DATA SCIENCE

BERBASIS PROYEK

PANDUAN PRAKTIS ANALISIS DATA

Ani Yoraeni, S.Pd., M.Kom.
Dr. H. Heri Kuswara, M.Kom.
Nicodias Palasara, M.Kom.
Herryansyah, M.Kom.
Yayan Hendrian, S.T., M.Kom.

DATA SCIENCE BERBASIS PROYEK

Panduan Praktis Analisis Data

Ani Yoraeni, S.Pd., M.Kom.
Dr. H. Heri Kuswara, M.Kom.
Nicodias Palasara, M.Kom.
Herryansyah, M.Kom.
Yayan Hendrian, S.T., M.Kom.

Takaza
INNOVATIX LABS

Data Science Berbasis Proyek: Panduan Praktis Analisis Data

Penulis	Ani Yoraeni, S.Pd., M.Kom. | Dr. H. Heri Kuswara, M.Kom. Nicodias Palasara, M.Kom. | Herryansyah, M.Kom. Yayan Hendrian, S.T., M.Kom.
Editor	Setia Purnama Wia Erlina
Desain Cover	Innovatix Labs Team
URL Buku	https://takazapublisher.com/shop/6aed539e-90d5-4253-a8bf-362b78cd5e8b
ISBN	978-634-7741-98-1
Ukuran	xi, 200, Uk: 15.5 × 23 cm
Cetakan Pertama	Juli 2026

Hak Cipta 2026, Pada Penulis

> *Hak cipta dilindungi undang-undang. Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari penerbit.*

Isi diluar tanggung jawab percetakan

Copyright © 2026 by Takaza Innovatix Labs
All Right Reserved

Penerbit Takaza Innovatix Labs
Anggota Ikatan Penerbit Indonesia (IKAPI) No.044/SBA/2023

Jl. Berlian Raya Blok M4, Pegambiran Ampalu Nan XX,
Lubuk Begalung, Kota Padang, Sumatera Barat
No Hp: +62 811 50321 47
Website: www.takaza.id
E-mail: bookspublishing@takaza.id

KATA PENGANTAR

Puji syukur senantiasa kita panjatkan kehadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga penyusunan buku referensi berjudul **"Data Science Berbasis Proyek: Panduan Praktis Analisis Data"** ini dapat diselesaikan dengan baik. Di era transformasi digital saat ini, penguasaan ilmu data tidak lagi cukup hanya sebatas pemahaman teoretis, melainkan menuntut kemampuan implementasi yang nyata. Oleh karena itu, buku ini dirancang secara khusus dengan pendekatan kolaboratif dan berbasis proyek untuk menjembatani kesenjangan antara konsep akademis seperti algoritma *machine learning* dan pemodelan prediktif dengan aplikasi praktis di dunia bisnis maupun penelitian. Melalui langkah-langkah yang terstruktur, pembaca akan dipandu untuk membongkar kerumitan data mentah menjadi wawasan strategis yang dapat langsung ditindaklanjuti.

Kehadiran literatur ini tentu tidak lepas dari kontribusi, dedikasi, serta diskusi panjang dengan berbagai rekan penulis dan akademisi yang turut memperkaya perspektif di setiap babnya. Kami berharap panduan praktis ini dapat menjadi literatur yang esensial dan inspiratif bagi mahasiswa, peneliti, hingga praktisi industri yang ingin menajamkan kompetensi analitis mereka. Menyadari bahwa disiplin ilmu pengetahuan ini terus berkembang dengan sangat cepat, kami senantiasa terbuka terhadap kritik dan saran yang membangun demi penyempurnaan karya ini di masa mendatang. Akhir kata, selamat membaca, mempraktikkan, dan mengeksplorasi potensi tak terbatas dari dunia analisis data.

Penulis

DAFTAR ISI

KATA PENGANTAR..... iii
DAFTAR ISI.....v
DAFTAR TABEL ix
DAFTAR GAMBAR.....xi
BAB I PENGANTAR DATA SCIENCE 1
A. Konsep dan Definisi Data Science 2
B. Peran Data dalam Era Digital 4
C. Komponen Utama Data Science 7
D. Siklus Hidup Proyek Data Science.....10
E. Peluang dan Tantangan Data Science 13
F. Studi Kasus Penggunaan Data Science 16
BAB II DASAR-DASAR STATISTIK UNTUK DATA SCIENCE.....19
A. Jenis dan Sumber Data.....19
B. Statistika Deskriptif 23
C. Distribusi Data 24
D. Pengukuran Variabilitas.....26
E. Korelasi dan Hubungan Antarvariabel.....28
F. Studi kasus.....30
BAB III DASAR-DASAR STATISTIK UNTUK DATA SCIENCE....33
A. Bahasa Pemrograman dalam Data Science 33
B. Instalasi dan Konfigurasi Lingkungan Kerja 38
C. Struktur Dasar Program.....41
D. Variabel dan Tipe Data 44
E. Operasi Data Dasar.....47
F. Studi kasus.....51

BAB IV AKUISISI DAN PENGUMPULAN DATA 55
A. Sumber Data Internal dan Eksternal..... 55
B. Teknik Pengumpulan Data..... 58
C. Penggunaan API dan Web Scraping..... 61
D. Data Open Source..... 64
E. Validasi Data..... 67
F. Studi kasus..... 70
BAB V DATA CLEANING DAN DATA PREPARATION 75
A. Pentingnya Kualitas Data..... 76
B. Penanganan Missing Values..... 78
C. Deteksi dan Penanganan Outlier..... 80
D. Transformasi Data..... 82
E. Integrasi Data..... 84
F. Studi kasus..... 87
BAB VI MANAJEMEN DATA DAN BASIS DATA..... 91
A. Konsep Basis Data..... 92
B. SQL Dasar..... 96
C. Pengelolaan Dataset Besar..... 100
D. Data Warehouse..... 104
E. Data Governance..... 107
F. Studi kasus..... 111
BAB VII EXPLORATORY DATA ANALYSIS (EDA)..... 115
A. Konsep EDA..... 116
B. Analisis Univariate..... 119
C. Analisis Bivariate..... 121
D. Analisis Multivariate..... 125
E. Interpretasi Hasil Analisis..... 128
F. Studi kasus..... 130

BAB VIII VISUALISASI DATA..... 133
A. Prinsip Visualisasi Data 134
B. Grafik dan Diagram Dasar..... 136
C. Dashboard Analitik..... 138
D. Storytelling dengan Data..... 141
E. Kesalahan Umum dalam Visualisasi..... 143
F. Studi kasus 145
BAB IX ANALISIS DATA UNTUK PENGAMBILAN KEPUTUSAN 149
A. Data-Driven Decision Making..... 150
B. Analisis Tren..... 152
C. Segmentasi Data..... 154
D. Analisis Performa..... 156
E. Penyusunan Insight Bisnis..... 158
F. Studi kasus 160
BAB X PENGANTAR MACHINE LEARNING 163
A. Konsep Machine Learning 164
B. Jenis-Jenis Machine Learning..... 165
C. Tahapan Pengembangan Model 166
D. Dataset Training dan Testing..... 172
E. Evaluasi Model..... 172
F. Studi kasus 173
BAB XI PENGANTAR MACHINE LEARNING 175
A. Konsep Machine Learning 176
B. Regresi Logistik 177
C. Decision Tree..... 178
D. Random Forest..... 180
E. Evaluasi Kinerja Model..... 181

F. Studi kasus 187
BAB XII UNSUPERVISED LEARNING 189
A. Clustering 190
B. K-Means 191
C. Hierarchical Clustering 192
D. Association Rule 194
E. Interpretasi Hasil Clustering 197
F. Studi kasus 198
DAFTAR PUSTAKA 201
BIOGRAFI PENULIS 207

DAFTAR TABEL

Tabel 1. Matriks Interaksi dan Karakteristik Komponen Utama Data Science 9
Tabel 2. Matriks Karakteristik Skala Pengukuran Data dan Operasi Statistik 22
Tabel 3. Dimensi Kualitas Data dalam Proses Validasi Akuisisi 69
Tabel 4. Perbandingan Algoritma Unsupervised Learning 200

DAFTAR GAMBAR

Gambar 1. Database Development.....93
Gambar 2. Data Warehouse Architecture.....104
Gambar 3. Diagram Pencar.....122

BAB I
PENGANTAR DATA SCIENCE

Di era informasi yang kita huni saat ini, data telah bermutasi menjadi komoditas paling berharga di muka bumi, sering kali disetarakan dengan minyak bumi di abad ke-20. Namun, data mentah betapapun besarnya hanyalah sekumpulan angka, teks, dan log yang sunyi dan tak memiliki makna inheren. Transformasi data mentah menjadi kebijaksanaan strategis inilah yang menjadi inti dari disiplin ilmu *Data Science*. Sebagai sebuah irisan multidisiplin yang mempertemukan matematika, statistika, pemrograman komputer, serta keahlian domain bisnis, *Data Science* bukan sekadar metode teknis, melainkan sebuah kerangka berpikir untuk mengurai kompleksitas dunia nyata menjadi wawasan yang dapat ditindaklanjuti.

Lahir dari konvergensi revolusi komputasi awan dan ledakan volume data digital, *Data Science* telah mengubah cara organisasi di berbagai sector mulai dari penerbitan akademik yang mengelola hak intelektual, hingga manajemen rantai pasok global dalam mengambil keputusan. Jika di masa lalu pengambilan keputusan didominasi oleh intuisi atau pengalaman heuristik belaka, hari ini kita berpindah ke era pengambilan keputusan berbasis bukti (*evidence-based decision making*). Bab ini akan memperkenalkan fondasi filosofis dan operasional dari *Data Science*, memetakan ekosistem kemampuannya, serta mendefinisikan peran krusial seorang ilmuwan data sebagai "penerjemah" antara bahasa mesin yang rumit dengan kebutuhan strategis manusia. Kita akan menelusuri bagaimana

perpaduan antara ketajaman matematis dan kreativitas analitis menjadi kunci utama dalam memecahkan masalah-masalah paling pelik di abad ke-21.

A. Konsep dan Definisi Data Science

Perkembangan teknologi informasi yang masif telah mengubah lanskap ilmu pengetahuan dan industri secara fundamental, melahirkan suatu disiplin ilmu baru yang kini dikenal sebagai *data science* (ilmu data). Secara epistemologis, *data science* tidak lahir dari ruang hampa, melainkan merupakan evolusi konvergen dari statistika terapan, ilmu komputer, matematika, dan analisis bisnis. Cao (2024) mendefinisikan *data science* sebagai disiplin ilmu yang mempelajari data secara komprehensif, mulai dari penciptaan, penangkapan, pemrosesan, analisis, hingga transformasi data menjadi pengetahuan empiris yang dapat ditindaklanjuti. Disiplin ini melampaui batas-batas analisis statistik konvensional dengan mengintegrasikan algoritma komputasi modern dan arsitektur komputasi skala besar guna mengekstrak pola tersembunyi (*latent patterns*) dari struktur data yang kompleks.

Dalam pemahaman yang lebih kontemporer, *data science* dipandang sebagai sebuah paradigma ilmiah keempat (*the fourth paradigm*) setelah paradigma empiris, teoritis, dan komputasional. Menurut Kelleher dan Tierney (2022), esensi utama dari *data science* adalah kemampuan untuk menavigasi ketidakpastian dan menemukan struktur di dalam data yang tidak terstruktur (*unstructured data*), seperti teks, gambar, video, dan sinyal digital. Pendekatan ini menuntut paradigma berpikir yang multidimensional, di mana seorang praktisi tidak hanya dituntut untuk menguasai aspek teknis algoritma, tetapi juga memiliki intuisi kritis dalam merumuskan masalah

ilmiah atau bisnis. Oleh karena itu, ilmu ini tidak sekadar berfokus pada alat (*tools*), melainkan pada metodologi pemecahan masalah secara saintifik.

Transformasi konseptual ini juga menggeser bagaimana organisasi memandang aset informasi mereka. Jika pada dekade sebelumnya data hanya berfungsi sebagai catatan historis pasif (sistem pelaporan), kini melalui *data science*, data bertransformasi menjadi instrumen prediktif dan preskriptif yang proaktif. Saltz dan Stanton (2021) menegaskan bahwa *data science* melibatkan proses iteratif yang ketat untuk menguji hipotesis berbasis data, di mana data mentah dieksplorasi secara mendalam guna meminimalkan bias kognitif manusia dalam pengambilan keputusan. Melalui integrasi teknik kecerdasan buatan (*artificial intelligence*) dan pembelajaran mesin (*machine learning*), disiplin ilmu ini mampu menghasilkan model-model matematika yang adaptif terhadap perubahan lingkungan data secara *real-time*.

Dilihat dari cakupan operasionalnya, perbedaan antara statistika tradisional dan *data science* terletak pada skala, variabilitas, dan tujuan akhirnya. Statistika tradisional umumnya bekerja dengan sampel data yang representatif di bawah asumsi-asumsi matematis yang ketat untuk melakukan inferensi populasi. Sebaliknya, *data science* beroperasi pada ekosistem data raya (*big data*) yang sering kali tidak memenuhi asumsi klasik statistika, sehingga memerlukan pendekatan algoritmik yang lebih fleksibel namun tetap valid secara empiris (Borgman, 2021). Hal ini menjadikan *data science* sebagai jembatan yang krusial dalam mentransfer teori-teori murni matematika ke dalam aplikasi praktis yang berdampak langsung pada efisiensi operasional dan inovasi di berbagai sektor.

Lebih lanjut, karakteristik utama dari *data science* kontemporer adalah penekanannya pada pendekatan berbasis proyek (*project-based approach*)

dan pemecahan masalah dunia nyata. Ilmu ini tidak lagi dikurung dalam laboratorium komputasi yang terisolasi, melainkan diintegrasikan langsung ke dalam alur kerja industri, kebijakan publik, dan riset biomedis. Mitchell (2023) menyatakan bahwa keberhasilan penerapan *data science* diukur dari sejauh mana model analitik yang dibangun mampu memberikan nilai tambah ekonomi atau sosial, serta memitigasi risiko kegagalan sistemik. Dengan demikian, pemahaman konseptual yang kokoh mengenai dinamika data menjadi prasyarat mutlak sebelum melangkah pada implementasi teknis yang lebih rumit.

Sebagai penutup dalam memahami konsep dasar ini, penting untuk digarisbawahi bahwa *data science* bersifat dinamis dan terus berevolusi seiring dengan lompatan teknologi perangkat keras dan lunak. Munculnya arsitektur model bahasa besar (*large language models*) dan komputasi awan (*cloud computing*) dalam beberapa tahun terakhir semakin memperluas batas kemampuan ilmu ini dalam memproses informasi berskala eksbita (Zheng, 2023). Mahasiswa dan praktisi yang mempelajari disiplin ini tidak hanya diajarkan untuk menjadi pengguna algoritma siap pakai, melainkan dibentuk untuk menjadi pemikir analitis yang mampu merancang solusi berbasis data secara etis, valid, dan dapat dipertanggungjawabkan secara ilmiah.

B. Peran Data dalam Era Digital

Di era transformasi digital yang masif, data telah memantapkan posisinya sebagai komoditas paling berharga, bahkan sering kali dianalogikan sebagai "bahan bakar baru" (*the new oil*) yang menggerakkan roda peradaban modern. Ledakan volume data ini dipicu oleh proliferasi perangkat internet untuk segala (*Internet of Things* - IoT), adopsi media

sosial yang eksponensial, serta digitalisasi layanan komersial dan pemerintahan. Davenport (2023) mengemukakan bahwa fenomena penumpukan data ini bukan sekadar masalah kuantitas, melainkan representasi dari pergeseran kultural di mana setiap aktivitas manusia kini meninggalkan jejak digital (*digital footprint*) yang kaya akan informasi potensial. Peran data telah berubah dari yang semula hanya sebagai produk sampingan transaksi administratif menjadi aset strategis utama bagi keberlanjutan organisasi.

Secara makro, eksistensi data dalam jumlah besar telah mengubah struktur kompetisi industri secara global. Perusahaan-perusahaan yang mengadopsi budaya berbasis data (*data-driven culture*) terbukti memiliki ketahanan yang lebih tinggi terhadap volatilitas pasar dan mampu mendeteksi perubahan preferensi konsumen secara instan. Ghasemaghaei dan Calic (2022) dalam penelitiannya menunjukkan bahwa kualitas dan kelancaran aliran data dalam suatu organisasi berkorelasi langsung dengan kemampuan inovasi produk dan efisiensi operasional. Data tidak lagi disimpan dalam repositori terisolasi (*data silos*), melainkan dialirkan ke seluruh lini organisasi untuk memandu setiap kebijakan eksekutif maupun operasional harian, meminimalkan ketergantungan pada intuisi subjektif.

Di sektor publik dan pemerintahan, peran data di era digital berpusat pada perwujudan tata kelola yang transparan dan pembuatan kebijakan berbasis bukti (*evidence-based policy*). Data demografi, kesehatan, transportasi, dan ekonomi yang terintegrasi memungkinkan pemerintah merancang program jaminan sosial yang tepat sasaran serta merespons krisis atau pandemi dengan kalkulasi yang presisi. Janssen et al. (2021) menekankan bahwa pemanfaatan data publik yang optimal dapat memitigasi

risiko korupsi dan meningkatkan partisipasi warga dalam pembangunan kota cerdas (*smart city*). Melalui visualisasi data yang dapat diakses publik, transparansi birokrasi dapat ditingkatkan secara signifikan, menciptakan ekosistem sosial yang lebih demokratis dan responsif.

Namun, peran data yang begitu dominan ini juga membawa konsekuensi epistemologis terhadap bagaimana kebenaran ilmiah dikonstruksikan. Dalam era digital, fenomena sosial maupun alam tidak lagi hanya diamati melalui pendekatan kualitatif naratif, melainkan dikuantifikasi secara masif untuk menemukan korelasi-korelasi baru yang sebelumnya tidak terdeteksi oleh panca indra manusia. Mayer-Schönberger dan Cukier (2021) menyatakan bahwa ketersediaan data dalam skala besar memungkinkan ilmuwan beralih dari pencarian kausalitas yang memakan waktu lama menuju pemanfaatan korelasi prediktif yang instan namun akurat untuk kebutuhan praktis. Hal ini mempercepat siklus riset dan pengembangan di berbagai sektor, mulai dari penemuan obat baru hingga mitigasi perubahan iklim.

Dari perspektif mikroekonomi, data memegang peran sentral dalam menciptakan pengalaman konsumen yang sangat terpersonalisasi (*hyper-personalization*). Setiap klik, durasi penayangan, dan riwayat transaksi digunakan oleh algoritma untuk menyusun profil preferensi individu, yang kemudian digunakan untuk merekomendasikan produk atau konten yang paling relevan. Zuboff (2021) mengidentifikasi fenomena ini sebagai bagian dari kapitalisme pengawasan (*surveillance capitalism*), di mana perilaku manusia dikomodifikasi melalui ekstraksi data perilaku. Realitas ini menunjukkan bahwa data di era digital memiliki pisau bermata dua: di satu sisi mendorong kenyamanan konsumen dan pertumbuhan ekonomi, di sisi

lain menuntut perhatian serius terhadap aspek privasi dan kedaulatan informasi individu.

Paragraf terakhir mengenai peran data ini menegaskan bahwa masa depan peradaban digital akan sangat ditentukan oleh bagaimana masyarakat mengelola, menyaring, dan memanfaatkan data. Kesenjangan dalam kemampuan mengolah data (*data literacy gap*) akan menjadi faktor pembeda utama antara negara maju dan berkembang, serta antara organisasi yang progresif dan yang stagnan. Oleh karena itu, investasi pada infrastruktur data keras (*hardware* dan jaringan) harus berjalan selaras dengan pengembangan kapasitas manusia yang mampu menerjemahkan angka-angka mentah menjadi narasi yang bermakna dan kebijakan yang humanis (Manyika et al., 2024). Data, dalam segala bentuknya, telah menjadi fondasi dari realitas baru yang kita tinggali saat ini.

Assets of Takaza INNOVATIX LABS

C. Komponen Utama Data Science

Keberhasilan sebuah proyek *data science* sangat bergantung pada sinergi tiga komponen fundamental yang saling beririsan. Komponen-komponen ini membentuk fondasi yang kokoh, memastikan bahwa analisis yang dihasilkan tidak hanya valid secara matematis, tetapi juga efisien secara komputasi dan relevan secara praktis. Kelleher dan Tierney (2022) memformulasikan tiga pilar utama tersebut sebagai berikut:

1. **Matematika dan Statistika (*Mathematics and Statistics*)**

   Komponen ini merupakan jantung teoritis dari *data science*. Tanpa fondasi matematika yang kuat khususnya aljabar linear, kalkulus, teori peluang, dan statistik inferensial seorang *data scientist* hanya akan menggunakan algoritma sebagai "kotak hitam" (*black box*) tanpa

memahami mekanisme kerja di dalamnya. Statistika menyediakan kerangka kerja untuk mengukur ketidakpastian, menguji hipotesis, dan memvalidasi apakah pola yang ditemukan pada data merupakan representasi dari realitas atau sekadar kebetulan statistik (*statistical noise*) (Saltz & Stanton, 2021). Aljabar linear dan kalkulus memainkan peran penting dalam proses optimasi algoritma pembelajaran mesin, terutama dalam penyesuaian bobot pada jaringan saraf tiruan (*neural networks*).

2. Ilmu Komputer dan Teknologi Informasi (*Computer Science and Information Technology*)

Pilar kedua ini menyediakan instrumen teknis untuk mengeksekusi teori-teori matematika pada data berskala besar. Komponen ini mencakup penguasaan bahasa pemrograman (seperti Python dan R), pemahaman tentang struktur data, algoritma komputasi, serta pengelolaan basis data baik relasional (SQL) maupun non-relasional (NoSQL). Di era *big data*, ilmu komputer juga mencakup kemampuan mengoperasikan arsitektur komputasi terdistribusi seperti Apache Hadoop dan Spark, serta pemanfaatan platform komputasi awan (Zheng, 2023). Ketangkasan dalam menulis kode yang efisien sangat krusial karena model matematika yang canggih tidak akan berguna jika tidak dapat dieksekusi dalam waktu yang masuk akal akibat algoritma yang tidak efisien.

3. Pengetahuan Domain Bisnis (*Domain Business Knowledge*)

Komponen ketiga yang sering kali menjadi penentu keberhasilan proyek adalah pemahaman mendalam mengenai domain spesifik tempat data tersebut berasal. Seorang ahli data tidak dapat menganalisis data

medis tanpa pemahaman dasar tentang bio-informatika, atau menganalisis data keuangan tanpa mengerti instrumen pasar modal. Pengetahuan domain berfungsi sebagai pemandu dalam merumuskan pertanyaan bisnis yang tepat, memilih fitur data yang relevan (*feature engineering*), dan menginterpretasikan hasil pemodelan menjadi strategi yang konkret (Cao, 2024). Tanpa komponen ini, analisis data berisiko menghasilkan kesimpulan yang secara teknis benar namun tidak memiliki nilai guna praktis bagi organisasi.

Untuk memberikan gambaran yang lebih sistematis mengenai bagaimana ketiga komponen ini berinteraksi dan membentuk peran-peran spesifik dalam ekosistem data, Tabel 1.1 di bawah ini menyajikan perbandingan komprehensif berdasarkan fokus keahlian, peran utama, instrumen yang digunakan, dan keluarannya (*output*).

Assets of Telkuza INNOVATIX LABS

Tabel 1. Matriks Interaksi dan Karakteristik Komponen Utama Data Science

Karakteristik / Dimensi	Matematika & Statistika	Ilmu Komputer & TI	Pengetahuan Domain Bisnis
Fokus Keahlian Utama	Validasi teoritis, pemodelan probabilitas, dan desain eksperimen.	Rekayasa perangkat lunak, arsitektur data, dan optimasi kode.	Kontekstualisasi masalah, perumusan KPI, dan nilai strategis.
Peran Utama dalam	*Data Scientist / Statistician*	*Data Engineer /*	*Business Analyst / Subject Matter*

Karakteristik / Dimensi	Matematika & Statistika	Ilmu Komputer & TI	Pengetahuan Domain Bisnis
Ekosistem		*Machine Learning Engineer*	*Expert*
Instrumen & Alat Kerja	R, SPSS, MATLAB, Python (SciPy, Statsmodels).	Python, SQL, Docker, Spark, AWS/GCP.	Dashboard BI (Tableau, PowerBI), Excel, Kerangka Kerja Bisnis.
Keluaran Utama (*Output*)	Formulasi algoritma, nilai signifikansi ($p-value$), matriks evaluasi model.	Pipeline data otomatis, API model siap pakai, arsitektur basis data.	Rekomendasi kebijakan, ROI proyek, visualisasi pemangku kepentingan.
Dampak Jika Komponen Absen	Model mengalami *overfitting*, bias, dan menghasilkan kesimpulan keliru.	Analisis berjalan lambat, tidak dapat diskalakan (*unscalable*), sistem sering eror.	Hasil analisis tidak relevan, tidak dapat diterapkan, dan membuang sumber daya.

Assets of Takaza INNOVATIX LABS

D. Siklus Hidup Proyek Data Science

Sebuah proyek *data science* yang sukses tidak berjalan secara acak, melainkan mengikuti metodologi terstruktur yang mengatur alur kerja dari

hulu ke hilir. Salah satu kerangka kerja yang paling banyak diadopsi dalam industri dan riset akademik adalah CRISP-DM (*Cross-Industry Standard Process for Data Mining*) yang telah dimodifikasi agar relevan dengan ekosistem modern, atau yang sering disebut sebagai *Modern Data Science Lifecycle*. Menurut Martinez-Plumed et al. (2021), siklus hidup ini bersifat iteratif, artinya hasil yang diperoleh pada suatu tahapan sering kali mengharuskan tim untuk kembali dan memperbaiki tahapan sebelumnya. Siklus ini memastikan bahwa setiap investasi teknologi dan waktu yang dikeluarkan selaras dengan tujuan strategis organisasi.

Tahapan-tahapan di bawah ini menjelaskan secara rinci siklus hidup proyek *data science* konvensional hingga modern (Saltz et al., 2023):

1. **Pemahaman Bisnis (*Business Understanding*)** Fase inisiasi ini berfokus pada perumusan masalah yang ingin diselesaikan dari perspektif bisnis atau riset. Aktivitas utama meliputi penetapan tujuan proyek, identifikasi sumber daya yang tersedia, pemetaan batasan teknis, dan penerjemahan tujuan bisnis menjadi tujuan analitik data yang spesifik. Kegagalan pada tahap ini akan menyebabkan seluruh tahapan berikutnya menjadi tidak relevan, karena algoritma terbaik sekalipun tidak akan menghasilkan nilai guna jika menyelesaikan masalah yang salah.
2. **Akuisisi dan Pemahaman Data (*Data Acquisition and Understanding*)** Setelah tujuan ditetapkan, langkah berikutnya adalah mengidentifikasi dan mengumpulkan sumber data yang relevan, baik dari basis data internal, API pihak ketiga, maupun hasil *web scraping*. Pada tahap ini, dilakukan pemeriksaan awal terhadap kualitas data untuk mendeteksi keberadaan nilai yang hilang

(*missing values*), pencilan (*outliers*), atau inkonsistensi format. Evaluasi awal ini krusial untuk menentukan apakah data yang tersedia cukup representatif untuk menjawab pertanyaan bisnis yang telah dirumuskan.

3. **Pembersihan dan Persiapan Data (*Data Cleaning and Preparation*)** Tahap ini sering kali memakan waktu hingga 70% dari total durasi proyek. Aktivitasnya meliputi pembersihan data dari kesalahan input, penanganan nilai yang hilang melalui teknik imputasi, standardisasi format, dan transformasi variabel. Termasuk di dalamnya adalah proses *feature engineering*, yaitu menciptakan variabel-variabel baru dari data mentah yang dapat meningkatkan daya prediksi dari model pembelajaran mesin yang akan dibangun.
4. **Eksplorasi Data (*Exploratory Data Analysis - EDA*)** EDA adalah proses investigasi awal pada data untuk menemukan pola, mendeteksi anomali, dan menguji hipotesis melalui ringkasan statistik dan visualisasi grafis. Tahap ini membantu *data scientist* mengembangkan intuisi yang mendalam tentang karakteristik data mereka, mengidentifikasi hubungan antarvariabel (korelasi), dan memilih teknik pemodelan yang paling sesuai untuk tahapan berikutnya.
5. **Pemodelan (*Modeling*)** Pada fase ini, berbagai algoritma pembelajaran mesin (*machine learning*) diterapkan pada dataset yang telah disiapkan. Tergantung pada karakteristik masalahnya, ilmuwan data dapat memilih teknik *supervised learning* (seperti regresi atau klasifikasi) atau *unsupervised learning* (seperti *clustering*). Proses ini melibatkan pembagian data menjadi *training*

set dan *testing set*, serta penyetelan hiperparameter (*hyperparameter tuning*) untuk mengoptimalkan performa model.

6. **Evaluasi Model (*Evaluation*)** Model yang menghasilkan akurasi tinggi pada data latih belum tentu dapat berkinerja baik di dunia nyata. Pada tahap evaluasi, model diuji menggunakan metrik yang ketat seperti *accuracy, precision, recall, F1-score*, atau MSE (*Mean Squared Error*) pada data uji yang belum pernah dilihat sebelumnya. Selain performa teknis, model juga dievaluasi apakah benar-benar menjawab kebutuhan bisnis yang ditetapkan pada fase pertama dan tidak mengandung bias yang merugikan.
7. **Penyebaran dan Pemantauan (*Deployment and Monitoring*)** Tahap akhir dari siklus ini adalah mengintegrasikan model yang telah divalidasi ke dalam lingkungan produksi (*production environment*), seperti aplikasi web, sistem otomatisasi industri, atau dashboard eksekutif. Proses ini tidak berhenti setelah model aktif; pemantauan secara berkala wajib dilakukan untuk mendeteksi penurunan performa model akibat perubahan karakteristik data riil di lapangan, sebuah fenomena yang dikenal sebagai *data drift* atau *concept drift*.

E. Peluang dan Tantangan Data Science

Penerapan *data science* membuka pintu peluang yang sangat luas bagi inovasi global, namun di saat yang sama memunculkan tantangan teknis dan etis yang kompleks. Dari sisi peluang, otomatisasi proses analitis berkemampuan tinggi memungkinkan organisasi melakukan efisiensi biaya secara radikal dan menciptakan aliran pendapatan baru (*new revenue*

streams) berbasis produk data. McKinsey (2024) melaporkan bahwa adopsi analitika tingkat lanjut berbasis kecerdasan buatan berkontribusi signifikan terhadap peningkatan produktivitas global, khususnya dalam optimalisasi rantai pasok (*supply chain*) dan personalisasi layanan pelanggan massal. Bagi profesional di bidang ini, tingginya permintaan pasar yang tidak sebanding dengan ketersediaan talenta menjadikan profesi *data scientist* sebagai salah satu karier paling menjanjikan dengan nilai kompensasi yang sangat kompetitif.

Di ranah riset ilmiah, peluang yang dihadapi juga tidak kalah revolusioner. *Data science* mempercepat penemuan di bidang genomika, astronomi, dan material baru melalui pemrosesan simulasi digital skala besar yang sebelumnya membutuhkan waktu puluhan tahun jika dilakukan secara manual di laboratorium fisik. Floridi (2023) mencatat bahwa kemampuan ilmu data dalam mengidentifikasi pola makro dari fenomena iklim global memberikan harapan baru bagi perancangan kebijakan mitigasi bencana yang lebih presisi. Fleksibilitas metode ini menjadikannya katalisator utama bagi lahirnya riset-riset interdisipliner yang mampu memecahkan masalah kemanusiaan yang kompleks.

Namun, di balik optimisme tersebut, tantangan besar menghadang pada aspek privasi data dan kepatuhan hukum. Implementasi pengumpulan data yang masif sering kali menabrak batas privasi individu, memicu lahirnya regulasi ketat seperti GDPR (*General Data Protection Regulation*) di Uni Eropa atau UU Perlindungan Data Pribadi (UU PDP) di Indonesia. Organisasi kini dihadapkan pada kewajiban hukum yang berat untuk memastikan bahwa data yang mereka kumpulkan diolah dengan persetujuan eksplisit pengguna (*informed consent*) dan dilindungi dari risiko kebocoran

data (*data breach*) oleh peretas yang semakin canggih (Jobin et al., 2021). Tantangan kepatuhan ini menuntut biaya investasi infrastruktur keamanan siber yang tidak sedikit.

Tantangan teknis lainnya berpusat pada masalah kualitas data mentah dan fenomena "sampah masuk, sampah keluar" (*garbage in, garbage out*). Banyak organisasi memiliki volume data yang melimpah namun tersimpan dalam format yang berantakan, tidak konsisten, dan terfragmentasi di berbagai sistem warisan (*legacy systems*). Penanganan data kotor ini membutuhkan energi dan waktu yang besar sebelum algoritma dapat dijalankan. Selain itu, ketergantungan pada model-model yang terlalu kompleks berisiko menghasilkan *overfitting*, di mana model tampak sempurna saat diuji di laboratorium namun gagal total ketika dihadapkan pada dinamika data dunia nyata yang fluktuatif.

Aspek etis dari penggunaan algoritma juga menjadi perhatian kritis komunitas ilmiah global saat ini. Algoritma pembelajaran mesin belajar dari data historis; jika data historis tersebut mengandung bias manusia seperti bias jender, ras, atau kelas sosial-ekonomi maka model yang dihasilkan akan melanggengkan bahkan memperkuat diskriminasi tersebut secara otomatis. O'Neil (2022) menyebut fenomena ini sebagai "senjata penghancur matematika" (*weapons of math destruction*), di mana keputusan otomatis seperti kelayakan kredit, rekrutmen karyawan, atau penilaian risiko kriminal dijalankan oleh algoritma bias yang tidak transparan. Kurangnya akuntabilitas model ini menciptakan tantangan besar dalam membangun *Explainable AI* (XAI), yaitu kecerdasan buatan yang keputusan-keputusannya dapat dinalar dan dijelaskan kepada publik.

Terakhir, tantangan institusional dalam bentuk resistensi kultural terhadap perubahan sering kali menjadi batu sandungan utama di lapangan. Banyak pemimpin organisasi yang masih enggan melepaskan intuisi personal atau pengalaman masa lalu mereka untuk tunduk pada rekomendasi obyektif yang dihasilkan oleh data analitik. Mengubah pola pikir organisasi tradisional menjadi entitas yang berbasis data membutuhkan kepemimpinan yang visioner dan program literasi data yang inklusif di semua level staf. Tanpa transformasi budaya internal, model *data science* secanggih apa pun hanya akan berakhir sebagai dokumen teknis yang tidak pernah diimplementasikan dalam kebijakan strategis.

F. Studi Kasus Penggunaan Data Science

Untuk memahami bagaimana teori-teori *data science* diaplikasikan secara konkret, kita dapat meninjau implementasinya pada tiga sektor utama: perdagangan elektronik (*e-commerce*), pelayanan kesehatan (*healthcare*), dan optimasi transportasi perkotaan. Pada sektor *e-commerce*, perusahaan raksasa seperti Amazon dan Tokopedia memanfaatkan sistem rekomendasi berbasis *collaborative filtering* dan *deep learning* untuk menyajikan katalog produk yang unik bagi setiap pengguna. Algoritma menganalisis kombinasi riwayat pencarian, durasi pandangan mata pada layar, transaksi masa lalu, dan profil demografi sejenis secara *real-time*. Menurut Aggarwal (2023), sistem rekomendasi berbasis data ini berkontribusi hingga 35% dari total volume penjualan platform perdagangan digital modern, membuktikan bahwa presisi analitik berkorelasi langsung dengan peningkatan profitabilitas bisnis.

Di sektor pelayanan kesehatan, *data science* telah membawa revolusi dalam deteksi dini penyakit kritis, salah satunya melalui analisis citra medis berbasis *computer vision*. Topol (2021) memaparkan bagaimana model pembelajaran mesin berbasis *Convolutional Neural Networks* (CNN) dilatih menggunakan jutaan data pemindaian sinar-X dan MRI untuk mendeteksi sel kanker dengan tingkat akurasi yang menyamai, bahkan dalam beberapa kasus melampaui, tim dokter spesialis radiologi. Keunggulan model ini terletak pada kemampuannya mendeteksi anomali piksel mikroskopis yang luput dari pandangan mata manusia. Implementasi ini tidak bertujuan menggantikan peran dokter, melainkan sebagai instrumen triase yang mempercepat penanganan pasien di ruang gawat darurat dan meminimalkan tingkat kesalahan diagnosis (*diagnostic errors*).

Sementara itu, di sektor transportasi dan tata kelola kota cerdas (*smart city*), pemerintah kota metropolitan seperti London dan Singapura menggunakan analisis data aliran lalu lintas publik untuk memitigasi kemacetan ekstrem. Sinyal GPS dari bus kota, taksi, dan aplikasi navigasi ponsel pintar warga dikumpulkan secara terpusat untuk memodelkan kemacetan secara dinamis. Dengan menerapkan algoritma prediksi deret waktu (*time-series forecasting*), sistem dapat memprediksi simpul kemacetan 30 menit sebelum terjadi, dan secara otomatis menyesuaikan durasi lampu lalu lintas atau mengalihkan rute transportasi publik (Zheng, 2023). Pendekatan berbasis data ini terbukti menghemat jutaan jam produktivitas warga yang biasanya terbuang di jalan raya dan menekan emisi karbon perkotaan secara signifikan.

Assets of Takaza INNOVATIX LABS

BAB II
DASAR-DASAR STATISTIK UNTUK DATA SCIENCE

Statistika merupakan fondasi epistemologis dan artikulasi matematis utama yang menggerakkan seluruh algoritma dalam *data science*. Tanpa pemahaman statistika yang kokoh, aktivitas analisis data hanya akan terjebak pada penggunaan perangkat lunak tanpa arah, yang sering kali menghasilkan kesimpulan semu (*spurious conclusions*). Bab ini dirancang untuk membekali mahasiswa dan praktisi dengan kerangka berpikir induktif, mulai dari pemahaman karakteristik data, ringkasan numerik, hingga pola distribusi dan asosiasi antarvariabel. Melalui pendekatan berbasis proyek, setiap konsep statistik dalam bab ini akan ditekankan pada bagaimana prinsip-prinsip matematis tersebut ditransformasikan menjadi keputusan bisnis atau kebijakan publik yang valid dan dapat dipertanggungjawabkan secara ilmiah.

A. Jenis dan Sumber Data

Eksplorasi dalam *data science* selalu bermula dari identifikasi jenis dan skala pengukuran data, karena hal ini menentukan jenis algoritma pembelajaran mesin atau uji statistik yang dapat diterapkan. Secara garis besar, data diklasifikasikan menjadi dua kategori utama, yaitu data kualitatif (kategorikal) dan data kuantitatif (numerik). Menurut Bruce et al. (2021), kesalahan paling umum dalam fase awal proyek data adalah memperlakukan

data kategorikal sebagai data numerik hanya karena data tersebut dikodekan dengan angka, yang pada akhirnya mendistorsi seluruh hasil pemodelan. Pemahaman yang keliru terhadap struktur data ini akan menyebabkan kegagalan sistemik pada fase *feature engineering*.

Untuk membedah data secara lebih spesifik, ilmu statistik membaginya ke dalam empat skala pengukuran yang diintroduksikan secara hierarkis: nominal, ordinal, interval, dan ratio. Skala nominal merupakan tingkatan paling dasar di mana angka atau label hanya berfungsi sebagai pembeda tanpa adanya tingkatan nilai, seperti jenis kelamin atau kategori produk. Skala ordinal melangkah lebih jauh dengan menyertakan unsur urutan atau tingkatan antar-kategori, namun jarak antar-tingkatan tersebut tidak dapat dikuantifikasi secara pasti, contohnya adalah tingkat kepuasan pelanggan (Heumann et al., 2023). Memahami batas-batas operasional dari skala kategorikal ini sangat krusial agar analis tidak terjebak melakukan operasi aritmatika yang mustahil, seperti menghitung rata-rata dari variabel nominal.

Assets of Takaza INNOVATIX LABS

Pada tingkatan yang lebih tinggi, terdapat data numerik yang diwakili oleh skala interval dan ratio. Skala interval memiliki jarak antar-nilai yang konsisten dan dapat diukur, namun tidak memiliki nilai nol mutlak, yang berarti ketiadaan nilai bukan mencerminkan ketiadaan atribut (seperti ukuran suhu dalam Celcius). Sebaliknya, skala ratio memiliki seluruh karakteristik skala interval ditambah dengan kepemilikan nilai nol mutlak, seperti pendapatan, berat badan, atau jumlah unduhan aplikasi (Navidi, 2024). Di dalam ekosistem *data science*, data ratio sangat disukai karena sifatnya yang fleksibel untuk ditransformasikan secara matematis menggunakan algoritma mutakhir yang sensitif terhadap jarak spasial.

Selain dari karakteristik skalanya, seorang praktisi data juga harus membedakan data berdasarkan dimensi waktu pengumpulannya, yaitu data silang (*cross-sectional*) dan data deret waktu (*time-series*). Data silang menangkap informasi dari banyak objek dalam satu titik waktu tertentu, memberikan potret statis dari suatu fenomena. Sementara itu, data deret waktu merekam informasi dari satu atau lebih objek secara berkesinambungan dalam interval waktu yang teratur (Hyndman & Athanasopoulos, 2021). Perbedaan dimensi ini menuntut arsitektur komputasi dan algoritma yang berbeda; data deret waktu membutuhkan penanganan khusus terhadap unsur tren dan musiman (*seasonality*) agar prediksi masa depan tidak bias.

Dari aspek asal-usulnya, sumber data dalam era modern dikelompokkan menjadi data internal organisasi dan data eksternal yang diperoleh dari ekosistem digital yang lebih luas. Data internal mencakup catatan transaksi finansial, log aktivitas pengguna sistem, dan basis data operasional perusahaan. Di sisi lain, data eksternal melibatkan data yang diekstrak dari media sosial, satelit cuaca, open data pemerintah, hingga hasil pemantauan sensor IoT (Davenport, 2023). Keberhasilan proyek *data science* berskala besar sering kali ditentukan oleh kemahiran tim dalam mengintegrasikan kedua sumber data ini guna memperoleh sudut pandang analitik yang komprehensif dan multiperspektif.

Sebagai panduan praktis dalam fase desain proyek, Tabel 2.1 di bawah ini meringkas karakteristik setiap skala pengukuran data beserta jenis operasi statistik yang diizinkan, sehingga mencegah terjadinya kesalahan metodologis dalam pemilihan visualisasi maupun pemodelan selanjutnya.

Tabel 2. Matriks Karakteristik Skala Pengukuran Data dan Operasi Statistik

Skala Pengukuran	Karakteristik Utama	Operasi Matematika yang Diizinkan	Contoh Sektor Industri	Statistik Deskriptif yang Sesuai
Nominal	Hanya klasifikasi label, tidak ada urutan atau nilai numerik instrinsik.	Persamaan (= ,≠)	Kategori industri saham, jenis metode pembayaran.	Modus, Frekuensi Persentase
Ordinal	Memiliki urutan logis, namun jarak antar-label tidak sama/terukur.	Perbandingan (>, <)	Peringkat kepuasan konsumen, tingkat risiko kredit.	Median, Modus, Rentang Interkuartil
Interval	Jarak antar-nilai konsisten, tidak memiliki nilai nol mutlak.	Penjumlahan, Pengurangan (+, -)	Skor indeks kepuasan kerja, temperatur server komputasi.	Mean, Median, Simpangan Baku
Ratio	Memiliki jarak konsisten dan nilai nol mutlak (arti ketiadaan fisik).	Perkalian, Pembagian x, ÷)	Nilai transaksi penjualan, jumlah retensi pengguna aktif.	Semua ukuran numerik dan transformasi logaritma

Sumber: Disintesis dari Bruce et al. (2021) dan Heumann et al. (2023)

B. Statistika Deskriptif

Statistika deskriptif merupakan fase krusial dalam *data science* yang bertujuan untuk merangkum, mengorganisasikan, dan menyajikan karakteristik utama dari suatu dataset tanpa menarik kesimpulan umum bagi populasi yang lebih luas. Melalui pendekatan deskriptif, kumpulan data mentah yang berjumlah jutaan baris ditransformasikan menjadi representasi numerik ringkas yang menceritakan struktur internal data tersebut. Field (2022) menegaskan bahwa melompati tahapan statistik deskriptif langsung menuju pemodelan prediktif yang kompleks adalah tindakan ceroboh, karena kesalahan input data atau anomali distribusi tidak akan terdeteksi tanpa adanya ringkasan numerik awal yang ketat.

Komponen pertama dari statistika deskriptif adalah ukuran pemusatan data (*measures of central tendency*), yang terdiri dari rata-rata (*mean*), median, dan modus. Rata-rata dihitung dengan menjumlahkan seluruh nilai data dibagi dengan jumlah observasi, memberikan gambaran pusat massa dari data kuantitatif. Namun, rata-rata memiliki kelemahan fatal yaitu sangat sensitif terhadap nilai-nilai ekstrem atau pencilan (*outliers*). Oleh karena itu, *data scientist* sering kali beralih ke median nilai tengah dari data yang telah diurutkan sebagai ukuran pemusatan yang lebih kokoh (*robust*) ketika menghadapi dataset yang menceng, seperti data distribusi kekayaan atau harga properti (Montgomery & Runger, 2022). Modus, sebagai nilai yang paling sering muncul, menjadi andalan utama untuk merangkum nilai pusat pada skala nominal.

