

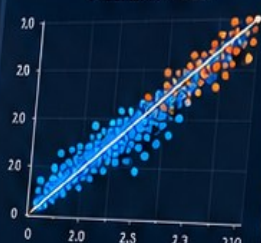
Machine Learning

Kasus Riset Kimia Menggunakan Python

Teori, Praktik, dan 15 Proyek Riset Riil untuk Mahasiswa, Peneliti, dan Praktisi

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.model_selection import train_test_split
4 from sklearn.ensemble import RandomForestRegressor
5 from sklearn.metrics import mean_squared_error, r2_score
6
7 # Load dataset
8 data = pd.read_csv('data.csv')
9 X = data.drop('target', axis=1)
10 y = data['target']
11
12 # Train model
13 X_train, X_test, y_train, y_test = train_test_split(
14     X, y, test_size=0.2, random_state=42)
15 model = RandomForestRegressor(n_estimators=200)
16 model.fit(X_train, y_train)
17
18 # Evaluate
19 y_pred = model.predict(X_test)
20 print("R2 Score:", r2_score(y_test, y_pred))
```

Prediksi vs Aktual



Feature Importance



Python

Machine Learning

Cheminformatics

Computational Chemistry

Data Science

KASMUI

MACHINE LEARNING KASUS RISET KIMIA MENGUNAKAN PYTHON

Buku Akademik-Praktis untuk Riset Kimia Berbasis Data

KASMUI
2026

Disusun dengan pendekatan praktik Python, data kimia terbuka, dan prinsip reproducible machine learning.

KATA PENGANTAR

Buku ini