Komponen kedua yang tidak kalah penting adalah analisis bentuk distribusi melalui pengukuran kemencengan (*skewness*) dan keruncingan (*kurtosis*). *Skewness* mengukur derajat ketidaksimetrisan suatu distribusi

data di sekitar rata-ratanya; nilai *skewness* positif mengindikasikan bahwa ekor distribusi memanjang ke arah kanan, yang berarti mayoritas data bernilai kecil namun terdapat segelintir data bernilai sangat besar. Sebaliknya, *kurtosis* mengukur derajat keruncingan atau "ketebalan ekor" (*heavy-tailed*) dari distribusi dibandingkan dengan distribusi normal (Navidi, 2024). Mengetahui nilai *kurtosis* suatu dataset membantu praktisi data mengantisipasi risiko kemunculan nilai ekstrem yang dapat merusak akurasi model regresi linier standar.

Dalam praktik komputasi modern menggunakan Python atau R, fungsi deskriptif otomatis seperti describe() atau summary() sering kali digunakan untuk memunculkan ringkasan lima angka (*five-number summary*). Ringkasan ini meliputi nilai minimum, kuartil pertama (Q_1), median (Q_2), kuartil ketiga (Q_3), dan nilai maksimum. Menurut Bruce et al. (2021), ringkasan lima angka ini memberikan fondasi geometris bagi pembuatan grafik *box plot*, yang merupakan alat visualisasi standar industri untuk mendeteksi pencilan secara instan. Dengan mengintegrasikan nilai numerik deskriptif ini ke dalam alur kerja analitik awal, tim proyek dapat merancang strategi penanganan data hilang secara lebih kontekstual dan akurat.

C. Distribusi Data

Distribusi data mendeskripsikan bagaimana nilai-nilai dari suatu variabel tersebar dan seberapa sering nilai-nilai tersebut muncul dalam sebuah ruang sampel. Memahami bentuk distribusi data merupakan salah satu pilar utama dalam *data science*, karena setiap algoritma pembelajaran mesin dibangun di atas asumsi-asumsi distribusi tertentu. Salah satu konsep paling fundamental dalam teori peluang dan statistika adalah Distribusi

Normal, atau yang sering disebut sebagai Distribusi Gauss (*Gaussian Distribution*). Distribusi ini berbentuk simetris menyerupai lonceng, di mana nilai rata-rata, median, dan modus berada tepat di satu titik pusat yang sama (Montgomery & Runger, 2022).

Eksistensi distribusi normal didasarkan pada sebuah prinsip matematis universal yang dikenal sebagai Teorema Limit Pusat (*Central Limit Theorem* - CLT). Teorema ini menyatakan bahwa jika kita mengambil sampel acak berukuran cukup besar dari populasi apa pun (bahkan populasi yang distribusinya sangat menceng), maka distribusi dari rata-rata sampel tersebut akan mendekati distribusi normal seiring bertambahnya ukuran sampel (Navidi, 2024). Di era *big data*, CLT menjadi penyelamat teoritis yang memungkinkan para analis melakukan inferensi statistik dan membangun model prediktif pada dataset raksasa tanpa perlu khawatir tentang ketidaknormalan bentuk asli data populasi di tingkat mikro.

Dalam ruang lingkup operasional, distribusi normal diatur oleh Aturan Empiris (*Empirical Rule*) atau aturan 68-95-99.7. Aturan ini menetapkan bahwa dalam dataset yang berdistribusi normal, sekitar 68% data akan jatuh di dalam rentang satu simpangan baku ($\pm 1\sigma$) dari rata-rata, 95% data berada di dalam rentang dua simpangan baku ($\pm 2\sigma$), dan 99.7% data akan berada di dalam rentang tiga simpangan baku ($\pm 3\sigma$) (Heumann et al., 2023). Praktisi *data science* memanfaatkan aturan empiris ini untuk mendeteksi anomali pada sistem keamanan siber atau transaksi perbankan; setiap data transaksi yang jatuh di luar rentang tiga simpangan baku akan langsung ditandai sebagai aktivitas mencurigakan (*fraud detection*).

Selain distribusi normal, dunia nyata sering kali menyajikan data yang mengikuti pola distribusi non-Gaussian, salah satunya adalah Distribusi Poisson. Distribusi Poisson digunakan untuk memodelkan probabilitas terjadinya sejumlah peristiwa dalam interval waktu atau ruang yang tetap, dengan asumsi peristiwa tersebut terjadi dengan laju rata-rata yang konstan dan saling independen (Bruce et al., 2021). Contoh konkret penerapannya dalam industri digital meliputi prediksi jumlah kunjungan pengguna ke suatu situs web per menit, jumlah panggilan darurat yang diterima pusat kendali per jam, atau jumlah kegagalan server komputasi awan dalam satu hari operasional.

Ada pula Distribusi Binomial yang berfokus pada eksperimen dengan hasil yang bersifat dikotomi atau biner, yaitu hanya ada dua kemungkinan hasil: "sukses" atau "gagal". Menurut Field (2022), distribusi binomial merupakan fondasi teoretis bagi pelaksanaan *A/B Testing*, sebuah eksperimen standar dalam industri digital untuk menguji efektivitas perubahan desain antarmuka aplikasi atau strategi kampanye pemasaran. Melalui pemahaman terhadap parameter probabilitas sukses (p) dan jumlah percobaan (n), tim pengembang produk dapat mengambil keputusan berbasis data apakah fitur baru yang dirilis benar-benar disukai pengguna atau hanya memberikan dampak semu.

D. Pengukuran Variabilitas

Jika statistika deskriptif berfokus pada pencarian titik pusat data, maka pengukuran variabilitas (*measures of dispersion*) bertujuan untuk mengkuantifikasi sejauh mana data tersebut tersebar atau bervariasi dari titik pusatnya. Dua dataset bisa saja memiliki nilai rata-rata yang sama persis,

namun memiliki karakteristik internal yang sepenuhnya berbeda akibat tingkat dispersi data yang kontras. Bruce et al. (2021) memberikan ilustrasi bahwa mengabaikan variabilitas dalam analisis bisnis adalah hal berbahaya; misalnya, laporan rata-rata waktu pengiriman barang sebesar 3 hari tidak bermakna jika variabilitasnya sangat tinggi (rentang pengiriman aktual berkisar antara 1 hingga 30 hari), karena hal itu mencerminkan ketidakstabilan sistem operasional.

Instrumen pengukuran variabilitas yang paling dasar adalah Jangkauan (*Range*), yang dihitung dari selisih antara nilai maksimum dan nilai minimum dalam dataset. Meskipun sangat mudah dipahami, *range* memiliki kelemahan besar karena hanya bergantung pada dua nilai ekstrem dan tidak mencerminkan variasi data yang berada di antaranya. Untuk mengatasi kelemahan tersebut, digunakan Rentang Interkuartil (*Interquartile Range* - IQR), yaitu selisih antara kuartil ketiga (Q_3) dan kuartil pertama (Q_1). IQR mengisolasi 50% data yang berada tepat di tengah distribusi, menjadikannya metrik yang sangat stabil dan kebal terhadap gangguan nilai pencilan ekstrem (Heumann et al., 2023).

Ukuran variabilitas yang paling ilmiah dan banyak digunakan dalam algoritma pembelajaran mesin adalah Ragam (*Variance*) dan Simpangan Baku (*Standard Deviation*). Ragam mengukur rata-rata dari kuadrat selisih setiap titik data terhadap nilai rata-ratanya; kuadrat dilakukan agar nilai selisih negatif tidak saling menghilangkan dengan nilai selisih positif. Namun, karena nilainya berbentuk kuadrat, satuan dari ragam menjadi berbeda dengan satuan asli data. Oleh karena itu, simpangan baku diturunkan dengan cara menarik akar kuadrat dari ragam, mengembalikan metrik dispersi ke dalam satuan dimensi asli yang sama dengan data mentah

(Navidi, 2024). Simpangan baku berfungsi sebagai tolok ukur utama volatilitas dalam analisis risiko keuangan dan kendali mutu industri.

Dalam arsitektur pembelajaran mesin, pemahaman variabilitas memegang peran vital melalui proses standardisasi data (*Z-score normalization*). Banyak algoritma sensitif terhadap perbedaan skala variabilitas antar-variabel, seperti algoritma *K-Means Clustering* atau *Support Vector Machines*. Jika satu variabel memiliki variabilitas bernilai ribuan (misal: pendapatan) dan variabel lain bernilai satuan (misal: jumlah anak), maka variabel berdispersion tinggi akan mendominasi perhitungan jarak geometris secara tidak adil. Dengan membagi selisih data terhadap rata-rata dengan simpangan bakunya, seluruh variabel ditransformasikan ke dalam skala seragam dengan rata-rata nol dan simpangan baku satu, memastikan keadilan kontribusi fitur dalam pemodelan (Field, 2022).

E. Korelasi dan Hubungan Antarvariabel

Analisis hubungan antarvariabel merupakan tahapan esensial dalam *data science* sebelum melangkah pada pembuatan model prediksi bervariat banyak. Eksplorasi ini bertujuan untuk mendeteksi apakah perubahan pada suatu variabel diikuti secara konsisten oleh perubahan pada variabel lainnya. Hubungan ini diukur secara numerik melalui Koefisien Korelasi, sebuah nilai standar yang berkisar antara -1.00 hingga $+1.00$. Nilai $+1.00$ mencerminkan korelasi positif sempurna, nilai 0.00 berarti tidak ada hubungan linier sama sekali, sedangkan nilai -1.00 menunjukkan korelasi negatif sempurna di mana kenaikan satu variabel berbanding terbalik dengan variabel pasangannya (Montgomery & Runger, 2022).

Metode penghitungan korelasi yang paling umum digunakan adalah Koefisien Korelasi Pearson (*Pearson's Product Moment Correlation*). Korelasi Pearson bekerja dengan asumsi bahwa kedua variabel berskala interval atau ratio dan memiliki hubungan yang bersifat linier serta berdistribusi normal. Namun, jika dataset yang dihadapi mengandung hubungan non-linier atau berskala ordinal, penggunaan Pearson akan menghasilkan nilai yang menyesatkan. Untuk mengatasi masalah tersebut, *data scientist* menggunakan Korelasi Peringkat Spearman (*Spearman's Rank Correlation*), yang menghitung keeratan hubungan berdasarkan peringkat data, menjadikannya non-parametrik dan lebih fleksibel terhadap keberadaan pencilan (Field, 2022).

Satu dogma krusial dalam statistika yang wajib diinternalisasi oleh setiap ilmuwan data adalah kalimat: *"Correlation does not imply causation"* (Korelasi tidak sama dengan kausalitas). Adanya hubungan statistik yang kuat antara dua variabel tidak secara otomatis membuktikan bahwa satu variabel menjadi penyebab langsung atas munculnya variabel lain. Menurut Pearl dan Mackenzie (2021), dua variabel yang tampak berkorelasi kuat sering kali dikendalikan oleh variabel ketiga tersembunyi yang tidak ikut diukur, sebuah fenomena yang dikenal sebagai Faktor Pengganggu (*Confounding Variable*). Tanpa desain eksperimen yang ketat, analisis korelasi dapat melahirkan korelasi semu yang menyesatkan strategi bisnis.

Di dalam alur kerja pemodelan prediktif, analisis korelasi digunakan secara intensif untuk mendeteksi masalah Multikolinieritas, yaitu situasi di mana terdapat korelasi yang terlalu kuat di antara sesama variabel independen (fitur prediksi). Multikolinieritas yang tinggi menyebabkan estimasi koefisien dalam model regresi menjadi tidak stabil dan sulit untuk

diinterpretasikan secara ilmiah (Navidi, 2024). Melalui pembuatan matriks korelasi visual (*heatmap*), seorang *data scientist* dapat melakukan seleksi fitur, mengeliminasi variabel yang redundan, dan mempertahankan fitur-fitur yang paling efektif untuk membangun model pembelajaran mesin yang efisien dan berdaya guna tinggi.

F. Studi kasus

Analisis Karakteristik Pola Belanja Pelanggan Minimarket "XYZ"

Latar Belakang Masalah Manajemen Minimarket "XYZ" menghadapi tantangan berupa fluktuasi nilai transaksi bulanan yang tidak menentu dan ketidakpastian pola pembelian barang oleh konsumen. Untuk mengoptimalkan manajemen stok gudang dan merancang program loyalitas pelanggan yang efektif, tim *data science* diminta melakukan analisis statistik deskriptif dan eksplorasi distribusi terhadap data transaksi dari 1.500 pelanggan yang tercatat selama kuartal pertama tahun berjalan.

Langkah Kerja Proyek 1. *Pengumpulan dan Pembersihan Data:* Langkah awal dimulai dengan mengekstrak data transaksi dari basis data POS (*Point of Sales*). Tim mengidentifikasi variabel utama: Kategori Keanggotaan (Nominal: Member/Non-Member), Usia Pelanggan (Ratio), Jumlah Barang yang Dibeli (Ratio), Total Pengeluaran per Transaksi dalam Rupiah (Ratio), dan Tingkat Kepuasan Pelanggan setelah bertransaksi (Ordinal).

2. *Eksplorasi Statistik Deskriptif:* Menggunakan pustaka Pandas pada Python, tim melakukan kalkulasi ringkasan numerik. Ditemukan bahwa nilai rata-rata (*mean*) total pengeluaran per transaksi adalah Rp 250.000,

sedangkan nilai median berada di angka Rp 175.000. Selisih yang signifikan antara *mean* dan median ini mengindikasikan adanya nilai kemencengan (*skewness*) positif yang tinggi pada data pengeluaran.

3. *Analisis Distribusi dan Variabilitas:* Setelah dihitung, nilai simpangan baku (*standard deviation*) pengeluaran mencapai Rp 150.000 dengan nilai maksimum transaksi sebesar Rp 1.200.000. Nilai maksimum ini dikonfirmasi sebagai pencilan (*outlier*) menggunakan metode IQR, mewakili kelompok kecil pelanggan institusional yang melakukan belanja grosir. Ketika visualisasi histogram dijalankan, bentuk distribusi total pengeluaran terbukti mengikuti pola Distribusi non-Gaussian dengan ekor memanjang ke kanan, sementara data usia pelanggan terdistribusi normal di sekitar usia 34 tahun.

4. *Analisis Korelasi:* Tim menghitung hubungan antarvariabel kuantitatif menggunakan metode Pearson untuk hubungan usia dengan total pengeluaran, dan menemukan korelasi positif lemah ($r = 0.21$). Namun, ketika menghitung hubungan antara Jumlah Barang yang Dibeli dengan Total Pengeluaran, ditemukan korelasi positif yang sangat kuat ($r = 0.88$). Sementara itu, hubungan antara Tingkat Kepuasan (Ordinal) dengan Kategori Keanggotaan dianalisis menggunakan pendekatan tabulasi silang (*cross-tabulation*).

Solusi dan Pengambilan Keputusan Berbasis Data Berdasarkan hasil ringkasan deskriptif dan pola korelasi, tim *data science* merekomendasikan keputusan strategis kepada pihak manajemen:

- Karena median pengeluaran berada di angka Rp 175.000 dan mayoritas data berkumpul di area tersebut, manajemen disarankan membuat paket promosi "Bundling Hemat" dengan target harga Rp

200.000 untuk mendorong rata-rata pengeluaran pelanggan kelas menengah ke atas.

- Keberadaan pencilan transaksi bernilai jutaan rupiah tidak dihapus dari dataset, melainkan diisolasi menjadi segmen pasar baru, yaitu segmen "B2B skala kecil", yang penanganannya dipisahkan dari pelanggan ritel biasa agar tidak merusak akurasi model stok barang.
- Program promosi tidak perlu disegmentasikan secara ketat berdasarkan usia pelanggan, mengingat nilai korelasi usia terhadap daya beli sangat lemah ($r = 0.21$). Fokus promosi sebaiknya dialihkan pada kuantitas barang yang dibeli per transaksi melalui program insentif "Beli Banyak Lebih Murah" yang terbukti memiliki korelasi linear sangat kuat terhadap total pendapatan minimarket.

Assets of Takaza INNOVATIX LABS

BAB III
DASAR-DASAR STATISTIK UNTUK DATA SCIENCE

Pemrograman merupakan instrumen teknis utama yang menjembatani antara teori statistik matematis dengan eksekusi komputasional pada volume data berskala besar. Dalam disiplin *data science*, bahasa pemrograman bukan sekadar alat bantu ketik, melainkan medium operasional untuk merepresentasikan logika manusia ke dalam algoritma yang dapat diproses oleh mesin. Bab ini secara komprehensif menguraikan fondasi pemrograman yang esensial bagi seorang praktisi data, yang mencakup pemilihan bahasa pemrograman yang relevan, perancangan arsitektur lingkungan kerja (*environment*), hingga pemahaman struktur sintaksis dan manipulasi tipe data. Melalui penguasaan materi pada bab ini, mahasiswa dan peneliti diharapkan tidak hanya mampu menulis kode yang beroperasi tanpa eror, tetapi juga mampu merancang skrip yang efisien secara memori, tangguh (*robust*) terhadap anomali data, dan mudah direproduksi oleh peneliti lain dalam ekosistem analitika kolaboratif.

A. Bahasa Pemrograman dalam Data Science

Pemrograman dalam konteks *data science* memiliki paradigma yang unik dibandingkan dengan rekayasa perangkat lunak tradisional (*traditional software engineering*). Jika rekayasa perangkat lunak berfokus pada pembangunan sistem dan antarmuka interaktif yang tangguh terhadap perilaku pengguna, maka pemrograman ilmu data berfokus pada eksplorasi,

transformasi, dan ekstraksi nilai informasi dari sekumpulan angka atau teks (VanderPlas, 2023). Hal ini menuntut seorang praktisi data untuk menggunakan bahasa pemrograman yang tidak hanya memiliki kecepatan eksekusi tinggi, tetapi juga didukung oleh ekosistem pustaka (*libraries*) yang kaya akan fungsi-fungsi matematika, statistik, dan visualisasi. Keberhasilan ekstraksi pola tersembunyi dari dalam data sangat ditentukan oleh seberapa efisien bahasa tersebut menangani struktur data multidimensional dan aljabar linear.

Di posisi puncak popularitas global, Python telah memantapkan dirinya sebagai bahasa utama, atau *lingua franca*, bagi komunitas kecerdasan buatan dan *data science*. Dominasi Python tidak lahir dari kecepatan komputasi mentahnya mengingat ia adalah bahasa pemrograman yang diinterpretasi (*interpreted language*) melainkan dari filosofi desainnya yang menekankan keterbacaan kode (*readability*) dan kesederhanaan sintaksis. McKinney (2022) menegaskan bahwa kekuatan sesungguhnya dari Python terletak pada ekosistem pustaka pihak ketiga seperti Pandas, NumPy, dan Scikit-Learn, yang sesungguhnya ditulis menggunakan bahasa C dan C++ tingkat rendah (*low-level*). Arsitektur ini memungkinkan Python berperan sebagai "bahasa lem" (*glue language*), memberikan antarmuka yang sangat ramah bagi ilmuwan data sambil tetap menjaga performa eksekusi komputasi aljabar di belakang layar.

Berada di posisi kedua namun sangat dominan di kalangan akademisi dan ahli biostatistik, bahasa R menawarkan pendekatan yang secara khusus dikembangkan oleh dan untuk para statistikawan. Berbeda dengan Python yang merupakan bahasa tujuan umum (*general-purpose*), R dirancang dengan paradigma komputasi statistik sebagai fitur bawaannya (*built-in*).

Wickham et al. (2023) menjelaskan bahwa keunggulan absolut bahasa R terletak pada ekosistem *Tidyverse*, sebuah koleksi paket yang menyediakan tata bahasa (*grammar*) yang konsisten untuk manipulasi data dan visualisasi tingkat lanjut (seperti ggplot2). Bagi proyek-proyek riset yang berpusat pada uji hipotesis yang ketat, inferensi ekonometrika, dan visualisasi berkualitas publikasi jurnal, bahasa R sering kali menjadi pilihan logis yang melampaui kapabilitas dasar Python.

Selain bahasa analitik murni, *Structured Query Language* (SQL) menduduki posisi fundamental dan tidak tergantikan dalam tahapan akuisisi data. Sebagian besar data transaksional di dunia industri tersimpan dalam Relational Database Management Systems (RDBMS) seperti PostgreSQL, MySQL, atau Oracle. SQL merupakan bahasa deklaratif; alih-alih menginstruksikan mesin *bagaimana* cara melakukan sesuatu secara spesifik melalui perulangan (*loops*), pengembang cukup mendeskripsikan *apa* hasil akhir yang diinginkan, dan mesin basis data akan mencari rute eksekusi yang paling optimal (DeBarros, 2022). Penguasaan fungsi-fungsi SQL tingkat lanjut, seperti *Window Functions* dan *Common Table Expressions* (CTE), mutlak dikuasai oleh seorang *data scientist* untuk memproses dan menyaring miliaran baris data langsung di dalam peladen basis data sebelum data tersebut ditarik ke lingkungan analitik lokal.

Seiring dengan meledaknya volume data dalam hitungan terabita hingga petabita, bahasa-bahasa fungsional seperti Scala dan bahasa komputasi ilmiah modern seperti Julia mulai merebut perhatian industri. Scala dirancang untuk berjalan di atas *Java Virtual Machine* (JVM) dan merupakan bahasa pemrograman utama yang mendukung ekosistem komputasi terdistribusi Apache Spark. Di sisi lain, bahasa Julia

dikembangkan di Massachusetts Institute of Technology (MIT) dengan tujuan menggabungkan kemudahan sintaksis Python dengan kecepatan eksekusi setara bahasa C (Roesch et al., 2023). Julia memecahkan fenomena yang dikenal sebagai "masalah dua bahasa" (*two-language problem*) di mana ilmuwan harus menulis prototipe di Python lalu merekayasa ulangnya ke dalam bahasa C++ untuk disebarkan pada fase produksi.

Perkembangan mutakhir dalam era digital menuntut organisasi untuk tidak fanatik terhadap satu bahasa pemrograman tunggal. Praktik terbaik saat ini mengarah pada paradigma *polyglot programming*, di mana berbagai bahasa digunakan secara bergantian dalam satu jalur pipa (*pipeline*) proyek yang sama sesuai dengan kekuatannya masing-masing. Misalnya, insinyur data dapat menggunakan Scala dan SQL untuk proses ETL (*Extract, Transform, Load*) berskala masif, ilmuwan data menggunakan Python untuk melatih model pembelajaran mesin, dan analis bisnis menggunakan R untuk merancang dasbor visualisasi interaktif eksekutif (Geron, 2023). Pendekatan modular ini mengoptimalkan efisiensi komputasi dan meminimalisasi waktu pengembangan (*time-to-market*).

Untuk mempertegas perbedaan fungsional dan operasional dari berbagai bahasa tersebut, Tabel 3.1 berikut menguraikan perbandingan karakteristik dan skenario penggunaan optimal dari empat bahasa pemrograman paling esensial dalam disiplin ilmu *data science*.

Assets of Talaza INNOVATIX LABS

Tabel 3.1: Perbandingan Karakteristik Bahasa Pemrograman Utama dalam Data Science

Parameter / Karakteristik	Python	R	SQL	Julia
Paradigma Utama	Berorientasi Objek & Fungsional (*General-purpose*)	Komputasi Statistik & Fungsional	Deklaratif (Relational Data)	Komputasi Kinerja Tinggi (*High Performance*)
Keunggulan Absolut	Ekosistem *Machine Learning* dan *Deep Learning* terlengkap.	Visualisasi data (*ggplot2*) dan model inferensi statistik.	Ekstraksi dan penggabungan (*join*) data di dalam server secara kilat.	Kecepatan eksekusi (*compiled*) tanpa mengorbankan keterbacaan kode.
Kurva Pembelajaran	Sangat landai (mudah dipelajari pemula).	Sedang hingga curam (khususnya bagi non-statistikawan).	Sangat mudah untuk level dasar, curam di level arsitektur.	Sedang (konsep tipe data komposit yang ketat).
Skenario Penggunaan	Pembuatan model prediktif,	Riset biomedis, bio-	Penarikan (*querying*) dataset	Simulasi fisika kuantum,

	pemrosesan bahasa alami (NLP), otomatisasi.	informatika, dan peramalan ekonometrika.	besar dari *Data Warehouse*.	pemodelan iklim makro berskala eksabita.
Kelemahan Utama	Performa bawaan lambat dan *Global Interpreter Lock* (GIL).	Manajemen memori buruk saat memproses data lebih besar dari RAM.	Tidak dirancang untuk algoritma *machine learning* kompleks	

B. Instalasi dan Konfigurasi Lingkungan Kerja

Lingkungan kerja (*environment*) merupakan ekosistem perangkat lunak fondasional tempat kode pemrograman ditulis, diuji, dan dieksekusi. Tanpa konfigurasi lingkungan kerja yang terstandarisasi, sebuah proyek *data science* rentan mengalami sindrom "kode ini berjalan di komputer saya, namun gagal di komputer Anda" (*it works on my machine syndrome*). Hal ini umumnya disebabkan oleh ketidakcocokan versi paket pihak ketiga atau perbedaan versi bahasa pemrograman inti yang diinstal pada sistem operasi pengguna yang berbeda (Severance, 2024). Oleh karena itu, langkah pertama yang mutlak dilakukan sebelum menulis satu baris kode pun adalah merancang dan mengisolasi lingkungan kerja agar seluruh dependensi proyek bersifat reprodusibel (*reproducible*) dan portabel.

Dalam ekosistem lokal (di dalam komputer pribadi atau *workstation*), manajer paket (*package manager*) dan distributor lingkungan seperti Anaconda memegang peranan krusial. Anaconda dirancang khusus untuk memenuhi kebutuhan komputasi ilmiah dengan menyertakan lebih dari 250

paket dasar secara otomatis. Manajer dependensi bawaannya, yaitu conda, secara dinamis memecahkan konflik antar-pustaka kompleks yang sering kali luput jika hanya menggunakan penginstal standar pip. Melalui pendekatan ini, ilmuwan data dapat dengan aman melakukan instalasi alat-alat analitik berat seperti TensorFlow atau PyTorch tanpa khawatir akan merusak stabilitas sistem operasi utama (VanderPlas, 2023). Pendekatan isolasi virtual ini merupakan standar keamanan esensial bagi setiap pengembang profesional.

Instrumen utama bagi ilmuwan data untuk berinteraksi dengan baris kode adalah *Integrated Development Environment* (IDE). Di kalangan akademisi dan peneliti eksperimental, Jupyter Notebook merupakan standar emas emas (*gold standard*). Jupyter menawarkan paradigma *Literate Programming*, di mana blok eksekusi kode dapat diselingi secara interaktif dengan teks naratif berbasis *Markdown*, persamaan matematika LaTeX, dan grafik visual dalam satu antarmuka peramban (*web browser*). Format ini sangat cocok untuk eksplorasi data secara iteratif, di mana pengguna dapat menjalankan, memeriksa, dan memodifikasi satu potongan kecil algoritma tanpa harus mengeksekusi ulang seluruh skrip dari awal (McKinney, 2022). Sebaliknya, bagi proyek yang berorientasi produksi dan rekayasa perangkat lunak berskala besar, IDE monolitik seperti Visual Studio Code (VS Code) dengan integrasi ekstensi *debugging* menjadi opsi yang lebih rasional.

Di era transisi menuju komputasi awan (*cloud computing*), instalasi lokal sering kali mulai ditinggalkan karena keterbatasan perangkat keras, khususnya ketiadaan *Graphics Processing Unit* (GPU) yang mahal. Lingkungan *notebook* berbasis awan seperti Google Colaboratory (Colab) atau Kaggle Notebooks telah mendemokratisasi akses analitik dengan

menyediakan lingkungan yang telah diprakonfigurasi dan memberikan akses GPU secara gratis. Pengguna hanya memerlukan peramban dan koneksi internet untuk melatih model jaringan saraf tiruan kompleks yang biasanya akan memakan waktu berminggu-minggu jika dieksekusi pada komputer portabel standar (Geron, 2023). Lingkungan awan ini menghilangkan gesekan teknis pada fase instalasi awal, memungkinkan para mahasiswa dan praktisi untuk langsung fokus pada logika matematika algoritma pemecahan masalah.

Integrasi lingkungan kerja tidak akan sempurna tanpa penerapan *Version Control System* (VCS), dengan Git sebagai perangkat lunak paling fundamental. Dalam proyek *data science* kolaboratif, sangat lazim bagi banyak analis untuk memodifikasi satu berkas kode yang sama secara bersamaan. Git mendokumentasikan setiap perubahan skrip secara granular (baris per baris), mengizinkan percabangan eksperimen (*branching*), dan memfasilitasi penggabungan kode (*merging*) ke dalam repositori utama, seperti GitHub atau GitLab, tanpa risiko tertimpanya (*overwriting*) hasil kerja kolega (Wickham et al., 2023). Penggunaan Git memisahkan praktik coba-coba yang amatir dari rekayasa *data science* berstandar industri.

Tingkatan paling lanjut dari konfigurasi lingkungan kerja melibatkan teknologi kontainerisasi (*containerization*) seperti Docker. Berbeda dengan lingkungan virtual standar yang hanya mengisolasi pustaka Python, Docker mengisolasi dan merekam konfigurasi pada tingkat sistem operasi. Gambar Docker (*Docker Image*) membungkus kode program beserta seluruh versi bahasa pemrograman, pustaka pihak ketiga, dan pengaturan sistem Linux yang diperlukan ke dalam satu unit portabel (DeBarros, 2022). Kontainerisasi ini memastikan bahwa model pembelajaran mesin yang

dibangun di laboratorium eksperimental riset akan berjalan dengan presisi dan kompatibilitas 100% saat dikerahkan ke dalam lingkungan produksi (server komputasi perusahaan atau peladen aplikasi peladen awan).

C. Struktur Dasar Program

Pemahaman terhadap struktur dasar sebuah program merupakan jembatan kognitif untuk menerjemahkan logika intuitif manusia ke dalam instruksi sekuensial yang dapat dicerna oleh unit pemrosesan sentral (CPU). Setiap baris kode yang ditulis sejatinya dikendalikan oleh seperangkat aturan sintaksis (tata bahasa) dan semantik (makna operasional) dari bahasa pemrograman tersebut. Dalam *data science*, meskipun penulisan kode sering kali bersifat skriptual dan sangat cepat, mengabaikan prinsip-struktur dasar akan berujung pada terciptanya "kode spageti" (*spaghetti code*) sebuah skrip yang tidak terstruktur, sulit dibaca, dan hampir mustahil untuk dipelihara (Severance, 2024). Struktur pemrograman yang baik mensyaratkan pembagian alur komputasi menjadi blok-blok instruksi yang koheren.

Paradigma pemrograman yang paling umum digunakan dalam fase pra-pemrosesan data adalah Pemrograman Prosedural (*Procedural Programming*). Paradigma ini memandang program sebagai urutan langkah operasional linier, dieksekusi baris demi baris dari atas ke bawah. Namun, seiring dengan meningkatnya kompleksitas model, ilmuwan data dituntut untuk mengadopsi elemen Pemrograman Berorientasi Objek (*Object-Oriented Programming* / OOP). Dalam OOP, data dan fungsi pembentuknya digabungkan ke dalam satu entitas yang disebut "Objek". Sebagai contoh, saat seorang ilmuwan data memanggil modul pada Scikit-Learn, ia sesungguhnya sedang melakukan instansiasi sebuah objek *machine learning*

(misalnya RandomForestClassifier) yang memiliki metode-metode internal seperti .fit() untuk proses pelatihan dan .predict() untuk menghasilkan luaran (Geron, 2023).

Komponen fundamental yang mengarahkan navigasi kode adalah struktur Kendali Aliran (*Control Flow*), khususnya pernyataan bersyarat seperti *if-elif-else*. Fungsi ini memberikan kapabilitas kognitif bagi program untuk mengambil cabang keputusan adaptif berdasarkan kondisi data (*state of the data*) secara *real-time*. Dalam pembersihan data, logika bersyarat diterapkan secara masif; misalnya, program diinstruksikan: *jika* nilai usia pelanggan lebih kecil dari nol atau bernilai kosong (*NaN*), *maka* ganti nilai tersebut dengan rata-rata populasi usia, *jika tidak*, biarkan data sebagaimana adanya (McKinney, 2022). Fleksibilitas navigasi ini memastikan bahwa pengecualian analitik dapat diselesaikan secara terprogram tanpa intervensi manual dari pengguna pada setiap baris data.

Selain pengambilan keputusan, kemampuan program mengotomatisasi pekerjaan yang membosankan terletak pada struktur Iterasi atau Perulangan (*Loops*), seperti for dan while. Iterasi memungkinkan program untuk menelusuri jutaan observasi data, mengekstrak baris, atau memproses sekumpulan fail (*files*) di dalam sebuah direktori secara berurutan. Meskipun secara konsep sangat intuitif, perulangan manual konvensional di dalam Python (seperti for loop) sangat dihindari ketika berhadapan dengan data berskala puluhan gigabita karena biaya komputasionalnya (*time complexity*) sangat lambat (VanderPlas, 2023). Praktik modern pada *data science* memandatkan penggantian iterasi konvensional dengan teknik Vektorisasi (*Vectorization*), di mana proses operasi matematis dilemparkan langsung ke level kompilator C agar dijalankan secara simultan pada matriks data.

Assets of Takaza INNOVATIX LABS

Ketika serangkaian logika bersyarat dan iterasi mulai digunakan berulang-ulang, struktur program yang ideal harus dienkapsulasi menjadi sebuah Fungsi Khusus (*Functions*). Pembuatan fungsi mengikuti filosofi desain perangkat lunak modern: *Don't Repeat Yourself* (DRY). Jika seorang analis menemukan dirinya menyalin dan menempel (*copy-pasting*) tiga baris kode transformasi data yang sama di berbagai bagian buku catatan proyeknya, maka blok kode tersebut harus dibungkus ke dalam satu fungsi terdefinisi (menggunakan perintah def di Python atau function di R). Pendekatan fungsi termodular ini mempersempit peluang terjadinya kesalahan ketik, menyederhanakan pelacakan kutu (*debugging*), dan mempermudah proses pembaharuan algoritma logika tanpa harus mencari lokasi salinan kode yang tersebar (Severance, 2024).

Tidak ada perangkat lunak yang sempurna, dan data mentah dari dunia nyata tidak pernah hadir dalam kondisi higienis. Oleh karena itu, struktur program ilmu data harus dilengkapi dengan mekanisme penanganan pengecualian atau toleransi kesalahan (*Exception Handling*), seperti blok try-except. Blok pelindung ini mencegah berhentinya sistem secara paksa (*program crash*) saat aplikasi mencoba memproses anomali data ekstrem (misalnya, pembagian dengan angka nol atau upaya mengubah teks acak menjadi numerik). Dengan struktur penanganan galat yang tepat, skrip tetap berjalan memproses sisa data, sembari sekadar mencetak peringatan sistematis pada *log file* bahwa suatu baris data tertentu tidak dapat dieksekusi (DeBarros, 2022).

Sebagai sentuhan akhir pada arsitektur struktur program, pengembang diwajibkan menyertakan dokumentasi leksikal yang memadai. Penulisan komentar sebaris (*inline comments*) dan blok referensi tebal (*docstrings*)

berfungsi sebagai penjelasan naratif meta-kognitif yang mengungkap "mengapa" (*why*) suatu algoritma diterapkan, alih-alih hanya menjelaskan "apa" (*what*) yang dikerjakan kode tersebut. Wickham et al. (2023) menyebutkan bahwa audiens utama dari sebuah kode bukanlah mesin kompilator, melainkan rekan sesama peneliti atau bahkan diri sang pemrogram sendiri enam bulan di masa depan. Kerapian indentasi dan kepatuhan terhadap pedoman gaya penulisan kode, seperti PEP 8 di Python, mencerminkan kedalaman profesionalisme seorang ilmuwan data.

D. Variabel dan Tipe Data

Inti dari semua komputasi digital adalah tentang menyimpan nilai di dalam memori dan memanipulasi nilai tersebut sesuai kebutuhan logika algoritma. Dalam pemrograman, "Variabel" berfungsi sebagai kontainer atau rujukan simbolis (*pointers*) yang menunjuk pada alamat spesifik di *Random Access Memory* (RAM) tempat data sesungguhnya berada. Penamaan variabel dalam proyek ilmu data bukan sekadar persoalan selera estetika, melainkan wajib mencerminkan makna empiris dari objek yang disimpannya (VanderPlas, 2023). Sebuah nama variabel seperti x1 adalah representasi nama yang buruk di ranah industri; ia harus diganti secara eksplisit menjadi customer_age atau transaction_amount agar skrip dapat dipahami tanpa ambiguitas oleh kolega lintas divisi.

Setiap variabel diikat oleh karakteristik fundamental yang disebut Tipe Data Primitif (*Primitive Data Types*). Tipe data kuantitatif dibedakan antara *Integer* untuk mendeskripsikan angka bulat (misalnya, jumlah anak, jumlah kunjungan halaman web) dan *Float* untuk menampung angka desimal atau pecahan dengan tingkat presisi berkelanjutan (misalnya, nilai

tukar mata uang, bobot algoritma saraf tiruan). Perbedaan tipe ini sangat krusial karena mesin komputasi mengalokasikan ruang memori yang berbeda dan cara kalkulasi biner yang terpisah untuk keduanya. Pemaksaan operasi memori yang keliru akan menyebabkan masalah *floating-point precision*, di mana kalkulasi desimal dalam jumlah besar perlahan-lahan menghasilkan margin eror atau bias kecil yang merusak hasil akhir peramalan keuangan (Severance, 2024).

Untuk merepresentasikan kelompok data kualitatif dan literal teks, pemrograman menyediakan tipe data String (*Text/String*). Pada lingkungan *big data*, pengelolaan string melibatkan kompleksitas yang signifikan akibat keragaman standar pengkodean karakter. Ilmuwan data modern secara mutlak bertumpu pada standar pengkodean Universal Character Set Transformation Format atau UTF-8 guna menangani alfabet di luar bahasa Inggris standar, memproses karakter emoji dari media sosial, serta aksara khusus di Asia dan Timur Tengah. Manipulasi dan dekode (*decoding*) tipe data string yang gagal akan melahirkan teks berupa sekumpulan simbol acak yang tidak bermakna (sering disebut *mojibake*), yang secara langsung akan menggagalkan algoritma Pemrosesan Bahasa Alami (NLP) dalam memahami konteks sentimen pelanggan (McKinney, 2022).

Sebagai instrumen pengendali alur logika kondisional, terdapat tipe data Boolean yang semata-mata menampung dua nilai mutlak: *True* (1) atau *False* (0). Dalam *data science*, operasi boolean tidak sekadar digunakan untuk mengevaluasi satu nilai, melainkan dimanfaatkan secara serentak sebagai "Topeng Pemfilteran" (*Boolean Masking*) pada jutaan deret data. Sebagai ilustrasi, ketika kita meminta program mencari semua pelanggan yang berusia di atas 30 tahun, kompilator secara harfiah akan menyusun satu

larik array baru berisi urutan jutaan nilai *True* dan *False*. Array boolean ini kemudian "ditumpangkan" ke atas dataset asli untuk mengekstrak hanya observasi yang bernilai *True* (Geron, 2023). Konsep *masking* ini adalah kunci kecepatan operasional algoritma penyaringan komputasi tinggi.

Lebih dari sekadar tipe data tunggal, kompleksitas realitas sosial dan empiris memaksa pemrograman merangkai nilai-nilai dasar ke dalam Struktur Data Majemuk atau Komposit. Bahasa Python menawarkan tiga struktur natif utama: *List*, *Tuple*, dan *Dictionary*. *List* adalah urutan linear yang sangat fleksibel (dapat diubah/ditambah nilainya atau *mutable*), cocok untuk menampung runtutan temporal data deret waktu sementara. Sebaliknya, *Tuple* merupakan urutan data yang tidak bisa dimodifikasi (*immutable*) semenjak ia dideklarasikan, sangat relevan untuk menjaga keamanan titik-titik koordinat spasial yang tidak boleh sengaja diubah oleh kesalahan baris kode yang lain. *Dictionary* (atau Hash Map) merepresentasikan data dalam format Pasangan Kunci-Nilai (*Key-Value Pairs*), yang merupakan tulang punggung arsitektur struktur penyimpanan objek JSON pada komunikasi antarmuka web modern (Wickham et al., 2023).

Namun, format primitif ini terbukti tidak efisien untuk kalkulasi aljabar linear dan kalkulus matriks, yang mendasari kecerdasan buatan. Oleh sebab itu, ekosistem ilmu data mengintroduksikan tipe data tingkat tinggi (*advanced data structures*) seperti *NumPy Arrays* (larik multidimensional bertenaga C) dan *Pandas DataFrames* (Struktur Data Tabular berindeks). *DataFrame* secara revolusioner menduplikasi karakteristik visual lembar kerja peranti lunak tabel (seperti Microsoft Excel) ke dalam bahasa pemrograman. Melalui tipe data DataFrame ini, seorang ilmuwan data dapat

melihat representasi matriks kolom (fitur/variabel) dan baris (observasi), lengkap dengan label teks dan nilai kategorial dalam satu entitas objek (McKinney, 2022). Kesuksesan awal proyek sangat ditentukan oleh kelancaran mengimpor berkas sumber mentah (.csv, .xlsx, atau basis data relasional) ke dalam bentuk *DataFrame*.

Pilar pemahaman terakhir mengenai tipe data bermuara pada Konversi Tipe Data (*Type Casting*) dan Manajemen Memori (*Memory Optimization*). Data dunia maya sering kali dikumpulkan secara tidak sempurna; tidak jarang kolom "Harga" terdeteksi oleh peranti lunak sebagai rentetan "Teks" karena adanya simbol mata uang atau tanda baca koma. Memaksa menghitung rata-rata dari "teks" akan menimbulkan eror fatal. Ilmuwan data harus melakukan *type casting* secara manual (menggunakan metode .astype() dalam Python) untuk membersihkannya menjadi *Float*. Lebih lanjut, proses mengubah tipe data skala besar (seperti mengubah *float64* yang berat menjadi tipe data khusus category untuk memori minimal) dapat menghemat pemakaian alokasi RAM secara ekstrem hingga 70-80%, menjaga agar mesin awan beranggaran rendah tidak mengalami kegagalan operasional tipe *Out of Memory* (OOM) (Roesch et al., 2023).

E. Operasi Data Dasar

Operasi data dasar merupakan jembatan fungsional yang mengubah himpunan angka atau label statis menjadi wawasan dinamis (*insights*). Berbeda dengan operasi di ilmu komputer klasik yang mengeksekusi instruksi pada satu variabel tunggal per siklus CPU (operasi sakelar/skalar), bahasa pemrograman *data science* memprioritaskan kalkulasi pada ratusan ribu elemen data sekaligus (operasi berbasis himpunan). Paradigma operasi

ini meminimalisasi kompleksitas dan sintaksis yang tebal; dengan satu ekspresi aritmatika tunggal, sebuah algoritma dapat menstandarisasi rentang jutaan nilai piksel pada sebuah gambar medis (VanderPlas, 2023). Penguasaan sintaks operasi operasional ini adalah kriteria mendasar kompetensi teknis dari pengembang data amatir menuju analis tingkat profesional menengah.

Di tingkat fundamental, Operasi Aritmetika mencakup pengintegrasian simbol tambah, kurang, kali, bagi, dan eksponensiasi untuk menciptakan metrik fitur yang baru (*feature engineering*). Salah satu konsep terpenting dalam operasi aritmatika pada tingkat larik array adalah Pemancaran (*Broadcasting*). Dalam Python dan R, jika Anda menambahkan sebuah skalar berukuran tunggal (misalnya angka 10) pada sebuah larik matriks yang memuat ribuan data harga, mesin bahasa tidak akan menghasilkan eror akibat perbedaan bentuk geometri, melainkan secara cerdas memancarkan (*broadcast*) dan menduplikasi angka 10 tersebut ke semua baris secara virtual untuk dijumlahkan (McKinney, 2022). Mekanisme tersembunyi (*under-the-hood*) ini membuat skrip aritmetika menjadi padat, cepat, dan terbebas dari jeratan iterasi yang memakan memori komputasi secara sia-sia.

Kelompok kedua merujuk pada Operasi Relasional dan Logika. Operasi ini digunakan untuk mengukur dan membandingkan relasi hierarki antardata, menggunakan operator lebih besar dari ($>$), kurang dari ($<$), maupun setara secara persis ($==$). Seperti yang telah dibahas sebelumnya terkait sifat data *boolean mask*, kombinasi gerbang logika AND (&), OR (|), dan NOT (~) diaplikasikan sebagai penyaring instan. Misalnya, operasi logika ini adalah satu-satunya metode praktis untuk mengisolasi sub-

himpunan demografi target: "*Saring dan tampilkan seluruh data di mana (Gaji > 10.000.000) DAN (Status == 'Menikah')*". Evaluasi relasional ini memangkas derau statistik awal dan memfokuskan sumber daya komputasi secara presisi hanya pada titik data populasi penelitian yang dituju (Geron, 2023).

Sementara itu, untuk mengelola kekacauan tekstual, Operasi Manipulasi String menjadi perisai utama. Mengingat lebih dari 80% data korporat tergolong sebagai teks tak terstruktur, ilmuwan data perlu secara sistematis memilah huruf kapital, menghilangkan karakter spasi kosong, dan memisahkan frasa kalimat menjadi token kata individual. Instrumen operasi tercanggih untuk teks adalah *Regular Expressions* (Regex). Regex adalah sub-bahasa sintaksis operasional independen yang digunakan untuk menemukan pola (*pattern matching*) di lautan karakter (Severance, 2024). Melalui ekspresi ringkas seperti ^\d{4}-\d{2}-\d{2}$, analis mampu mengekstrak format tanggal universal dari dalam ribuan paragraf artikel berita tanpa perlu membaca manual satu per satu, mengotomatisasi pembersihan yang mustahil dilakukan oleh tenaga manusia.

Pengambilan rujukan data dilakukan lewat Operasi Pengindeksan dan Pemotongan (*Indexing and Slicing*). Analogi dari proses ini adalah ibarat memotong koordinat spesifik dari sebuah kue kubus raksasa. Pengindeksan dalam kerangka data modern (*DataFrames*) dibedakan secara tegas antara posisi numerik implisit (*integer-based indexing*, yakni urutan baris ke-1, ke-2, dst) dan posisi berlabel eksplisit (*label-based indexing*, pemanggilan berbasis nama kunci atau waktu indeks) (Wickham et al., 2023). Fungsi .loc[] dan .iloc[] pada pustaka Pandas memfasilitasi pengambilan spesifik atas titik koordinat nilai, memotong rentang observasi satu minggu ke

belakang, hingga menarik fitur kolom yang letaknya tidak berurutan, tanpa harus mengutak-atik struktur file sistem basis data yang mentah secara permanen.

Pada tahap yang merangkum hasil agregat, arsitektur dasar bertumpu penuh pada Operasi Pengelompokan Berbasis Agregasi (*Split-Apply-Combine*), yang biasanya ditangani oleh klausa operasional .groupby(). Konsep ini diadaptasi dari SQL klasik, di mana sekumpulan himpunan data raya "dipecah" (*split*) ke dalam keranjang atau ember kategorikal (misalnya berdasarkan nama Provinsi). Kemudian, bahasa pemrograman "menerapkan" (*apply*) fungsi operasi aritmetika deskriptif seperti menghitung nilai total, rata-rata deviasi, atau minimum pada keranjang setiap provinsi. Hasil perhitungannya "dikombinasikan" (*combine*) kembali ke dalam sebuah format ringkasan laporan utuh (McKinney, 2022). Kapabilitas semantik dari .groupby() sejatinya merangkum esensi dari ilmu statistik analisis multivarian.

Terakhir, efisiensi penulisan seluruh operasi ini disempurnakan melalui pendekatan Pelekatan Metode Rantai Kode (*Method Chaining*). Alih-alih menulis ulang nama variabel di setiap baris dan menyimpan langkah intermediasi memori secara bertahap, fungsi operasi berantai dapat ditulis layaknya susunan blok LEGO, mengalir mulus dengan memanggil satu metode setelah metode lainnya secara instan. Teknik gaya operasional perangkaian ini menghapus keberadaan puluhan variabel sementara di memori lokal, menghasilkan skrip transformasi data pipa (*pipeline ETL*) yang sangat anggun secara estetik dan luar biasa efisien secara eksekusi algoritma alokasi memori (DeBarros, 2022).

Assets of Takaza INNOVATIX LABS

F. Studi kasus

Inisiasi Pipa Analisis Data Prediksi "Churn" Pegawai dengan Python

Latar Belakang Masalah Departemen Sumber Daya Manusia (HRD) Perusahaan Teknologi Inovatif (PTI) sedang menghadapi lonjakan tingkat pengunduran diri karyawan yang tidak lazim dalam dua kuartal terakhir (*employee churn*). Sebagai konsultan ilmu data yang baru direkrut, Anda diminta untuk menulis purwarupa skrip pemrograman awal dalam bahasa Python yang berfungsi untuk membaca data mentah karyawan, membersihkan format yang anomali, memberikan tipe data yang benar, serta melakukan ekstraksi fitur (*feature engineering*) sederhana sebelum masuk ke tahap eksplorasi statistik lanjut.

Langkah Kerja Proyek Terprogram 1. *Setup dan Instalasi Lingkungan Kerja:* Untuk memastikan skrip tidak mengintervensi sistem kerja perusahaan, Anda terlebih dahulu membuat lingkungan virtual kondensator lokal dengan menjalankan perintah conda create -n hr_churn python=3.10 di terminal konsol, kemudian mengaktifkannya. Anda memasang perpustakaan ekosistem primer, yakni pandas dan numpy. Eksplorasi koding ini lantas dituangkan ke dalam IDE interaktif Visual Studio Code (VS Code) yang disambungkan pada ekstensi Jupyter Notebook agar alur pemikiran logika dapat dinarasikan baris demi baris menggunakan anotasi teks Markdown. 2. *Impor Pustaka dan Pembacaan Data:* Skrip Anda memanggil dependensi pustaka dasar di blok awal (import pandas as pd, import numpy as np). Data sekunder HRD tersimpan di dalam berkas berformat *.csv* berisi rincian seperti "ID Pegawai", "Masa Kerja", "Gaji Terakhir", dan label teks "Resign/Stay". Melalui operasi masukan

pd.read_csv("hr_data.csv"), teks dipindahkan ke dalam struktur data tingkat lanjut berupa *DataFrame*. Saat diinspeksi lewat perintah dasar .info(), terlihat kerancuan tipe data: kolom "Gaji Terakhir" terbaca sebagai objek *String* alih-alih data kuantitatif (*Float*), karena terdapat ketikan simbol mata uang 'Rp' dan tanda titik pemisah ribuan. 3. *Operasi Pembersihan dan Konversi (Type Casting):* Melalui operasi manipulasi *string* berbasis metode berantai, variabel gaji tersebut dibersihkan secara matematis. Anda menuliskan fungsi operasional: df['Gaji Terakhir'] = df['Gaji Terakhir'].str.replace('Rp', '').str.replace('.', '').astype(float). Dengan dua metode *replace* dan satu metode *type casting*, semua simbol valuta hilang dengan rapi, dan kolom tersebut sukses bertransformasi menjadi besaran variabel rasio rasional (*float64*) yang siap dilemparkan ke dalam mesin agregasi perhitungan aljabar. 4. *Manipulasi Logika dan Pengindeksan Masking Boolean:* Anda diberikan informasi domain bisnis bahwa pengunduran diri hanya signifikan terjadi pada karyawan dengan pengalaman masa kerja di bawah 5 tahun. Oleh karenanya, Anda menerapkan operasi pemfilteran (*Boolean masking*): df_junior = df[df['Masa Kerja'] < 5]. Variabel struktur penyangga yang baru tercipta ini mengindeks dan mengisolasi khusus memori segmen profil junior tersebut secara spasial tanpa harus merusak struktur penyimpanan data primer untuk karyawan senior perusahaan. 5. *Agregasi GroupBy:* Untuk mengekstrak konklusi dari baris per baris data, Anda menjalankan operasi pengelompokan .groupby('Departemen')['Gaji Terakhir'].mean(). Secara instan, terminal konsol melontarkan keluaran ringkas yang mengungkapkan fakta nyata lapangan bahwa departemen Riset & Pengembangan (R&D) mendominasi nilai rata-rata gaji terendah dibanding departemen pemasaran, memberikan

Assets of Takaza INNOVATIX LABS

indikasi awal kepada dewan direktur bahwa keluhan perihal kompensasi mungkin menjadi katalis utama tingginya tingkat perputaran tenaga kerja di klaster teknologi.

Solusi dan Pengambilan Keputusan Berbasis Data Pemrograman dalam ilmu data bukan sekadar menulis deretan sintaks mesin; dari baris kode yang singkat di atas, departemen HRD telah mampu mengkonfirmasi anomali struktur gaji karyawan tanpa harus repot menyaring kolom lembar kerja baris per baris secara manual. Berbekal perombakan tipe variabel string ke dalam bentuk numerik yang tepat, jalur pipa ini kini telah bersih dan kompatibel untuk disuapkan ke dalam kerangka pengembangan algoritma kecerdasan buatan (*machine learning classifier*) guna memprediksi identitas nama-nama karyawan krusial yang berisiko tinggi untuk mengajukan pengunduran diri pada bulan yang akan datang.

Assets of Takaza INNOVATIX LABS

BAB IV
AKUISISI DAN PENGUMPULAN DATA

Setelah memahami infrastruktur pemrograman dan fondasi statistik, tahapan operasional pertama dalam siklus hidup proyek *data science* adalah akuisisi dan pengumpulan data. Dalam komunitas akademik dan industri analitik, terdapat sebuah aksioma fundamental yang berbunyi *"garbage in, garbage out"* (sampah yang masuk, sampah pula yang keluar). Sebegitu canggihnya arsitektur algoritma pembelajaran mesin atau pemodelan statistik yang dibangun, kelayakan dan akurasi model tersebut mutlak ditentukan oleh kualitas dan kuantitas data mentah yang diakuisisi pada tahap awal. Bab ini akan membedah secara komprehensif taksonomi sumber data, mekanisme ekstraksi informasi melalui antarmuka pemrograman (API) dan teknik *web scraping*, pemanfaatan repositori data terbuka (*open source*), hingga protokol validasi ketat yang harus diterapkan saat data pertama kali memasuki jalur pipa komputasi.

A. Sumber Data Internal dan Eksternal

Di dalam ekosistem bisnis dan penelitian modern, data tidak lagi dipandang sebagai entitas monolitik, melainkan sebuah aset hibrida yang harus dikonstruksikan dari berbagai titik awal. Pengenalan terhadap lanskap sumber data dimulai dengan mengklasifikasikannya ke dalam dua domain spasial utama: sumber data internal dan eksternal. Pemisahan ini bukan sekadar persoalan lokasi fisik peladen penyimpanannya, melainkan berkaitan

erat dengan hak kepemilikan, tingkat kendali atas format data, serta relevansi kontekstualnya terhadap masalah yang sedang dipecahkan. Seorang praktisi data dituntut untuk memiliki kemampuan memetakan kedua ranah ini secara presisi guna membangun kerangka analitik yang holistik dan terbebas dari bias pandangan tunggal (*myopic bias*).

Data internal merujuk pada seluruh aset informasi yang secara eksklusif dihasilkan, dimiliki, dan dipelihara oleh suatu organisasi di dalam batasan sistem operasional mereka sendiri. Menurut Baesens et al. (2022), data internal merupakan tulang punggung dari analisis deskriptif dasar karena memiliki tingkat akurasi historis yang sangat tinggi. Contoh paling klasik dari data internal adalah basis data transaksional yang bersumber dari sistem *Enterprise Resource Planning* (ERP) atau *Customer Relationship Management* (CRM). Data ini merekam secara rinci siapa pelanggan yang bertransaksi, apa produk yang dibeli, kapan waktu pembelian terjadi, hingga riwayat keluhan layanan pelanggan. Tingkat granularitas yang ditawarkan oleh data internal menjadikannya fondasi terkuat untuk membangun model retensi pelanggan maupun peramalan arus kas perusahaan.

Selain data transaksional, sistem internal modern juga menghasilkan volume data perilaku (*behavioral data*) berskala masif yang sering kali terabaikan. Data ini mencakup log aktivitas peladen (*server logs*), durasi kunjungan pada setiap halaman situs web perusahaan, hingga rekam jejak presensi dan metrik kinerja harian karyawan di departemen sumber daya manusia. Berbeda dengan data transaksional yang terstruktur rapi di dalam tabel relasional, data perilaku internal sering kali berwujud semi-terstruktur atau tidak terstruktur, sehingga membutuhkan proses pembersihan tambahan. Walaupun demikian, ekstraksi nilai dari *log* internal ini mampu memberikan

wawasan tak ternilai terkait kendala pengalaman pengguna (*user experience friction*) atau titik-titik kebocoran dalam corong konversi pemasaran digital perusahaan.

Terlepas dari kekayaannya, mengandalkan data internal secara eksklusif dapat menciptakan kebutaan organisasional terhadap dinamika pasar makro. Di sinilah letak urgensi integrasi dengan data eksternal, yaitu informasi yang bersumber dari luar tembok organisasi, sering kali dari domain publik, entitas pemerintah, atau pialang data komersial pihak ketiga. Data eksternal berfungsi sebagai variabel penjelas (*explanatory variables*) yang memberikan konteks lingkungan atas fenomena yang direkam oleh data internal. Sebagai ilustrasi, penurunan drastis pada data penjualan ritel internal tidak dapat dianalisis penyebabnya secara utuh tanpa menyandingkannya dengan data eksternal berupa indikator inflasi nasional, perubahan regulasi pajak, atau lonjakan data iklim ekstrem yang mengganggu jalur logistik (Davenport, 2023).

Akuisisi data eksternal dalam dekade terakhir sangat didominasi oleh proliferasi data media sosial dan umpan (*feed*) berita digital. Platform seperti X (sebelumnya Twitter), Instagram, dan TikTok menghasilkan aliran data teks, gambar, dan video yang merepresentasikan sentimen publik dan tren konsumsi secara aktual. Penggunaan data eksternal tak terstruktur ini memungkinkan organisasi untuk melakukan *social listening*, memantau reputasi merek, dan mendeteksi pergeseran preferensi konsumen jauh sebelum hal tersebut tercermin dalam laporan kuartalan internal. Namun, pemanfaatan data sosial menuntut infrastruktur analitik yang lebih berat, khususnya dalam penerapan algoritma Pemrosesan Bahasa Alami (*Natural Language Processing* / NLP) dan visi komputer (*computer vision*).

Tantangan strategis terbesar bagi seorang *data scientist* bukanlah pada cara menemukan data tersebut, melainkan pada bagaimana menyinergikan data internal dan eksternal secara koheren. Perbedaan arsitektur basis data, ketidakselarasan format stempel waktu (*timestamp*), serta ketidakcocokan definisi operasional variabel kerap kali memicu konflik integritas data saat proses penggabungan (*data merging*). Menurut Kelleher dan Tierney (2022), peleburan yang keliru antara data demografi eksternal dan catatan transaksi internal dapat melahirkan simpulan korelasi semu. Oleh karena itu, perancangan tata kelola data (*data governance*) yang menyatukan glosarium definisi antar-sumber menjadi prasyarat mutlak sebelum tahap pengumpulan data dieksekusi secara teknis.

Assets of Takaza INNOVATIX LABS

B. Teknik Pengumpulan Data

Teknik pengumpulan data merujuk pada metodologi spesifik yang digunakan untuk memindahkan informasi dari sumber asalnya ke dalam media penyimpanan analitik. Dalam evolusinya, teknik pengumpulan informasi telah bergeser dari paradigma pencatatan manual yang lambat menuju otomatisasi digital berbasis komputasi tingkat tinggi. Pemilihan teknik pengumpulan tidak boleh dilakukan secara serampangan; ia harus dijustifikasi berdasarkan volume data yang diincar, kecepatan penyegaran informasi yang dibutuhkan (*data velocity*), serta batasan anggaran komputasi. Secara konseptual, metodologi ini dibagi ke dalam dua kelompok utama, yaitu teknik pengumpulan aktif dan teknik pengumpulan pasif, di mana keduanya memiliki protokol validasi dan tingkat kerumitan arsitektur yang sangat berbeda.

Teknik pengumpulan data aktif melibatkan intervensi langsung dan partisipasi sadar dari subjek penelitian atau sumber data. Metode ini mencakup pelaksanaan survei digital berstruktur, wawancara mendalam yang ditranskripsi, eksperimen laboratorium, hingga uji lapangan seperti pengujian A/B (*A/B testing*) pada antarmuka aplikasi. Dalam riset akademis yang menggunakan pendekatan *Structural Equation Modeling* (SEM) atau analisis faktor, kuesioner berskala likert dikumpulkan secara aktif untuk mengukur variabel laten seperti persepsi, kepuasan, atau niat perilaku yang tidak dapat direkam secara otomatis oleh sensor mesin. Keunggulan mutlak dari teknik aktif ini adalah tingkat relevansinya yang sangat tinggi terhadap rumusan masalah, karena parameter dan butir pertanyaan dirancang khusus oleh peneliti untuk membidik hipotesis tertentu secara presisi (Hair et al., 2022).

Di sisi lain, kelemahan dari pengumpulan aktif adalah biaya operasional yang mahal, potensi bias respons dari partisipan, dan batasan skala ukuran sampel. Untuk menanggulangi hal tersebut, disiplin *data science* lebih banyak bersandar pada teknik pengumpulan data pasif (*passive data collection*). Dalam skenario ini, data diakumulasikan secara otomatis di latar belakang tanpa memerlukan keterlibatan aktif dari manusia. Implementasi utamanya terlihat pada ekosistem *Internet of Things* (IoT), di mana ribuan sensor suhu, kelembapan, atau geolokasi GPS secara konstan mengirimkan metrik telemetri ke peladen awan setiap detiknya. Pengumpulan pasif mampu menghasilkan ekosistem *big data* bervolume ekstrem yang terbebas dari intervensi emosional manusia, menjadikannya sangat ideal untuk memodelkan sistem fisika, rute logistik, dan deteksi anomali keamanan siber.

Dari sudut pandang arsitektur perangkat lunak, teknik memindahkan data ke dalam gudang data (*data warehouse*) dikategorikan berdasarkan dimensi waktunya, yang pertama adalah pemrosesan tumpak (*batch processing*). Pada metode ini, data dikumpulkan, diendapkan sementara, lalu dipindahkan dalam satu bongkahan besar pada jadwal waktu yang telah ditentukan, misalnya setiap jam 12 malam ketika beban lalu lintas jaringan sedang rendah. Pemrosesan tumpak sangat efisien secara biaya dan komputasi untuk menangani laporan rekapitulasi penjualan harian atau perhitungan penggajian bulanan. Alat bantu seperti Apache Hadoop dan arsitektur siklus ETL (*Extract, Transform, Load*) klasik sangat mendominasi ranah pengumpulan data berskala historis ini.

Ketika kebutuhan bisnis menuntut pengambilan keputusan dalam hitungan milidetik, teknik pengumpulan beralih pada arsitektur pemrosesan aliran data (*real-time streaming ingestion*). Dalam skenario seperti perdagangan algoritma frekuensi tinggi di bursa saham atau deteksi penipuan transaksi kartu kredit, menunggu proses *batch* tengah malam adalah sebuah kegagalan operasional fatal. Teknologi pengumpulan aliran seperti Apache Kafka dan Amazon Kinesis memungkinkan aliran data ditangkap, divalidasi, dan dianalisis seketika saat data tersebut tercipta (Zheng, 2023). Tantangan utama dalam membangun sistem pengumpulan aliran adalah tingginya kompleksitas rekayasa perangkat lunak untuk menjaga agar tidak ada paket data yang hilang (*packet loss*) di tengah fluktuasi koneksi jaringan internet.

Tidak kalah penting dari aspek teknis, seluruh teknik pengumpulan data modern terikat erat pada dimensi etika dan kepatuhan hukum perlindungan privasi. Proses pengumpulan pasif yang merekam jejak digital pengguna sering kali berbenturan dengan regulasi ketat seperti *General Data*

Protection Regulation (GDPR) di ranah global atau Undang-Undang Perlindungan Data Pribadi (UU PDP) di Indonesia. Setiap organisasi kini diwajibkan secara legal untuk mengimplementasikan mekanisme persetujuan eksplisit (*informed consent*), memberikan kejelasan mengenai tujuan pengumpulan, serta mengonfigurasi fitur penganoniman (*data anonymization*) secara otomatis pada saat data diakuisisi (Jobin et al., 2021). Mengabaikan landasan etis dalam teknik pengumpulan bukan hanya mencederai validitas riset, melainkan dapat berujung pada sanksi pidana dan kehancuran reputasi institusi.

C. Penggunaan API dan Web Scraping

Dalam era ketika data telah menjadi komoditas terdesentralisasi yang tersebar di ribuan aplikasi berbeda, metode salin-tempel manual atau pengunduhan fail statis tidak lagi memadai. Untuk menjawab tantangan ekstraksi data terprogram antarsistem, industri perangkat lunak memperkenalkan *Application Programming Interface* (API). Secara leksikal, API adalah seperangkat aturan, protokol, dan definisi yang bertindak sebagai "pelayan restoran", yang menerima pesanan data dari aplikasi milik pengguna, mengirimkan pesanan tersebut ke peladen penyedia data, lalu membawa data yang diminta kembali kepada pengguna dalam format yang rapi (Mitchell, 2023). Penggunaan API merupakan standar industri tertinggi untuk akuisisi data sekunder karena ia menawarkan jalur komunikasi yang sah, aman, dan berstruktur tinggi.

Arsitektur API yang paling dominan digunakan dalam proyek *data science* saat ini adalah *RESTful API* (Representational State Transfer). REST beroperasi di atas protokol komunikasi HTTP standar yang sama dengan

yang digunakan peramban web. Ketika seorang ilmuwan data mengirimkan permintaan (sering disebut sebagai metode GET) ke sebuah titik akhir (*endpoint*) API misalnya titik akhir bursa saham atau data cuaca BMKG peladen akan merespons dengan mengirimkan data yang umumnya berformat *JavaScript Object Notation* (JSON). Format JSON sangat disukai karena struktur pasangan kunci dan nilainya (*key-value pairs*) yang hierarkis sangat mudah dikonversi secara langsung menjadi tipe data *Dictionary* pada Python atau *DataFrame* pada Pandas tanpa memerlukan proses pembersihan sintaksis yang rumit (Lawson, 2022).

Meskipun sangat efisien, penggunaan API tidak sepenuhnya bebas tanpa batas. Penyedia layanan lazimnya memberlakukan mekanisme otentikasi ketat, seperti token OAuth atau kunci enkripsi akses (*API Keys*), guna mengidentifikasi siapa yang sedang menarik data. Selain itu, untuk menjaga stabilitas peladen mereka dari serangan *Denial of Service* (DoS), penyedia menerapkan kebijakan pembatasan laju (*rate limiting*). Seorang praktisi data mungkin hanya diizinkan untuk mengeksekusi maksimal 100 permintaan per menit. Jika skrip pemrograman mencoba menarik data melebihi batas tersebut, peladen akan memblokir akses dan memunculkan galat "Error 429: Too Many Requests". Hal ini menuntut perancang skrip analitik untuk memasukkan logika perulangan yang memiliki waktu jeda otomatis (*sleep function*) guna mengakali restriksi tersebut secara elegan.

Namun, tidak semua informasi berharga di internet difasilitasi dengan layanan API publik yang rapi. Terkadang, satu-satunya cara organisasi atau kompetitor menampilkan informasi adalah melalui antarmuka halaman situs web HTML standar. Ketika situasi ini terjadi, para analis beralih pada teknik *Web Scraping*. *Web scraping* adalah proses otomatisasi ekstraksi data tak

terstruktur dari halaman web ke dalam format tabular lokal. Teknik ini mensimulasikan perilaku manusia yang sedang berselancar di internet menggunakan peranti lunak khusus (seperti perpustakaan BeautifulSoup, Scrapy, atau Selenium di Python) untuk membaca, mengurai (*parsing*), dan memotong struktur hierarki dokumen HTML atau *Document Object Model* (DOM) pada situs web sasaran (VanderPlas, 2023).

Mekanisme teknis *web scraping* sangat bergantung pada stabilitas kode HTML situs web sumber. Skrip pemrograman diinstruksikan untuk mencari kelas (*class*) atau penanda tag HTML spesifik misalnya, mencari seluruh tag untuk mengekstrak ribuan katalog harga kompetitor dalam hitungan detik. Meskipun terdengar sangat bertenaga, pendekatan ini luar biasa rapuh (*brittle*). Jika pengembang web dari pihak kompetitor mengubah sedikit saja desain tata letak antarmuka web mereka (seperti mengubah nama tag HTML), maka seluruh skrip ekstraksi data yang telah dibangun akan mengalami kegagalan operasional total. Kondisi ini mengharuskan insinyur data untuk terus memantau dan memperbarui skrip *scraping* secara berkala guna mengimbangi dinamika perubahan situs target.

Dimensi paling krusial dari *web scraping* berada pada garis demarkasi abu-abu hukum dan etika. Mengambil data dari layar situs publik tidak secara otomatis menjadikannya legal secara absolut. Sebelum mengeksekusi skrip penarikan data, sebuah program yang beretika wajib membaca berkas robots.txt milik situs target, yaitu sebuah pedoman teks tersembunyi yang mengatur direktori mana yang boleh atau diharamkan untuk diindeks oleh bot otomasi. Pemaksaan ekstraksi data yang membebani peladen (*server overload*) atau mencuri material berhak cipta intelektual (*intellectual*

property) dengan dalih penelitian dapat memicu konsekuensi litigasi perdata hukum korporasi dan pemblokiran alamat IP secara permanen (O'Neil, 2022).

D. Data Open Source

Sejalan dengan desakan transparansi publik dan kolaborasi ilmiah tanpa batas, dekade terakhir menjadi saksi lahirnya gerakan Data Terbuka (*Open Data Movement*). Berbeda dengan data kepemilikan privat (*proprietary data*) yang dilindungi oleh dinding berbayar korporasi, data sumber terbuka adalah dataset berskala besar yang disediakan secara bebas di domain publik. Filosofi utamanya adalah bahwa data khususnya yang dikumpulkan menggunakan dana pajak negara atau riset hibah public harus dapat diakses, digunakan ulang, dan didistribusikan ulang oleh siapa saja tanpa batasan kekayaan intelektual (Borgman, 2021). Kehadiran data terbuka telah mendemokratisasi ranah kecerdasan buatan, memungkinkan mahasiswa dan perusahaan rintisan kecil untuk melatih algoritma pembelajaran mesin tingkat lanjut tanpa harus mengalokasikan investasi miliaran rupiah untuk proses pengumpulan sampel di lapangan.

Pilar terbesar dari ekosistem data terbuka disediakan oleh lembaga pemerintah dan lembaga swadaya masyarakat supranasional. Inisiatif pemerintah seperti *Satu Data Indonesia*, *data.gov* di Amerika Serikat, atau repositori data dari Bank Dunia dan Perserikatan Bangsa-Bangsa (PBB) menyajikan data empiris yang luar biasa masif. Informasi yang ditawarkan merentang dari data sensus kependudukan demografis, koordinat spasial pemetaan wilayah administratif, hingga rekaman riwayat epidemiologi penyakit berskala nasional. Aksesibilitas terhadap data agregat publik ini

berperan sentral bagi peneliti untuk melakukan analisis tren kebijakan publik, memprediksi laju pertumbuhan ekonomi wilayah, serta mengevaluasi efektivitas distribusi subsidi secara transparan dan akuntabel.

Selain inisiatif pemerintahan, komunitas akademik sains dan praktisi industri teknologi memiliki repositorinya sendiri yang lebih terfokus pada spesifikasi format pemodelan algoritma. *UCI Machine Learning Repository* telah berpuluh tahun menjadi standar emas penyimpanan rujukan (*benchmark*) dataset akademik. Sementara itu, di ranah modern, platform kompetisi ilmu data seperti Kaggle secara konsisten merilis ribuan dataset riil (yang telah dianomimkan) sumbangan dari korporasi teknologi global. Ketersediaan kumpulan data terkurasi ini memungkinkan ilmuwan data membandingkan tingkat efikasi metrik akurasi dari berbagai varian arsitektur jaringan saraf tiruan (seperti CNN atau RNN) menggunakan ukuran sampel standar yang persis sama, menjamin validitas tinjauan sejawat (*peer review*) dalam literatur riset (Mitchell, 2023).

Aspek legalitas yang mengikat perizinan distribusi diatur melalui berbagai struktur lisensi hak cipta progresif, dengan *Creative Commons* (CC) sebagai standar global yang paling diakui. Lisensi seperti CC0 (domain publik absolut) membebaskan pengguna untuk memodifikasi dan mengkomersialkan turunan data tanpa syarat. Sebaliknya, atribut lisensi CC-BY-NC mengharuskan analis untuk memberikan kutipan sitasi asli kepada sang pembuat data, sekaligus melarang data tersebut dirangkai menjadi produk turunan yang berorientasi laba finansial (Jobin et al., 2021). Pemahaman terhadap matriks lisensi terbuka ini amatlah vital; mengkomersialkan model algoritma prediktif korporasi yang kebetulan dilatih menggunakan data akademik berlisensi non-komersial adalah sebuah

Assets of Takaza INNOVATIX LABS

cacat hukum kepatuhan yang dapat berujung pada penuntutan denda pelanggaran hak cipta.

Meskipun menawarkan kelimpahan informasi secara cuma-cuma, penggunaan data terbuka menyimpan jebakan metodologis yang berbahaya bagi peneliti pemula. Karakteristik utama dari sumber data nirbayar ini adalah ketiadaan pihak yang bertanggung jawab atas proses perawatan kualitas data secara terus-menerus (*data maintenance*). Banyak dataset yang diunggah ke portal publik adalah proyek terbengkalai, yang mana metadatanya tidak didokumentasikan dengan baik, makna singkatan variabelnya ambigu, atau metodologi pengumpulan aslinya mengandung cacat bias penyampelan. Menggunakan data yang diunduh secara serampangan dari repositori publik tanpa verifikasi kritis hanya akan mentransfer seluruh kecacatan laten data tersebut ke dalam parameter algoritma analitik yang baru dibangun.

Sebagai solusi strategis terbaik, praktik *data science* kontemporer jarang sekali menggunakan data terbuka sebagai satu-satunya pilar utama sumber analitik final. Organisasi modern memfungsikan repositori publik sebagai data suplementer atau penguat matriks penjelas. Misalnya, algoritma prediksi gagal bayar kredit mikro (UMKM) milik sebuah platform *fintech* tidak hanya bergantung pada profil slip gaji nasabah, tetapi diperkaya silang dengan data terbuka geografis curah hujan BMKG untuk menilai tingkat probabilitas gagal panen dari nasabah yang berprofesi sebagai petani. Sinergisme penyilangan bobot parameter privat dan atribut terbuka inilah yang menjadi katalis terciptanya model wawasan prediktif tingkat tinggi (*high-fidelity models*).

Assets of Takaza INNOVATIX LABS

E. Validasi Data

Akuisisi informasi dari sumber mana pun belum dianggap tuntas tanpa melewati gerbang pemrosesan kritis yang disebut Validasi Data. Validasi merupakan tahapan protektif penyaringan otomatis yang diimplementasikan seketika saat arus data memasuki sistem basis data analitik. Mengabaikan validasi dan membiarkan data kotor mengendap di dalam repositori sentral akan memicu efek bola salju: eror kecil pada tahapan masukan akan beramplifikasi dan merusak keandalan dasbor analitik interaktif eksekutif yang digunakan dalam penetapan keputusan harian. Menurut Squire (2021), validasi tidak ditujukan untuk memperbaiki data yang salah (yang merupakan tugas pembersihan data di fase selanjutnya), melainkan berfungsi sebagai palang pintu diagnostik untuk mendeteksi, menolak, atau menandai secara sistematis setiap baris masukan yang melanggar aturan logika bisnis.

Dimensi kualitas yang pertama kali dievaluasi adalah Akurasi dan Kelengkapan Data. Akurasi memastikan bahwa data yang direkam benar-benar merepresentasikan wujud fisik objek realitas. Validasi akurasi melibatkan pencocokan nilai masukan terhadap batas wajar rentang ukuran (*range constraints*); misalnya, persentase nilai tukar tidak boleh melebihi 100%, atau tahun lahir pelanggan tidak mungkin di bawah angka 1900. Sementara itu, kelengkapan data berfokus pada deteksi nilai selang (*missing variables*). Meskipun sistem mengizinkan formulir dikosongkan, arsitektur tata kelola basis data wajib membedakan secara tegas apakah sebuah bidang sengaja tidak diisi oleh pengguna atau gagal ditransfer karena terjadi jeda koneksi API sistem (Kelleher & Tierney, 2022).

Dimensi struktural selanjutnya berkaitan dengan Konsistensi dan Ketepatan Waktu (*Timeliness*). Konsistensi mensyaratkan berlakunya kaidah

penyelarasan format logika atribut lintas-tabel. Jika status seorang nasabah tercatat "Belum Menikah" di sistem registrasi aplikasi seluler, namun tabel asuransi tanggungan mencatatkan adanya klaim asuransi istri atas nama nasabah yang sama, sistem harus seketika mendeteksi inkonsistensi kontradiktif ini. Sedangkan dimensi ketepatan waktu mengevaluasi kebaruan informasi model analisis sentimen saham membutuhkan aliran data maksimal beberapa detik setelah harga tutup bursa; data yang tertunda 3 jam telah kehilangan nilai utilitasnya dan harus dieliminasi dari memori komputasi agar tidak mendistorsi peramalan (Baesens et al., 2022).

Dalam praktiknya, validasi otomatis dijalankan dengan memformulasikan Skema Data (*Data Schemas*). Skema bertindak laksana "paspor imigrasi" yang menetapkan struktur prasyarat ketat, meliputi: tipe data dasar (harus *integer*), format karakter khusus, dan klausul tidak boleh kosong (*not null*). Dalam ekosistem Python, perpustakaan validasi kontrak berkinerja tinggi seperti *Pydantic* atau *Great Expectations* kerap kali disisipkan ke dalam jalur pipa (*pipeline*) rekayasa data. Setiap kali sistem *web scraping* atau API menarik ribuan kolom, paket ini akan memverifikasi kesesuaian bentuk data. Jika kolom yang dijanjikan berisi angka namun secara tiba-tiba tersisipi aksara abjad, proses injeksi data digagalkan parsial seketika sebelum ia menularkan korupsi format ke seluruh ekosistem gudang data operasional.

Skenario mitigasi kesalahan penolakan harus dirancang agar tidak melumpuhkan sistem operasional pusat secara keseluruhan. Ketika proses validasi skema mengidentifikasi ribuan deret anomali selama proses pengumpulan tumpak (*batch*), arsitektur ETL modern menerapkan mekanisme Karantina Data. Rekaman yang dinyatakan cacat dialihkan

(*redirected*) menuju folder isolasi sementara ("*dead-letter queue*"). Pemisahan karantina ini menjamin bahwa model pembelajaran analitik yang dijadwalkan dapat terus berlatih menggunakan sisa porsi data yang valid dan steril, sementara anomali ekstrem di folder karantina akan ditinjau keesokan harinya untuk diselidiki apakah terjadi kebocoran serangan keamanan peretas atau sekadar kesalahan versi pembaruan aplikasi pihak ketiga.

Sebagai ringkasan komprehensif, standar kualitas yang harus dipenuhi dalam alur masuk data diorganisasikan ke dalam matriks dimensi pada Tabel 4.1 di bawah ini. Tabel ini berfungsi sebagai pedoman teknis bagi ilmuwan data maupun manajer basis data dalam memformulasikan rancangan aturan pemfilteran pra-analisis.

Assets of Takaza INNOVATIK LABS

Tabel 3. Dimensi Kualitas Data dalam Proses Validasi Akuisisi

Dimensi Kualitas	**Definisi Konseptual**	**Contoh Deteksi Kasus Anomali (Validasi Sistem)**	**Strategi Mitigasi Awal di Tingkat Akuisisi**
Akurasi (*Accuracy*)	Kesesuaian nilai yang dicatat dengan kondisi fakta empiris di lapangan.	Sistem mencatat koordinat bujur dan lintang toko berada di tengah lautan.	Penerapan batasan rentang nilai (*range constraints*) toleransi geometris geolokasi.
Kelengkapan (*Completeness*)	Ketersediaan pencatatan seluruh parameter atribut kunci yang	Rekam medis dikirim tanpa mencantumkan label diagnosis kategori akhir.	Penolakan masuk otomatis (*schema rejection*) untuk atribut *Not-Null*.

	diwajibkan.		
Konsistensi (*Consistency*)	Keselarasan relasi kebenaran nilai satu variabel terhadap variabel pengikat lain.	Usia pelanggan tertulis 15 tahun, namun memiliki riwayat kartu kredit platinum 5 tahun.	Uji logika silang (*cross-field validation*) berbasis aturan matriks domain bisnis.
Integritas Format (*Conformity*)	Kepatuhan mutlak nilai data terhadap standar kaidah sintaksis struktur yang ditetapkan.	Tanggal transaksi berformat "DD-MM-YYYY" bercampur dengan "MM/DD/YY".	Penerapan konversi ekspresi reguler (Regex) untuk penyeragaman sintaks standar ISO.
Kebaruan (*Timeliness*)	Keterikatan relevansi makna data terhadap rentang jeda waktu saat pengumpulan.	Sistem lelang otomotif menarik data harga suku cadang berdasarkan katalog 4 tahun lalu.	Menandai log dengan stempel waktu ekstraksi (*timestamping*) kedaluwarsa sist

F. Studi kasus

Akuisisi Data Terpadu untuk Memprediksi Preferensi Konsumen UMKM Fashion di Indonesia

Latar Belakang Masalah Sebuah jenama Usaha Mikro Kecil dan Menengah (UMKM) *fashion* lokal yang berbasis di Bandung mengalami kemandekan pertumbuhan dan ketidakmampuan membaca pergeseran tren estetika konsumen Gen-Z secara cepat. Pemilik usaha berinisiatif merancang

riset mandiri berjudul *"Predicting Consumer Preferences for Indonesian Fashion MSMEs using Machine Learning"* guna mendeteksi tren warna, model potongan, dan sensitivitas harga yang diproyeksikan meledak di musim liburan mendatang. Tantangan utama penelitian ini bukan pada pembuatan arsitektur mesin prediksi, melainkan pada kebingungan tim IT dalam mengakuisisi data mentah lintas platform sosial karena preferensi audiens mereka sangat terfragmentasi di berbagai ekosistem digital.

Langkah Kerja Akuisisi Data 1. *Pemetaan Sumber Internal dan Eksternal:* Tim mengawali alur riset dengan menetapkan pilar arsitektur sumber komposit. Sumber internal disedot langsung dari platform niaga elektronik Shopee, memanfaatkan basis data riwayat pembelian enam bulan ke belakang dan catatan inventori gudang mereka sendiri. Namun, memprediksi selera masa depan tak bisa sekadar berpijak pada riwayat masa lalu. Maka, mereka menyuntikkan sumber variabel prediktif eksternal yang diakuisisi dari media sosial Instagram, guna memantau sentimen ulasan serta popularitas tagar gaya berpakaian (*fashion hashtags*) yang sedang digandrungi oleh pempengaruh (*influencers*) lokal. 2. *Ekstraksi Data Terstruktur via API Niaga-el:* Untuk pengumpulan aliran dari sisi internal, pengembang UMKM mengajukan otorisasi kunci pengembang API *Shopee Open Platform*. Melalui metode *RESTful API* tipe. GET(), sistem diinstruksikan menyedot agregasi data transaksi secara tumpak (*batch*) harian. Respons API berbentuk format dokumen JSON yang rapi mengalirkan variabel krusial seperti: SKU barang terjual, nominal diskon yang dipakai pelanggan, demografi usia pembeli, serta frekuensi konversi geranjang belanja berstatus sukses. Data berskala gigabita ini seketika dikonversi bersih menjadi struktur *DataFrame Pandas* numerik. 3.

Otomatisasi Web Scraping pada Domain Sosial: Karena Instagram menerapkan perlindungan protektif pembatasan laju (*rate limits*) API publik yang ketat, tim berinisiatif menyusun skrip *Web Scraping* semi-pasif menggunakan kombinasi pustaka *Selenium* pada Python. Skrip diotomatisasi secara bergilir membuka profil sepuluh pempengaruh busana *fashion* lokal terbesar setiap akhir pekan, menyisir elemen tag kerangka DOM web, lalu menggores (*scrape*) tautan deskripsi gambar, jumlah iterasi tanda *Likes*, serta ratusan blok teks komentar pujian dan keluhan pelanggan terhadap desain pakaian kompetitor tanpa melanggar parameter aturan di *robots.txt*. 4. *Penerapan Gerbang Validasi Awal:* Jutaan baris teks yang tercampur kode emoji divalidasi keras melalui pintu imigrasi awal. Tim merancang filter Regex otomatis untuk mengkarantina karakter aneh tak bermakna (*Mojibake*) dan menolak injeksi data tanggal komentar berformat anonim. Validasi Konsistensi juga menggagalkan skrip penipu kompetitor (bot ulasan palsu) yang memborbardir kolom komentar ulasan dalam sepersekian detik. Teks kotor hasil keruk laman situs segera dipisahkan ke modul *NLP Stemming* bahasa Indonesia untuk mencari lema kata seperti "oversize", "rajut", atau "kemahalan".

Solusi dan Pengambilan Keputusan Berbasis Data Melalui perpaduan apik antara arsitektur API transaksi terstruktur komersial (*hard data*) dan ekstrak kualitatif penambangan teks sosial (*soft data*), model *machine learning* berhasil menemukan pola korelasi silang yang revolusioner. Sistem prediktif mengungkap sentimen preferensi di media sosial terhadap desain "baju rajut bumi (*earth-tone*)" secara konsisten meledak satu bulan sebelum lonjakan aktual permintaan barang di keranjang aplikasi Shopee. Pemilik UMKM mengkapitalisasi wawasan keunggulan

temporal akuisisi ini dengan segera menyesuaikan persediaan bahan pabrik jahit sebulan lebih cepat sebelum gelombang lonjakan musim tren tiba, secara sukses menghapus risiko barang stok tidak laku mati di gudang secara permanen.

Assets of Takaza INNOVATIX LABS

BAB V
DATA CLEANING DAN DATA PREPARATION

Dalam ekosistem *data science* dan analitik bisnis modern, sering kali muncul sebuah adagium klasik yang menyatakan bahwa seorang ilmuwan data menghabiskan hingga 80% waktu mereka murni untuk membersihkan, merapikan, dan menyiapkan data, sementara hanya 20% sisa waktu yang tersedia untuk benar-benar membangun dan melatih model prediktif. Proporsi waktu yang timpang ini sama sekali bukanlah sebuah kebetulan matematis atau indikasi inefisiensi kerja, melainkan sebuah refleksi nyata dari realitas lapangan. Data yang tersedia di dunia nyata baik yang ditarik dari *database* transaksional perusahaan, diekstraksi melalui metode *web scraping*, maupun direkam dari *log* sensor hampir selalu berada dalam kondisi yang sangat "kotor", berantakan, tidak lengkap, atau bahkan mengandung distorsi bias yang parah.

Mengabaikan proses ini sama halnya dengan membangun sebuah gedung pencakar langit di atas fondasi pasir. Sebegitu canggihnya arsitektur algoritma *machine learning* yang digunakan, ia tetap akan menyesatkan pengambil keputusan jika disuapi dengan data yang cacat. Bab ini didedikasikan untuk memandu Anda secara mendalam melalui seni sekaligus sains dalam melakukan preparasi data (*data wrangling*). Kita akan membedah metodologi sistematis lapis demi lapis untuk mentransformasikan data mentah yang kacau balau menjadi sebuah *dataset* yang terstruktur, bersih, konsisten, dan siap secara komputasional untuk diolah. Tujuannya

adalah menghasilkan wawasan bisnis dan prediksi perilaku konsumen yang tingkat akurasinya dapat dipertanggungjawabkan secara akademis maupun manajerial.

A. Pentingnya Kualitas Data

Kualitas data tidak boleh sekadar dipandang sebagai variabel pendukung atau langkah opsional dalam jalur pipa (*pipeline*) rekayasa data. Kualitas data adalah fondasi ontologis utama yang secara absolut menentukan integritas, validitas, dan reliabilitas seluruh proyek analitik. Mengumpankan data yang berkualitas rendah ke dalam sistem akan memicu fenomena yang dikenal luas sebagai *"garbage in, garbage out"*. Hal ini berarti model akan menghasilkan korelasi yang *spurious* (korelasi semu) atau prediksi yang bias, yang pada akhirnya dapat merusak strategi bisnis secara fatal. Praktisi data dituntut untuk memiliki kepekaan analitis bahwa kualitas data harus dievaluasi dan diukur melalui beberapa dimensi kritis sebelum proses pemodelan matematis dieksekusi.

Berikut adalah pemisahan dimensi utama kualitas data yang wajib divalidasi secara ketat:

- **Akurasi (*Accuracy*):** Dimensi ini mengukur sejauh mana data yang tercatat di sistem secara presisi mencerminkan realitas fisik atau kejadian faktual dari objek yang diamati. Ketidakakuratan lazimnya bersumber dari kesalahan input tipografi manual (*human error*), sensor IoT yang terkalibrasi dengan buruk, atau malfungsi pada sistem *Application Programming Interface* (API) saat transmisi data.
- **Kelengkapan (*Completeness*):** Evaluasi kelengkapan bertujuan memastikan bahwa tidak ada kekosongan nilai (*missing values*) pada

variabel-variabel prediktor krusial yang diwajibkan untuk analisis. Sebuah baris data pelanggan yang kehilangan informasi "total pengeluaran bulanan" akan menjadi tidak berguna jika tujuan utama riset adalah melakukan segmentasi daya beli.

- **Konsistensi (*Consistency*):** Dimensi ini menuntut adanya keselarasan format dan makna logika data ketika dilakukan pengecekan silang antar berbagai sumber, tabel, atau *database* relasional. Sebagai ilustrasi, status "Aktif" pada tabel profil pengguna di sistem *front-end* harus memiliki kodifikasi dan representasi hierarki yang sama persis di dalam tabel riwayat transaksi *back-end*.
- **Validitas (*Validity*):** Validitas menguji apakah nilai data telah sepenuhnya patuh terhadap batasan format sintaksis dan aturan bisnis (*business rules*) yang telah dideklarasikan sebelumnya. Contoh paling nyata adalah rentang usia responden yang tidak mungkin bernilai negatif, atau format penulisan alamat surel yang secara mutlak harus mengandung karakter "$\dots$@$\dots$.com".
- **Keunikan (*Uniqueness*):** Dimensi ini berfokus pada sterilisasi *dataset* dari keberadaan entitas data duplikat. Jika sebuah data riwayat pembelian dari satu pelanggan tercatat dua atau tiga kali akibat *glitch* pada peladen, algoritma analitik akan memberikan pembobotan statisik yang berlebih (*overweighting*) pada pelanggan tersebut, mendistorsi seluruh parameter peramalan.

Assets of Takaza INNOVATIX LABS

B. Penanganan Missing Values

Fenomena *missing values* atau hilangnya titik data tertentu adalah tantangan universal yang tidak mungkin dihindari dalam setiap siklus proyek *data science*. Kekosongan informasi ini dapat dipicu oleh rupa-rupa faktor operasional: terputusnya koneksi jaringan internet saat proses ekstraksi, subjek penelitian yang menolak atau sengaja melompati pertanyaan sensitif pada instrumen kuesioner, hingga kelemahan desain arsitektur basis data itu sendiri. Sebelum seorang analis gegabah menjatuhkan pilihan pada teknik penanganan tertentu, langkah diagnostik pertama yang wajib dilakukan adalah mengidentifikasi mekanisme probabilitas di balik hilangnya data tersebut. Menerapkan strategi imputasi yang keliru tanpa memahami mekanismenya justru berisiko menginjeksi bias sistematis baru yang jauh lebih berbahaya.

Assets of Takaza INNOVATIX LABS

Terdapat tiga taksonomi mekanisme utama hilangnya data yang harus dipahami secara konseptual:

1. **MCAR (*Missing Completely at Random*):** Pada skenario ini, probabilitas hilangnya data terjadi secara acak murni dan tidak memiliki korelasi sedikit pun dengan variabel pengamatan lainnya, baik variabel yang diamati maupun yang tidak diamati. Ini adalah kondisi paling ideal secara statistik karena mengeliminasi baris data ini tidak akan mengubah distribusi probabilitas sampel asli, sehingga bebas dari bias turunan.
2. **MAR (*Missing at Random*):** Skenario ini mengindikasikan bahwa data hilang memang memiliki pola tertentu, namun pola kehilangannya tersebut dapat dijelaskan secara rasional oleh variabel penjelas lain yang datanya tersedia lengkap di dalam *dataset*.

Contoh klasiknya adalah pola respons demografis: responden dengan gender laki-laki mungkin secara persentase lebih jarang mengisi kolom "pengeluaran produk perawatan kulit bulanan" dibandingkan dengan responden perempuan.

3. **MNAR (*Missing Not at Random*):** Ini adalah kondisi patologis yang paling problematik dan sangat sulit untuk ditangani. Data dinyatakan MNAR jika alasan di balik hilangnya data berkaitan langsung dengan nilai intrinsik dari data yang hilang itu sendiri. Sebagai contoh empiris, individu yang berada di kelompok desil pendapatan paling tinggi sering kali secara proaktif menolak untuk mengisi form survei terkait jumlah kekayaan riil mereka karena alasan privasi.

Untuk menetralisasi masalah tersebut, teknik penanganan *missing values* dikategorikan ke dalam metode berikut:

- **Penghapusan (*Deletion*):** Teknik ini membuang baris data secara penuh (*listwise deletion*) atau mengabaikan kolom spesifik (*pairwise deletion*). Pendekatan konservatif ini hanya dapat dijustifikasi secara statistik jika jumlah data yang hilang sangat marjinal (kurang dari 5% dari total populasi sampel) dan telah terbukti memenuhi asumsi MCAR.
- **Imputasi Sederhana Univariat:** Sebuah langkah kompromi cepat dengan mengisi sel yang kosong menggunakan nilai proksi statistik deskriptif sentral. Penggantian menggunakan nilai Rata-rata (*Mean*) cocok diaplikasikan pada distribusi data yang normal sempurna. Nilai Tengah (*Median*) dipilih jika distribusi data terindikasi menceng (*skewed*) atau terkontaminasi pencilan, sedangkan Nilai

Assets of Takaza INNOVATIX LABS

Modus (*Mode*) adalah satu-satunya opsi logis untuk mengimputasi variabel yang berjenis kategorikal.

- **Imputasi Multivariat Tingkat Lanjut:** Jika korelasi antar-fitur sangat kuat, praktisi wajib menggunakan algoritma prediktif iteratif. Algoritma *K-Nearest Neighbors* (K-NN) mencari jarak terdekat dari baris data yang utuh untuk "meminjam" nilainya, sedangkan metode *Multivariate Imputation by Chained Equations* (MICE) menjalankan serangkaian regresi berantai pada setiap variabel yang hilang secara berulang hingga mencapai titik konvergensi statistik.

C. Deteksi dan Penanganan Outlier

Dalam leksikon statistika, *outlier* atau pencilan didefinisikan sebagai titik obeservasi data yang menyimpang dan menjauh secara sangat ekstrem dari kumpulan pusat observasi mayoritas lainnya. Tantangan filosofis terbesar bagi seorang ilmuwan data adalah membedakan antara *outlier* yang terlahir dari malfungsi proses pencatatan (misalnya, harga sebuah tas ransel tercatat Rp 1) dengan *outlier* yang merepresentasikan fenomena langka namun benar-benar terjadi di dunia nyata (seperti sebuah lonjakan transaksi bernilai miliaran rupiah yang merupakan indikator aktivitas pencucian uang). Mengabaikan keberadaan pencilan berpotensi menghancurkan asumsi dasar normalitas varians, yang akan merusak algoritma sensitif seperti Regresi Linier. Sebaliknya, menghapus seluruh pencilan tanpa proses investigasi mendalam sama saja dengan membuang sinyal-sinyal anomali berharga.

Berikut adalah taksonomi teknik deteksi matematis yang kerap diimplementasikan:

- **Z-Score (Standard Score):** Teknik parametrik ini mengkuantifikasi seberapa jauh margin deviasi sebuah titik data terhadap nilai rata-rata populasinya, direpresentasikan dalam satuan simpangan baku. Secara matematis ditulis sebagai $Z = \frac{x - \mu}{\sigma}$. Dalam konsensus akademik umum, setiap titik observasi yang menghasilkan nilai Z-score di atas angka +3 atau berada di bawah -3 akan langsung diklasifikasikan sebagai *outlier* ekstrem.
- **Rentang Interkuartil / Interquartile Range (IQR):** Metode ini mengandalkan pendekatan statistika *robust* (kebal) yang tidak mudah terdistorsi oleh data ekstrem seperti halnya *mean*. IQR dihitung dari selisih kuartil atas dan kuartil bawah. Sebuah data divonis sebagai pencilan jika posisinya terdampar di luar wilayah batas aman operasional, yaitu lebih kecil dari formula $Q1 - 1.5 \times IQR$ atau melampaui ambang batas formula $Q3 + 1.5 \times IQR$.
- **Isolation Forest:** Untuk *dataset* berdimensi tinggi (*high-dimensional space*) di mana pencilan tidak terlihat di satu variabel namun aneh secara kombinasi multivariabel, metode statistik tradisional akan gagal. *Isolation Forest* merespons ini dengan menggunakan pendekatan *machine learning* berbasis *ensemble*, di mana algoritma akan membangun ratusan pohon keputusan acak (*random decision trees*) untuk secara struktural mengisolasi atau "memenjarakan" titik-titik data minoritas yang secara anomali sangat mudah dipisahkan dari kelompok mayoritas.

Setelah terdeteksi secara meyakinkan, protokol penanganan dapat dieksekusi melalui strategi ini:

Assets of Takaza INNOVATIX LABS

1. **Pemangkasan (*Trimming*):** Pendekatan destruktif di mana observasi pencilan sepenuhnya direduksi dan dihapus dari kumpulan data. Ini direkomendasikan secara khusus jika analis 100% yakin bahwa anomali tersebut murni berasal dari eror mesin sensor atau kelalaian manusia.
2. **Pemotongan / Pembatasan (*Winsorizing / Capping*):** Alih-alih menghapus data dan kehilangan ukuran sampel, teknik ini membendung nilai yang kelewat batas dengan menariknya kembali ke ambang persentil yang wajar. Sebagai contoh, seluruh angka penjualan yang menembus persentil ke-99 akan ditimpa dan disamaratakan dengan nilai maksimum pada persentil ke-99 tersebut.
3. **Transformasi Skala Penekanan:** Mengaplikasikan fungsi matematika tingkat lanjut, seperti transformasi logaritmik atau akar kuadrat (*square root*), dengan tujuan mengompresi jarak interval antara titik-titik rentang data yang ekstrem agar efek distorsinya pada model menjadi jinak, tanpa harus secara fisik menghapus atau mengubah esensi data aslinya.

Assets of Takaza INNOVATIX LABS

D. Transformasi Data

Fase transformasi data menempati kedudukan yang sangat sentral karena ia bertugas menjembatani perbedaan "bahasa" antara format data mentah manusiawi dengan format komputasi yang diwajibkan oleh arsitektur algoritma pembelajaran mesin. Realitas matematis menunjukkan bahwa algoritma berbasis kalkulasi jarak ruang vector seperti *Support Vector Machines* (SVM) atau pengelompokan *K-Means* memiliki sensitivitas yang

berlebihan terhadap disparitas unit metrik. Jika fitur "Umur" diukur dalam skala 1 hingga 100, sementara fitur "Gaji Tahunan" melentur dalam puluhan juta rupiah, algoritma secara keliru akan memandang "Gaji" sebagai variabel tunggal terpenting hanya karena angkanya masif. Transformasi memastikan arena bermain yang adil (*level playing field*) bagi seluruh atribut data.

Teknik-teknik operasional transformasi yang mutlak harus dikuasai meliputi:

- **Penskalaan Vektor Numerik (*Scaling*):**
  - *Min-Max Normalization:* Proses kompresi matematis linier yang meremas seluruh distribusi rentang nilai orisinal agar muat dan persis berada ke dalam kotak buatan berskala absolut $[0, 1]$. Teknik ini sangat presisi digunakan apabila praktisi memiliki pengetahuan pasti tentang nilai pijakan minimum dan atap maksimum teoritis dari sebuah variabel pengamatan.
  - *Standarisasi / Z-Score Scaling:* Sebuah mekanisme penyesuaian distribusi di mana keseluruhan himpunan data digeser dan disejajarkan sehingga titik rata-ratanya berlabuh tepat pada angka nol (0) dengan deviasi standar seragam sebesar satu (1). Formulasi ini mempertahankan bentuk kurva *outlier* dan direkomendasikan secara *default* untuk mayoritas algoritma jaringan saraf tiruan (*neural networks*).
- **Pengkodean Variabel Kategorikal (*Encoding*):**
  - *One-Hot Encoding:* Digunakan khusus untuk menguraikan variabel bertipe nominal murni yang tidak memiliki strata hierarki atau urutan alamiah (misalnya: warna mobil atau nama provinsi). Algoritma akan menciptakan matriks kolom biner (berisi angka 1

dan 0) yang ekuivalen dengan jumlah kategori unik yang eksis, demi menghindari model yang salah menafsirkan angka sebagai sebuah urutan kasta.

- *Label / Ordinal Encoding:* Diimplementasikan secara ketat pada data kualitatif yang mengusung tingkatan derajat berurutan atau interval hierarki (*ordinal*). Misalnya, kategori kuesioner "Sangat Buruk, Buruk, Cukup, Baik" akan direpresentasikan secara urut dengan rentetan angka matematis "0, 1, 2, 3" agar model sanggup mencerna arah eskalasi nilainya.

- **Transformasi Distribusi Logaritmik:**
  - Dalam banyak kasus data bisnis di lapangan, kurva frekuensi tidak pernah membentuk lonceng simetris, melainkan memiliki ekor yang menjuntai terlampau panjang ke arah kanan (*right-skewed*), seperti yang sering ditemui pada distribusi pendapatan per kapita atau jumlah penonton video viral. Menyuntikkan fungsi $log(x)$ akan menarik ekor distribusi yang ekstrem tersebut kembali ke tengah, memaksa struktur distribusinya menyerupai kurva Gaussian (normal) agar kompatibel dengan asumsi parametrik.

E. Integrasi Data

Seiring dengan masifnya proses digitalisasi, sebuah entitas korporasi lazimnya tidak lagi mengandalkan satu pilar sistem penyimpan data tunggal. Data kini terfragmentasi dalam silo-silo departemen yang terisolasi; mulai dari *database* SQL transaksional, platform sistem pengelolaan hubungan pelanggan (CRM) berbasis awan, hingga *log* perilaku navigasi pengunjung aplikasi berformat JSON. Proses integrasi data (*data integration*) menjadi

krusial karena ia merupakan satu-satunya jembatan untuk menyatukan potongan-potongan mozaik tersebut agar dapat menampilkan potret holistik perilaku pengguna atau kinerja perusahaan. Namun, upaya mensintesis data eksternal dan internal ini sering kali melahirkan mimpi buruk sinkronisasi dan anomali kardinalitas yang kompleks.

Tantangan arsitektural utama yang harus ditaklukkan dalam proses fusi data meliputi:

- **Pemetaan Skema Atribut (*Schema Mapping*):** Benturan terminologi terjadi ketika dua departemen menamai objek yang sama dengan leksikon yang sama sekali berbeda. Umpamanya, sistem penjualan di cabang pertama menamai sebuah kolom kunci dengan istilah Cust_ID, sementara departemen pemasaran digital melabelinya sebagai User_Identity. Sistem integrasi wajib memformulasikan kamus metadata yang mengatur translasi antarkolom agar penyatuan antar-sumber *(join operations)* tidak menemui kegagalan struktur.
- **Resolusi Entitas (*Entity Resolution & Record Linkage*):** Sebuah problem algoritmis yang menuntut kecerdasan sistem untuk menganalisis dan mendeduksi apakah dua baris catatan dari dua sumber independen sejatinya menunjuk pada satu objek individu fisik yang sama. Kelemahan entri manual kerap menyebabkan penulisan "PT. Inovasi UMKM", "PT Inovasi UMKM", atau "Inovasi UMKM, PT". Model *fuzzy matching* sering dikerahkan di sini guna mendeteksi koefisien kemiripan string agar identitas ganda dapat direkonsiliasi.

- **Deduplikasi Konjungtif:** Konsekuensi matematis yang tidak bisa dihindari saat menggabungkan dua himpunan tabel dengan metode perkalian kartesian silang adalah munculnya kloning data, alias duplikasi entitas tak disengaja. Algoritma perapian harus dirancang menggunakan parameter waktu kejadian (stempel waktu / *timestamp*) untuk mengeliminasi salinan usang dan hanya menyimpan rekaman versi yang paling mutakhir.

Sebagai panduan operasional komprehensif, berikut adalah tabel matriks ringkasan strategi krusial di ranah pembersihan dan prapemrosesan data:

Tabel 5.1: Matriks Strategi Data Cleaning dan Preparation

Fokus Tahapan Operasional	Tujuan Fundamental Modul	Algoritma & Teknik Eksekusi Populer
1. Penanganan *Missing Values*	Melengkapi kekosongan informasi numerik/kategorikal tanpa merusak distribusi asli populasi sampel.	*Mean/Median/Mode Imputation, Multiple Imputation by Chained Equations (MICE), K-NN Imputer*.
2. Deteksi & Peredaman *Outlier*	Memitigasi efek distorsif atau reduksi *noise* ekstrem tanpa menghilangkan esensi anomali penting.	*Interquartile Range (IQR) Fences, Z-Score Thresholds, Algoritma Isolation Forest, Winsorization.*
3. Penskalaan & Transformasi	Mensinkronisasi perbedaan skala unit dimensi untuk memfasilitasi komputasi berbasis vektor	*Standardization (Z-Score), Min-Max Scaling [0,1], Logarithmic Transformation, Robust Scaler*.

	kedekatan.	
4. Pengkodean (*Encoding*)	Mengkonversi tekstur string leksikal ke dalam format representasi biner matematika vektor spasial.	*One-Hot Encoding, Ordinal/Label Encoding, Target Mean Encoding.*
5. Fusi & Integrasi Data	Merajut dan memadukan silo data yang terdesentralisasi agar menjadi satu korpus tabel relasional utuh.	*Left/Inner Join SQL, Fuzzy String Matching, Algoritma Deduplication berbasis Timestamp*.

F. Studi kasus

Pembersihan Data Agregator E-Commerce "Toko Serba Ada"

Latar Belakang Masalah:

Perusahaan "Toko Serba Ada", yang beroperasi sebagai platform penyedia agregator untuk berbagai produk ekosistem UMKM (Usaha Mikro, Kecil, dan Menengah), mendapati bahwa tingkat retensi pengguna mereka menurun tajam. Tim manajemen proyek dan administrator pemasaran menginisiasi sebuah riset saintifik guna membangun model *machine learning* untuk memprediksi probabilitas probabilitas churn pelanggan yakni kecenderungan pelanggan untuk berhenti berlangganan atau berbelanja. Bahan baku analitik yang diekstraksi mencakup log riwayat transaksi kumulatif selama 6 bulan terakhir, berisikan variabel krusial seperti ID pelanggan unik, agregat jumlah pembelian kotor bulanan, proksi durasi sesi navigasi saat *login*, serta label kategori demografi regional. Sayangnya, karena platform menampung ratusan data penjual berskala UMKM dengan

kedisiplinan *input* yang bervariasi, wujud data mentah tersebut sangat porak-poranda.

Langkah Pengerjaan dan Implementasi Sistematis:

1. **Auditing dan Pemprofilan Eksploratif (*Profiling*):** Langkah investigatif awal menggunakan modul statistik deskriptif mengungkap temuan mencemaskan: tercatat 15% dari total sel pada kolom metrik "jumlah pembelian" berstatus nihil (*Null/NaN*). Lebih parahnya, pemeriksaan kualitas menemukan adanya fenomena numerik mustahil pada fitur "durasi login", yang mana ratusan baris merekam nilai waktu negatif absolut sebuah indikasi jelas terjadinya disfungsi pada arsitektur pencatatan waktu peladen (*server logging error*). Di sisi lain, deteksi tebaran menemukan puluhan nilai ekstrem di mana jumlah nominal pembelanjaannya tidak logis untuk ukuran pembeli retail.
2. **Imputasi Presisi untuk *Missing Values*:** Tim ilmuwan data merumuskan aturan bisnis untuk mengisi kekosongan informasi secara diferensial. Mengingat sebagian besar entitas pelanggan yang absen log aktivitasnya dapat ditafsirkan sebagai pengguna pasif seutuhnya, sistem mengaplikasikan imputasi konstanta absolut di mana *missing values* pada lajur aktivitas diisi paksa dengan angka nol (0). Sebaliknya, sel-sel kosong yang merusak kelengkapan pada atribut kualitatif demografi dikonsolidasikan dan ditambal menggunakan pendekatan sentralitas modus (kategori kemunculan terbanyak di provinsi tersebut) agar pola sosiodemografis dasar tidak terdistorsi.

Assets of Takaza INNOVATIX LABS

3. **Sterilisasi *Outlier* yang Kontekstual:** Berhadapan dengan data cacat "durasi login" yang bernilai negatif, diputuskan sebuah prosedur koreksi transformasi fungsi absolut ($\vert{}x\vert{}$), mengingat selisih waktu terbalik di sistem masih mewakili durasi interaksi riil pengguna. Sementara itu, untuk menata letak angka pembelanjaan hiper-ekstrem di atas persentil 99, analisis lanjutan menyingkap bahwa angka-angka raksasa ini sebenarnya merepresentasikan segmen pelanggan *whale* (misalnya pembeli kain eceran grosir/distributor tangan kedua). Karena profil *whale* ini sangat dipertahankan agar tidak teranomali oleh model, titik datanya tidak dibuang melalui pemangkasan, melainkan diredam melalui teknik *Winsorizing* batas atas.
4. **Eksekusi Transformasi Sintaksis:** Atribut regional yang merepresentasikan "kategori demografi" spasial (contoh: Jawa, Sumatra, Bali) diekspos melalui pemrosesan *One-Hot Encoding*, yang memecahnya menjadi deretan kolom indikator biner ortogonal yang rapi, memastikan model tidak menjatuhkan asumsi palsu bahwa pulau yang satu lebih "tinggi" nilainya dari yang lain. Untuk fitur finansial "jumlah pembelian" yang kurvanya menukik curam condong ke kanan alias sarat dengan konsumen mikro bergaji standar, pengaplikasian transformasi logaritma natural seketika menjinakkannya, membentangkan data menjadi bentuk bel Gaussian yang ideal untuk ditelaah.
5. **Fusi Integratif Multitabel:** Tahap krusial penutup adalah merajut lajur historis transaksi temporer tersebut dengan katalog master data profil preferensi pelanggan secara permanen menggunakan agregasi

Assets of Takaza INNOVATIX LABS

Customer_ID sebagai Kunci Utama atau pengait primer (*Primary Key*). Untuk menggagalkan potensi penggandaan basis baris prapenggabungan, rutinitas algoritma deduplikasi dieksekusi secara ketat dengan memberikan parameter perioritas kepada stempel waktu transaksi yang menyandang atribut temporal paling terbaru (*latest entry update*).

Melalui kedisiplinan dan orkestrasi pembersihan multi-tahap tersebut, luaran proyek berupa *dataset* korporasi raksasa tersebut akhirnya sukses bertransformasi menjadi sebuah material yang terstandarisasi sempurna, steril dari segala manifestasi bising (*noise*), dan sangat siap secara metodologis untuk diumpankan ke dalam algoritma klasifikasi *Logistic Regression* maupun pembentukan *Random Forest* guna meramal potensi pelarian pelanggan.

Assets of Takaza INNOVATIX LABS

BAB VI
MANAJEMEN DATA DAN BASIS DATA

Setelah kita melampaui fase kritis pembersihan dan preparasi data pada bab sebelumnya, tantangan rasional berikutnya yang menanti seorang praktisi analitik adalah persoalan persistensi dan orkestrasi infrastruktur penyimpanannya. Data yang telah dibersihkan hingga berwujud paripurna tidak akan memiliki nilai keabadian apabila ia hanya bersemayam sesaat di dalam memori akses acak (RAM) komputer saat skrip algoritma dieksekusi (Kleppmann, 2023). Dalam spektrum industri modern, mulai dari manufaktur global hingga penerbitan akademik yang mengelola ribuan hak kekayaan intelektual (HKI), data menuntut sebuah ekosistem habitat yang permanen, aman, dan dapat diakses kembali dalam kecepatan presisi tinggi. Bab ini menandai transisi penting dari sekadar "mengolah data" menjadi "mengelola aset data" berskala enterprise.

Manajemen data dan fondasi basis data adalah pilar penyangga dari segala inovasi kecerdasan buatan (*Artificial Intelligence*) dan otomasi *machine learning* yang kita kenal saat ini (Coronel & Morris, 2022). Tanpa arsitektur basis data yang kokoh, algoritma prediktif paling canggih sekalipun akan mengalami kelaparan informasi strategis. Bab ini akan mengelaborasi secara komprehensif evolusi teknologi penyimpanan, mulai dari konsep fundamental relasional, penguasaan bahasa kueri SQL, penjinakan gelombang *Big Data*, perancangan *Data Warehouse* untuk analitik tingkat tinggi, hingga pemberlakuan tata kelola (*Data Governance*)

demi menjamin kepatuhan hukum perlindungan privasi. Penguasaan atas ranah manajemen ini adalah prasyarat mutlak bagi siapa pun yang ingin merancang sistem pengambil keputusan otomatis berbasis bukti faktual.

A. Konsep Basis Data

Secara fundamental, basis data (*database*) bukanlah sekadar kumpulan angka dan teks yang ditumpuk secara pasif di dalam sebuah media penyimpanan cakram keras. Ia adalah sebuah representasi digital yang terstruktur, sistematis, dan termodelkan dari sebuah dunia nyata, yang dirancang sedemikian rupa untuk menyimpan, memperbarui, dan mengekstraksi kembali informasi secara sangat efisien (Elmasri & Navathe, 2021). Di masa lampau, organisasi menyimpan catatan menggunakan format fail datar (*flat files*) seperti hamparan lembar kerja *spreadsheet* tradisional. Namun, seiring dengan kompleksitas operasional seperti melacak daftar buku referensi, mengaitkan nomor ISBN, dan memantau royalti penulis secara bersamaan pendekatan *flat file* menjadi sangat primitif dan rentan terhadap anomali struktural (Coronel & Morris, 2022). Kelemahan bawaan seperti redudansi data masif, inkonsistensi penamaan, dan ketiadaan sistem pelindung dari akses serentak, memaksa lahirnya arsitektur yang lebih cerdas.

Gambar 1. Database Development

Sebagai solusi atas kekacauan tersebut, industri rekayasa perangkat lunak memperkenalkan *Database Management System* (DBMS). DBMS bertindak sebagai perangkat lunak maestro, sebuah lapisan perantara (*middleware*) yang memediasi komunikasi antara pengguna manusia (atau aplikasi front-end) dengan kepingan data mentah yang tersimpan di dalam memori peladen fisik (Kleppmann, 2023). Melalui antarmuka DBMS, seorang administrator tidak perlu lagi memikirkan di sektor cakram mana data direkam secara biner. DBMS mengambil alih seluruh tugas tingkat rendah tersebut dan menyajikan sebuah abstraksi visual logis berbentuk tabel-tabel atau dokumen. Fungsionalitas inti DBMS juga mencakup mekanisme pencadangan otomatis (*automated backups*), pemulihan bencana (*disaster recovery*), dan manajemen hak akses otorisasi yang sangat ketat untuk mencegah kebocoran data rahasia institusi (Elmasri & Navathe, 2021).

Paradigma paling dominan yang merajai sejarah komputasi hingga hari ini adalah model Basis Data Relasional (*Relational Database Management System* / RDBMS), yang pertama kali digagas oleh ilmuwan

Edgar F. Codd pada dekade 1970-an (Coronel & Morris, 2022). Model relasional sangat elegan karena ia mengorganisasikan data ke dalam struktur tabular dua dimensi yang terdiri dari baris (rekaman) dan kolom (atribut). Kekuatan magis dari sistem ini terletak pada kemampuannya menciptakan relasi atau hubungan matematis antar-tabel secara dinamis. Tabel-tabel independen ini direkatkan satu sama lain melalui penggunaan Kunci Primer (*Primary Key*) yakni identitas unik sebuah baris data, dan Kunci Asing (*Foreign Key*) yakni kolom yang merujuk pada Kunci Primer di tabel lain. Desain ini memastikan bahwa informasi tidak perlu diulang-ulang secara mubazir di berbagai tempat.

Dalam perancangan RDBMS, ilmuwan data wajib mematuhi kaidah Normalisasi (*Normalization*). Normalisasi adalah sebuah proses audit struktural matematis berlapis (mulai dari 1NF, 2NF, hingga 3NF) yang bertujuan melucuti segala bentuk redudansi anomali dari skema tabel (Elmasri & Navathe, 2021). Jika sebuah entitas bisnis memiliki tabel yang mencampuradukkan data biodata pengguna, alamat domisili, dan riwayat transaksi dalam satu baris panjang, normalisasi akan membongkar tabel raksasa tersebut menjadi tiga tabel terpisah yang saling bertautan logis. Hal ini menjamin bahwa jika seorang pengguna berpindah alamat domisili, administrator hanya perlu mengubah satu buah baris di tabel profil pengguna, tanpa perlu merombak ratusan baris data transaksi yang pernah dilakukan oleh individu tersebut di masa lalu.

Keandalan RDBMS modern dipertahankan melalui garansi kepatuhan terhadap empat prinsip filosofis yang disingkat ACID: *Atomicity*, *Consistency*, *Isolation*, dan *Durability* (Kleppmann, 2023). *Atomicity* menggaransi bahwa sebuah transaksi yang terdiri dari banyak langkah harus

berhasil seluruhnya atau digagalkan sama sekali. *Consistency* memastikan validitas aturan bisnis tidak pernah dilanggar pascatransaksi. *Isolation* memampukan ribuan pengguna menyunting data di detik yang sama tanpa sistem mengalami tabrakan data (*race conditions*). Terakhir, *Durability* berjanji bahwa apabila transaksi telah terekam dan sukses, data tersebut akan tetap permanen terselamatkan meskipun sedetik kemudian peladen pusat mengalami pemadaman listrik total (Coronel & Morris, 2022).

Meskipun tangguh, era internet melahirkan tipe data tak beraturan yang menggoyahkan fondasi RDBMS yang kaku. Menghadapi ledakan lalu lintas e-commerce dan aplikasi media sosial, arsitektur NoSQL (*Not Only SQL*) muncul sebagai antitesis evolusioner (White, 2022). Berbeda dengan RDBMS yang mewajibkan penentuan skema secara kaku sejak hari pertama (*schema-on-write*), NoSQL menawarkan fleksibilitas ekstrem tanpa skema (*schema-less*). Sistem NoSQL dikategorikan menjadi empat aliran besar: Berbasis Dokumen (seperti MongoDB), *Key-Value Stores* (seperti Redis), *Wide-Column Stores* (seperti Cassandra), dan *Graph Databases* (seperti Neo4j untuk memetakan jejaring koneksi relasi antar-entitas) (Kleppmann, 2023).

Evolusi penyimpanan kini mulai bersinggungan secara agresif dengan teknologi *Blockchain* dan penerapan *Smart Contracts*. Desentralisasi basis data (*Distributed Ledger Technology*) ini tidak lagi memiliki satu administrator pusat tunggal, melainkan disalin dan divalidasi oleh ribuan titik simpul (*nodes*) independen di seluruh dunia berbasis algoritma konsensus kriptografi (Ladley, 2022). Dalam konteks validasi aset digital atau keabsahan kontrak bisnis, *blockchain* menciptakan buku besar imutabel di mana data tidak bisa diedit atau dihapus secara sepihak setelah direkam

(*append-only ledger*). Sistem ini membuka era revolusi desentralisasi di dunia digital, menjamin transparansi tanpa memerlukan perantara otorisasi konvensional.

Oleh karenanya, menguasai konsep basis data tidak hanya sekadar belajar teknis menyimpan baris teks, melainkan memahami bagaimana fondasi arsitektur ini membentuk model operasional institusi (Coronel & Morris, 2022). Pemilihan antara RDBMS, NoSQL, atau *Blockchain* akan secara absolut menentukan seberapa cepat sebuah sistem bisa beroperasi, seberapa kebal data terhadap upaya peretasan global, dan seberapa lincah sebuah institusi dapat melakukan manuver analitik. Organisasi yang gagal mendesain arsitektur basis datanya di fase embrio akan berakhir dengan utang teknis (*technical debt*) menumpuk, sistem peladen yang melambat drastis, dan ketidakmampuan beradaptasi dengan disrupsi industri (Kleppmann, 2023).

Assets of Takaza INNOVATIX LABS

B. SQL Dasar

Di tengah pergantian ribuan bahasa pemrograman yang silih berganti kehilangan popularitasnya setiap dekade, *Structured Query Language* (SQL) tetap berdiri kokoh sebagai *lingua franca* atau bahasa universal untuk berkomunikasi dengan basis data relasional sejak era 1970-an (Nield, 2021). Secara konseptual, SQL bukanlah bahasa pemrograman prosedural konvensional seperti Python atau C++, di mana pengguna harus mendikte komputer langkah demi langkah tentang *bagaimana* cara mengeksekusi sebuah perintah. SQL adalah bahasa deklaratif; pengguna cukup mendeskripsikan secara eksplisit *apa* wujud hasil akhir data yang diinginkan, dan mesin optimasi basis data (DBMS) akan memutar otak

secara internal untuk merancang rute matematis paling efisien guna mengambil data tersebut dari peladen (Elmasri & Navathe, 2021).

Konstruksi leksikal SQL dibagi ke dalam beberapa pilar fungsional yang memiliki yurisdiksi tersendiri, dimulai dengan *Data Definition Language* (DDL) (Nield, 2021). DDL bertindak sebagai arsitek perancang fondasi bangunan yang bertugas menciptakan, mengubah, atau menghancurkan kerangka struktur wadah tabel itu sendiri. Perintah CREATE TABLE dieksekusi di hari pertama untuk mendefinisikan kolom, mewajibkan tipe data tertentu (seperti VARCHAR untuk teks panjang, atau INT untuk bilangan bulat), serta menetapkan pengekang primer (*Primary Key constraints*). Seiring berjalannya dinamika organisasi, apabila sebuah kolom baru wajib ditambahkan, maka perintah ALTER TABLE akan disuntikkan. Apabila ada tabel usang yang tidak lagi relevan, sintaks DROP TABLE akan mengeksekusi penghapusan permanen dari antarmuka skema (Coronel & Morris, 2022).

Pilar kedua adalah *Data Manipulation Language* (DML), yang mewakili aktivitas operasional harian, yakni mengelola baris data riil di dalam wadah tabel yang telah dibuat oleh DDL (Elmasri & Navathe, 2021). Rutinitas memasukkan baris data informasi baru dilayani oleh perintah INSERT INTO. Kebutuhan untuk memperbarui data, misalnya mengoreksi entri tipografi yang salah dari basis pengguna, diselesaikan secara presisi melalui klausa UPDATE yang selalu dikawinkan dengan parameter kondisi spesifik. Perintah DELETE disediakan secara hati-hati untuk membersihkan baris data yang cacat. Karena DML bersifat mengubah fakta riil, pengembang mewajibkan pelapisan klausa filter secara teliti; mengeksekusi

sebuah UPDATE tanpa filter akan secara fatal mengubah seluruh baris data dalam satu eksekusi (Nield, 2021).

Bagi seorang ilmuwan data dan analis intelijen bisnis, pilar ketiga yaitu *Data Query Language* (DQL) melalui sintaks SELECT adalah area pertempuran utama yang paling sering dikunjungi (Nield, 2021). Kerangka anatomi kueri dasar selalu dimulai dengan SELECT (mendeklarasikan kolom apa saja yang mau dilihat), diikuti dengan klausa FROM (dari tabel mana asal data tersebut), dan dipersempit ruang lingkupnya dengan klausa WHERE (aturan filter kondisi matematis atau teks). Jika seorang analis ingin melihat khusus data transaksi pada periode tertentu, klausa WHERE bertugas memindai jutaan baris seketika untuk hanya meloloskan hasil yang memenuhi parameter pencarian tersebut, memberikan respons wawasan seketika kepada pengguna.

Setelah ekstraksi mentah berhasil dilakukan, SQL dibekali dengan modul fungsional penyortiran dan agregasi komputasional tingkat menengah yang sangat digdaya (Coronel & Morris, 2022). Klausa ORDER BY difungsikan untuk mengurutkan hasil penyajian, entah itu secara menanjak alfabetis (ASC) atau menurun secara besaran metrik (DESC). Namun, kemampuan SQL yang paling vital untuk rekapitulasi analitik terletak pada fungsi agregasi kolektif seperti SUM() untuk menjumlahkan total keseluruhan, AVG() untuk mencari rata-rata, atau COUNT() untuk menghitung populasi entitas. Klausa sakti bernama GROUP BY kemudian akan mengkategorikan dan memecah agregasi agregat kasar tersebut ke dalam keranjang-keranjang segmen unik, seperti menampilkan perhitungan jumlah total pengguna per wilayah geografis (Nield, 2021).

Assets of Takaza INNOVATIX LABS

Tantangan analitik kompleks jarang sekali bisa dijawab hanya dengan mengandalkan data terisolasi dari satu buah tabel tunggal. Untuk merajut kepingan informasi yang tersebar, SQL memfasilitasi operasi penggabungan yang disebut *JOIN* (Elmasri & Navathe, 2021). Melalui sintaks INNER JOIN, sistem akan mengawinkan dua buah tabel yang berbeda berdasarkan sebuah kolom kunci irisan yang sama (seperti ID Pengguna). Terdapat pula LEFT JOIN yang dirancang untuk mempertahankan seluruh baris data dari tabel utama sebelah kiri secara utuh, sembari tetap menempelkan data dari tabel referensi tambahan jika ada; sangat berguna untuk melakukan audit rekonsiliasi data, mencari entitas mana yang terdaftar namun kehilangan atribut pasangannya di sistem lain (Nield, 2021).

Evolusi SQL tidak berhenti pada komputasi dasar, melainkan terus berekspansi ke wilayah fungsionalitas analitis tinggi (*advanced analytics*). Fitur Kueri Bersarang (*Subqueries* atau *Common Table Expressions* / CTE) memampukan analis untuk menanamkan kueri ekstraksi kecil di dalam sebuah wadah kueri utama raksasa, menciptakan proses pemfilteran berlapis-lapis secara matematis dan modular yang sangat mudah dibaca (Coronel & Morris, 2022). Lebih jauh lagi, fitur Fungsi Jendela (*Window Functions*) memfasilitasi komputasi baris demi baris secara relasional tanpa perlu meruntuhkan dan menyusutkan baris data asli menjadi satu agregat kaku seperti sifat destruktif dari GROUP BY.

Kepiawaian merangkai kueri SQL bukan semata-mata tentang mendapatkan *output* data yang akurat, tetapi tentang efisiensi kalkulasi memori komputasi secara elegan (*query optimization*) (Kleppmann, 2023). Kueri yang berantakan dengan perancangan *Join* yang salah kaprah dapat menyiksa memori RAM peladen hingga berjam-jam akibat anomali produk

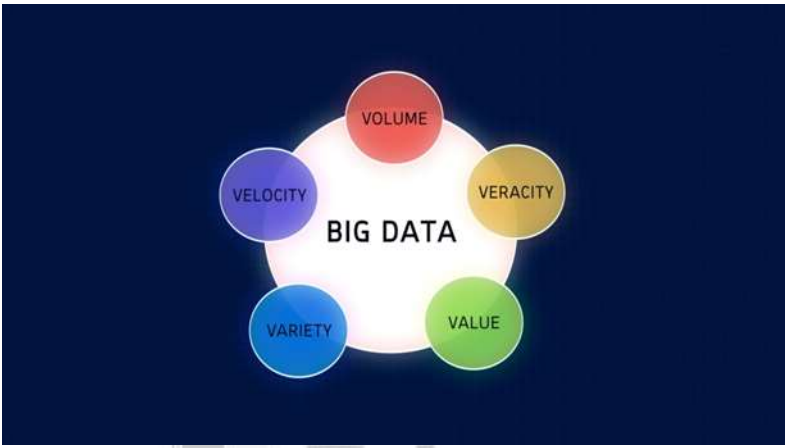
kartesian, sementara kueri logis yang memanfaatkan indeks *B-Tree* yang tepat mampu memuntahkan hasil dari miliaran baris hanya dalam rentang waktu sekian milidetik. Singkatnya, SQL adalah instrumen presisi bedah bagi analis data, memberikan keleluasaan absolut untuk memahat sebongkah besar data kotor menjadi wawasan bisnis yang tervalidasi (Nield, 2021).

C. Pengelolaan Dataset Besar

Istilah *Big Data* sering kali disalahpahami oleh masyarakat awam sebagai semata-mata persoalan ukuran kapasitas cakram (*storage*) yang menggelembung menjadi terabita atau petabita. Dalam ekosistem rekayasa informasi tingkat lanjut, *Big Data* didefinisikan secara holistik oleh para pakar melalui parameter dimensi fundamental yang populer dengan sebutan 3V: *Volume* (ukuran kuantitas fisik yang luar biasa masif), *Velocity* (kecepatan aliran penciptaan data yang terjadi secara seketika atau *real-time stream*), dan *Variety* (keberagaman wujud format yang kacau, mulai dari fail teks, sensor log, hingga multimedia) (White, 2022). Mengelola ketiganya secara simultan melampaui kemampuan intelektual dari RDBMS tradisional manapun.

Ketika entitas bisnis berhadapan dengan fenomena *Big Data*, arsitektur teknologi konvensional mengalami sebuah kebuntuan paradigmatik yang disebut limitasi penskalaan vertikal (*vertical scaling limitations*) (Kleppmann, 2023). Secara historis, saat tumpukan data sebuah peladen penuh, langkah yang diambil adalah menanamkan prosesor yang lebih kuat atau memori yang lebih massif sebuah solusi brutal secara finansial dan memiliki batas fisika yang tak bisa dilewati. Pengelolaan *Big Data* meruntuhkan pendekatan tersebut dan mengadopsi prinsip penskalaan

horizontal (*horizontal scaling*). Arsitektur modern menautkan ribuan peladen kelas pekerja menengah (*commodity hardware*) menjadi satu klaster terdistribusi yang sangat toleran terhadap kerusakan perangkat (*fault-tolerant*) (White, 2022).



Gambar 1. Pengelolaan Dataset Besar

Revolusi besar pertama dalam pengelolaan ekosistem klaster horizontal ini dipicu oleh kelahiran proyek *Apache Hadoop* (White, 2022). Arsitektur ini memperkenalkan *Hadoop Distributed File System* (HDFS), sebuah mekanisme brilian di mana sebuah fail raksasa dicincang secara matematis menjadi puluhan kepingan blok berukuran seragam. Kepingan-kepingan tersebut lantas didistribusikan secara acak ke penjuru klaster peladen, dan secara simultan sistem akan mereplikasi salinan gandanya untuk berjaga-jaga jika sewaktu-waktu salah satu peladen mengalami malfungsi fisis. Pada infrastruktur HDFS ini, data didistribusikan secara desentralistik namun dikelola secara terpusat oleh *NameNode* (Kleppmann, 2023).

Untuk mengolah fail cacahan yang tersebar di HDFS tersebut, diperkenalkanlah paradigma komputasi paralel *MapReduce* (White, 2022). Dibandingkan dengan sistem lama yang memaksakan seluruh data yang tersebar ditarik berpindah menuju satu mesin pemrosesan pusat, *MapReduce* membalik logikanya: program algoritmanyalah yang dikirim mendatangi masing-masing simpul peladen untuk melakukan komputasi lokal (*Map*), lalu merangkum hasil akhirnya secara kolektif (*Reduce*). Meskipun sangat tangguh secara arsitektur dan anti-gagal, *MapReduce* klasik bekerja dengan ritme moderat karena sangat mengandalkan sirkulasi tulis-baca pada keping piringan keras mekanik (*disk I/O*) di setiap tahapan tugasnya (Kleppmann, 2023).

Kelemahan latensi baca-tulis tersebut melahirkan generasi revolusioner berikutnya, yaitu mesin komputasi *Apache Spark* (White, 2022). Mengungguli pendahulunya, Spark menyuntikkan paradigma pengolahan data murni di dalam lautan memori akses acak (*in-memory processing paradigm*). Dengan memproyeksikan seluruh beban operasional komputasi ke dalam ruang RAM klaster, Spark mampu melakukan operasi iteratif, termasuk menjalankan skrip algoritma pembelajaran mesin tingkat tinggi, hingga berlipat kali lebih cepat melampaui arsitektur MapReduce lawas (Kleppmann, 2023). Dalam lanskap komputasi modern, Spark adalah instrumen mutlak untuk merajut saluran pipa pengolahan arus data masif.

Terkait wujud penampungan massal, industri korporasi raksasa bermigrasi meninggalkan paradigma basis data yang terkekang menuju hamparan Danau Data (*Data Lakes*) (Inmon, 2021). *Data Lake* merupakan entitas reservoir berbiaya sangat ekonomis yang mengizinkan pembuangan dan pengendapan aset log data kasar dalam rupa dan wujud natural aslinya,

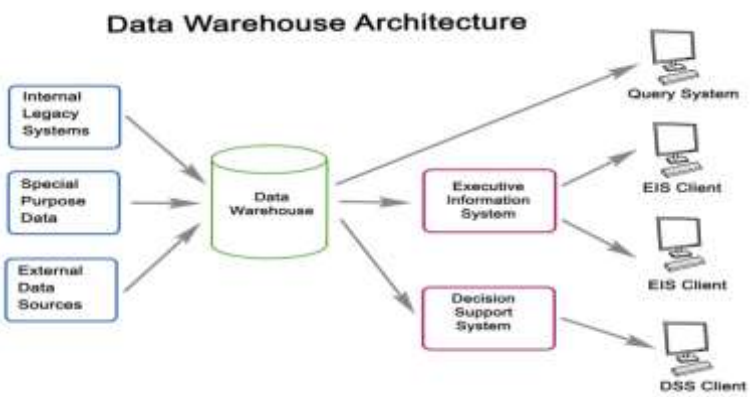
tanpa kewajiban merapikan atau mendefinisikan skema baris-kolom pada hari pertama penyalinan. Kelenturan arsitektur ini membebaskan ilmuwan data untuk menyimpan miliaran himpunan rekaman yang kelak dapat diurai dan distrukturkan kembali kapan pun insting analitik membutuhkannya untuk dieksplorasi (White, 2022).

Dari dimensi optimalisasi ruang dan analitik tingkat tinggi, format penyusunan tabular kuno model baris (seperti format fail CSV murni) telah dielevasi dengan dominasi standar penyimpanan format Kolumnar (*Columnar Storage format*) layaknya kepingan fail *Apache Parquet* (Kleppmann, 2023). Konstruksi format berkas Parquet menyimpan kepingan entitas per blok vertikal. Struktur geometri ini memicu percepatan signifikan saat mesin mengeksekusi kueri agregasi, sebab *reader engine* hanya perlu meluncur mengekstrak spesifik satu lajur kolom target saja tanpa perlu membaca seluruh himpunan variabel kolom tak berguna lainnya, menghasilkan efisiensi komputasi awan yang luar biasa (White, 2022).

Keseluruhan orkestrasi orkestra *Big Data* ini bukan sekadar pamer ketangguhan spesifikasi perangkat keras, melainkan pondasi operasional mutlak guna memuluskan pemodelan prediktif tingkat lanjut (Coronel & Morris, 2022). Rangkaian *dataset* super masif berskala petabita kini dapat disaring dan diringkas secara simultan untuk selanjutnya disuntikkan menuju model peramalan *machine learning*. Penguasaan tata kelola atas dimensi *Volume, Velocity, dan Variety* ini adalah parameter keberhasilan yang akan menentukan keakuratan dan kapabilitas daya respons suatu organisasi analitik terhadap ledakan informasi digital di era kontemporer.

D. Data Warehouse

Apabila *Database* relasional (OLTP - *Online Transaction Processing*) dirancang dengan fokus sempit untuk melayani ribuan operasi tulis dan sunting rekaman transaksional instan per detik secara cepat tanpa hambatan, maka *Data Warehouse* lahir untuk tujuan fungsional yang sepenuhnya berbeda (Kimball & Ross, 2023). *Data Warehouse* (OLAP - *Online Analytical Processing*) adalah sebuah mahakarya arsitektur repositori terpusat, yang dikhususkan murni untuk mengakomodasi beban kueri analitik pencarian fakta historis berskala masif. Sistem OLTP akan seketika mengalami degradasi kinerja jika dipaksa menyajikan laporan tren bisnis multi-tahun, di situlah *Data Warehouse* bertindak sebagai wadah terpisah yang dioptimalkan khusus untuk operasi baca (*read-heavy operations*) miliaran baris (Inmon, 2021).



Gambar 2. Data Warehouse Architecture

Secara taksonomi sejarah pembentukannya, data mentah tidak diekstraksi begitu saja secara instan menuju struktur *Data Warehouse*.

Terdapat sebuah jembatan prapemrosesan higienis mahapenting yang dijuluki saluran ETL (*Extract, Transform, Load*) (Kimball & Ross, 2023). Dalam fase *Extract*, mesin perangkat lunak akan menyedot cuplikan data dari belasan silo sistem sumber operasional yang terpisah-pisah. Pada etape penyucian *Transform*, format penanggalan yang berbeda dikonversi menjadi standar tunggal, anomali dibersihkan, dan konflik redudansi entitas diselesaikan. Terakhir, pada stasiun *Load*, leburan data yang sudah termodifikasi dan terintegrasi tersebut dipompakan masuk secara periodik ke dalam struktur skema dimensional *Data Warehouse* pusat (Inmon, 2021).

Untuk mewujudkan struktur baca secepat kilat tersebut, para arsitek menanggalkan doktrin normalisasi kaku ala RDBMS 3NF, beralih pada penerapan filosofi Pemodelan Dimensional (*Dimensional Modeling*) yang dipopulerkan melalui konsep Skema Bintang (*Star Schema*) (Kimball & Ross, 2023). Di titik pusat orbit tata surya skema ini bertengger sebuah Tabel Fakta (*Fact Table*) yang memuat metrik transaksional bernilai agregat matematis objektif, seperti baris angka kuantitas atau kumulasi nominal transaksi. Mengelilingi tabel fakta tersebut adalah konstelasi Tabel Dimensi (*Dimension Tables*) yang menyimpan deskripsi atribut kualitatif untuk konteks pemfilteran; seperti dimensi profil wilayah, dimensi demografi, atau dimensi kalender hierarki waktu (Coronel & Morris, 2022).

Terkadang, sebuah korporasi raksasa menilai *Data Warehouse* tunggal terlampau besar, kompleks, dan pelan adaptasinya untuk setiap divisi operasional teknis (Inmon, 2021). Sebagai kompromi struktural, arsitek data membangun miniatur analitik terdesentralisasi yang dikenal sebagai *Data Mart*. *Data Mart* adalah sub-fragmen atau irisan modular dari entitas *Data Warehouse* yang difokuskan pada pemenuhan matriks metrik eksklusif milik

satu departemen spesifik. Sebagai contoh, departemen pemasaran digital dapat memiliki *Data Mart* tersendiri yang dioptimalkan struktur kolomnya hanya untuk melacak metrik kampanye konversi iklan tanpa perlu memuat data logistik pergudangan, mempercepat waktu respons kueri departemen tersebut (Kimball & Ross, 2023).

Memasuki dekade pilar komputasi awan (*cloud computing*), konsep fundamental penyedia layanan telah merombak total doktrin monolitik konvensional. Inovasi arsitektural kontemporer memperkenalkan Pemisahan Kapasitas Komputasi CPU dan Kapasitas Wadah Penyimpanan *Disk Storage* (*Separation of Storage and Compute*) (Kleppmann, 2023). Di era on-premise, pelebaran kapasitas penyimpanan memaksa pembelian perangkat keras secara utuh. Kini, di platform seperti Snowflake atau Google BigQuery, biaya retensi volume penyimpanan terlepas secara operasional dari ketersediaan mesin komputasi; memungkinkan perusahaan membayar daya komputasi ekstrem secara elastis hanya pada saat kueri agregasi sedang dieksekusi (Coronel & Morris, 2022).

Evolusi arsitektur awan ini juga memicu fenomena pergeseran aliran metodologi pipa integrasi dari sistem ETL konvensional, bertransformasi menuju rute pipa ELT (*Extract, Load, Transform*) (Inmon, 2021). Dalam aliran ELT modern, organisasi memompa muatan data mentah secara masif masuk menembus *Data Warehouse* awan (*Load*), barulah menyuruh mesin *warehouse* awan raksasa tersebut melakukan perombakan skema dan pencucian format (*Transform*) secara internal. Pendekatan ini secara agresif memanfaatkan kekuatan komputasi paralel dan skalabilitas platform awan untuk mereduksi hambatan proses transformasi data yang secara historis sering menjadi kemacetan antrean (Kimball & Ross, 2023).

Pada puncak rantai piramida pemanfaatan nilainya, *Data Warehouse* berfungsi sebagai jantung mekanis yang memompa sirkulasi metrik operasional secara stabil menuju berbagai peranti antarmuka visual dasbor *Business Intelligence* (BI) (Coronel & Morris, 2022). Dasbor interaktif tersebut menyajikan visualisasi grafis analitik tingkat eksekutif. Apabila para analis mampu melihat fluktuasi penurunan metrik performa dalam hitungan detik saat peninjauan berkala, hal itu sepenuhnya bertumpu pada stabilitas relasi desain OLAP dan kematangan arsitektur skema dimensi di dalam fondasi *Data Warehouse* (Kimball & Ross, 2023).

Tanpa disiplin perancangan *Data Warehouse* yang sistematis, sebuah organisasi analitik hanya akan memiliki serakan *spreadsheet* dan laporan basis data harian yang tak selaras (Inmon, 2021). Kesatuan arsitektur ini memfasilitasi rekaman perbandingan periode dari tahun ke tahun secara andal. Menguasai arsitektur pergudangan data modern adalah kompetensi sentral untuk mentransisikan entitas organisasi dari sekadar entitas penyimpan rekaman menjadi entitas yang dikendalikan murni oleh wawasan intelijen data (*data-driven intelligence*) secara komprehensif.

E. Data Governance

Di tengah gegap gempita pengerukan data analitik berskala besar, aspek *Data Governance* atau tata kelola data sering kali dikesampingkan sebagai lapis birokrasi legalitas yang memperlambat laju eksplorasi tim *data science* (Ladley, 2022). Namun, pandangan rabun tersebut adalah jalan tol menuju potensi bencana operasional dan reputasi institusional. *Data Governance* sesungguhnya adalah sebuah kerangka doktrin operasional komprehensif yang menjembatani interaksi harmonis antara subjek entitas

Orang (*People*), Prosedur Standar Operasional (*Process*), serta Eksekusi Instrumen Teknologi (*Technology*). Tata kelola bertindak sebagai fondasi yurisprudensi internal yang menjamin kemurnian kualitas, ketersediaan, dan keamanan utilitas aset data (Coronel & Morris, 2022).

Konstitusi struktural tata kelola data diinisiasi dengan mendefinisikan pembagian kewenangan tanggung jawab kasta matriks secara presisi eksplisit (Ladley, 2022). Ekosistem ini lazimnya dikomandoi oleh peran lapis pimpinan fungsional yang dinamakan Pemilik Data (*Data Owners*), yang memegang mandat persetujuan strategis atas sebuah domain data spesifik. Di tingkat operasional teknis, mereka mendelegasikan pengawasan harian kepada *Data Stewards*, yaitu praktisi spesialis hibrida yang diamanatkan menjamin glosarium definisi makna atribut metrik tersebut selalu seragam, konsisten, dan dipatuhi oleh seluruh lintas departemen operasional (Coronel & Morris, 2022).

Assets of Pakasa

Diagram concept with Data Governance text and keywords. EPS 10. Sumber: Shutterstock

Fokus teknis dan krusial pertama dari komite tata kelola adalah penciptaan pilar Manajemen Master Data (*Master Data Management* / MDM) (Ladley, 2022). MDM berupaya mengonsolidasikan referensi universal untuk mendirikan satu pilar Sumber Kebenaran Tunggal (*Single Source of Truth*) bagi entitas-entitas terpenting penggerak roda sistem operasi organisasi, seperti nomor unik pengguna atau identitas aset utama. Tanpa kedisiplinan pemeliharaan referensi master data yang terpusat, segenap divisi organisasi akan rentan terhadap perselisihan metrik pelaporan analitik, di mana setiap departemen menggunakan referensi basis acuan data yang bersilangan atau mengalami duplikasi historis tak terdeteksi.

Tata kelola juga bertanggung jawab penuh merumuskan standardisasi kontrol untuk kerangka *Data Quality Management* (Ladley, 2022). Ia menggariskan metrik pengujian validitas dalam jalur pipa integrasi sistem; menetapkan ambang batas toleransi kesalahan (*error tolerance thresholds*) untuk ketiadaan isi sel dalam repositori *dataset*. Modul kualitas ini berfungsi sebagai penjaga gerbang yang memutus mata rantai penyakit mal-input data sejak dari sistem pencatatan sumber terdepan, guna memastikan kebenaran kalkulasi statistik algoritma analitik di rantai ujung (Coronel & Morris, 2022).

Dimensi regulasi yang paling mendesak di era digital modern dari pelaksanaan tata kelola ini adalah kepatuhan mutlak secara hukum terhadap privasi data pribadi (*Data Privacy & Compliance Regulations*) (Kleppmann, 2023). Dengan pemberlakuan perundangan pengamanan privasi global dan nasional, kerangka tata kelola dituntut memformulasikan aturan penganoniman sandi otomatis (*data anonymization & masking*). Hak akses visibilitas kolom informasi sensitif demografi *PII (Personally Identifiable*

Information) diwajibkan tertutup atau terenkripsi lapis ganda, membatasi eksposur visibilitas hanya pada pihak-pihak berotoritas sah secara hukum atau dengan tujuan spesifik audit (Ladley, 2022).

Di sektor industri yang mengandalkan legalitas kekayaan intelektual digital atau perikatan digital *smart contracts*, penerapan *Data Governance* tidak hanya terbatas pada privasi konsumen (Ladley, 2022). Tata kelola turut mencakup penjagaan integritas ketertelusuran dokumen (*document provenance*) dan pembatasan akses amandemen rekaman log digital. Kerangka ini harus menancapkan prosedur audit persetujuan otorisasi berganda untuk modifikasi aset bernilai legal demi mencegah intrusi sabotase data perikatan yang rentan terkonfrontasi dari eksploitasi peretasan ancaman eksternal maupun internal organisasi (Coronel & Morris, 2022).

Setiap napas perpindahan rute integrasi akses data, modifikasi, hingga tindakan penghancuran struktur atribut basis data tabel, secara prosedural diwajibkan terekam abadi dalam mekanisme pencatatan riwayat silsilah (*Data Lineage*) dan Log Audit (Kleppmann, 2023). Kemampuan pelacakan keturunan silsilah informasi ini bersifat fundamental untuk memfasilitasi penelusuran asal muasal anomali data, membedah alur waktu spesifik, dan memvalidasi rute aliran penggabungan transformasi variabel sejak dari hulu sumber aplikasi hingga bermuara ke dalam laporan dasbor akhir (Ladley, 2022).

Pada konklusi muaranya, perwujudan ekosistem fondasi tata kelola data mustahil tercipta hanya dengan menerbitkan lembar kebijakan prosedur teknis di atas kertas (Ladley, 2022). *Data Governance* harus meresap mentransformasi literasi kesadaran fundamental setiap anggota institusi. Saat seluruh unit elemen pengumpul log metrik lapangan mempraktikkan

penginputan formulir secara bertanggung jawab untuk menjaga kualitas kebenaran keutuhan data, barulah sistem manajemen dan algoritma analitik buatan organisasi sanggup merumuskan intelijen peramalan ekstrapolatif dengan derajat validitas akademis dan utilitas praktis paripurna (Coronel & Morris, 2022).

F. Studi kasus

Integrasi Sistem untuk Penerbitan Buku Kolaborasi dan Pemasaran Digital

Latar Belakang Operasional: Sebuah konsorsium penerbitan intelektual akademik terkemuka yang diampu oleh berbagai pengurus proyek kolaboratif sering kali menghadapi hambatan skalabilitas struktural operasional. Konsorsium ini sangat terfokus mengelola penerbitan ratusan judul buku kolaborasi antar-kampus regional dan karya pengubah bahan pengajaran menjadi buku referensi ber-HKI (Hak Kekayaan Intelektual). Dengan semakin dominannya arus transisi komersialisasi pemasaran modern, divisi operasional periklanan dituntut untuk tidak lagi hanya memasarkan produk intelektual ini di katalog konservatif, melainkan menyebarkannya via penetrasi komersial promosi e-commerce afiliasi digital yang menyasar khalayak akademis di pelbagai etalase platform interaktif sosial yang dinamis (Kleppmann, 2023).

Identifikasi Penyakit Manajemen Data: Problem kritis institusi ini adalah kelumpuhan arsitektur akibat fragmentasi penyimpanan operasional (silo data) (Ladley, 2022). Basis data relasional internal HKI dan registrasi penerbitan mereka tercampur baur pada fail lembar kerja tak terstruktur; memuat tumpang tindih ribuan profil kepakaran akademis penulis

kolaborasi, hierarki struktural kepangkatan edukator, hingga catatan penetapan status royalti. Di sisi seberang, jutaan log keping data laporan pengiklanan lalu lintas jaring navigasi antarmuka pemasaran digital bertumpuk di media penyimpanan awan terdistribusi secara parsial tanpa format konsisten. Tim analitik sistem gagal menemukan korelasi metrik antara gelombang tren promosi penelusuran "Pemasaran Digital" di media e-commerce sosial dengan rasio kesuksesan permintaan cetak ulang buku-buku referensi (Inmon, 2021).

Implementasi Tahap Pertama: RDBMS untuk Katalog dan HKI: Tim rekayasa data segera mengeksekusi proses normalisasi sistem basis data skema relasional terpusat SQL untuk jantung administrasi pendaftaran dan perlindungan rekaman aset (Elmasri & Navathe, 2021). Tabel masif yang kusut dibelah secara prosedural menjadi relasi entitas terpisah: (1) Tabel Entitas Profil Akademis Penulis, (2) Tabel Register Master Atribut ISBN/HKI, dan (3) Tabel Pertautan Perikatan Kontrak Persentase Hak Distribusi Royalti. Kini, jika terdapat urgensi kebutuhan koreksi identitas atau titel hierarki penulis (misalnya pembaruan *record* melalui DML klausa UPDATE), operasi mutasi cukup dieksekusi di satu titik sel koordinat tabel referensi primer, memutus ancaman anomali pembaruan asinkron (*update anomaly*) di seluruh ekosistem penerbitan (Coronel & Morris, 2022).

Implementasi Tahap Kedua: Ekskavasi SQL untuk Otomasi Bisnis: Dengan tegaknya fondasi relasional yang kokoh, tim fungsional operasional kini mengeksploitasi fleksibilitas kueri SQL dasar terstruktur guna merilis modul otomatisasi pembagian persentase hasil bagi-komisi seluruh kolektif penulis lintas region (Nield, 2021). Melalui implementasi fungsionalitas deklaratif gabungan relasional tingkat lanjut INNER JOIN

bertingkat antara agregasi data *Tabel Penetapan Harga Pokok*, *Tabel Kalkulasi Riwayat Kuantitas Penjualan Bersih*, serta *Tabel Metrik Margin Ketetapan Royalti*, sistem konsorsium kini sanggup merekapitulasi transparansi dasbor ringkasan omzet dan pendistribusian kalkulatif alokasi moneter secara otomatis dari pangkalan basis datanya secara *real-time* (Elmasri & Navathe, 2021).

Implementasi Tahap Ketiga: Pipa Injeksi Big Data Eksternal: Guna menjaring dan mencerna parameter gejolak arus ulasan sentimen teks interaksi massa pembeli di ranah ekosistem pemasaran media sosial digital yang bervolume masif, fondasi RDBMS diputuskan tak relevan mengakomodasi beban *velocity* dan *variety* (White, 2022). Institusi mendirikan arsitektur platform *Data Lake* komputasi awan. Berbagai log data nir-struktur (*unstructured data*) berisikan umpan balik teks testimoni ditarik dari API eksternal dan dialirkan langsung menuju *Data Lake*, kemudian dijinakkan dengan integrasi orkestrasi paralel mesin *Apache Spark* guna membersihkan anomali deret abjad teks bising (spam) menjadi matriks prediktor opini analitis (Kleppmann, 2023).

Implementasi Tahap Keempat: Konstruksi Otak Data Warehouse: Bahan galian konversi rekaman transaksi operasional terstruktur internal dari DBMS direkatkan dengan hasil saringan indikator opini agregat sentimen big data eksternal di dalam lingkungan arsitektur *Cloud Data Warehouse* melalui pipa terotomasi ELT (Kimball & Ross, 2023). Arsitektur dirancang menggunakan kerangka pemodelan dimensi *Star Schema*; di mana *Tabel Fakta Penjualan* terkonsolidasi dikelilingi kerangka *Tabel Dimensi Atribut Penulis*, *Tabel Dimensi Parameter Gelombang Waktu Kalender Promosi*, dan *Tabel Dimensi Jejak Kanal Afiliasi Distribusi E-*

Commerce. Modul ini bertindak sebagai otak pelaporan super-cepat lintas sistem yang melayani mesin intelijen dasbor bisnis harian (Inmon, 2021).

Implementasi Tahap Kelima: Kepatuhan Tata Kelola Siber Digital: Sebagai langkah memitigasi bahaya pelanggaran perdata etika kepatuhan proteksi perlindungan identitas pribadi (PII) terkait data transaksi komersial pelanggan dan identitas privasi profesor penulis, struktur eksekutif membentuk gugus tugas *Data Governance* (Ladley, 2022). Tata kelola ini mengimplementasikan mekanisme enkripsi limitasi visibilitas kriptografi (*masking*) atas kolom sensitif pajak serta informasi otentikasi identitas bank penerima royalti di basis pangkalan penyimpanan. Penerapan pencatatan prosedur *Data Lineage* aktif menancapkan segel lacak balik riwayat jejak audit modifikasi metadata HKI guna mencegah intrusi amandemen ilegal di dalam pangkalan data pusat (Coronel & Morris, 2022).

Konklusi Transformasional: Dalam tempo setengah tahun operasional, modernisasi restrukturisasi infrastruktur data berskala komprehensif ini mengubah haluan kapal penerbit akademik tersebut (Kimball & Ross, 2023). Pengambilan keputusan logistik prapencetakan buku referensi tak lagi dijalankan dengan insting heuristik pasif, melainkan dipandu penuh oleh korelasi agregat metrik *Data Warehouse*. Sistem dasbor sanggup menyorot sinyal korelasi sentimen publik positif di kanal ulasan promosi digital dan secara proaktif mengantisipasi proyeksi gelombang peningkatan pemesanan harian logistik resi e-commerce sekian hari sebelum lonjakan volume pemesanan cetak mewujud di sistem keranjang transaksi operasional (Ladley, 2022).

BAB VII
EXPLORATORY DATA ANALYSIS (EDA)

Setelah infrastruktur penyimpanan data (Basis Data) berhasil dibangun dan diorkestrasi pada bab sebelumnya, tantangan intelektual berikutnya bagi seorang ilmuwan data adalah menembus lapisan kekakuan data mentah menjadi wawasan yang memiliki daya gerak (*actionable insights*). Data yang tersimpan rapi di dalam *Data Warehouse* hanyalah sekumpulan bit dan byte yang bisu jika tidak diberikan pertanyaan yang tepat dan metode eksplorasi yang tajam. Bab ini berfokus pada seni dan sains Exploratory Data Analysis (EDA) sebuah fase kritis di mana kita berperan sebagai detektif data yang berusaha mengungkap pola tersembunyi, anomali yang mencurigakan, serta hubungan antar-variabel sebelum melangkah lebih jauh ke fase pemodelan prediktif yang lebih kompleks (Tukey, 1977).

EDA bukan sekadar ritual rutin sebelum membangun model *Machine Learning*; ia adalah fondasi pemahaman domain. Tanpa melakukan EDA secara mendalam, praktisi data sering kali terjebak dalam bias asumsi menganggap data mengikuti distribusi normal padahal kenyataannya tidak, atau gagal menyadari keberadaan *outlier* yang dapat merusak akurasi model nantinya. Melalui bab ini, kita akan membedah metodologi eksplorasi dari yang paling sederhana (univariat) hingga yang melibatkan interaksi variabel yang sangat kompleks (multivariat). Kita akan mempelajari bagaimana visualisasi bukan hanya sekadar estetika, melainkan instrumen kognitif

untuk memvalidasi hipotesis awal sebelum kita melangkah menuju kesimpulan yang lebih formal (Wickham, 2014).

A. Konsep EDA

Exploratory Data Analysis (EDA) adalah pendekatan analitis untuk mengeksplorasi kumpulan data guna memahami karakteristik utamanya, sering kali menggunakan metode visual. Dicetuskan pertama kali oleh John Tukey, EDA menekankan pada pentingnya "melihat" data sebelum melakukan uji statistik formal atau pemodelan (Tukey, 1977). Dalam EDA, kita tidak berangkat dengan hipotesis yang kaku atau kaku untuk dibuktikan (*confirmatory analysis*), melainkan dengan pola pikir terbuka untuk menemukan apa yang "diperintahkan" oleh data kepada kita. Ini adalah proses iteratif; sering kali, hasil eksplorasi pertama akan memicu pertanyaan baru yang menuntun kita kembali ke proses pembersihan atau pengayaan data.

Assets of Takaza INNOVATIX LABS

Tujuan utama dari EDA adalah deteksi anomali, pengujian asumsi, dan pemilihan model yang tepat. Dengan EDA, kita bisa mendeteksi nilai-nilai ekstrem (*outliers*) yang mungkin merupakan kesalahan input atau, sebaliknya, merupakan penemuan berharga (seperti pola kecurangan transaksi). Selain itu, EDA membantu kita menguji apakah data yang kita miliki memenuhi asumsi-asumsi statistik dasar seperti normalitas atau homoskedastisitas. Jika asumsi ini dilanggar, kita tahu bahwa kita harus beralih ke metode statistik non-parametrik atau melakukan transformasi data terlebih dahulu sebelum memasukkannya ke dalam algoritma *Machine Learning* (James et al., 2021).

Proses EDA juga mencakup pembersihan dan transformasi data secara tidak langsung. Selama eksplorasi, sering kali kita menemukan data yang hilang (*missing values*) atau ketidakkonsistenan format yang luput dari tahap *data cleaning* awal. EDA memungkinkan kita untuk melihat distribusi data dengan lebih nyata, misalnya melalui *boxplots* atau histogram, yang secara langsung memperlihatkan apakah data kita memiliki *skewness* (kemiringan) yang ekstrem. Visualisasi dalam EDA bertindak sebagai "cermin" yang memantulkan kondisi riil data di lapangan, memberikan gambaran yang jauh lebih intuitif dibandingkan hanya mengandalkan nilai *summary statistics* semata (Wickham, 2014).

Dalam konteks manajemen data korporasi, EDA berfungsi sebagai fase "inspeksi kualitas" sebelum aset data dideploy ke dalam sistem operasional atau dasbor eksekutif. Jika seorang analis data tidak melakukan EDA, ia berisiko membangun sistem pelaporan berbasis metrik yang salah atau menyesatkan. EDA memaksa analis untuk "berkenalan" dengan data memahami konteks bisnis di balik angka-angka tersebut. Apakah lonjakan penjualan terjadi karena strategi pemasaran yang sukses, atau karena adanya anomali sistemik dalam pencatatan transaksi? Pertanyaan-pertanyaan seperti inilah yang harus dijawab dalam fase EDA agar pengambilan keputusan berbasis data menjadi valid.

Selain aspek teknis, EDA juga melatih intuisi analitis seorang data scientist. Dengan sering melakukan eksplorasi, analis akan terbiasa mengenali pola-pola yang tidak wajar hanya dengan melihat sebaran data secara sekilas. Ini adalah keterampilan taktis yang sulit diajarkan hanya melalui buku teks. Penguasaan EDA mencakup pemahaman mendalam tentang statistik deskriptif dan keterampilan menggunakan perangkat lunak

visualisasi (seperti Matplotlib, Seaborn, Tableau, atau PowerBI). Analis yang mahir EDA tahu kapan harus menggunakan grafik batang untuk kategori, dan kapan harus beralih ke *scatter plot* untuk melihat korelasi antar-variabel numerik (James et al., 2021).

EDA juga memainkan peran krusial dalam rekayasa fitur (*feature engineering*). Sering kali, hubungan antar-variabel tidak bersifat linear. Melalui eksplorasi visual, kita mungkin menemukan bahwa logaritma dari sebuah variabel memiliki korelasi yang jauh lebih kuat dengan variabel target dibandingkan nilai aslinya. Pengetahuan ini sangat berharga saat kita membangun model prediktif. Tanpa EDA, kita mungkin melewatkan transformasi fitur yang berpotensi meningkatkan performa model secara signifikan. Dengan demikian, EDA bukan hanya tahap awal, melainkan fase yang menentukan kesuksesan seluruh siklus hidup proyek data.

Pentingnya EDA dalam organisasi data-driven tidak bisa diremehkan. Ia adalah garis pertahanan pertama melawan "sampah masuk, sampah keluar" (*Garbage In, Garbage Out*). Dalam ekosistem *Big Data*, di mana data sering kali tidak terstruktur dan bervolume masif, EDA membantu kita melakukan *sampling* secara cerdas untuk mendapatkan wawasan cepat. Dengan teknik *downsampling* atau *feature selection* berdasarkan eksplorasi, kita bisa memahami karakteristik data besar tanpa harus memproses seluruh dataset secara penuh setiap kali melakukan eksperimen, yang secara signifikan menghemat biaya komputasi (Tukey, 1977).

Akhirnya, EDA adalah tentang komunikasi. Hasil eksplorasi sering kali menjadi materi utama saat mempresentasikan temuan kepada pemangku kepentingan (*stakeholders*) non-teknis. Visualisasi yang menarik dan intuitif dari fase EDA dapat menceritakan kisah di balik data dengan jauh lebih

efektif dibandingkan tabel-tabel angka yang membosankan. Melalui EDA, kita tidak hanya menemukan kebenaran statistik, tetapi juga mengemas kebenaran tersebut menjadi narasi bisnis yang meyakinkan, yang pada akhirnya memicu aksi nyata untuk perbaikan performa perusahaan (Wickham, 2014).

B. Analisis Univariate

Analisis univariat adalah bentuk analisis data paling dasar di mana kita mengamati satu variabel pada satu waktu. Tujuannya adalah untuk memahami distribusi, tendensi sentral, dan penyebaran variabel tersebut tanpa mempedulikan hubungannya dengan variabel lain. Dalam konteks data numerik, analisis univariat berfokus pada nilai rata-rata (*mean*), median, modus, serta ukuran penyebaran seperti standar deviasi, varians, rentang (*range*), dan *interquartile range* (IQR) (James et al., 2021).

Visualisasi utama untuk data numerik dalam analisis univariat adalah histogram dan *box plot*. Histogram memungkinkan kita melihat bentuk distribusi data apakah ia berdistribusi normal, *skewed* ke kanan atau kiri, atau multimodal (memiliki lebih dari satu puncak). *Box plot*, di sisi lain, sangat efektif untuk mengidentifikasi nilai-nilai pencilan (*outliers*) yang mungkin tersembunyi. Keberadaan *outlier* yang signifikan sering kali menjadi sinyal bagi analis untuk kembali memeriksa kualitas data atau mengevaluasi apakah *outlier* tersebut adalah kesalahan sistem atau fenomena nyata yang harus diperlakukan secara khusus (Tukey, 1977).

Untuk data kategorikal, analisis univariat dilakukan melalui tabel frekuensi dan visualisasi seperti *bar chart* atau *pie chart*. Di sini, kita mencari tahu seberapa sering setiap kategori muncul dalam dataset. Apakah

distribusi kategori seimbang (*balanced*) atau terjadi ketimpangan ekstrem (*imbalanced*)? Dalam proyek klasifikasi *Machine Learning*, analisis univariat pada variabel target sangat krusial; dataset yang sangat tidak seimbang (misalnya, 99% kelas A dan 1% kelas B) akan memerlukan teknik *resampling* khusus seperti SMOTE untuk memastikan model tidak bias ke kelas mayoritas (Wickham, 2014).

Analisis univariat juga mencakup pemeriksaan terhadap *missing values* pada setiap variabel. Kita harus mengetahui persentase data yang hilang pada setiap kolom. Apakah datanya hilang secara acak (*Missing Completely at Random*), atau ada pola tertentu yang menyebabkan data hilang? Misalnya, dalam data survei, responden mungkin enggan menjawab pertanyaan tentang pendapatan, sehingga menciptakan pola *missing* yang sistematis. Memahami distribusi *missing values* secara univariat adalah langkah pertama sebelum memutuskan metode imputasi yang tepat apakah menggunakan mean, median, atau metode lain (James et al., 2021).

Variabel waktu (*time series*) juga memerlukan analisis univariat yang spesifik, yaitu melihat tren, musiman (*seasonality*), dan siklus. Meskipun hanya melihat satu variabel, kita memetakannya terhadap waktu untuk melihat apakah ada pola pertumbuhan atau penurunan yang stabil. Jika kita melihat fluktuasi yang tidak menentu, mungkin kita perlu melakukan dekomposisi data untuk memisahkan antara tren jangka panjang dan efek musiman (misalnya, peningkatan penjualan buku setiap semester baru). Pemahaman ini sangat vital sebelum membangun model peramalan (*forecasting*).

Penggunaan ukuran dispersi seperti standar deviasi memberikan gambaran tentang volatilitas data. Jika standar deviasi sangat tinggi, berarti

data tersebut sangat bervariasi dan tidak konsisten. Sebaliknya, standar deviasi rendah menunjukkan stabilitas. Analis harus peka terhadap interpretasi ukuran ini; misalnya, dalam manajemen kualitas buku, deviasi ketebalan buku yang tinggi bisa menunjukkan proses produksi yang tidak presisi. Analisis univariat memberikan alarm awal bagi manajemen untuk melakukan audit proses (Tukey, 1977).

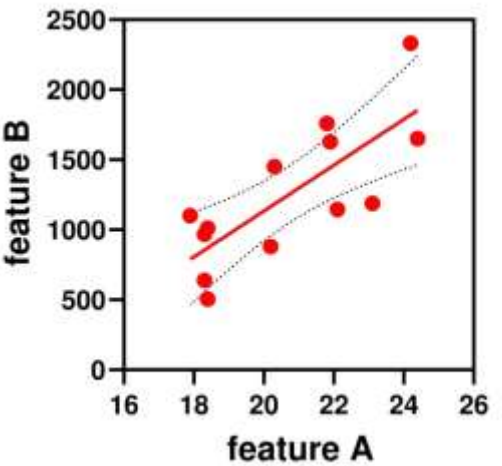
Selain itu, analisis univariat juga mencakup pemeriksaan *skewness* (kemiringan) dan *kurtosis* (keruncingan). Data yang memiliki *skewness* tinggi sering kali memerlukan transformasi logaritmik atau *Box-Cox* agar berdistribusi lebih normal, sehingga model statistik linear bisa bekerja lebih optimal. *Kurtosis* memberi tahu kita tentang keberadaan ekor yang tebal (*heavy tails*), yang berarti ada kemungkinan kejadian ekstrem yang lebih besar daripada yang diprediksi oleh distribusi normal. Memahami profil ini secara univariat adalah prasyarat teknis yang sering diabaikan namun berakibat fatal jika dilewati (James et al., 2021).

Kesimpulannya, analisis univariat adalah fondasi dari setiap proyek analitik. Meskipun sederhana, ia tidak boleh disepelekan karena memberikan profil kesehatan variabel secara individual. Ia adalah langkah pertama "kenalan" dengan data sebelum kita mulai mencampurkan variabel satu dengan yang lain dalam analisis bivariat atau multivariat. Tanpa penguasaan analisis univariat yang kokoh, seorang ilmuwan data akan membangun fondasi bangunan di atas tanah yang tidak stabil, yang akan menyebabkan keruntuhan pada saat fase pemodelan (Wickham, 2014).

C. Analisis Bivariate

Analisis bivariat melangkah satu tingkat lebih jauh dengan mengeksplorasi hubungan antara dua variabel. Tujuannya adalah untuk

menentukan apakah terdapat korelasi atau asosiasi antara variabel independen dan dependen, atau antar variabel independen itu sendiri. Hubungan ini bisa berbentuk linear, non-linear, atau berupa perbedaan signifikan antar kelompok. Analisis ini sangat krusial untuk mengidentifikasi variabel mana yang paling berpengaruh terhadap target yang ingin diprediksi atau dioptimalkan dalam bisnis (James et al., 2021).



Gambar 3. Diagram Pencar

Untuk dua variabel numerik, instrumen visual utama adalah *scatterplot*. *Scatterplot* memungkinkan kita melihat secara langsung pola hubungan: apakah positif (jika X naik, Y naik), negatif (jika X naik, Y turun), atau tidak ada pola sama sekali. Secara statistik, kita mengukurnya menggunakan koefisien korelasi (seperti Pearson untuk hubungan linear atau Spearman untuk hubungan monoton). Penting untuk diingat bahwa korelasi tidak sama dengan kausalitas. Dua variabel bisa bergerak bersamaan, namun bukan berarti satu menyebabkan yang lainnya; mungkin ada variabel ketiga

yang tidak teramati (*confounding variable*) yang menggerakkan keduanya (Tukey, 1977).

Dalam hubungan antara variabel numerik dan kategorikal, kita menggunakan teknik visualisasi yang berbeda. Kita bisa menggunakan *box plot* yang dikelompokkan berdasarkan kategori, atau *violin plot*. Misalnya, jika kita ingin melihat hubungan antara "Tipe Kampus" (kategorikal) dan "Penjualan Buku" (numerik), *box plot* akan memperlihatkan perbandingan distribusi penjualan antar-tipe kampus secara berdampingan. Secara statistik, kita bisa menggunakan uji T (*T-Test*) jika hanya ada dua kategori, atau ANOVA (*Analysis of Variance*) jika terdapat lebih dari dua kategori, untuk melihat apakah perbedaan rata-rata tersebut signifikan secara statistik (James et al., 2021).

Untuk dua variabel kategorikal, kita menggunakan tabel kontingensi (*crosstab*) dan visualisasi seperti *stacked bar chart* atau *heatmap* (dengan uji Chi-Square). Tabel kontingensi menyajikan frekuensi kemunculan pasangan kategori. Misalnya, kita bisa melihat hubungan antara "Wilayah Distribusi" dan "Metode Pembayaran". Apakah pelanggan di wilayah tertentu cenderung lebih menyukai pembayaran digital dibandingkan tunai? Chi-Square test akan membantu kita menentukan apakah perbedaan proporsi ini terjadi karena faktor kebetulan atau memang ada hubungan yang signifikan antar-kategori tersebut (Wickham, 2014).

Analisis bivariat juga membantu kita mendeteksi kolinearitas (*multicollinearity*), yaitu kondisi di mana dua variabel independen memiliki hubungan yang sangat kuat satu sama lain. Dalam model regresi, adanya multikolinearitas yang tinggi antar variabel independen bisa menyebabkan estimasi parameter menjadi tidak stabil dan sulit diinterpretasikan. Oleh

karena itu, melalui analisis bivariat, kita sering kali harus melakukan seleksi fitur memilih salah satu dari variabel yang berkorelasi sangat tinggi dan membuang yang lainnya untuk menyederhanakan model (James et al., 2021).

Pentingnya analisis bivariat juga terletak pada kemampuannya memberikan konteks. Dalam pemasaran, mengetahui bahwa total pengeluaran iklan berkorelasi positif dengan volume penjualan adalah wawasan yang baik. Namun, analisis bivariat memungkinkan kita menggali lebih dalam: apakah korelasi ini konsisten di seluruh segmen pelanggan? Kita bisa melakukan analisis bivariat pada tiap segmen untuk melihat apakah variabel iklan tersebut berperilaku sama. Jika ternyata iklan hanya efektif pada satu segmen, maka strategi pemasaran bisa dibuat lebih personal dan efisien, alih-alih melakukan *blanket campaign* yang mahal namun sia-sia.

Dalam konteks manajemen risiko, analisis bivariat sering digunakan untuk memvalidasi indikator-indikator peringatan dini (*early warning indicators*). Misalnya, hubungan antara keterlambatan pembayaran cicilan dengan skor kredit pelanggan. Jika ada hubungan negatif yang kuat, maka skor kredit menjadi variabel prediktor yang valid untuk risiko gagal bayar. Tanpa analisis bivariat, kita hanya akan melihat data secara terfragmentasi. Analisis bivariat adalah jembatan yang menghubungkan titik-titik data, membentuk garis narasi yang menjelaskan mengapa sesuatu terjadi di dunia nyata (Tukey, 1977).

Singkatnya, analisis bivariat adalah tentang mencari jawaban atas pertanyaan "apakah X mempengaruhi Y?". Ini adalah fase di mana kita mulai membangun pemahaman tentang mekanisme sistem yang kita pelajari. Dengan memahami interaksi antar-dua variabel, kita dapat mulai

mengidentifikasi penggerak utama (*key drivers*) dari hasil bisnis yang kita amati. Namun, kita harus tetap rendah hati dengan hasil analisis ini; dalam dunia yang kompleks, jarang ada fenomena yang hanya dipengaruhi oleh satu variabel saja. Inilah yang menuntun kita pada kebutuhan untuk melakukan analisis yang lebih komprehensif: analisis multivariat (Wickham, 2014).

D. Analisis Multivariate

Analisis multivariat adalah metode statistik dan visualisasi yang melibatkan lebih dari dua variabel secara bersamaan. Dalam dunia nyata, fenomena jarang bersifat univariat atau bivariat; hampir semua hasil bisnis (seperti kepuasan pelanggan atau volume penjualan) dipengaruhi oleh sistem yang kompleks dari banyak variabel yang saling berinteraksi. Analisis multivariat memungkinkan kita untuk memahami interaksi kompleks ini dan memberikan pandangan yang lebih holistik tentang struktur data (James et al., 2021).

Salah satu alat utama dalam analisis multivariat adalah *Pairplot* (atau *Scatterplot Matrix*). *Pairplot* memungkinkan kita untuk melihat korelasi antara semua pasangan variabel numerik dalam satu tampilan visual yang terpadu. Ini sangat berguna untuk mendapatkan gambaran cepat tentang mana variabel yang saling berkaitan kuat dan mana yang tidak. Selain *pairplot, Heatmap* korelasi adalah standar industri untuk memvisualisasikan matriks korelasi, di mana intensitas warna mewakili kekuatan hubungan antara variabel-variabel tersebut (Wickham, 2014).

Selain visualisasi, kita juga menggunakan teknik reduksi dimensi seperti *Principal Component Analysis* (PCA). Ketika kita memiliki dataset

dengan puluhan atau ratusan fitur (misalnya, data perilaku pengguna yang mencakup ribuan aksi klik), mustahil bagi manusia untuk memvisualisasikannya secara langsung. PCA mengubah fitur-fitur yang banyak tersebut menjadi beberapa komponen utama (*principal components*) yang tetap mempertahankan sebagian besar varians data. Ini memungkinkan kita untuk mereduksi kompleksitas data ke dalam dua atau tiga dimensi, sehingga kita bisa memplotnya dan melihat klaster atau pola yang tersembunyi (James et al., 2021).

Analisis multivariat juga mencakup teknik klastering (*clustering*) seperti K-Means atau Hierarchical Clustering. Klastering bukan tentang memprediksi sebuah label, melainkan menemukan pengelompokan alami dalam data. Misalnya, dalam pemasaran buku, kita bisa menggunakan analisis multivariat untuk mengelompokkan pelanggan berdasarkan kombinasi usia, wilayah, jenis buku yang dibeli, dan waktu pembelian. Hasilnya mungkin mengungkap segmen "Mahasiswa yang Membeli Buku di Malam Hari" atau "Akademisi yang Membeli Buku Referensi di Akhir Pekan". Segmentasi ini jauh lebih kaya dibandingkan sekadar membagi pelanggan berdasarkan satu kriteria saja (Tukey, 1977).

Selain itu, analisis multivariat memfasilitasi deteksi interaksi variabel yang tidak terlihat pada analisis bivariat. Misalnya, mungkin variabel "Harga" tidak berpengaruh signifikan pada "Penjualan" secara keseluruhan (analisis bivariat), tetapi ketika kita memasukkan variabel "Promosi" dan "Tipe Wilayah" (analisis multivariat), kita mungkin menemukan bahwa Harga berpengaruh negatif pada Penjualan *hanya* jika tidak ada Promosi di wilayah perkotaan. Inilah yang disebut efek moderasi atau interaksi, yang

hanya bisa ditemukan melalui teknik multivariat seperti regresi berganda (*multiple regression*) (James et al., 2021).

Dalam lingkungan *Big Data*, analisis multivariat juga mencakup penggunaan *Random Forest* atau *Gradient Boosting* untuk menentukan *feature importance*. Algoritma ini secara otomatis mengevaluasi kontribusi setiap variabel terhadap prediksi target dalam konteks kehadiran variabel lainnya. Ini memberikan wawasan tentang fitur mana yang sebenarnya "berharga" dan fitur mana yang hanyalah "noise". Pemahaman ini sangat membantu dalam menyederhanakan model tanpa mengorbankan akurasi, yang merupakan kunci efisiensi dalam sistem *Machine Learning* skala besar (Wickham, 2014).

Tantangan terbesar dalam analisis multivariat adalah interpretasi. Semakin banyak variabel yang terlibat, semakin sulit untuk memberikan interpretasi kausalitas yang sederhana. Oleh karena itu, sangat penting untuk tetap memiliki kerangka berpikir teoritis yang kuat. Jangan hanya "melemparkan" semua variabel ke dalam algoritma multivariat dan berharap mendapatkan jawaban. Analis harus memiliki hipotesis tentang bagaimana variabel-variabel tersebut berinteraksi, dan menggunakan teknik multivariat untuk menguji dan memperkuat hipotesis tersebut (Tukey, 1977).

Secara keseluruhan, analisis multivariat adalah puncak dari eksplorasi data. Ia memberikan gambaran tentang "ekosistem" variabel yang kita pelajari. Dengan menguasai teknik ini, analis dapat beralih dari sekadar mendeskripsikan data menjadi memahami dinamika sistem yang kompleks. Ini adalah kemampuan tingkat tinggi yang membedakan seorang analis pemula dengan seorang ilmuwan data senior kemampuan untuk melihat hutan, bukan sekadar pohon-pohon yang menyusunnya (James et al., 2021).

E. Interpretasi Hasil Analisis

Interpretasi adalah fase paling krusial dalam EDA, di mana angka-angka dan grafik bertransformasi menjadi narasi yang bermakna bagi bisnis. Interpretasi bukanlah sekadar membaca nilai p-value atau koefisien korelasi; ia adalah proses menggabungkan hasil statistik dengan pengetahuan domain (*domain knowledge*). Seorang analis yang mahir statistik tetapi buta terhadap konteks bisnis akan menghasilkan interpretasi yang kering, bahkan bisa menyesatkan (Wickham, 2014).

Langkah pertama dalam interpretasi adalah memastikan relevansi. Apakah temuan yang kita dapatkan secara statistik memang memiliki signifikansi praktis dalam dunia nyata? Sebuah korelasi yang signifikan secara statistik ($p < 0.05$) mungkin tidak berarti apa-apa secara bisnis jika efeknya sangat kecil (*effect size* rendah). Misalnya, perubahan tata letak tombol web mungkin meningkatkan konversi secara signifikan secara statistik, tetapi jika peningkatannya hanya 0.001%, apakah itu layak untuk diimplementasikan secara teknis? Interpretasi yang baik harus selalu mempertimbangkan *Cost-Benefit Analysis* (James et al., 2021).

Selanjutnya, analis harus berhati-hati terhadap bias kognitif dalam interpretasi. Kita sering kali cenderung melihat apa yang ingin kita lihat (*confirmation bias*). Jika kita memiliki hipotesis bahwa iklan sosial media sangat efektif, kita mungkin cenderung mengabaikan data yang menunjukkan sebaliknya. Untuk melawan ini, kita harus menerapkan pola pikir skeptis. Selalu tanyakan: "Apakah ada penjelasan alternatif untuk pola ini?" atau "Data apa yang mungkin hilang atau terdistorsi sehingga saya melihat pola ini?". Melibatkan pihak lain untuk meninjau interpretasi kita adalah cara terbaik untuk meminimalkan bias pribadi (Tukey, 1977).

Konteks temporal dan spasial juga harus diperhatikan dalam interpretasi. Data yang dikumpulkan pada tahun lalu mungkin tidak lagi relevan dengan perilaku pasar saat ini. Interpretasi harus selalu dibatasi oleh ruang dan waktu. Selain itu, perhatikan pula anomali eksternal yang tidak terekam dalam data. Jika terjadi lonjakan penjualan buku secara drastis, interpretasi tidak bisa hanya berhenti pada variabel internal; kita harus memeriksa apakah ada faktor eksternal seperti liburan nasional, kebijakan pemerintah, atau tren viral di media sosial yang mungkin menjadi pemicu utamanya (Wickham, 2014).

Dalam mengomunikasikan hasil interpretasi kepada pemangku kepentingan, gunakan prinsip *storytelling*. Jangan mulai dengan teknis statistik yang rumit. Mulailah dengan temuan utama (*the 'so-what'*), diikuti dengan bukti visual, dan diakhiri dengan implikasi bisnis. Eksekutif tidak ingin tahu tentang asumsi normalitas residual; mereka ingin tahu apa yang harus dilakukan perusahaan untuk meningkatkan profit. Analis yang sukses adalah penerjemah yang mampu mengubah bahasa teknis yang kompleks menjadi bahasa strategi yang sederhana namun akurat (James et al., 2021).

Interpretasi juga harus mencakup diskusi tentang keterbatasan data. Jangan takut untuk mengakui jika data yang dimiliki memiliki keterbatasan dalam hal cakupan atau kualitas. Kejujuran intelektual mengenai apa yang *tidak bisa* dijawab oleh data adalah tanda profesionalisme. Jika data tidak cukup untuk menyimpulkan hubungan kausal, katakanlah demikian. Memberikan rekomendasi berdasarkan data yang cacat tanpa menyebutkan keterbatasannya adalah tindakan yang lebih berbahaya bagi bisnis daripada memberikan rekomendasi yang tidak pasti (Tukey, 1977).

Akhirnya, interpretasi harus mengarah pada rekomendasi yang dapat ditindaklanjuti. EDA yang baik diakhiri dengan daftar langkah konkret. Jika analisis menunjukkan bahwa pelanggan di wilayah X memiliki retensi rendah, jangan hanya melaporkan itu. Rekomendasikan tindakan: "Lakukan kampanye promosi khusus wilayah X" atau "Evaluasi efektivitas kurir logistik di wilayah X". EDA bukan untuk berakhir di laci laporan; ia adalah bahan bakar untuk setiap keputusan strategis yang diambil organisasi (Wickham, 2014).

F. Studi kasus

Exploratory Data Analysis pada Konsorsium Penerbitan

Latar Belakang: Konsorsium penerbitan intelektual yang kita bahas sebelumnya (dari Bab 6) ingin memahami mengapa angka penjualan buku referensi untuk kategori "Pemasaran Digital 5.0" sangat fluktuatif di berbagai wilayah. Mereka memiliki dataset yang mencakup: ID Transaksi, Wilayah, Tanggal, Harga, Jumlah Penjualan, Rating Ulasan, dan Data Iklan Digital (biaya/klik). Tim analis ditugaskan untuk melakukan EDA menyeluruh sebelum manajemen membuat keputusan alokasi anggaran cetak untuk kuartal berikutnya.

Langkah 1: Analisis Univariat (Profil Variabel): Analis memulai dengan memeriksa distribusi "Jumlah Penjualan". Histogram menunjukkan distribusi yang *skewed* ke kanan (banyak transaksi kecil, sedikit transaksi besar). Analis mendeteksi beberapa *outlier* ekstrem yang ternyata adalah pesanan borongan dari institusi pendidikan. Mereka memutuskan untuk membuat dua dataset terpisah: dataset ritel dan dataset institusi, karena perilaku pembelian keduanya sangat berbeda. Analisis pada variabel "Rating Ulasan"

menunjukkan mayoritas rating adalah 4 dan 5, dengan sangat sedikit rating rendah. Ini memberikan keyakinan bahwa kualitas produk secara umum sangat diterima pasar (Tukey, 1977).

Langkah 2: Analisis Bivariat (Korelasi & Hubungan): Selanjutnya, analis melakukan analisis bivariat antara "Biaya Iklan" dan "Jumlah Penjualan". *Scatterplot* menunjukkan korelasi positif yang kuat, namun ada beberapa titik di mana biaya iklan tinggi tetapi penjualan rendah. Setelah diselidiki melalui *crosstab* dengan variabel "Wilayah", ditemukan bahwa iklan di wilayah pedesaan memiliki *Return on Ad Spend* (ROAS) yang rendah dibandingkan wilayah perkotaan. Ini adalah temuan kritis: strategi iklan saat ini mungkin terlalu dipaksakan di wilayah yang akses logistiknya terbatas (James et al., 2021).

Langkah 3: Analisis Multivariat (Interaksi Kompleks): Analis menggunakan *heatmap* untuk melihat korelasi antara "Harga", "Jumlah Penjualan", "Biaya Iklan", dan "Rating". Mereka menemukan bahwa "Harga" berkorelasi negatif kuat dengan "Jumlah Penjualan" hanya pada segmen pelanggan baru, namun tidak berpengaruh pada pelanggan lama yang loyal (yang lebih mementingkan rating). Analis melakukan analisis segmentasi multivariat dan menemukan bahwa pelanggan loyal lebih responsif terhadap kualitas rating, sementara pelanggan baru lebih responsif terhadap diskon harga. Ini memberikan dasar untuk strategi pemasaran yang dibedakan: diskon untuk akuisisi, dan testimoni/ulasan untuk retensi (Wickham, 2014).

Langkah 4: Interpretasi dan Rekomendasi Bisnis: Hasil EDA disajikan kepada dewan direksi. Analis menjelaskan bahwa fluktuasi penjualan bukan karena ketidakstabilan pasar, melainkan karena strategi pemasaran yang

"satu ukuran untuk semua" (*one-size-fits-all*). Rekomendasi konkret yang diberikan: (1) Hentikan iklan berbiaya tinggi di wilayah dengan logistik rendah, (2) Fokuskan anggaran diskon untuk akuisisi di wilayah perkotaan, dan (3) Bangun sistem manajemen ulasan untuk menjaga loyalitas pelanggan lama. Dewan direksi setuju, dan implementasi strategi ini diprediksi akan meningkatkan efisiensi biaya pemasaran sebesar 20% pada kuartal mendatang.

Konklusi Kasus: Studi kasus ini menunjukkan bahwa EDA bukan sekadar memproses data, melainkan proses mengurai kompleksitas menjadi strategi. Tanpa EDA, perusahaan mungkin akan terus membakar uang di wilayah yang salah. EDA memberikan "peta" bagi manajemen untuk menavigasi pasar. Analis data di sini bukan lagi sekadar operator statistik, melainkan mitra strategis bisnis yang mampu mengubah data menjadi profitabilitas melalui pemahaman mendalam tentang perilaku pasar (Tukey, 1977; Wickham, 2014).

Assets of Takaza INNOVATIX LABS

BAB VIII
VISUALISASI DATA

Visualisasi data bukanlah sekadar proses akhir dari analisis statistik, melainkan sebuah jembatan kognitif yang krusial antara kumpulan data mentah yang kompleks dengan persepsi manusia yang intuitif. Di dalam dunia profesional, data sering kali tersimpan dalam bentuk baris dan kolom yang "mati"an gka-angka yang tidak memiliki suara jika tidak diberikan konteks naratif yang kuat. Sebagaimana dinyatakan oleh Edward Tufte (2001), visualisasi data bukan sekadar hiasan estetika; ia adalah instrumen kognitif untuk menalar, membandingkan, menemukan anomali, dan mengomunikasikan informasi yang tertanam dalam dataset yang masif.

Di era di mana organisasi dihujani oleh *Big Data*, kemampuan untuk mensintesis jutaan baris data ke dalam sebuah representasi visual yang dapat dipahami dalam hitungan detik atau bahkan milidetik adalah kompetensi strategis yang tak ternilai bagi seorang manajer atau analis. Visualisasi yang buruk dapat menyebabkan salah paham fatal, sementara visualisasi yang presisi dapat mempercepat pengambilan keputusan secara signifikan. Bab ini akan membedah secara komprehensif bagaimana prinsip psikologi persepsi, pemilihan diagram yang tepat, arsitektur *dashboard* yang fungsional, serta seni *storytelling* dapat mengubah angka-angka yang "bisu" menjadi pendorong keputusan bisnis yang kuat, akurat, dan dapat dipertanggungjawabkan.

.

A. Prinsip Visualisasi Data

Visualisasi data yang efektif berakar pada pemahaman mendalam tentang psikologi persepsi visual manusia. Tujuan utamanya bukanlah untuk memamerkan kecanggihan teknis, melainkan untuk meminimalkan *cognitive load* (beban kognitif) pengguna. Ketika audiens melihat sebuah grafik, otak mereka harus bekerja untuk memproses informasi, bukan untuk menebak-nebak apa yang dimaksud oleh grafik tersebut. Analis harus memastikan fokus pengguna tertuju sepenuhnya pada *insight* yang relevan, bukan pada elemen dekoratif yang tidak perlu (Few, 2013).

Salah satu pilar utama dalam prinsip ini adalah *Data-Ink Ratio* (Tufte, 2001). Konsep ini mengajukan premis bahwa proporsi tinta yang digunakan untuk menampilkan data harus jauh lebih besar daripada tinta yang digunakan untuk elemen non-data. Garis kisi (*gridlines*) yang terlalu tebal, latar belakang yang mencolok, gradasi warna yang tidak perlu, atau legenda yang redundan adalah *noise* visual yang mengalihkan perhatian dari tren data itu sendiri. Setiap elemen yang tidak menambah nilai informasi adalah beban yang harus dieliminasi tanpa ampun demi kejelasan pesan.

Selain itu, kita harus memahami *Preattentive Attributes*. Otak manusia secara alami memproses informasi melalui atribut pra-atentif yaitu elemen visual yang diproses secara instan sebelum kita memfokuskan perhatian secara sadar. Atribut ini meliputi perbedaan warna, bentuk, ukuran, posisi, dan intensitas. Analis yang mahir menggunakan atribut ini secara strategis, misalnya, dengan menyoroti baris "kerugian" menggunakan warna merah kontras di antara baris "keuntungan" yang berwarna biru, untuk memandu audiens ke poin utama secara instan tanpa perlu penjelasan panjang lebar.

Prinsip integritas visual, atau yang sering disebut sebagai *Lie Factor*, juga menjadi imperatif. Sebuah visualisasi dianggap gagal total jika ia mendistorsi realitas data. Contoh klasik adalah *truncated axes* (memotong sumbu Y), yang sering digunakan secara manipulatif untuk membesar-besarkan perbedaan kecil agar terlihat drastis. Prinsip integritas menuntut kejujuran intelektual: data harus merepresentasikan angka aslinya, bukan dimanipulasi untuk mendukung agenda tertentu atau menipu audiens agar percaya pada tren yang sebenarnya tidak ada.

Dalam ranah psikologi bentuk, kita juga harus menerapkan prinsip Gestalt. Prinsip ini menjelaskan bagaimana manusia mengelompokkan elemen visual. Contohnya adalah prinsip kedekatan (*proximity*) dan kesamaan (*similarity*). Elemen yang diletakkan berdekatan atau memiliki warna/bentuk yang serupa akan dipersepsikan oleh otak sebagai satu kesatuan kelompok. Dengan mengoptimalkan prinsip-prinsip ini, analis dapat memastikan bahwa grafik bukan hanya terlihat indah, tetapi juga jujur, mudah dibaca, dan memberikan pemahaman yang objektif mengenai realitas bisnis yang sedang dianalisis (Cairo, 2016).

Warna juga merupakan alat komunikasi yang sangat kuat namun berbahaya jika disalahgunakan. Penggunaan warna harus didasarkan pada makna, bukan sekadar selera. Dalam dunia bisnis, warna merah umumnya diasosiasikan dengan peringatan atau penurunan, sementara hijau diasosiasikan dengan keberhasilan atau pertumbuhan. Jika analis menggunakan warna secara konsisten dengan konvensi ini, mereka mempercepat proses pemahaman. Namun, jika warna digunakan secara sembarangan, hal ini akan membingungkan audiens dan mengaburkan pesan.

Interaktivitas juga harus diperhatikan dalam prinsip visualisasi modern. Tidak semua data perlu ditampilkan sekaligus. Penggunaan teknik *drill-down* atau *tooltips* memungkinkan analis menyajikan detail tanpa menumpuk semua informasi dalam satu grafik tunggal yang menyesakkan. Kunci dari visualisasi modern adalah kesederhanaan yang kaya akan konteks. Dengan menyembunyikan detail teknis di balik lapisan interaksi, audiens dapat fokus pada gambaran besar terlebih dahulu sebelum memutuskan untuk menggali lebih dalam ke detail mikro.

Sebagai kesimpulan dari prinsip-prinsip ini, keberhasilan visualisasi diukur dari seberapa cepat dan akurat audiens memahami *insight* yang disajikan. Visualisasi yang baik adalah visualisasi yang "menghilang", dalam artian perhatian audiens tidak tertuju pada grafik itu sendiri, melainkan pada kebenaran yang diungkapkan oleh data tersebut. Oleh karena itu, disiplin dan pengendalian diri dalam mendesain adalah kunci. Analis harus berani menghapus elemen-elemen yang "terlihat bagus" namun tidak berkontribusi pada penyampaian pesan utama

Assets of Takaza INNOVATIX LABS

B. Grafik dan Diagram Dasar

Memilih diagram yang tepat adalah seni pemilihan konteks. Berdasarkan intensi analisisnya, kita dapat mengategorikan penggunaan grafik ke dalam empat kelompok utama agar praktisi dapat menentukan visualisasi yang paling efektif (Few, 2013). Berikut adalah pembagian dan penjelasan mendalam mengenai kategori tersebut:

1. **Kategori Perbandingan (*Comparison*)** Digunakan untuk mengukur perbedaan antar item guna menentukan peringkat, membandingkan performa, atau melihat selisih nilai.

- **Bar Chart (Horizontal):** Sangat efektif untuk kategori dengan label teks yang panjang, karena memberikan ruang horizontal yang luas bagi teks sehingga mudah dibaca tanpa perlu memutar kepala.
- **Column Chart (Vertical):** Standar emas untuk membandingkan nilai antar kategori. Penting untuk selalu melakukan *sorting* (pengurutan) data dari yang tertinggi ke terendah agar mata audiens tidak bekerja keras mencari nilai maksimum.
- **Grouped Bar Chart:** Digunakan untuk membandingkan sub-kategori dalam satu kategori utama (misalnya: membandingkan performa penjualan buku per genre di tahun 2025 vs 2026). Ini memungkinkan perbandingan lintas waktu dan kategori sekaligus.

2. **2. Kategori Komposisi (*Composition*)** Menunjukkan bagaimana bagian-bagian membentuk sebuah keseluruhan (*part-to-whole*).
   - **Stacked Bar Chart:** Menunjukkan proporsi bagian terhadap total, sekaligus membandingkan total antar grup. Sangat baik untuk melihat struktur data.
   - **Treemap:** Sangat efisien untuk menampilkan data hierarkis dengan banyak kategori di mana proporsi tiap bagian ditunjukkan melalui luas area persegi. Ini jauh lebih informatif daripada *Pie Chart* jika variabelnya banyak.
   - **Donut Chart:** Alternatif modern dari *Pie Chart*. Ruang tengahnya bisa dimanfaatkan untuk menampilkan angka

Assets of Takaza INNOVATIX LABS

total atau narasi penting, yang memperkuat konteks nilai agregat.

3. **3. Kategori Hubungan (*Relationship*)** Mengidentifikasi korelasi, tren, atau interaksi antar variabel numerik.
    - **Scatter Plot:** Digunakan untuk melihat pola hubungan antar dua variabel (misalnya: apakah kenaikan biaya iklan berkorelasi dengan kenaikan penjualan?). Ini mengungkap korelasi yang mungkin tidak terlihat dalam tabel.
    - **Bubble Chart:** Modifikasi *Scatter Plot* dengan dimensi ketiga (ukuran lingkaran) untuk merepresentasikan metrik tambahan seperti "Profit Margin" atau volume transaksi.
4. **4. Kategori Distribusi (*Distribution*)** Memahami sebaran data dan keberadaan nilai ekstrem atau *outlier*.
    - **Histogram:** Menunjukkan frekuensi kemunculan nilai dalam rentang tertentu, sangat berguna untuk melihat apakah data berdistribusi normal atau *skewed*.
    - **Box Plot:** Alat yang sangat kuat untuk melihat sebaran data (median, kuartil, dan *outlier*) dalam satu tampilan yang ringkas, membantu analis mendeteksi ketidakwajaran data tanpa perlu melihat ribuan baris data mentah.

C. Dashboard Analitik

Dashboard adalah pusat komando visual yang mengintegrasikan berbagai metrik kunci ke dalam satu layar tunggal. Ini adalah alat komunikasi paling krusial bagi manajemen. Arsitektur *dashboard* yang baik harus mampu menjawab pertanyaan "apa yang terjadi saat ini?" tanpa

memaksa pengguna melakukan kueri manual atau mencari-cari data di berbagai laporan terpisah (Few, 2013). Tanpa arsitektur yang baik, *dashboard* hanya akan menjadi kumpulan grafik yang tidak saling terhubung dan membingungkan penggunanya.

Prinsip pertama dalam membangun *dashboard* adalah "Prinsip 5 Detik". Seorang pengguna, baik itu staf operasional maupun direktur, harus dapat memahami pesan utama dari *dashboard* tersebut dalam waktu kurang dari lima detik. Jika mereka membutuhkan waktu lebih lama untuk mengartikan apa yang mereka lihat, maka *dashboard* tersebut gagal memenuhi fungsinya sebagai alat bantu pengambilan keputusan cepat. Keseimbangan antara data detail dan ringkasan eksekutif adalah kunci utamanya.

Selanjutnya, kita harus memperhatikan Hierarki Visual atau *Top-Left Priority*. Mata manusia secara alami membaca dari kiri atas ke kanan bawah (mengikuti pola Z). Oleh karena itu, KPI (*Key Performance Indicators*) yang paling kritis harus selalu diletakkan di bagian kiri atas. Informasi pendukung atau detail teknis harus diletakkan di bagian bawah atau di halaman sekunder yang dapat diakses melalui *drill-down*. Mengatur tata letak berdasarkan hierarki ini membantu mengarahkan alur pikir pengguna secara logis.

Konsistensi desain adalah faktor yang sering diabaikan. Menggunakan palet warna, jenis huruf, dan gaya penempatan yang seragam di seluruh halaman *dashboard* akan sangat mengurangi *mental overhead*. Ketika pengguna berpindah antar halaman atau melihat berbagai *widget*, mereka tidak perlu menyesuaikan persepsi mereka jika gaya visualnya konsisten. Ini membangun kepercayaan pengguna terhadap data yang disajikan, karena *dashboard* terlihat profesional dan terpercaya.

Assets of Takaza INNOVATIX LABS

Interaktivitas yang terkendali adalah aspek yang membedakan *dashboard* statis dengan *dashboard* analitik modern. Berikan filter (seperti rentang tanggal, wilayah, atau kategori produk) agar pengguna bisa melakukan *drill-down* dan eksplorasi data secara mandiri. Namun, ada bahaya dalam memberikan terlalu banyak pilihan. Terlalu banyak filter justru akan membingungkan pengguna. Pilihlah filter yang paling relevan untuk menjawab pertanyaan bisnis utama yang paling sering muncul.

Berdasarkan audiensnya, *dashboard* dibagi menjadi tiga kategori utama. Pertama, *Operational Dashboards* yang fokus pada *real-time data* untuk staf lapangan. Mereka membutuhkan data yang diperbarui setiap detik atau menit. Kedua, *Strategic Dashboards* yang fokus pada KPI jangka panjang untuk eksekutif/direktur, dengan pembaruan yang lebih lambat namun lebih stabil. Ketiga, *Analytical Dashboards* yang fokus pada korelasi dan tren untuk analis data, yang mendukung eksplorasi mendalam.

Kualitas data yang mendasari *dashboard* juga tidak boleh diabaikan. *Dashboard* yang indah namun datanya tidak akurat adalah bahaya nyata. *Dashboard* harus didukung oleh *pipeline* data yang bersih dan terverifikasi. Seringkali, masalah dalam *dashboard* bukan terletak pada visualisasinya, melainkan pada data mentah yang tidak terstandardisasi. Oleh karena itu, integrasi antara ilmu data dan desain visual adalah keharusan mutlak.

Terakhir, *dashboard* bukanlah produk sekali jadi. Ia harus berevolusi sesuai dengan perubahan kebutuhan bisnis. Lakukan evaluasi secara berkala dengan pengguna untuk menanyakan apakah grafik yang disajikan masih relevan. Jika sebuah grafik tidak lagi digunakan atau tidak membantu dalam pengambilan keputusan, jangan ragu untuk menghapusnya. *Dashboard* yang bersih, relevan, dan terupdate adalah aset strategis yang membedakan

organisasi yang dikelola secara intuitif dengan organisasi yang dikelola secara berbasis data.

D. Storytelling dengan Data

Visualisasi data yang paling efektif adalah yang mampu menceritakan kisah. Menurut Knaflic (2015), *storytelling* dengan data bukanlah sekadar membuat grafik, melainkan membangun alur narasi yang logis yang menghubungkan angka dengan realitas bisnis. Alurnya mengikuti pola: **Konteks → Konflik (Insight) → Resolusi (Aksi)**. Tanpa narasi, data hanyalah deretan angka yang dingin dan tidak menggugah empati atau urgensi manajemen.

Langkah pertama, Konteks, adalah fondasi. Analis harus mampu menjelaskan siapa audiensnya, apa masalah bisnis yang sedang dihadapi, dan apa tujuan yang ingin dicapai melalui data tersebut. Tanpa konteks yang jelas, audiens akan tersesat dalam lautan angka. Menjelaskan "mengapa" data ini penting dan "mengapa" ini relevan bagi audiens adalah langkah awal yang menentukan keberhasilan penyampaian pesan.

Langkah kedua adalah Konflik, atau *Insight*. Di sinilah visualisasi digunakan untuk menonjolkan bagian yang paling krusial atau "mengganggu" dari data. Di sinilah teknik *pre-attentive processing* berperan vital; kita menggunakan warna kontras misalnya, abu-abu untuk data pembanding (latar belakang), dan warna cerah (seperti biru atau oranye) untuk data yang ingin disorot agar perhatian audiens langsung tertuju pada masalah utama atau anomali yang ditemukan.

Langkah ketiga, Resolusi atau Aksi, adalah klimaks dari narasi. Setelah audiens memahami masalah (konflik) melalui visualisasi yang jelas,

langkah terakhir adalah memberikan rekomendasi atau tindakan konkret yang harus diambil. Banyak analis berhenti pada poin menyajikan grafik; namun, *storyteller* yang hebat akan melanjutkan dengan memberikan saran. "Karena data menunjukkan penurunan di wilayah ini, maka tindakan yang disarankan adalah X."

Sebuah *story* yang kuat sangat bergantung pada anotasi. Jangan pernah memaksa audiens menafsirkan grafik sendiri. Gunakan teks penjelasan langsung di dalam grafik untuk memandu mata dan pikiran audiens. Berikan penjelasan naratif yang mengikat data tersebut ke realitas bisnis. Anotasi yang tepat mengubah grafik dari sekadar gambar menjadi argumen yang persuasif dan sulit dibantah.

Selain itu, *storytelling* dengan data juga melibatkan pemilihan media yang tepat. Apakah ini akan disampaikan dalam rapat presentasi, dalam *dashboard* interaktif, atau dalam laporan PDF? Setiap media memiliki teknik *storytelling* yang berbeda. Dalam presentasi langsung, kita bisa membangun narasi secara bertahap (animasi grafik). Dalam laporan statis, kita harus memastikan semua informasi penting dapat dipahami dalam satu pandangan mata.

Emosi juga memainkan peran penting. Meskipun *data science* sering dianggap sebagai bidang yang dingin dan objektif, *storytelling* mengakui bahwa pengambilan keputusan adalah proses manusiawi. Menggunakan data untuk menyoroti keberhasilan atau kegagalan yang berdampak pada orang nyata seperti efisiensi kerja atau kepuasan pelanggan akan membuat pesan lebih berkesan dan lebih mungkin menghasilkan tindakan.

Terakhir, *storytelling* adalah keterampilan yang harus dilatih terus-menerus. Ia membutuhkan kemampuan untuk menyederhanakan

kompleksitas tanpa menghilangkan kebenaran. Menjadi seorang *data storyteller* berarti menjadi penerjemah antara bahasa mesin yang rumit dan bahasa bisnis yang pragmatis. Jika Anda berhasil menguasai ini, Anda tidak hanya memberikan data, Anda memberikan arah bagi organisasi.

E. Kesalahan Umum dalam Visualisasi

Kesalahan dalam visualisasi sering kali lahir dari keinginan untuk membuat grafik "terlihat menarik" tanpa mempertimbangkan fungsinya (Cairo, 2016). Keinginan untuk menambahkan estetika berlebihan sering kali merusak integritas data. Berikut adalah beberapa jebakan klasik yang harus dihindari oleh setiap analis data yang bertanggung jawab:

Kesalahan pertama adalah penggunaan *Pie Chart* yang berlebihan. Manusia sangat buruk dalam membandingkan luas sudut atau area melingkar. Ketika kita mencoba membandingkan potongan-potongan kue yang ukurannya hampir sama, mata kita akan gagal memberikan estimasi yang akurat. Gunakan *Bar Chart* jika perbedaan nilainya tipis karena mata manusia jauh lebih efisien dalam membandingkan panjang garis pada sumbu yang sama daripada membandingkan sudut.

Kesalahan kedua adalah *Truncated Axes* atau pemotongan sumbu. Memulai sumbu Y bukan dari nol adalah teknik manipulasi visual yang umum digunakan untuk membuat perubahan kecil terlihat sangat drastis atau dramatis. Meskipun ada kasus tertentu di mana ini diperbolehkan (misalnya dalam bidang saintifik yang sangat spesifik), dalam bisnis, ini sering dianggap tidak etis karena menyesatkan pembaca tentang skala perubahan yang sebenarnya terjadi.

Ketiga adalah *Chart Junk*. Ini adalah penambahan elemen dekoratif yang tidak perlu, seperti efek bayangan 3D, tekstur, atau gambar latar belakang yang ramai. Grafik 3D untuk data 2D hanya akan membuat data sulit dibaca karena efek perspektif yang mendistorsi nilai proporsional data. Selalu pertahankan desain 2D yang bersih, sederhana, dan fokus pada data itu sendiri. Estetika yang baik adalah estetika yang fungsional, bukan dekoratif.

Keempat adalah penggunaan warna yang tidak konsisten. Contohnya, menggunakan warna berbeda untuk kategori yang sama di halaman yang berbeda, atau menggunakan skala warna yang tidak logis (misalnya menggunakan merah untuk nilai positif atau biru untuk nilai negatif). Konsistensi warna adalah bahasa universal dalam visualisasi. Jika Anda menggunakan biru untuk "Penjualan" di grafik pertama, jangan gunakan warna merah untuk "Penjualan" di grafik kedua, karena ini akan membingungkan audiens.

Kelima, *Over-plotting* atau penumpukan data. Ketika kita memiliki ribuan titik data dalam satu grafik *scatter plot*, titik-titik tersebut akan saling menumpuk dan pola hubungan menjadi tidak terbaca. Analis harus menggunakan teknik *jittering*, transparansi, atau *sub-sampling* untuk memastikan pola data tetap terlihat. Jangan pernah menyajikan grafik yang tampak seperti "noda tinta" yang tidak teratur karena hal tersebut gagal menyampaikan informasi apapun.

Keenam, label yang membingungkan atau terlalu banyak teks. Grafik harus bisa berbicara sendiri sebagian besar dari isinya. Jika audiens harus membaca paragraf panjang untuk memahami apa arti sebuah grafik, maka grafik tersebut gagal. Gunakan judul yang informatif (misalnya: "Penjualan

Meningkat 20% pada Q1" daripada hanya "Grafik Penjualan") agar pesan utama langsung tersampaikan.

Ketujuh, salah memilih tipe grafik untuk jenis data. Menggunakan *line chart* untuk data kategorikal yang tidak berurutan, atau menggunakan *bar chart* untuk data deret waktu yang sangat rapat, adalah kesalahan fundamental. Memahami sifat data (apakah *time-series*, kategorikal, atau kontinu) adalah prasyarat untuk memilih grafik yang tepat.

Terakhir, kurangnya anotasi. Membiarkan grafik berdiri sendiri tanpa penjelasan adalah kesalahan fatal dalam konteks *data storytelling*. Selalu tambahkan catatan singkat, panah penunjuk, atau teks penjelas pada titik-titik anomali atau data penting. Ingat, tujuan akhir dari visualisasi bukan untuk membuat grafik yang indah secara seni, melainkan grafik yang efektif secara komunikasi.

Assets of Takaza INNOVATIX LABS

F. Studi kasus

Permasalahan: Manajemen sebuah konsorsium penerbitan buku mengalami stagnasi dalam pengambilan keputusan. Data tersebar secara silo (terpisah) antara divisi penjualan, divisi pemasaran digital, dan gudang logistik. Pimpinan konsorsium tidak bisa memahami mengapa buku berjudul "Referensi Akademik" mengalami penurunan penjualan yang tajam di wilayah Jawa Barat, padahal promosi digital sedang gencar dilakukan. Mereka terjebak dalam perdebatan tanpa solusi karena setiap divisi membawa data yang berbeda-beda ke meja rapat.

Implementasi: Tim analis data memutuskan untuk membangun sebuah *Analytical Dashboard* terpusat yang mengintegrasikan ketiga sumber data tersebut. Mereka menerapkan prinsip *Data-Ink Ratio* dengan sangat

ketat. Mereka menghilangkan semua garis kisi yang tidak perlu, menghapus latar belakang yang berwarna-warni, dan hanya menggunakan warna biru netral sebagai warna utama untuk metrik penjualan. Warna merah *bold* hanya digunakan secara khusus untuk menandai metrik penjualan yang berada di bawah target mingguan.

Analisis dan Narasi:

1. **Analisis Awal:** *Dashboard* secara instan menunjukkan tren penurunan penjualan di wilayah Jawa Barat melalui *heat map* wilayah yang disinkronkan dengan data transaksi harian.
2. **Interaksi Drill-Down:** Analis menggunakan fitur *filter* pada *dashboard* untuk menelusuri data per kota dan per kategori buku. Ternyata, penurunan spesifik hanya terjadi pada jenis buku "Referensi Akademik" di wilayah Bandung, bukan di seluruh Jawa Barat.
3. **Penyimpulan (*Storytelling*):** Analis menambahkan anotasi pada grafik penjualan tersebut: "Penurunan penjualan di Bandung bertepatan dengan keterlambatan pengiriman logistik selama 2 minggu akibat perbaikan akses jalan utama."
4. **Keputusan:** Manajemen segera mengalihkan prioritas kurir logistik dan memberikan diskon khusus bagi pelanggan di Bandung untuk memulihkan kepercayaan pelanggan.

Konklusi: *Dashboard* ini membuktikan bahwa visualisasi data yang baik bukan sekadar alat pemantau yang pasif, melainkan instrumen komunikasi yang memaksa manajemen untuk segera mengambil tindakan berbasis fakta, bukan asumsi. Kemampuan untuk menelusuri "mengapa" di balik penurunan angka (dari wilayah → kategori → keterlambatan logistik)

Assets of Takaza INNOVATIX LABS

adalah bukti kekuatan visualisasi data dalam mengubah ketidakpastian menjadi strategi yang dapat dieksekusi secara nyata dalam hitungan jam, bukan minggu.

Strategi ini mengubah dinamika rapat manajemen. Sebelumnya, mereka menghabiskan waktu berjam-jam untuk berdebat tentang data mana yang benar. Sekarang, mereka menghabiskan waktu untuk merancang strategi berdasarkan data yang sudah disepakati kebenarannya di layar *dashboard*. Visualisasi telah mengubah budaya kerja dari "menebak-nebak" menjadi "berbasis bukti". Ini adalah demonstrasi nyata bagaimana *Data Science* dipadukan dengan desain visual yang tepat dapat memberikan dampak bisnis yang sangat signifikan.

Keberhasilan kasus ini juga menggarisbawahi pentingnya kolaborasi antar departemen. *Dashboard* bertindak sebagai "bahasa pemersatu" (common language) bagi divisi penjualan, pemasaran, dan logistik. Dengan melihat satu kebenaran (single source of truth), setiap departemen dapat memahami bagaimana peran mereka mempengaruhi kinerja organisasi secara keseluruhan, dan ini pada akhirnya meningkatkan efisiensi operasional secara kolektif di seluruh konsorsium.

Di masa depan, konsorsium ini berencana untuk menambahkan modul prediksi berbasis *Machine Learning* ke dalam *dashboard* tersebut. Dengan integrasi ini, mereka tidak hanya bisa melihat apa yang *telah* terjadi, tetapi juga memprediksi potensi masalah logistik sebelum terjadi. Ini adalah evolusi alami dari visualisasi data: dari pelaporan masa lalu menuju pandangan ke masa depan, yang semuanya dimulai dari prinsip visualisasi yang sederhana namun sangat kuat.

Assets of Takaza INNOVATIX LABS

BAB IX
ANALISIS DATA UNTUK PENGAMBILAN KEPUTUSAN

Setelah visualisasi data berhasil menerjemahkan angka ke dalam bahasa yang dapat dilihat oleh mata, langkah selanjutnya yang jauh lebih krusial adalah mengubah "apa yang terlihat" menjadi "apa yang harus dilakukan". Bab ini, Analisis Data untuk Pengambilan Keputusan, berfungsi sebagai inti dari setiap alur kerja *Data Science* dalam dunia bisnis. Jika visualisasi adalah alat komunikasi, maka analisis adalah proses berpikir. Di sini, kita akan mengeksplorasi bagaimana data bertransformasi dari sekadar aset informasi menjadi pendorong utama strategi organisasi. Sebagaimana dijelaskan oleh Davenport & Harris (2007), keunggulan kompetitif di era modern bukan lagi ditentukan oleh seberapa banyak data yang dimiliki, melainkan seberapa efektif data tersebut dianalisis untuk mendukung keputusan.

Pengambilan keputusan berbasis data bukan sekadar tentang menggunakan perangkat lunak canggih atau memiliki *dashboard* yang estetik. Ini adalah tentang menanamkan logika empiris ke dalam setiap urat nadi organisasi, di mana setiap kebijakan yang diambil didasarkan pada probabilitas, bukti, dan pola, bukan sekadar intuisi atau pendapat subjektif dari pemangku kepentingan tertinggi. Bab ini akan membedah metodologi analisis dari sisi tren, segmentasi, hingga pengukuran performa, serta bagaimana menyintesis semua temuan tersebut menjadi *insight* bisnis yang

memiliki nilai ekonomi tinggi bagi perusahaan. Proses ini menuntut ketajaman analitis yang didukung oleh pemahaman mendalam terhadap domain bisnis yang bersangkutan (Provost & Fawcett, 2013).

A. Data-Driven Decision Making

Data-Driven Decision Making (DDDM) adalah sebuah paradigma di mana keputusan organisasi dipandu oleh pengumpulan data, analisis fakta, dan interpretasi metrik daripada sekadar mengandalkan firasat atau pengalaman historis. Di dunia yang volatil, DDDM memungkinkan organisasi untuk mengurangi ketidakpastian secara signifikan. Prinsip dasarnya adalah bahwa keputusan yang didukung oleh data lebih cenderung menghasilkan hasil yang konsisten, terukur, dan dapat diulang, karena ia menghilangkan bias kognitif yang sering menghinggapi pengambilan keputusan manusia yang bersifat intuitif (Kahneman, 2011).

Namun, mengadopsi DDDM memerlukan perubahan budaya yang radikal. Ini bukan hanya soal teknologi, melainkan tentang menciptakan lingkungan di mana setiap orang memiliki *data literacy* kemampuan untuk membaca, bekerja dengan, menganalisis, dan berargumentasi menggunakan data. Organisasi harus bergerak menjauh dari model HiPPO (*Highest Paid Person's Opinion*), di mana keputusan hanya diambil berdasarkan opini pimpinan tertinggi, menuju model demokratis berbasis bukti di mana argumen yang didukung oleh data menang terlepas dari siapa yang mengucapkannya (Brynjolfsson & McElheran, 2016).

Implementasi DDDM juga menuntut adanya transparansi dalam pengumpulan data. Data tidak boleh "dijemput bola" hanya untuk membenarkan keputusan yang sudah dibuat sebelumnya (*confirmation bias*).

Sebaliknya, data harus dikumpulkan secara objektif untuk menjawab pertanyaan strategis. Proses ini dimulai dari mendefinisikan masalah, mengumpulkan data relevan, membersihkan data dari *noise*, menganalisis pola, dan akhirnya mengambil langkah konkret berdasarkan hasil analisis tersebut, diikuti dengan evaluasi berkelanjutan terhadap dampak keputusan tersebut.

Selain itu, DDDM harus mempertimbangkan keterbatasan data. Data masa lalu tidak selalu menjadi prediktor sempurna untuk masa depan, terutama di pasar yang disruptif. Oleh karena itu, pengambilan keputusan berbasis data yang matang harus memadukan "Data" (sebagai kompas) dengan "Domain Expertise" (sebagai nakhoda). Data memberikan arah, namun pengalaman manusia memberikan konteks mengenai nuansa pasar, etika, dan nilai-nilai organisasi yang tidak bisa diwakili oleh angka saja.

Salah satu tantangan terbesar dalam DDDM adalah *Analysis Paralysis*, yaitu kondisi di mana pimpinan terjebak dalam memproses terlalu banyak data tanpa mampu menarik kesimpulan. Untuk menghindarinya, DDDM harus fokus pada *Key Performance Indicators* (KPI) yang relevan. Fokus adalah mata uang dari analisis data; mengetahui apa yang tidak perlu diukur sama pentingnya dengan mengetahui apa yang harus diukur. Organisasi harus belajar untuk menolak data yang tidak memberikan nilai tambah bagi tujuan strategis utama.

Penerapan DDDM juga menuntut sistem *feedback loop* yang kuat. Keputusan yang diambil berdasarkan data harus dimonitor hasilnya secara ketat. Apakah hasil di lapangan sesuai dengan prediksi model? Jika tidak, ini bukanlah kegagalan, melainkan peluang untuk melakukan kalibrasi ulang terhadap model atau asumsi yang digunakan. Budaya belajar dari kegagalan

data ini adalah ciri khas organisasi yang benar-benar matang dalam menggunakan data.

Terakhir, DDDM harus didukung oleh integrasi data antar departemen. Data yang terisolasi di departemen pemasaran, penjualan, atau keuangan akan memberikan gambaran yang parsial. Keputusan strategis yang holistik memerlukan integrasi data lintas fungsi. Ketika semua departemen berbicara menggunakan bahasa data yang sama, organisasi dapat bergerak dengan kecepatan dan akurasi yang jauh lebih tinggi.

Pada akhirnya, DDDM adalah tentang meminimalisir risiko. Di pasar yang kompetitif, organisasi yang cepat belajar dari data akan selalu mengungguli mereka yang bergerak berdasarkan asumsi. DDDM bukan tentang menghapus kreativitas atau intuisi, melainkan memberikan kerangka kerja yang solid agar kreativitas dan intuisi tersebut diarahkan ke tujuan yang paling mungkin membawa keberhasilan.

Assets of Takaza INNOVATIX LABS

B. Analisis Tren

Analisis tren adalah seni membedah arah pergerakan data dari waktu ke waktu (*time-series analysis*). Tujuannya adalah untuk memahami apakah kinerja organisasi sedang meningkat, menurun, atau stagnan, serta mengidentifikasi apakah pola tersebut bersifat sementara atau struktural. Memahami tren memungkinkan perusahaan untuk melakukan perencanaan kapasitas, manajemen inventaris, dan strategi pemasaran yang proaktif alih-alih reaktif (Box et al., 2015).

Dasar dari analisis tren adalah memisahkan *signal* (sinyal nyata) dari *noise* (fluktuasi acak). Seringkali, data harian bisa menunjukkan volatilitas yang tinggi karena faktor-faktor luar yang tidak relevan. Analis harus

menggunakan teknik seperti *Moving Average* (rata-rata bergerak) untuk menghaluskan data tersebut sehingga tren jangka panjang yang lebih signifikan dapat terlihat dengan jelas. Tanpa penghalusan, kita cenderung bereaksi berlebihan terhadap perubahan kecil yang sebenarnya tidak berarti.

Penting untuk membedakan antara musiman (*seasonality*) dan tren. Tren bersifat jangka panjang dan arahnya cenderung konsisten, sementara musiman bersifat siklikal dan berulang dalam periode tertentu (misalnya, lonjakan penjualan setiap akhir tahun atau saat libur panjang). Analis yang gagal membedakan keduanya akan sering membuat kesalahan fatal: menganggap lonjakan musiman sebagai tren peningkatan permanen, sehingga perusahaan melakukan investasi berlebihan pada kapasitas yang tidak dibutuhkan saat musim berakhir.

Analisis tren juga harus melibatkan identifikasi *titik balik* (*inflection points*). Ini adalah momen di mana arah tren berubah secara drastis, misalnya ketika pertumbuhan produk mulai melambat (kejenuhan pasar) atau ketika tiba-tiba terjadi penurunan permintaan akibat faktor eksternal seperti regulasi baru. Mengidentifikasi titik-titik ini lebih awal memungkinkan organisasi untuk melakukan pivot atau penyesuaian strategi sebelum krisis terjadi.

Dalam lingkungan modern, analisis tren tidak lagi terbatas pada data internal. Kita juga harus memasukkan *External Data* (seperti tren pencarian Google, sentimen media sosial, atau data makroekonomi). Menggabungkan tren internal dengan tren eksternal memberikan konteks yang jauh lebih kaya. Misalnya, penurunan penjualan mungkin bukan karena kualitas produk menurun, melainkan karena perubahan tren gaya hidup yang dideteksi melalui data tren media sosial jauh sebelum terlihat di laporan penjualan.

Assets of Takaza INNOVATIX LABS

Teknik pemodelan statistik seperti *Regression Analysis* sering digunakan untuk memperkirakan kelanjutan tren. Namun, peringatan penting bagi analis: korelasi masa lalu bukanlah jaminan masa depan. Selalu gunakan interval kepercayaan (*confidence intervals*) saat memprediksi tren masa depan. Analisis tren yang baik harus selalu menyediakan skenario (optimis, moderat, pesimis) agar manajemen memiliki rencana kontinjensi.

Pentingnya *granularity* (tingkat kedalaman) dalam analisis tren tidak boleh diremehkan. Tren tingkat perusahaan mungkin terlihat positif, namun jika dibedah ke tingkat regional atau per-produk, kita mungkin menemukan bahwa tren positif tersebut hanya disumbang oleh satu wilayah saja, sementara wilayah lain sedang mengalami penurunan tajam. Analisis tren harus dilakukan secara bertingkat untuk mendapatkan gambaran yang jujur.

Sebagai penutup, analisis tren adalah tentang menceritakan masa depan berdasarkan jejak masa lalu. Ini adalah alat bantu navigasi utama. Dengan memahami tren, organisasi tidak lagi berjalan dalam kegelapan, melainkan memiliki peta jalan yang memungkinkan mereka untuk mengantisipasi tantangan dan memanfaatkan peluang sebelum kompetitor menyadarinya.

C. Segmentasi Data

Segmentasi data adalah proses membagi dataset yang besar dan heterogen menjadi sub-kelompok yang lebih kecil, homogen, dan memiliki karakteristik serupa. Dalam konteks bisnis, ini hampir selalu berarti segmentasi pelanggan (*customer segmentation*). Prinsip "satu ukuran untuk semua" adalah resep untuk kegagalan di pasar modern; segmentasi adalah

kunci untuk personalisasi, yang merupakan fondasi dari loyalitas pelanggan dan efisiensi pemasaran (Wedel & Kamakura, 2000).

Metode klasik dalam segmentasi adalah RFM (*Recency*, *Frequency*, *Monetary*). *Recency* mengukur seberapa baru pelanggan melakukan pembelian, *Frequency* mengukur seberapa sering, dan *Monetary* mengukur total nilai uang yang dihabiskan. Dengan teknik ini, kita bisa memisahkan pelanggan "juara" (sering belanja, baru saja belanja, nilai tinggi) dari pelanggan "berisiko" (lama tidak belanja, frekuensi rendah). Strategi komunikasi untuk kedua kelompok ini harus benar-benar berbeda.

Selain demografi (usia, lokasi, gender), segmentasi modern beralih ke *behavioral segmentation* (berdasarkan perilaku). Apa yang dilakukan pelanggan di situs web? Produk apa yang mereka lihat sebelum memutuskan untuk tidak membeli? Segmen berdasarkan perilaku jauh lebih akurat dalam memprediksi pembelian masa depan dibandingkan dengan segmentasi demografis semata. Seseorang mungkin memiliki demografi yang sama, namun perilaku pembeliannya bisa bertolak belakang karena motivasi atau kebutuhan yang berbeda.

Dalam analisis data, kita sering menggunakan algoritma *Clustering* (seperti K-Means) untuk menemukan segmen yang mungkin tidak terlihat oleh mata manusia. Algoritma ini mengelompokkan data berdasarkan kemiripan atribut secara matematis. Seringkali, hasil *clustering* ini mengungkapkan "tipe pelanggan" yang tidak terduga, yang kemudian bisa dieksplorasi oleh tim pemasaran untuk kampanye yang lebih spesifik.

Segmentasi juga krusial untuk *pricing strategy*. Tidak semua pelanggan sensitif terhadap harga. Dengan segmentasi yang tepat, perusahaan dapat menerapkan *price discrimination* yang etis dan

menguntungkan memberikan penawaran harga yang berbeda berdasarkan kesediaan membayar (*willingness to pay*) segmen tertentu. Ini memaksimalkan *revenue* tanpa kehilangan basis pelanggan yang lebih sensitif terhadap harga.

Namun, segmentasi memiliki jebakan: *over-segmentation*. Terlalu banyak segmen akan membuat kampanye pemasaran menjadi tidak efisien karena biaya operasional untuk mengelola setiap segmen membengkak. Segmentasi harus memiliki "skala ekonomi". Jika sebuah segmen terlalu kecil untuk memberikan dampak finansial yang berarti, mungkin lebih baik untuk menggabungkannya kembali ke segmen yang lebih besar.

Segmentasi juga bersifat dinamis. Pelanggan bisa berpindah dari satu segmen ke segmen lain seiring perubahan kebutuhan hidup atau preferensi. Oleh karena itu, segmentasi bukanlah proses sekali jalan, melainkan proses berkelanjutan. Data harus disegmentasi ulang secara berkala untuk menangkap dinamika perubahan perilaku pelanggan yang sangat cepat.

Kesimpulannya, segmentasi data mengubah massa pelanggan yang tidak dikenal menjadi individu-individu dengan kebutuhan spesifik. Ia memungkinkan perusahaan untuk berhenti "berteriak" kepada semua orang dan mulai "berbisik" kepada pelanggan yang tepat dengan pesan yang relevan. Personalisasi inilah yang mengubah transaksi menjadi hubungan jangka panjang.

D. Analisis Performa

Analisis performa adalah proses evaluasi keberhasilan organisasi dalam mencapai tujuan strategisnya dengan membandingkan hasil aktual terhadap target yang telah ditetapkan. Di sinilah *Key Performance Indicators*

(KPI) berperan sebagai kompas. Namun, analisis performa yang mendalam tidak berhenti pada angka "tercapai" atau "tidak tercapai", melainkan menggali akar penyebab mengapa hasil tersebut terjadi.

Perbedaan fundamental dalam analisis performa adalah antara *lagging indicators* dan *leading indicators*. *Lagging indicators* (seperti pendapatan bulanan) memberi tahu kita apa yang sudah terjadi, namun bersifat historis dan tidak bisa diubah. *Leading indicators* (seperti jumlah prospek baru atau tingkat *engagement* di media sosial) adalah prediktor masa depan. Analisis performa yang hebat selalu menyeimbangkan keduanya untuk memberikan gambaran kesehatan organisasi yang komprehensif (Kaplan & Norton, 1996).

Proses ini sering menggunakan *Benchmarking*. Membandingkan performa internal dengan standar industri atau kompetitor memberikan konteks apakah pertumbuhan 10% itu "hebat" atau justru "di bawah rata-rata". Tanpa *benchmark*, kita sering terjebak dalam rasa puas diri (*complacency*) karena melihat angka pertumbuhan yang positif, padahal pasar secara keseluruhan mungkin tumbuh lebih cepat.

Analisis performa juga harus melibatkan *Root Cause Analysis* (Analisis Akar Masalah). Jika sebuah metrik performa turun, kita tidak bisa langsung mencari kambing hitam. Kita harus menggunakan teknik seperti "5 Whys" atau *Fishbone Diagram* untuk membedah data dan menemukan akar penyebabnya. Apakah masalahnya pada proses operasional, kualitas produk, atau perubahan perilaku pasar?

Penting juga untuk menyadari *The Cobra Effect* atau konsekuensi yang tidak disengaja dari metrik. Jika kita terlalu fokus pada satu metrik, staf mungkin akan memanipulasi data agar terlihat bagus di metrik tersebut,

namun merusak aspek lain yang tidak terukur. Misalnya, jika KPI adalah "jumlah panggilan terjawab", staf mungkin akan menjawab dengan cepat lalu menutup telepon sebelum masalah terselesaikan. Analis performa harus selalu memantau metrik secara holistik.

Analisis performa yang efektif selalu mencakup analisis efisiensi biaya (*Cost-Effectiveness*). Performa bukan hanya tentang pendapatan, tetapi tentang profitabilitas. Menggunakan data untuk mengidentifikasi inefisiensi seperti biaya akuisisi pelanggan (CAC) yang terlalu tinggi dibandingkan dengan nilai seumur hidup pelanggan (CLV) adalah kunci untuk menjaga keberlanjutan bisnis.

Visualisasi dalam analisis performa harus fokus pada "Gap Analysis". Menampilkan selisih antara target dan realisasi dengan jelas akan menciptakan urgensi dan fokus. Menggunakan warna seperti merah untuk defisit dan hijau untuk surplus memberikan sinyal visual instan yang memudahkan manajer untuk mengidentifikasi area mana yang memerlukan perhatian segera.

Akhirnya, analisis performa adalah tentang akuntabilitas. Data menghilangkan perdebatan subjektif tentang siapa yang bekerja keras dan siapa yang tidak. Dengan analisis performa yang transparan, organisasi dapat menciptakan budaya kinerja yang sehat di mana setiap individu memahami bagaimana kontribusi mereka terhadap KPI perusahaan.

E. Penyusunan Insight Bisnis

Menyusun *insight* bisnis adalah proses mengubah hasil analisis teknis menjadi narasi strategis yang dapat dimengerti oleh pengambil keputusan. Ini adalah jembatan antara dunia analis data dan dunia eksekutif. Sebuah

insight bukan sekadar observasi; *insight* adalah kesimpulan yang menyiratkan tindakan. "Penjualan turun" adalah observasi. "Penjualan turun karena pelanggan beralih ke kompetitor X yang menawarkan harga lebih murah, sehingga kita perlu menyesuaikan paket harga" adalah *insight*.

Ciri utama *insight* yang baik adalah kemampuannya untuk memicu perubahan. Jika setelah membaca laporan Anda, pimpinan tidak tahu apa yang harus dilakukan, maka itu bukan *insight*, melainkan sekadar informasi tambahan. Sebuah *insight* yang kuat harus menjawab pertanyaan "So what?" (Lalu apa?). Analis harus mampu menghubungkan temuan data dengan implikasi finansial atau operasional yang nyata (Knaflic, 2015).

Dalam menyusun *insight*, struktur piramida (Barbara Minto's Pyramid Principle) sangat disarankan. Mulailah dengan kesimpulan utama di awal (*executive summary*), diikuti oleh pendukung utama, dan baru kemudian detail data teknis sebagai lampiran. Eksekutif biasanya tidak memiliki waktu untuk membaca metodologi analisis Anda dari awal hingga akhir; mereka membutuhkan "inti" dari temuan tersebut agar bisa segera mengambil keputusan.

Penting untuk menjaga netralitas dan objektivitas. Jangan memaksakan data untuk mendukung prasangka pribadi. Jika data menunjukkan bahwa strategi yang disukai pimpinan ternyata tidak efektif, analis memiliki kewajiban profesional untuk menyampaikannya dengan jujur namun taktis. Gunakan bahasa yang objektif dan fokus pada fakta, bukan opini.

Penyusunan *insight* juga membutuhkan empati terhadap audiens. Gunakan bahasa bisnis, bukan bahasa statistik yang terlalu teknis. Hindari istilah seperti "regresi linear multivariat" atau "p-value" di hadapan manajer

non-teknis kecuali diperlukan. Fokuslah pada dampak: "Efisiensi meningkat 15%" lebih baik daripada "Terdapat penurunan varians signifikan pada distribusi input".

Visual storytelling adalah pendukung utama dalam penyusunan *insight*. Gunakan bagan yang mendukung argumen Anda saja. Jangan membuat audiens membaca sepuluh grafik jika satu grafik ringkasan sudah cukup. Setiap grafik harus memiliki pesan tunggal yang jelas. Jika grafik Anda memerlukan interpretasi yang kompleks, maka grafik itu belum cukup sederhana untuk audiens eksekutif.

Ingatlah selalu tentang *Actionability*. Sebuah *insight* yang baik harus memiliki rekomendasi yang spesifik. Hindari saran yang normatif seperti "kita harus meningkatkan kualitas". Sebaliknya, gunakan saran seperti "kita harus meningkatkan kualitas pada produk X karena data menunjukkan bahwa 40% keluhan pelanggan berasal dari sana". Semakin spesifik rekomendasi Anda, semakin besar kemungkinan itu akan dieksekusi.

Sebagai penutup, menyusun *insight* adalah tentang membangkitkan rasa urgensi. Jika Anda memiliki data yang menunjukkan tren yang berbahaya di masa depan, tugas Anda adalah mengomunikasikannya sedemikian rupa sehingga audiens merasa perlu untuk bertindak. *Data Science* yang sukses adalah yang mampu memicu perubahan nyata di dunia nyata melalui kekuatan kata-kata dan bukti numerik.

F. Studi kasus

Transformasi Ritel Berbasis Data

Permasalahan: Sebuah jaringan toko ritel besar mengalami penurunan profitabilitas sebesar 15% dalam dua tahun terakhir. Meskipun volume

penjualan tetap stabil, margin keuntungan tergerus akibat biaya operasional yang meningkat dan persaingan ketat dari platform *e-commerce*. Pimpinan perusahaan terjebak dalam dilema: apakah harus menurunkan harga (untuk bersaing) atau memotong biaya operasional (yang berisiko menurunkan kualitas layanan).

Metodologi Analisis: Tim analis data menerapkan pendekatan *Data-Driven Decision Making* yang ketat. Pertama, mereka melakukan **Analisis Performa**: Membedah profitabilitas per cabang dan per produk. Ditemukan bahwa 20% toko menyumbang 80% kerugian operasional. Kedua, melakukan **Segmentasi Data**: Melakukan segmentasi pelanggan berdasarkan perilaku belanja. Ditemukan bahwa segmen pelanggan setia tidak sensitif terhadap harga, namun sangat peduli pada kenyamanan (*convenience*).

Ketiga, melakukan **Analisis Tren**: Menganalisis tren pembelian. Ternyata, tren pelanggan bergeser dari pembelian mingguan skala besar menjadi pembelian harian skala kecil. Dari data ini, tim merumuskan **Penemuan (*Insight*)**: Data menunjukkan bahwa model bisnis "toko besar di pinggiran kota" tidak lagi relevan dengan tren belanja harian pelanggan yang sibuk. Pelanggan setia lebih memilih toko yang lebih kecil dan dekat, bahkan jika harganya sedikit lebih mahal.

Keputusan Strategis: Manajemen tidak menurunkan harga, melainkan melakukan restrukturisasi aset. Mereka menutup 20% toko yang tidak menguntungkan di pinggiran kota dan mengalihkan investasinya untuk membuka *mini-market* di pusat pemukiman penduduk yang padat (*micro-fulfillment centers*).

Hasil: Dalam satu tahun, biaya operasional turun drastis, sementara margin keuntungan meningkat karena toko-toko kecil memiliki *throughput* pelanggan yang lebih tinggi dan biaya sewa yang lebih rendah. Analisis data telah menyelamatkan perusahaan dari keputusan salah yang didasarkan pada asumsi tradisional bahwa "harga murah adalah segalanya".

Konklusi: Studi kasus ini menegaskan bahwa analisis data yang mendalam mampu membalikkan asumsi bisnis yang sudah bertahan selama puluhan tahun. Keputusan berbasis data mengubah nasib perusahaan dari penurunan profitabilitas menjadi pertumbuhan yang berkelanjutan. Inilah esensi dari bab ini: data bukan sekadar untuk mencatat masa lalu, melainkan untuk merancang strategi masa depan yang lebih cerdas.

Ringkasan Metodologi Analisis Data

Metodologi	Tujuan Utama	Indikator Utama
DDDM	Mengurangi bias dalam pengambilan keputusan	Efektivitas Keputusan, ROI
Analisis Tren	Memprediksi arah pasar dan pola musiman	Growth Rate, Time-series variance
Segmentasi	Personalisasi layanan & Efisiensi Pemasaran	CLV, Churn Rate, Conversion Rate
Analisis Performa	Mengevaluasi ketercapaian target	KPI, Benchmarking, Root Cause

BAB X
PENGANTAR MACHINE LEARNING

Setelah memahami bagaimana data dianalisis untuk pengambilan keputusan strategis dalam bab sebelumnya, kita kini melangkah ke ranah yang lebih otonom dan prediktif: *Machine Learning* (ML). Jika analisis data tradisional sering kali bersifat deskriptif (menjelaskan apa yang telah terjadi), maka *Machine Learning* membawa organisasi ke tingkat selanjutnya, yakni prediktif dan preskriptif (memprediksi apa yang akan terjadi dan menentukan tindakan terbaik). Dalam paradigma ini, komputer tidak lagi sekadar menjalankan instruksi *if-then-else* yang kaku, melainkan belajar dari data untuk menemukan pola yang tersembunyi dan melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya. Sebagaimana dikemukakan oleh Mitchell (1997), sebuah program komputer dikatakan belajar dari pengalaman E sehubungan dengan tugas T dan ukuran kinerja P, jika kinerjanya dalam menyelesaikan tugas T, sebagaimana diukur dengan P, meningkat seiring dengan pengalaman E.

Bab ini akan membimbing pembaca memahami fondasi teknis dan konseptual *Machine Learning*, mulai dari filosofi dasar hingga penerapan praktis. Kita akan membedah bagaimana algoritma matematika bekerja untuk meminimalkan *error* dan memaksimalkan akurasi, serta pentingnya siklus hidup pengembangan model untuk menghindari jebakan umum seperti *overfitting*. Dengan memahami bab ini, pembaca akan memiliki kerangka

kerja untuk mengintegrasikan kecerdasan buatan ke dalam alur kerja bisnis, mengubah data mentah menjadi aset prediktif yang bernilai tinggi.

A. Konsep Machine Learning

Machine Learning adalah sub-bidang dari kecerdasan buatan (*Artificial Intelligence*) yang berfokus pada pengembangan algoritma dan model statistik yang memungkinkan sistem komputer untuk meningkatkan kinerjanya secara otomatis melalui pengalaman. Berbeda dengan pemrograman tradisional di mana pengembang menulis aturan spesifik untuk setiap skenario (pendekatan *top-down*), dalam *Machine Learning*, pengembang memberikan data dan model tersebut "belajar" untuk membangun logikanya sendiri melalui pendekatan *bottom-up* (Bishop, 2006). Inti dari konsep ini adalah deteksi pola atau *pattern recognition* yang kompleks, di mana mesin mengidentifikasi hubungan korelasi antarvariabel yang mungkin terlalu rumit atau terlalu subtil untuk dikenali oleh logika manusia.

Assets of Takaza INNOVATIX LABS

Dalam dunia bisnis, konsep ini merevolusi cara kita memproses informasi dan mengambil keputusan strategis. Misalnya, daripada membuat aturan manual untuk mendeteksi penipuan kartu kredit, sistem ML akan mempelajari jutaan transaksi historis untuk mengidentifikasi karakteristik transaksi yang "normal" dan "mencurigakan". Ketika transaksi baru masuk, model akan memberikan skor probabilitas apakah transaksi tersebut merupakan penipuan atau bukan. Fleksibilitas ini memungkinkan sistem untuk beradaptasi secara dinamis terhadap perubahan perilaku pasar, tren konsumsi, atau serangan siber yang terus berevolusi tanpa perlu melakukan *re-coding* secara manual oleh tim IT (Russell & Norvig, 2010).

Selain itu, *Machine Learning* sangat bergantung pada kualitas dan kuantitas data. Prinsip *Garbage In, Garbage Out* berlaku mutlak di sini; model yang dibangun di atas data yang bias, kotor, atau tidak relevan akan menghasilkan prediksi yang keliru dan berbahaya. Oleh karena itu, penguasaan atas konsep *feature engineering* yakni proses mengubah data mentah menjadi representasi yang lebih informatif bagi model menjadi sangat krusial. Seorang praktisi *Machine Learning* harus mampu memahami domain masalah secara mendalam untuk memilih fitur-fitur yang benar-benar berpengaruh terhadap target variabel yang ingin diprediksi, sehingga model dapat belajar dengan efisiensi optimal (Hastie et al., 2009).

B. Jenis-Jenis Machine Learning

Secara garis besar, metode *Machine Learning* dikategorikan berdasarkan cara model tersebut belajar dari data. Kategorisasi ini sangat menentukan dalam pemilihan algoritma yang tepat untuk suatu masalah bisnis.

1. **Supervised Learning (Pembelajaran Terbimbing):** Model dilatih menggunakan dataset yang telah diberi label (memiliki jawaban yang benar). Tugas model adalah memetakan input ke output berdasarkan contoh yang diberikan. Contohnya adalah klasifikasi email (spam atau bukan) dan regresi (memprediksi harga properti).
2. **Unsupervised Learning (Pembelajaran Tak Terbimbing):** Model diberikan data tanpa label dan harus menemukan struktur atau pola tersembunyi di dalamnya secara mandiri. Metode ini sering digunakan untuk segmentasi pelanggan (*clustering*) atau reduksi dimensi data.

3. **Reinforcement Learning (Pembelajaran Penguatan):** Model belajar melalui proses *trial-and-error* dengan berinteraksi dengan lingkungan. Model akan menerima "reward" jika mengambil tindakan yang tepat dan "penalti" jika sebaliknya. Ini sering digunakan dalam robotika, sistem rekomendasi yang kompleks, dan *game theory* (Russell & Norvig, 2010).

Jenis ML	**Data Input**	**Tujuan Utama**	**Contoh Penerapan**
Supervised	Berlabel (Labeled)	Prediksi & Klasifikasi	Churn Prediction, Credit Scoring
Unsupervised	Tak Berlabel (Unlabeled)	Penemuan Pola & Klaster	Segmentasi Pasar, Deteksi Anomali
Reinforcement	Interaksi (Agent/Env)	Optimasi Keputusan	Robotika, Algoritma Trading

Assets of Takaza INNOVATIV LABS

C. Tahapan Pengembangan Model

Berikut adalah versi penjelasan mengenai **Tahapan Pengembangan Model** yang telah diperluas secara komprehensif, disusun dalam paragraf-paragraf panjang, menggunakan *body note* (kutipan dalam teks), dan dikemas tanpa menggunakan rumus matematika agar sesuai dengan format buku referensi akademik Anda:

Pengembangan model *Machine Learning* atau pembelajaran mesin sejatinya bukanlah sekadar serangkaian proses penulisan kode pemrograman yang mekanis, melainkan sebuah siklus sistematis yang sangat iteratif dan multidisiplin. Banyak proyek kecerdasan buatan di industri gagal di tengah jalan karena tim pengembang terlalu terburu-buru melompat ke tahap penulisan algoritma tanpa mengindahkan kerangka kerja metodologis yang terstruktur (Wirth & Hipp, 2000). Tahapan pengembangan ini harus dilakukan secara disiplin, terdokumentasi, dan komprehensif untuk memastikan bahwa model yang dihasilkan tidak hanya menunjukkan tingkat akurasi yang tinggi secara metrik statistik di dalam lingkungan laboratorium, tetapi juga benar-benar dapat dioperasionalkan serta memberikan manfaat strategis bagi keberlangsungan bisnis perusahaan (Provost & Fawcett, 2013).

1. **Definisi Masalah**

   Tahapan paling fundamental yang sering kali dipandang sebelah mata oleh praktisi pemula adalah pendefinisian masalah secara presisi. Sebelum menyentuh baris data apa pun, seorang ilmuwan data wajib duduk bersama dengan para pemangku kepentingan bisnis untuk merumuskan apa yang sebenarnya ingin dicapai dan bagaimana menetapkan tolak ukur kesuksesannya. Proses ini melibatkan penerjemahan tujuan bisnis yang bersifat konseptual menjadi rumusan masalah *Machine Learning* yang terukur secara kuantitatif (Kelleher et al., 2015). Pada fase ini, tim harus memutuskan secara definitif apakah mereka sedang menghadapi masalah prediksi nilai angka berkesinambungan yang membutuhkan pendekatan regresi, masalah pengkategorian identitas yang membutuhkan algoritma klasifikasi, atau justru masalah pencarian pola tersembunyi yang mengarah pada

pembelajaran tanpa pengawasan. Kegagalan dalam mendefinisikan masalah operasional dengan tepat di awal akan selalu mengakibatkan pemborosan sumber daya komputasi dan waktu finansial yang sangat masif di kemudian hari.

2. Pengumpulan & Pembersihan Data

Setelah ruang lingkup masalah terdefinisi dengan sangat tajam, langkah krusial selanjutnya adalah pengumpulan dan pembersihan data, yang secara empiris diakui sebagai tahapan paling melelahkan dan menyita lebih dari tujuh puluh persen waktu dari keseluruhan proyek analitik. Mengumpulkan data sering kali merupakan tugas yang kompleks karena informasi historis biasanya tersebar di berbagai basis data yang terisolasi antardepartemen, memiliki format penyimpanan yang berbeda-beda, dan penuh dengan ketidakkonsistenan pencatatan (McKinney, 2012). Prinsip utama yang berlaku mutlak dan tidak bisa ditawar dalam tahap ini adalah *Garbage In, Garbage Out* (GIGO), yang bermakna bahwa model algoritma secanggih apa pun akan selalu menghasilkan prediksi yang buruk jika dilatih menggunakan bahan baku data yang kotor. Oleh karena itu, para praktisi harus melakukan serangkaian teknik sanitasi data yang ketat, seperti menangani data yang kosong melalui teknik imputasi cerdas, menetralisir anomali pencilan yang dapat merusak pola, serta melakukan standardisasi skala agar seluruh variabel masukan memiliki bobot evaluasi yang seimbang di mata komputer (Han et al., 2011).

3. Feature Engineering

Ketika pangkalan data sudah dalam kondisi steril dan terstruktur, proses berikutnya yang membutuhkan tingkat kreativitas serta intuisi keilmuan

tertinggi adalah rekayasa fitur atau *Feature Engineering*. Berbeda dengan tahapan lain yang dapat diserahkan sepenuhnya pada tenaga komputasi mesin murni, rekayasa fitur sangat menuntut perpaduan yang harmonis antara keterampilan teknis pemrograman dan pemahaman domain bisnis yang mendalam dari pakarnya (Zheng & Casari, 2018). Pada esensinya, model pembelajaran mesin tidak memiliki nalar kognitif layaknya manusia; algoritma hanya mampu mengkalkulasi dan mencari pola dari apa yang secara eksplisit disuapkan kepadanya. Oleh sebab itu, analis harus jeli memilih variabel yang paling relevan, menciptakan kombinasi metrik turunan baru, atau mentransformasi variabel mentah sedemikian rupa agar struktur hierarki yang tersembunyi dari data tersebut menjadi jauh lebih terekspos dan mudah diserap oleh model. Rekayasa fitur yang dieksekusi dengan brilian selalu menjadi faktor pembeda utama antara model prediktif yang berkinerja biasa-biasa saja dengan model yang memiliki tingkat presisi tingkat tinggi.

Assets of Takaza INNOVATIX LABS

4. Pemilihan Model

Memasuki tahapan komputasional, pemilihan model menuntut kebijaksanaan analitik yang tinggi karena kenyataannya tidak ada satu pun algoritma ajaib tunggal yang paling sempurna untuk menyelesaikan semua jenis masalah komputasi yang ada di dunia; sebuah prinsip universal yang dalam literatur akademik kecerdasan buatan dikenal dengan sebutan Teorema *No Free Lunch* (Wolpert, 1996). Tim pengembang harus dengan cermat mempertimbangkan dan membandingkan secara paralel berbagai macam arsitektur algoritma, mulai dari yang paling sederhana dan mudah diinterpretasikan seperti

regresi logistik, hingga metode ansambel gabungan yang sangat rumit seperti hutan acak atau *Gradient Boosting*. Pemilihan ini mutlak didasarkan pada karakteristik intrinsik dari bentuk data yang dimiliki, keterbatasan spesifikasi daya komputasi yang tersedia, serta tingkat kemudahan interpretasi atau kewajiban transparansi audit yang dituntut oleh pihak manajemen operasional dan regulator pemerintah (Molnar, 2020). Dalam banyak kasus industri yang sangat teregulasi seperti perbankan, model yang sedikit kurang akurat namun alur keputusannya mudah dijelaskan secara manusiawi jauh lebih disukai daripada model kotak hitam canggih yang cara kerja pengambil keputusannya tidak dapat dipertanggungjawabkan sama sekali.

5. Pelatihan (Training) & Evaluasi

Begitu kandidat arsitektur model dipilih, siklus pengembangan akan memasuki fase iterasi pelatihan dan evaluasi yang berjalan secara berkesinambungan. Dalam fase kritis ini, algoritma disuapi dengan puluhan ribu bahkan jutaan baris data historis agar sistem jaringan matematisnya dapat menyesuaikan bobot internalnya dan mengenali pola secara mandiri. Meskipun demikian, untuk menjamin bahwa model tersebut benar-benar belajar memahami inti permasalahan dan bukan sekadar menghafal mati keseluruhan data latih, performanya harus terus diuji secara ketat dan disimulasikan menggunakan sekumpulan data pengujian baru yang belum pernah dilihat sama sekali oleh model selama proses pelatihan awal berlangsung (Goodfellow et al., 2016). Berbagai eksperimen validasi silang dan penyetelan parameter hiper dilakukan berulang kali secara otomatis maupun manual dalam tahap ini untuk menemukan konfigurasi pengaturan paling optimal. Proses ini

bertindak sebagai penjaga gawang untuk memastikan model tidak jatuh ke dalam perangkap *overfitting* di mana model terlalu kaku terhadap data lama, atau sebaliknya mengalami *underfitting* di mana model terlalu dangkal dan gagal menangkap kompleksitas fenomena yang sedang diteliti.

6. Deployment & Monitoring

Akhirnya, keseluruhan siklus ini memuncak pada tahap *Deployment* atau peluncuran model ke sistem produksi, yang menandai transisi nyata dari lingkungan laboratorium data yang sangat terkendali menuju kejamnya dinamika lalu lintas data di dunia nyata. Mengimplementasikan model ke dalam arsitektur sistem perangkat lunak yang hidup mengharuskan adanya kolaborasi yang erat dan mulus antara ilmuwan data dan insinyur perangkat lunak, guna menjamin bahwa model algoritma dapat merespons dan memproses permintaan prediksi dari ribuan pengguna secara instan atau *real-time* tanpa mengganggu kestabilan peladen (Sculley et al., 2015). Kendati demikian, tanggung jawab tim analitik sama sekali tidak berhenti saat peluncuran selesai dilakukan. Mengingat dunia nyata terus bergerak dan tren perilaku manusia yang menjadi sumber data terus mengalami pergeseran ekstrem, data-data baru lambat laun pasti akan memiliki struktur pola yang menyimpang dari data awal, memunculkan fenomena degradasi performa yang disebut sebagai *data drift*. Untuk mengantisipasi keruntuhan sistem ini, pengawasan sistem infrastruktur pemeliharaan harus dibangun sedemikian rupa agar performa algoritma selalu dimonitor, memastikan model dapat dilatih ulang secara periodik

Assets of Takaza INNOVATIX LABS

menggunakan data terkini sehingga keandalan prediksinya tetap tangguh melawan ujian waktu.

D. Dataset Training dan Testing

Salah satu kesalahan paling fatal dalam pengembangan *Machine Learning* adalah mengevaluasi model menggunakan data yang sama dengan yang digunakan untuk melatihnya. Hal ini akan menyebabkan *overfitting*, yaitu kondisi di mana model hanya "menghafal" data pelatihan (termasuk *noise* di dalamnya) tetapi gagal melakukan generalisasi pada data dunia nyata yang baru. Untuk mencegah hal ini, dataset harus dibagi menjadi dua bagian utama: *Training Set* dan *Testing Set* (Hastie et al., 2009).

Training Set digunakan oleh model untuk mempelajari pola dan menyesuaikan bobot (parameter) internalnya. Sebaliknya, *Testing Set* (atau seringkali *Validation Set*) berperan sebagai "ujian akhir" bagi model. Karena model tidak pernah melihat data *testing* selama proses pelatihan, kinerja model pada dataset ini memberikan estimasi yang jujur tentang bagaimana model akan berperforma di masa depan. Praktik umum yang sering digunakan untuk dataset yang terbatas adalah *k-fold cross-validation*, di mana data dibagi menjadi beberapa bagian (*folds*) secara rotasi untuk memastikan bahwa kinerja model konsisten di berbagai subset data dan tidak bergantung pada kebetulan pembagian data.

E. Evaluasi Model

Bagaimana kita tahu apakah sebuah model sudah "baik"? Evaluasi model harus disesuaikan dengan jenis masalah yang diselesaikan karena satu metrik tidak bisa menjelaskan segalanya. Untuk masalah klasifikasi, kita

tidak bisa hanya mengandalkan *accuracy* jika data tidak seimbang (*imbalanced data*). Kita harus menggunakan metrik seperti *Precision, Recall,* dan *F1-Score*. Misalnya, dalam deteksi kanker atau deteksi transaksi penipuan, *Recall* lebih penting daripada akurasi karena kita tidak ingin melewatkan satu pun kasus positif (menghindari *false negative*), meskipun mungkin ada sedikit *false positive* yang dihasilkan.

Sedangkan untuk masalah regresi, kita menggunakan metrik seperti *Mean Absolute Error* (MAE) yang memberikan interpretasi kesalahan rata-rata dalam unit yang sama dengan data target, atau *Mean Squared Error* (MSE) yang memberikan penalti lebih besar pada kesalahan besar. Penggunaan *Confusion Matrix* juga sangat disarankan untuk memvisualisasikan di mana model membuat kesalahan klasifikasi, sehingga pengembang dapat melakukan perbaikan yang lebih spesifik pada bagian data yang sulit dipelajari oleh model, bukan sekadar melihat satu angka performa global saja (Bishop, 2006).

Assets of Takaza INNOVATIX LABS

F. Studi kasus

Prediksi Churn Pelanggan

Sebuah perusahaan telekomunikasi menghadapi masalah tingginya tingkat pelanggan yang berhenti berlangganan (*churn*). Perusahaan ingin menggunakan *Machine Learning* untuk memprediksi pelanggan mana yang akan *churn* dalam 30 hari ke depan agar tim pemasaran dapat melakukan intervensi.

- **Definisi:** Masalah klasifikasi biner (*Churn* vs *Tidak Churn*).
- **Data:** Histori penggunaan data, durasi kontrak, frekuensi keluhan pelanggan, dan tagihan bulanan.

- **Preprocessing:** Menangani data yang tidak seimbang (*imbalance*) karena jumlah pelanggan yang *churn* biasanya jauh lebih sedikit daripada pelanggan setia.
- **Model:** Menggunakan algoritma *Random Forest* karena kemampuannya menangani hubungan non-linear antar variabel yang kompleks.
- **Evaluasi:** Fokus pada *Recall*. Perusahaan lebih memilih menebak pelanggan akan *churn* (padahal tidak) daripada melewatkan pelanggan yang benar-benar akan *churn* (melewatkan pelanggan *churn* jauh lebih mahal secara bisnis).
- **Hasil:** Perusahaan dapat memberikan promosi proaktif atau penawaran khusus kepada pelanggan dengan probabilitas *churn* tinggi sebelum mereka benar-benar membatalkan langganan. Strategi berbasis model ini terbukti meningkatkan retensi pelanggan hingga 12% dalam satu tahun fiskal, menunjukkan nilai bisnis nyata dari penerapan *Machine Learning*.

Assets of Takaza INNOVATIX LABS

BAB XI
PENGANTAR MACHINE LEARNING

Supervised Learning atau pembelajaran terbimbing merupakan paradigma paling fundamental dalam *Machine Learning* yang menjadi tulang punggung bagi mayoritas aplikasi kecerdasan buatan di sektor industri saat ini. Konsep dasar dari *supervised learning* adalah melatih algoritma menggunakan dataset berlabel (*labeled data*), di mana setiap instans data telah memiliki "kunci jawaban" yang tepat atau target variabel yang ingin diprediksi. Selama proses pelatihan, algoritma akan mempelajari pemetaan fungsi antara fitur input (X) dan variabel target (y) dengan meminimalkan selisih antara prediksi model dan nilai aktual melalui fungsi kerugian (*loss function*). Sebagaimana dijelaskan oleh Hastie et al. (2009), keberhasilan model dalam *supervised learning* sangat bergantung pada representasi fitur yang dipilih serta kuantitas data yang tersedia untuk memastikan model mampu melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya.

Dalam konteks bisnis, *supervised learning* berfungsi sebagai alat prediktif yang memungkinkan organisasi untuk melakukan estimasi kuantitatif maupun klasifikasi kualitatif. Keputusan strategis seperti penentuan harga produk, deteksi risiko kredit, pemeliharaan mesin preventif, hingga prediksi loyalitas pelanggan, semuanya berakar pada kemampuan algoritma untuk mengekstraksi pola dari data historis. Bab ini akan membedah secara mendalam empat pilar algoritma *supervised learning* yang

paling sering digunakan dalam industri: Regresi Linear, Regresi Logistik, *Decision Tree*, dan *Random Forest*. Kita akan meninjau bagaimana setiap algoritma bekerja, kapan harus menggunakannya, dan bagaimana mengevaluasi performanya secara objektif untuk memastikan hasil analisis memberikan nilai ekonomi yang nyata bagi perusahaan.

A. Konsep Machine Learning

Regresi Linear adalah metode statistik yang digunakan untuk memodelkan hubungan linear antara variabel dependen kontinu (y) dan satu atau lebih variabel independen (x). Persamaan dasarnya sering dituliskan sebagai $y = \beta0 + \beta1\ x + \varepsilon$, di mana $\beta0$ merupakan *intercept*, $\beta1$ adalah koefisien kemiringan, dan ϵ merepresentasikan *error* atau residu yang tidak dapat dijelaskan oleh model. Tujuan utama dari algoritma ini adalah mencari garis "terbaik" (*line of best fit*) yang meminimalkan jumlah kuadrat dari residu antara prediksi model dan titik data aktual (metode *Ordinary Least Squares*). Sebagaimana ditegaskan oleh Montgomery et al. (2012), regresi linear bukan sekadar alat prediksi, melainkan juga instrumen diagnostik yang kuat untuk memahami kekuatan hubungan antarvariabel bisnis.

Dalam implementasi praktisnya, regresi linear memerlukan pemenuhan asumsi klasik seperti linearitas, independensi residu, homoskedastisitas (varians residu yang konstan), dan normalitas residu. Jika asumsi-asumsi ini dilanggar misalnya ketika data memiliki pola non-linear atau *outlier* yang ekstrem model akan menghasilkan prediksi yang tidak akurat dan bias. Penting bagi analis data untuk melakukan transformasi data atau menggunakan teknik regresi yang lebih robust jika data tidak memenuhi

asumsi tersebut. Meskipun tergolong algoritma yang sederhana, interpretabilitas regresi linear sangat tinggi, menjadikannya pilihan utama ketika pemangku kepentingan bisnis membutuhkan alasan logis di balik setiap prediksi, bukan sekadar "kotak hitam" yang memberikan jawaban tanpa penjelasan.

Selain itu, perlu dicatat bahwa dalam situasi di mana terdapat banyak variabel prediktor, kita sering beralih ke *Multiple Linear Regression*. Tantangan utama di sini adalah *multikolinieritas*, yaitu ketika variabel-variabel independen saling berkorelasi kuat satu sama lain, yang dapat mendistorsi nilai koefisien regresi. Analis harus menggunakan teknik seperti *Variance Inflation Factor* (VIF) untuk mendeteksi dan menghapus fitur yang redundan. Dengan penggunaan yang tepat, regresi linear mampu memberikan *insight* berharga, seperti mengukur elastisitas harga atau memprediksi pengaruh biaya pemasaran terhadap volume penjualan jangka panjang, menjadikannya fondasi wajib dalam *data-driven decision making*.

Assets of Takaza INNOVATIX LABS

B. Regresi Logistik

Meskipun namanya mengandung kata "regresi", Regresi Logistik sebenarnya adalah algoritma klasifikasi yang digunakan ketika variabel target bersifat kategorikal, biasanya biner (misalnya: Ya/Tidak, Lulus/Gagal, *Churn*/*Stay*). Regresi logistik tidak mencoba memprediksi nilai kontinu langsung, melainkan memprediksi probabilitas (antara 0 dan 1) bahwa suatu pengamatan termasuk ke dalam kategori tertentu. Hal ini dicapai dengan mentransformasi *log-odds* menggunakan fungsi sigmoid (fungsi logistik), yang memastikan output prediksi selalu berada dalam rentang [0,1] (James et al., 2013). Bentuk fungsi ini adalah P (y=1\vert{}x) = \frac{1}{1 + e^{-

(\beta_0 + \beta_1x)}}$, yang secara grafis membentuk kurva berbentuk huruf S yang sangat efisien dalam membedakan dua kelas.

Dalam praktiknya, untuk mengubah probabilitas menjadi klasifikasi biner, analis harus menetapkan ambang batas (*threshold*), yang secara *default* sering dipasang pada 0.5. Namun, pemilihan *threshold* ini sangat bergantung pada konteks bisnis dan toleransi risiko. Jika sebuah perusahaan mendeteksi transaksi penipuan, mereka mungkin lebih memilih ambang batas yang lebih rendah untuk meningkatkan sensitivitas, meskipun meningkatkan risiko *false positive*. Fleksibilitas dalam penentuan *decision boundary* inilah yang membuat regresi logistik menjadi standar industri dalam berbagai aplikasi klasifikasi seperti *credit scoring* dan *medical screening.*

Kelebihan utama dari regresi logistik terletak pada kemudahan interpretasinya melalui rasio peluang (*odds ratio*). Koefisien yang dihasilkan menunjukkan seberapa besar pengaruh variabel input terhadap kemungkinan terjadinya peristiwa. Selain itu, model ini relatif efisien secara komputasi dan tidak membutuhkan asumsi normalitas pada variabel input, menjadikannya sangat populer untuk dataset berskala besar. Namun, regresi logistik memiliki kelemahan yakni kesulitan dalam menangani hubungan non-linear yang kompleks secara langsung; untuk mengatasinya, analis seringkali perlu menambahkan fitur polinomial atau interaksi antar fitur sebelum memasukkannya ke dalam model.

C. Decision Tree

Decision Tree atau pohon keputusan adalah algoritma *supervised learning* non-parametrik yang bekerja dengan memecah dataset menjadi sub-

himpunan yang lebih kecil secara berulang berdasarkan aturan *if-then-else* yang dipelajari dari data. Struktur pohon ini terdiri dari *root node* (akar), *internal nodes* (keputusan), dan *leaf nodes* (hasil klasifikasi atau nilai prediksi). Keputusan untuk memecah simpul dilakukan berdasarkan metrik kemurnian (*purity metrics*) seperti *Gini Impurity* atau *Information Gain* (Entropi Shannon), yang bertujuan untuk memaksimalkan homogenitas dalam setiap kelompok hasil (Quinlan, 1986). Kemampuan pohon keputusan untuk meniru cara berpikir manusia dalam mengambil keputusan menjadikannya salah satu algoritma yang paling mudah dipahami oleh audiens non-teknis.

Salah satu tantangan terbesar dalam *decision tree* adalah kecenderungan model untuk *overfitting*. Karena sifatnya yang rekursif, pohon keputusan dapat terus tumbuh hingga mencakup setiap detail kecil dan *noise* dalam data pelatihan, yang berakibat pada kegagalan prediksi saat dihadapkan pada data baru. Untuk mengatasi masalah ini, teknik *pruning* (pemangkasan) atau pembatasan kedalaman pohon (*max depth*) sangat disarankan. Dengan membatasi kompleksitas pohon, kita memaksa model untuk menangkap pola umum yang lebih luas daripada menghafal setiap pengecualian dalam dataset.

Secara teknis, algoritma ini sangat tangguh dalam menangani data bertipe campuran (numerik dan kategorikal) tanpa memerlukan normalisasi atau standarisasi yang ketat seperti pada algoritma berbasis jarak (misalnya KNN). Selain itu, *decision tree* memiliki fitur bawaan untuk melakukan seleksi fitur secara otomatis, karena simpul-simpul teratas secara alami akan memilih variabel yang paling berpengaruh secara informatif. Namun, perlu diwaspadai bahwa perubahan kecil pada data dapat menghasilkan struktur

pohon yang sangat berbeda, sebuah karakteristik yang dikenal sebagai instabilitas model, yang kemudian memotivasi pengembangan metode *ensemble* seperti *Random Forest*.

D. Random Forest

Random Forest adalah algoritma *ensemble learning* yang membangun "hutan" dari sekumpulan *decision tree* yang saling independen dan kemudian menggabungkan hasil prediksi mereka untuk mendapatkan satu output yang lebih akurat dan stabil. Prinsip dasarnya adalah *bagging* (*Bootstrap Aggregating*), di mana setiap pohon dilatih menggunakan subset data yang berbeda (diambil secara acak dengan penggantian) dan menggunakan subset fitur yang berbeda pada setiap *split*. Teknik ini sangat efektif dalam mengurangi varians model tanpa meningkatkan bias secara signifikan, sebuah pendekatan yang sering disebut sebagai "kebijaksanaan orang banyak" (*wisdom of the crowd*) dalam statistik (Breiman, 2001).

Keunggulan utama *Random Forest* adalah ketahanannya terhadap *overfitting* yang sering menjadi kelemahan pohon keputusan tunggal. Karena model melakukan agregasi (melalui *voting* untuk klasifikasi atau *averaging* untuk regresi), kesalahan yang dibuat oleh satu pohon dapat dikompensasi oleh pohon lainnya, sehingga menghasilkan prediksi yang jauh lebih robust. Algoritma ini juga sangat baik dalam menangani dataset dengan banyak variabel yang tidak relevan atau *missing values*, karena mekanisme seleksi fitur acak memastikan bahwa pohon tidak hanya bergantung pada fitur dominan yang mungkin bias.

Meskipun performanya seringkali superior, *Random Forest* sering dianggap sebagai model *black box* karena interpretasinya lebih sulit

dibandingkan pohon keputusan tunggal. Kita tidak lagi bisa melihat satu "peta jalan" keputusan yang jelas, melainkan harus mengandalkan metrik seperti *feature importance* untuk memahami variabel mana yang paling berkontribusi terhadap hasil akhir. Bagi analis, *Random Forest* adalah "pisau tentara Swiss" dalam dunia *data science* seringkali menjadi algoritma pertama yang dicoba karena kemampuannya memberikan hasil yang kompetitif pada berbagai jenis data tanpa memerlukan penyetelan parameter (*hyperparameter tuning*) yang sangat rumit.

E. Evaluasi Kinerja Model

Berikut adalah versi penjelasan mengenai Evaluasi Kinerja Model yang telah diperluas, disusun dalam paragraf-paragraf panjang tanpa menggunakan rumus matematika, dan dilengkapi dengan *body note* (kutipan dalam teks) layaknya standar penulisan buku referensi akademik:

Evaluasi model bukanlah sekadar garis akhir atau formalitas pemenuhan syarat dalam sebuah alur kerja sains data, melainkan fase paling krusial yang menentukan apakah sebuah algoritma layak diterapkan di dunia nyata. Banyak praktisi pemula sering kali terjebak pada euforia ketika model selesai dilatih, namun gagal menyadari bahwa model yang tampak brilian di lingkungan pengembangan bisa menjadi bencana besar jika langsung diimplementasikan tanpa proses evaluasi yang ketat (Provost & Fawcett, 2013). Proses evaluasi ini secara esensial berfungsi sebagai jembatan penyambung antara metrik komputasi di layar komputer dan dampak bisnis di lapangan. Tujuannya adalah untuk memastikan bahwa sistem kecerdasan buatan yang dibangun benar-benar menyelesaikan masalah secara efektif,

bukan justru menciptakan risiko operasional baru, bias sistematis, atau bahkan kerugian finansial akibat prediksi yang meleset dari ekspektasi awal (Hastie et al., 2009).

Dalam ranah klasifikasi, di mana sebuah model bertugas untuk mengkategorikan data ke dalam kelompok-kelompok tertentu, tahapan evaluasi biasanya selalu berpusat pada sebuah tabel performa yang sangat terkenal dengan sebutan Matriks Kebingungan atau *Confusion Matrix* (Ting, 2011). Konsep konseptual dari matriks ini adalah memetakan seluruh tebakan yang dihasilkan oleh model ke dalam empat skenario fundamental yang sangat memengaruhi langkah strategis bisnis selanjutnya. Keempat skenario tersebut meliputi *True Positives* ketika model berhasil menebak kelas positif dengan tepat, *True Negatives* saat model dengan benar mengenali kelas negatif, *False Positives* yang merupakan kondisi alarm palsu di mana model menebak adanya kejadian padahal aslinya tidak ada, serta *False Negatives* yang merupakan kondisi paling berbahaya karena model gagal mendeteksi atau melewatkan sebuah kejadian krusial yang sebenarnya nyata-nyata terjadi (Fawcett, 2006). Pemahaman yang komprehensif dan mendalam terhadap keempat kuadran dasar ini adalah fondasi wajib sebelum seorang analis merumuskan metrik evaluasi turunan lainnya.

Berangkat dari matriks dasar tersebut, metrik turunan yang paling sering disebut pertama kali oleh masyarakat umum adalah Akurasi, yang sekadar menghitung persentase total keseluruhan tebakan yang benar dibandingkan dengan total seluruh data. Sayangnya, metrik akurasi sering kali menjadi jebakan ilusi performa yang sangat berbahaya, terutama jika analis berhadapan dengan kumpulan data yang sangat tidak seimbang atau

imbalanced dataset di dunia nyata (Chawla et al., 2002). Sebagai ilustrasi, dalam sebuah sistem deteksi penipuan transaksi perbankan di mana persentase penipuan asli biasanya sangatlah kecil, sebuah model pemalas yang selalu menebak "tidak ada penipuan" pada setiap transaksi akan terlihat memiliki akurasi di atas sembilan puluh persen. Model tersebut seolah-olah terlihat berkinerja luar biasa di atas kertas, padahal kenyataannya sama sekali tidak berguna dalam menangkap satu pun penipu di lapangan.

Oleh karena kelemahan akurasi tersebut, para analis data yang berpengalaman akan selalu beralih pada dua metrik yang jauh lebih spesifik, yaitu Presisi dan *Recall* atau Sensitivitas (Sokolova & Lapalme, 2009). Metrik Presisi berfungsi untuk mengukur seberapa tajam dan dapat dipercayanya setiap peringatan positif yang dikeluarkan oleh model; ini menjawab pertanyaan tentang berapa banyak kasus positif tebakan model yang memang benar-benar positif di dunia nyata. Di sisi lain, metrik *Recall* berfungsi untuk mengukur daya tangkap atau jangkauan kewaspadaan model, guna memastikan agar tidak ada satu pun kasus positif penting yang lolos dari radarnya (Powers, 2011). Pemilihan antara memprioritaskan Presisi atau *Recall* sangat bergantung pada konteks masalah bisnis yang dihadapi; misalnya, filter *spam email* lebih mengutamakan Presisi agar pesan penting tidak masuk ke folder sampah, sementara diagnosis medis akan selalu mengutamakan *Recall* agar penyakit kritis pasien tidak terlewatkan.

Dalam praktiknya di lapangan operasional, sering kali terjadi fenomena tarik-ulur atau *trade-off* yang sangat ketat antara nilai Presisi dan *Recall* (Buckland & Gey, 1994). Jika seorang pengembang merancang sebuah model untuk bersikap sangat berhati-hati dan konservatif agar tidak memberikan peringatan palsu sama sekali, model tersebut secara otomatis

akan menjadi kurang peka dan berisiko melewatkan beberapa kejadian positif yang sebenarnya. Sebaliknya, jika model tersebut disetel agar menjadi sangat sensitif untuk menangkap seluruh kemungkinan yang ada, model tersebut pasti akan menghasilkan banyak sekali peringatan palsu yang mengganggu efisiensi kerja. Untuk menjembatani dilema klasik ini, para ilmuwan data secara rutin menggunakan metrik penyeimbang yang disebut sebagai *F1-Score* (Van Rijsbergen, 1979). Metrik penengah ini bertindak sebagai sebuah standar evaluasi yang sangat ketat, di mana nilainya akan langsung anjlok drastis jika salah satu dari Presisi atau *Recall* memiliki performa yang buruk, sehingga metrik ini mampu memberikan gambaran keseimbangan model yang jauh lebih jujur dibandingkan sekadar melihat akurasi biasa.

Sementara itu, apabila diskusi beralih ke ranah pemodelan regresi yang tujuannya adalah memprediksi sebuah angka kontinu seperti prediksi harga properti, perkiraan suhu cuaca, atau proyeksi jumlah penjualan produk bulan depan, metodologi evaluasinya berubah drastis secara fundamental (James et al., 2013). Dalam regresi, evaluasi tidak lagi berbicara tentang benar atau salah secara mutlak seperti mengkategorikan hitam dan putih, melainkan berbicara tentang seberapa jauh nilai prediksi tersebut menyimpang atau meleset dari kenyataan angka yang sebenarnya. Metrik evaluasi yang paling intuitif dan mudah dipahami dalam konteks ini adalah rata-rata kesalahan absolut atau *Mean Absolute Error* (MAE). Keunggulan utama dari metrik MAE adalah kemampuannya memberikan gambaran rata-rata penyimpangan dalam satuan ukur yang sama persis dengan data aslinya, sehingga laporan evaluasi menjadi sangat mudah dipahami saat dipresentasikan kepada para pemangku kepentingan bisnis yang mungkin

Assets of Takaza INNOVATIX LABS

tidak memiliki latar belakang keilmuan teknis yang mendalam (Willmott & Matsuura, 2005).

Meskipun MAE sangat mudah diinterpretasikan, dalam berbagai skenario bisnis yang memiliki tingkat risiko tinggi, perusahaan sangat menghindari adanya tebakan model yang meleset terlalu jauh atau ekstrem. Untuk mengakomodasi kebutuhan kehati-hatian tersebut, para analis akan lebih memilih menggunakan *Mean Squared Error* (MSE) atau akar kuadratnya yang dikenal sebagai *Root Mean Squared Error* (RMSE) (Chai & Draxler, 2014). Mekanisme perhitungan metrik-metrik regresi kuadratik ini dirancang secara khusus untuk memberikan hukuman atau penalti matematis yang berlipat ganda pada setiap tebakan model yang meleset dengan jarak yang sangat lebar. Dengan menggunakan RMSE sebagai standar evaluasi utama, algoritma secara internal akan dipaksa untuk terus berlatih memprioritaskan stabilitas prediksi secara merata di seluruh baris data pelatihan, serta secara aktif menghindari pembuatan prediksi pencilan yang sangat melenceng yang dapat berakibat fatal bagi keputusan investasi perusahaan.

Selain menggunakan metrik yang berfokus pada penghitungan seberapa besar jarak kesalahan prediksi, evaluasi performa model regresi juga sangat sering menyertakan sebuah metrik proporsional yang populer disebut sebagai Koefisien Determinasi atau *R-Squared* (Cameron & Windmeijer, 1997). Alih-alih menghitung jumlah kerugian atau kesalahan absolut, metrik ini mencoba mengevaluasi kualitas model dari sudut pandang yang berbeda, yakni seberapa tangguh kinerja model yang dibangun jika dibandingkan dengan model paling sederhana yang hanya menebak menggunakan nilai rata-rata dari seluruh data. Nilai *R-Squared* yang tinggi

mengindikasikan bahwa model prediktif telah berhasil mendeteksi pola yang tersembunyi dan mampu menjelaskan hampir seluruh variasi pergerakan data dengan sangat baik. Walaupun terdengar menjanjikan, analis akademik harus tetap waspada karena tingginya nilai ini tidak secara otomatis membebaskan model dari potensi bias atau pemilihan fitur variabel yang keliru (Montgomery et al., 2021).

Lebih jauh lagi, perlu disadari bahwa seluruh metrik evaluasi absolut yang telah dibahas sebelumnya tidak akan memiliki tingkat keabsahan metodologis yang solid jika data pengujiannya tidak dipersiapkan dengan kerangka kerja yang kokoh. Di sinilah metode validasi silang atau *k-fold Cross-Validation* memainkan peranan yang esensial dalam memastikan stabilitas performa sistem kecerdasan buatan (Kohavi, 1995). Daripada hanya membelah data menjadi satu set pelatihan dan satu set pengujian secara kaku yang hasilnya sangat rentan dipengaruhi oleh kebetulan acak, metode validasi silang ini secara komputasional memecah data menjadi banyak lipatan bagian. Model kemudian dilatih dan diuji berulang kali secara bergiliran menggunakan lipatan data yang berbeda-beda, sehingga menjamin bahwa model berkinerja tinggi di berbagai variasi skenario, bukan sekadar menghafal informasi spesifik pada sebagian kecil sampel data (Arlot & Celisse, 2010).

Sebagai pelengkap evaluasi tingkat lanjut dan lapisan pertahanan analitik pamungkas, khususnya untuk kasus klasifikasi biner yang kompleks, seorang praktisi pembelajaran mesin yang andal umumnya tidak akan melewatkan proses analisis menggunakan kurva *ROC-AUC* atau *Receiver Operating Characteristic - Area Under Curve* (Bradley, 1997). Instrumen analitik berbentuk visual ini sangat luar biasa karena mampu mengevaluasi

ketangguhan performa model secara menyeluruh pada rentetan berbagai skenario ambang batas probabilitas yang sangat dinamis. Luasan area yang berada di bawah kurva tersebut pada intinya merepresentasikan kemampuan fundamental dan intrinsik dari model itu sendiri dalam memisahkan dan membedakan antara contoh data kelas positif dan kelas negatif secara meyakinkan (Fawcett, 2006).

Kesimpulannya, evaluasi kinerja model tidak boleh dipandang sebagai prosedur matematis tunggal, melainkan sebuah ekosistem pengujian komprehensif yang berlapis, transparan, objektif, dan senantiasa disesuaikan dengan konteks domain pengaplikasiannya. Pemilihan dan penentuan prioritas metrik harus dilakukan dengan tingkat presisi akademik tinggi yang selaras dengan tujuan strategis operasional, selalu memperhitungkan dampak turunan dari setiap bentuk kesalahan prediksi di dunia nyata. Penerapan kerangka kerja evaluasi yang ketat dan holistik ini adalah garansi absolut yang memastikan bahwa setiap model analitik yang diimplementasikan akan benar-benar memberikan nilai tambah pengetahuan yang nyata, tangguh, dan dapat dipertanggungjawabkan secara ilmiah (Provost & Fawcett, 2013).

Assets of Takaza INNOVATIX LABS

F. Studi kasus

Sebuah lembaga keuangan ingin mengotomatisasi proses persetujuan pinjaman mereka untuk mengurangi tingkat gagal bayar (*default rate*). Mereka memiliki dataset historis yang mencakup riwayat pembayaran, jumlah pendapatan, durasi pekerjaan, dan total hutang pelanggan. Permasalahan utama adalah menyeimbangkan antara persetujuan pinjaman (profitabilitas) dengan risiko gagal bayar (kerugian). Jika bank terlalu ketat,

mereka kehilangan nasabah potensial; jika terlalu longgar, mereka menghadapi risiko kerugian finansial yang besar.

Metodologi yang dipilih dimulai dengan pembersihan data dan penanganan *imbalanced data* (karena kasus gagal bayar biasanya jauh lebih sedikit daripada pelanggan yang patuh). Tim data kemudian melatih model *Random Forest* karena kemampuannya menangani hubungan non-linear antar variabel (misalnya, pengaruh pendapatan terhadap kemampuan bayar mungkin tidak linear). Sebagai perbandingan, model *Regresi Logistik* juga dikembangkan untuk menyediakan *baseline* yang lebih mudah diinterpretasi oleh tim kepatuhan bank.

Hasil evaluasi menunjukkan bahwa *Random Forest* memberikan skor *AUC-ROC* sebesar 0.88, secara signifikan mengungguli model logistik. Namun, saat diuji dengan *Cost-Benefit Matrix*, tim menyadari bahwa kesalahan klasifikasi "nasabah yang sebenarnya akan gagal bayar tetapi diprediksi aman" (*False Negative*) memiliki biaya 10 kali lebih besar daripada kesalahan klasifikasi "nasabah yang aman tetapi diprediksi gagal bayar" (*False Positive*). Oleh karena itu, tim menyesuaikan *threshold* pada model untuk memprioritaskan *Recall* pada kelas gagal bayar.

Keputusan akhir diambil dengan mengintegrasikan model ke dalam sistem *underwriting* bank. Dengan menggunakan pendekatan *supervised learning* ini, bank berhasil menurunkan tingkat gagal bayar sebesar 20% dalam enam bulan pertama tanpa mengurangi volume persetujuan pinjaman secara drastis. Studi kasus ini membuktikan bahwa *supervised learning* bukan hanya soal akurasi statistik, tetapi juga tentang bagaimana mengintegrasikan pemahaman bisnis ke dalam desain metrik evaluasi model untuk mencapai tujuan ekonomi yang optimal

BAB XII
UNSUPERVISED LEARNING

Setelah pada bab sebelumnya kita membahas *Supervised Learning* yang bergantung pada label data, kini kita melangkah ke domain yang menantang namun sangat kaya akan penemuan: *Unsupervised Learning* (pembelajaran tak terbimbing). Berbeda dengan *Supervised Learning* yang memiliki "guru" atau label untuk memandu proses pembelajaran, *Unsupervised Learning* beroperasi dalam lingkungan tanpa arahan eksplisit. Di sini, algoritma diberikan dataset yang tidak berlabel dan ditugaskan untuk menemukan struktur, pola, atau hubungan tersembunyi di dalamnya secara mandiri. Sebagaimana dinyatakan oleh Bishop (2006), tujuan utama dari pembelajaran tak terbimbing adalah untuk memodelkan kepadatan probabilitas yang mendasari data atau untuk melakukan pengelompokan (kategorisasi) yang secara alami ada dalam kumpulan data yang besar dan kompleks.

Dalam dunia bisnis, *Unsupervised Learning* adalah instrumen eksplorasi yang sangat kuat. Algoritma ini memungkinkan perusahaan untuk memahami basis pelanggan mereka tanpa praduga, menemukan anomali dalam transaksi keuangan yang mungkin merupakan indikasi kecurangan, atau menyederhanakan data dengan ribuan variabel menjadi dimensi yang lebih mudah dikelola. Bab ini akan membedah algoritma utama dalam *Unsupervised Learning*, mulai dari teknik *clustering* hingga analisis aturan asosiasi. Kita akan mempelajari bagaimana data yang tampak acak

sebenarnya memiliki struktur internal yang, jika berhasil diungkap, dapat memberikan keunggulan kompetitif yang signifikan bagi organisasi melalui segmentasi pasar yang lebih tajam dan sistem rekomendasi yang lebih personal (Hastie et al., 2009).

A. Clustering

Clustering atau pengelompokan adalah proses membagi kumpulan data menjadi beberapa kelompok (klaster) sedemikian rupa sehingga objek-objek dalam satu kelompok memiliki kemiripan yang tinggi satu sama lain, sementara objek di kelompok yang berbeda memiliki perbedaan yang signifikan. Berbeda dengan klasifikasi, di mana kategori sudah ditentukan sebelumnya, *clustering* bersifat deskriptif dan eksploratif; kita membiarkan data "berbicara sendiri" untuk menentukan batasan kategorinya. Dasar teknis dari proses ini adalah pengukuran jarak atau kesamaan (*distance metrics*), seperti jarak *Euclidean*, *Manhattan*, atau *Cosine Similarity*. Pilihan metrik jarak ini sangat krusial karena ia menentukan bagaimana definisi "kemiripan" dioperasionalkan dalam ruang data multidimensi (James et al., 2013).

Tantangan mendasar dalam *clustering* adalah fakta bahwa tidak ada ukuran objektif mutlak untuk kebenaran hasil pengelompokan karena ketiadaan label data. Oleh karena itu, efektivitas sebuah model *clustering* sering diukur berdasarkan kohesi (seberapa dekat titik-titik dalam klaster) dan separasi (seberapa jauh jarak antar klaster). Dalam konteks bisnis, ini berarti analis harus memiliki pemahaman mendalam tentang domain data untuk memvalidasi apakah hasil klaster yang terbentuk benar-benar merepresentasikan fenomena nyata. Sebagai contoh, jika kita melakukan

clustering pada pelanggan, kita harus memastikan bahwa klaster yang terbentuk bukan sekadar artefak matematis, melainkan memiliki karakteristik perilaku yang konsisten dan dapat ditindaklanjuti secara pemasaran.

Selain itu, *clustering* memerlukan penanganan data yang sangat hati-hati, terutama dalam hal penskalaan fitur. Karena banyak algoritma *clustering* berbasis pada perhitungan jarak, variabel dengan rentang nilai yang besar (seperti pendapatan) akan mendominasi variabel dengan rentang kecil (seperti usia), sehingga hasil klaster menjadi bias. Normalisasi atau standarisasi data (seperti *Z-score scaling*) adalah langkah wajib sebelum menjalankan algoritma. Dengan teknik yang tepat, *clustering* mampu mengubah data mentah yang kacau menjadi segmentasi yang terstruktur, memberikan landasan bagi strategi personalisasi yang lebih efektif dan efisien di tingkat operasional.

Assets of Takaza INNOVATIX LABS

B. K-Means

K-Means adalah algoritma *clustering* yang paling populer dan efisien karena kesederhanaan logikanya dalam mempartisi data menjadi K klaster yang tidak tumpang tindih. Prosesnya dimulai dengan menentukan jumlah klaster K secara manual, kemudian algoritma secara acak menempatkan titik pusat (*centroid*) di ruang data. Selanjutnya, setiap titik data akan ditugaskan ke *centroid* terdekat berdasarkan jarak *Euclidean*. Setelah semua titik ditugaskan, *centroid* akan dihitung ulang sebagai titik rata-rata dari semua data yang ada dalam klaster tersebut. Proses ini diulang secara iteratif hingga posisi *centroid* stabil (konvergen) atau tidak ada perubahan penugasan lagi (MacQueen, 1967).

Meskipun sangat populer, *K-Means* memiliki kelemahan signifikan terkait penentuan nilai K. Jika K dipilih terlalu kecil, kita akan kehilangan detail segmentasi; jika terlalu besar, kita akan mengalami *over-segmentation* yang tidak efisien. Teknik yang sering digunakan untuk menentukan K yang optimal adalah *Elbow Method* (Metode Siku), di mana kita memplot total *Within-Cluster Sum of Squares* (WCSS) terhadap berbagai nilai K. Titik di mana kurva penurunan WCSS mulai melandai (seperti bentuk siku tangan) dianggap sebagai jumlah klaster yang optimal secara matematis, karena penambahan klaster lebih lanjut memberikan keuntungan penurunan varians yang semakin kecil.

Kelemahan lain dari *K-Means* adalah sensitivitasnya terhadap *outlier* dan inisialisasi awal *centroid*. Karena *centroid* adalah nilai rata-rata, satu titik data yang ekstrem dapat menarik *centroid* menjauh dari posisi yang seharusnya, sehingga merusak kualitas klaster. Untuk memitigasi hal ini, praktisi sering menggunakan *K-Means++*, sebuah varian inisialisasi yang menempatkan *centroid* awal sejauh mungkin satu sama lain, atau melakukan *preprocessing* penghapusan *outlier* secara ketat sebelum pemodelan. Meskipun memiliki keterbatasan, kecepatan *K-Means* dalam menangani dataset yang masif menjadikannya pilihan utama untuk aplikasi *real-time* seperti segmentasi pengunjung situs web secara dinamis.

C. Hierarchical Clustering

Hierarchical Clustering adalah metode yang membangun hierarki klaster dalam bentuk pohon yang disebut dendrogram. Berbeda dengan *K-Means* yang membagi data dalam satu langkah iterasi, pendekatan ini bersifat bertingkat. Ada dua jenis utama: *Agglomerative* (bottom-up), di

mana setiap data dimulai sebagai satu klaster dan perlahan digabungkan, dan *Divisive* (top-down), di mana seluruh dataset dimulai sebagai satu klaster besar yang kemudian dipecah. Keunggulan utamanya adalah kita tidak perlu menentukan jumlah klaster K di awal, melainkan bisa memotong dendrogram pada tingkat kedalaman tertentu sesuai dengan kebutuhan interpretasi bisnis (Murtagh & Legendre, 2014).

Proses penggabungan dalam *agglomerative clustering* sangat bergantung pada kriteria *linkage* yang digunakan, seperti *single linkage* (jarak antar titik terdekat), *complete linkage* (jarak antar titik terjauh), atau *Ward's method* (meminimalkan varians dalam klaster). Pemilihan *linkage* ini akan menghasilkan struktur dendrogram yang berbeda secara drastis. *Single linkage* cenderung menghasilkan klaster yang panjang dan tipis (rantai), sedangkan *Ward's method* cenderung menghasilkan klaster yang lebih kompak dan bulat. Pilihan ini harus didasarkan pada karakteristik data yang sedang dianalisis; apakah kita mencari pengelompokan yang tersebar luas atau pengelompokan yang padat.

Kelebihan utama *Hierarchical Clustering* dibandingkan *K-Means* adalah kemampuannya memberikan gambaran relasi antar-objek yang lebih mendalam melalui struktur pohon. Ini sangat berharga dalam bidang seperti biologi (taksonomi) atau riset pasar di mana struktur hierarkis mungkin memang ada dalam data. Namun, kelemahannya terletak pada kompleksitas komputasinya yang tinggi, biasanya $O(n^3)$ atau $O(n^2 \log n)$, yang membuatnya kurang efisien dibandingkan *K-Means* untuk dataset dengan jutaan baris data. Oleh karena itu, metode ini biasanya diterapkan pada sampel data atau dataset berukuran menengah di mana detail relasi antar kelompok menjadi prioritas utama dibandingkan kecepatan pemrosesan.

Assets of Takaza INNOVATIX LABS

D. Association Rule

Association Rule Learning atau pembelajaran aturan asosiasi adalah teknik *unsupervised machine learning* yang dirancang untuk menemukan hubungan tersembunyi, pola, atau "aturan" menarik di antara variabel-variabel berskala besar dalam sebuah dataset. Sering disebut sebagai Market Basket Analysis (Analisis Keranjang Belanja), teknik ini sangat populer dalam industri ritel untuk menjawab pertanyaan strategis seperti: *"Jika pelanggan membeli produk A, seberapa besar kemungkinan mereka juga akan membeli produk B?"*.

Algoritma klasik yang menjadi pionir dalam teknik ini adalah Apriori (Agrawal et al., 1993). Algoritma ini bekerja dengan prinsip *bottom-up*, yaitu mencari *frequent itemsets* (kumpulan item yang sering muncul bersamaan) dalam database transaksi secara iteratif. Selain Apriori, saat ini juga banyak digunakan algoritma FP-Growth *(Frequent Pattern Growth)* yang dinilai lebih cepat dan efisien dalam memproses dataset raksasa karena menggunakan struktur pohon data (*tree-based*) sehingga tidak perlu memindai ulang database berkali-kali.

Dalam analisis asosiasi, untuk menentukan apakah sebuah aturan (misalnya: Jika membeli Roti $\rightarrow$ maka membeli Selai) benar-benar valid dan tidak terjadi hanya karena kebetulan, terdapat tiga metrik utama yang menjadi pilar evaluasi dasar, serta metrik tambahan untuk analisis tingkat lanjut:

1. **Support (Dukungan):**

   Mengukur seberapa sering suatu kombinasi item muncul dalam seluruh transaksi. Ini menunjukkan tingkat popularitas sebuah kombinasi produk. Aturan dengan nilai *support* yang terlalu rendah

seringkali diabaikan (dipangkas) karena dianggap tidak signifikan secara statistik atau hanya mewakili anomali pembelian.

Rumus: (Jumlah transaksi mengandung A dan B) / (Total seluruh transaksi)

2. **Confidence (Kepastian/Kepercayaan)**

   Mengukur seberapa sering item B muncul dalam transaksi yang sudah terkonfirmasi mengandung item A. Ini pada dasarnya adalah estimasi probabilitas bersyarat. Meskipun nilainya tinggi, *confidence* terkadang bisa menyesatkan jika item B ternyata memang produk yang sangat umum (seperti kantong plastik di minimarket).

   Rumus: (Jumlah transaksi mengandung A dan B) / (Jumlah transaksi mengandung A)

3. **Lift (Daya Angkat)**

   Metrik paling krusial untuk mengukur seberapa kuat hubungan antar item secara murni, dengan mempertimbangkan popularitas masing-masing item.

   - **Lift = 1:** Item A dan B saling independen (tidak ada hubungan).
   - **Lift > 1:** Hubungan positif (adanya A secara signifikan meningkatkan kemungkinan pembelian B). Semakin tinggi angkanya, semakin kuat hubungannya.
   - **Lift < 1:** Hubungan negatif atau substitusi (pembelian A justru menurunkan kemungkinan pembelian B, contohnya: Kopi Merek X dan Kopi Merek Y).

4. **Leverage & Conviction (Metrik Lanjutan):**

   Analis tingkat lanjut juga sering menggunakan *Leverage* (mengukur selisih antara frekuensi kemunculan A dan B bersama-sama dengan

Assets of Takaza INNOVATIX LABS

frekuensi harapan jika keduanya independen) dan *Conviction* (mengukur tingkat ketergantungan aturan; nilai yang tinggi berarti B sangat bergantung pada A).

Tantangan Implementasi:

Tantangan teknis terbesar dalam implementasinya adalah terjadinya ledakan kombinasi item (combinatorial explosion). Sebagai bayangan, jika sebuah toko fisik memiliki 10.000 produk *(SKU)*, jumlah kemungkinan kombinasinya sangatlah masif (mencapai miliaran kombinasi). Jika tidak dibatasi, komputasi akan memakan waktu berhari-hari dan menghasilkan jutaan aturan acak (*spurious rules*). Oleh karena itu, analis data (*data scientist*) harus menerapkan teknik *Pruning* (pemangkasan) dengan menetapkan ambang batas *(threshold)* minimum untuk *Support* dan *Confidence* secara bijak.

Assets of Takaza INNOVATIX LABS

Aplikasi di Dunia Nyata:

Penggunaan aturan asosiasi kini telah berevolusi jauh melampaui rak-rak toko ritel fisik. Beberapa penerapan utamanya meliputi:

- **Tata Letak Toko Fisik (Store Layout):** Menempatkan produk yang memiliki korelasi *Lift* tinggi secara berdekatan (misalnya, menempatkan pasta gigi di sebelah sikat gigi) untuk memicu pembelian impulsif.
- **Sistem Rekomendasi Digital:** Menjadi fondasi bagi fitur *"Frequently bought together"* (Sering dibeli bersamaan) di platform *e-commerce* seperti Amazon atau Shopee, serta rekomendasi konten di layanan *streaming* seperti Netflix dan Spotify.

- **Strategi Promosi (Bundling):** Menciptakan paket promosi silang (*cross-selling*) dan peningkatan penjualan (*up-selling*) dengan menggabungkan produk utama dan produk komplementer.
- **Web Usage Mining:** Menganalisis *clickstream* atau pola navigasi pengguna di sebuah situs web (Jika pengunjung membuka Halaman A, mereka cenderung akan membuka Halaman B), guna memperbaiki desain *User Experience* (UX).

Dengan interpretasi metrik yang tepat dan kombinasi pengetahuan bisnis, *Association Rule* merupakan senjata analitik yang sangat cerdas untuk secara otomatis meningkatkan metrik bisnis seperti *Average Order Value* (AOV) dan kepuasan pelanggan melalui penyajian produk yang sangat relevan.

Assets of Takaza INNOVATIX LABS

E. Interpretasi Hasil Clustering

Interpretasi adalah langkah krusial dalam *clustering* yang membedakan antara sekadar kalkulasi matematis dan *insight* bisnis yang dapat dieksekusi. Setelah algoritma selesai mengelompokkan data, langkah selanjutnya adalah memberikan "label" atau profil pada setiap klaster. Analis harus memeriksa karakteristik statistik (seperti rata-rata, median, distribusi) dari setiap variabel dalam klaster tersebut dibandingkan dengan populasi keseluruhan. Sebagai contoh, jika sebuah klaster pelanggan memiliki nilai rata-rata belanja yang tinggi dan frekuensi kunjungan yang rendah, kita bisa melabelinya sebagai segmen "Pelanggan VIP yang tidak sering belanja". Tanpa proses profilisasi ini, hasil *clustering* hanyalah angka yang tidak bermakna bagi pemangku kepentingan (Provost & Fawcett, 2013).

Seringkali, hasil *clustering* perlu divisualisasikan menggunakan teknik reduksi dimensi seperti *Principal Component Analysis* (PCA) atau *t-SNE*. Karena data bisnis seringkali memiliki puluhan fitur, kita tidak bisa melihat klaster dalam ruang 2D atau 3D secara langsung. PCA membantu memproyeksikan data ke dalam dimensi yang lebih rendah sambil mempertahankan sebanyak mungkin varians data. Visualisasi ini sangat membantu untuk melihat apakah klaster terpisah dengan jelas atau tumpang tindih. Jika klaster tumpang tindih secara signifikan, ini mungkin indikasi bahwa pemilihan fitur (*feature selection*) kurang tepat atau jumlah klaster yang ditentukan tidak ideal.

Selain itu, stabilitas klaster harus diuji untuk memastikan bahwa hasil yang didapat bukan karena fluktuasi acak dalam data. Teknik seperti *bootstrap* (melakukan *clustering* pada subset data yang berbeda secara berulang) dapat digunakan. Jika klaster tetap konsisten di berbagai subset, maka model kita dapat dikatakan stabil. Interpretasi yang baik juga harus mempertimbangkan etika; kita harus memastikan bahwa klaster yang terbentuk tidak menggunakan variabel sensitif (seperti etnis atau agama) yang bisa mengarah pada diskriminasi algoritma. Pada akhirnya, interpretasi adalah tentang menjembatani logika mesin dengan realitas pasar, memastikan setiap segmen yang ditemukan memiliki cerita dan strategi pemasarannya sendiri.

F. Studi kasus

Sebuah perusahaan *e-commerce* besar ingin meningkatkan efektivitas kampanye pemasaran mereka. Selama ini, mereka memperlakukan semua pelanggan secara sama dengan mengirimkan email promosi yang seragam,

yang mengakibatkan tingkat *click-through rate* (CTR) yang rendah. Mereka memiliki data perilaku pelanggan berupa frekuensi belanja, nilai uang yang dihabiskan, dan kategori produk yang sering dibeli. Tim data memutuskan untuk menggunakan teknik *Unsupervised Learning* untuk melakukan segmentasi pelanggan guna menciptakan kampanye yang lebih personal.

Metodologi yang diterapkan adalah *K-Means Clustering*. Data dibersihkan dan distandarisasi menggunakan *Z-score scaling* untuk memastikan fitur seperti "Total Belanja" (dalam jutaan) dan "Frekuensi" (dalam hitungan) memiliki bobot yang seimbang. Melalui *Elbow Method*, tim menentukan bahwa $K=4$ adalah jumlah klaster yang paling optimal. Setelah algoritma dijalankan, keempat klaster tersebut diprofilkan:

1. **Klaster 1 (High Value/Loyal):** Sering belanja, nilai tinggi. Strategi: Program loyalitas eksklusif.
2. **Klaster 2 (Budget Seekers):** Sering belanja, nilai rendah/diskon. Strategi: Kampanye *flash sale*.
3. **Klaster 3 (At-Risk/Churn):** Lama tidak belanja, nilai historis sedang. Strategi: Email *win-back* dengan voucher.
4. **Klaster 4 (New/Explorers):** Baru belanja sekali, kategori produk acak. Strategi: *Onboarding* dan pengenalan produk terlaris.

Hasil dari segmentasi ini sangat signifikan. Setelah menerapkan kampanye pemasaran yang dipersonalisasi berdasarkan segmen-segmen tersebut, perusahaan melihat peningkatan CTR sebesar 35% dan konversi penjualan sebesar 15% dalam satu kuartal. Studi kasus ini membuktikan bahwa *Unsupervised Learning* mampu memberikan *insight* yang tidak terlihat oleh mata manusia dalam data mentah. Dengan mengelompokkan pelanggan berdasarkan perilaku nyata, perusahaan mampu berpindah dari

pemasaran massal yang tidak efisien menuju pemasaran berbasis segmen yang sangat relevan dan menguntungkan.

Tabel 4. Perbandingan Algoritma Unsupervised Learning

Algoritma	Tipe Data Utama	Kebutuhan K/Threshold	Interpretasi	Kasus Penggunaan Utama
K-Means	Numerik	Ya (K)	Mudah (Centroid)	Segmentasi Pasar, Kompresi Gambar
Hierarchical	Numerik/ Campuran	Tidak (Dendrogram)	Sangat Tinggi (Struktur)	Taksonomi, Riset Pasar
Apriori	Kategorikal	Ya (Support/Conf)	Tinggi (Rules)	

Assets of Takaza INNOVATIX LABS

DAFTAR PUSTAKA

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, 207-216.

Baesens, B., De Winne, S., & Souter, L. (2022). *Analytics in a Big Data World: The Essential Guide to Data Science and its Applications* (2nd Edition). John Wiley & Sons.

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13, 281-305.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Borgman, C. L. (2021). *Big Data, Little Data, No Data: Scholarship in the Networked World*. MIT Press.

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). Wiley. https://doi.org/10.1002/9781118619193

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.

Bruce, P., Bruce, A., & Gedeck, P. (2021). *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python* (2nd Edition). O'Reilly Media.

Bruce, P., Bruce, A., & Gedeck, P. (2021). *Practical Statistics for Data Scientists*. O'Reilly Media.

Brynjolfsson, E., & McElheran, K. (2016). The Rapid Adoption of Data-Driven Decision-Making. *American Economic Review*, 106(5), 133-139. https://doi.org/10.1257/aer.p20161053

Coronel, C., & Morris, S. (2022). *Database Systems: Design, Implementation, & Management* (14th Edition). Cengage Learning.

Davenport, T. H. (2023). *All-in on AI: How Smart Companies Win Big with Artificial Intelligence*. Harvard Business Review Press.

Davenport, T. H. (2023). *All-in on AI: How Smart Companies Win Big with Artificial Intelligence*. Harvard Business Review Press.

Davenport, T. H., & Harris, J. G. (2007). *Competing on Analytics: The New Science of Winning*. Harvard Business Review Press.

DeBarros, A. (2022). *Practical SQL, 2nd Edition: A Beginner's Guide to Storytelling with Data*. No Starch Press.

Elmasri, R., & Navathe, S. B. (2021). *Fundamentals of Database Systems* (8th Edition). Pearson.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861-874.

Field, A. (2022). *Discovering Statistics Using IBM SPSS Statistics* (6th Edition). Sage Publications.

Geron, A. (2023). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd Edition). O'Reilly Media.

Géron, A. (2023). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.

Hair, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2022). *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)* (3rd Edition). Sage Publications.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

Heumann, C., Schomaker, M., & Shalabh. (2023). *Introduction to Statistics and Data Analysis: With Exercises, Solutions and Applications in R*. Springer Nature.

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and Practice* (3rd Edition). OTexts.

Inmon, W. H. (2021). *Building the Data Warehouse* (4th Edition). John Wiley & Sons.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*. Springer.

Jobin, A., Ienca, M., & Vayena, E. (2021). The global landscape of AI ethics guidelines. *Nature Machine Intelligence*, 1(9), 389-399.

Kahneman, D. (2011). *Thinking, Fast and Slow*. Farrar, Straus and Giroux.

Kaplan, R. S., & Norton, D. P. (1996). *The Balanced Scorecard: Translating Strategy into Action*. Harvard Business School Press.

Kelleher, J. D., & Tierney, B. (2022). *Data Science* (Revised Edition). MIT Press.

Kimball, R., & Ross, M. (2023). *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling* (4th Edition). John Wiley & Sons.

Kleppmann, M. (2023). *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*. O'Reilly Media.

Knaflic, C. N. (2015). *Storytelling with Data: A Data Visualization Guide for Business Professionals*. Wiley. https://doi.org/10.1002/9781119055860

Ladley, J. (2022). *Data Governance: How to Design, Deploy, and Sustain an Effective Data Governance Program* (3rd Edition). Morgan Kaufmann.

Lawson, R. (2022). *Web Scraping with Python: Collecting More Data from the Modern Web* (3rd Edition). O'Reilly Media.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 1, 281-297.

McKinney, W. (2022). *Python for Data Analysis*. O'Reilly Media.

Assets of Takaza

McKinney, W. (2022). *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter* (3rd Edition). O'Reilly Media.

Mitchell, R. (2023). *Web Scraping and API Fundamentals for Data Professionals*. O'Reilly Media.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Montgomery, D. C., & Runger, G. C. (2022). *Applied Statistics and Probability for Engineers* (8th Edition). John Wiley & Sons.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to Linear Regression Analysis* (5th ed.). Wiley.

Murtagh, F., & Legendre, P. (2014). Ward's hierarchical agglomerative clustering method: Which algorithms implement Ward's criterion? *Journal of Classification*, 31(3), 274-295.

Navidi, W. (2024). *Statistics for Engineers and Scientists* (6th Edition). McGraw-Hill Education.

Nield, T. (2021). *Getting Started with SQL: A Hands-On Approach for Beginners*. O'Reilly Media.

O'Neil, C. (2022). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown Publishing.

Pearl, J., & Mackenzie, D. (2021). *The Book of Why: The New Science of Cause and Effect*. Basic Books

Provost, F., & Fawcett, T. (2013). *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media.

Provost, F., & Fawcett, T. (2013). *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media.

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*, 1(1), 81-106.

Roesch, C., Sengupta, S., & Boudreau, J. (2023). *Julia for Data Analysis*. Manning Publications.

Russell, S. J., & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach* (3rd ed.). Pearson Education.

Russell, S. J., & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach* (3rd ed.). Pearson Education.

Severance, C. (2024). *Python for Everybody: Exploring Data Using Python 3*. CreateSpace Independent Publishing Platform.

Squire, M. (2021). *Clean Data: The Art and Science of Data Quality*. O'Reilly Media.

Squire, M. (2021). *Clean Data: The Art and Science of Data Quality*. O'Reilly Media.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press.

VanderPlas, J. (2023). *Python Data Science Handbook: Essential Tools for Working with Data* (2nd Edition). O'Reilly Media.

VanderPlas, J. (2023). *Python Data Science Handbook: Essential Tools for Working with Data* (2nd Edition). O'Reilly Media.

Wedel, M., & Kamakura, W. A. (2000). *Market Segmentation: Conceptual and Methodological Foundations*. Kluwer Academic Publishers. https://doi.org/10.1007/978-1-4615-4655-9

White, T. (2022). *Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale* (5th Edition). O'Reilly Media.

Wickham, H., Çetinkaya-Rundel, M., & Grolemund, G. (2023). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data* (2nd Edition). O'Reilly Media.

Zheng, X. (2023). *Cloud Computing and Big Data Analytics: Current Trends and Future Directions*. Academic Press.

BIOGRAFI PENULIS

Ani Yoraeni, S.Pd., M.Kom.

Adalah dosen Program Studi Sistem Informasi di Universitas Nusa Mandiri. Ia menyelesaikan pendidikan Sarjana Pendidikan Matematika dan Magister Ilmu Komputer dengan konsentrasi Manajemen Sistem Informasi. Bidang yang menjadi fokusnya meliputi Data Science, Artificial Intelligence, Machine Learning, Sistem Pendukung Keputusan, dan Analitik Data.

Selain aktif mengajar, penulis juga terlibat dalam berbagai kegiatan penelitian dan pengabdian kepada masyarakat, khususnya pada pengembangan solusi digital untuk mendukung pengambilan keputusan dan pembangunan berkelanjutan. Melalui karya-karya ilmiah dan buku yang ditulisnya, penulis berharap dapat memberikan kontribusi bagi pengembangan ilmu pengetahuan serta menjadi referensi bagi mahasiswa, akademisi, dan praktisi.

Takaza
INNOVATIX LABS

Nicodias Palasara, M.Kom.

Seorang akademisi, dosen, dan peneliti yang memiliki fokus kepakaran di bidang *Decision System Support* (Sistem Pendukung Keputusan) dan *Software Engineering* (Rekayasa Perangkat Lunak). Saat ini, beliau tercatat aktif mengabdi sebagai dosen pada Program Studi D3 Sistem Informasi di Universitas Bina Sarana Informatika (UBSI) Kampus Kabupaten Karawang. Beliau memiliki rekam jejak Tridharma Perguruan Tinggi yang ekstensif, terbukti dari profil akademiknya (SINTA ID: 5991211) dan (ID Scopus: 57220181348) yang memuat berbagai luaran penelitian, pengabdian masyarakat, hingga perolehan Hak Kekayaan Intelektual (HKI).

Berdasarkan pengalaman akademik dan penelitiannya, beliau telah menghasilkan berbagai kontribusi penting, antara lain:

- Menjadi peneliti dalam mengevaluasi keberhasilan teknologi publik melalui "Evaluasi Keberhasilan Aplikasi Qlue Dengan Model ITPOSMO Pada Pemerintahan Provinsi DKI Jakarta" yang didanai melalui hibah Penelitian Dosen Pemula (PDP)-Dikti pada tahun 2019.
- Aktif menghasilkan berbagai karya tulis ilmiah di jurnal nasional dan prosiding internasional (seperti Jurnal Sisfokom, Jurnal Informatika, dan Journal of Physics) yang berfokus pada implementasi algoritma K-Means, Regresi Linear, Topsis, dan *Analytical Hierarchy Process* (AHP).
- Mengembangkan dan meneliti berbagai perancangan sistem informasi, *clustering* data, kepuasan pengguna (menggunakan Webqual), hingga sistem pemesanan dan monitoring yang bermanfaat untuk dunia usaha dan pelayanan kesehatan.
- Konsisten dalam kegiatan pengabdian kepada masyarakat, seperti pemberdayaan kemandirian wirausaha berbasis internet dan pelatihan pemanfaatan teknologi digital untuk memperkecil kesenjangan digital di berbagai komunitas.

Herryansyah, M.Kom.

Pria berdarah Bugis lulusan Universitas Gunadarma dan Universitas Nusa Mandiri ini aktif sebagai dosen tetap Universitas Bina Sarana Informatika serta memiliki minat pada bidang pemrograman dan *ART-Crafting*. Buku ini merupakan karya pertamanya yang diterbitkan dan menjadi langkah awal dalam perjalanan kepenulisannya. Melalui buku ini, ia berharap dapat memberikan manfaat, memperluas wawasan pembaca, serta menginspirasi masyarakat untuk terus belajar dan berkembang.

Assets of Takaza INNOVATIX LABS

Yayan Hendrian, S.T., M.Kom.

Penulis adalah seorang Dosen sekaligus peneliti dan pembicara di beberapa seminar nasional khususnya yang berkaitan dengan Ilmu Komputer. Penulis menyelesaikan Program pendidikannya Studi Teknik di Elektro Universitas Tama Jagakarsa pada tahun 2009 dan Pendidikan Pasca Sarjana Magister Ilmu Komputer di STMIK Nusa Mandiri pada tahun 2012. Sejak 2010 menjadi Dosen tetap di Universitas Bina Sarana Informatika pada Prodi Informatika. Selain itu, beliau juga seorang penulis buku yang sudah diterbitkan secara nasional, diantaranya buku Organisasi dan Perancangan Arsitektur Komputer, Teknik Elektro Dalam Era Industri 4.0, Kecerdasan Buatan: Dasar-dasar, Metode Pencarian dan Machine Learning.

Assets
of
Takaza
INNOVATIX LABS

DATA SCIENCE BERBASIS PROYEK
PANDUAN PRAKTISI ANALISIS DATA

Buku ini dirancang sebagai buku ajar yang mengintegrasikan konsep teoritis dan praktik nyata dalam bidang data science melalui pendekatan pembelajaran berbasis proyek (Project-Based Learning). Buku ini membahas secara sistematis tahapan data science mulai dari pengenalan konsep dasar, statistika, pemrograman, pengumpulan dan pembersihan data, eksplorasi serta visualisasi data, hingga penerapan machine learning untuk menghasilkan insight yang mendukung pengambilan keputusan. Setiap materi dilengkapi dengan studi kasus, latihan, dan proyek yang mencerminkan permasalahan nyata di dunia bisnis, pendidikan, pemerintahan, maupun industri.

9 786347 741981

IKAPI
No. 44/ SBA / 2023

Google Play Books

Takaza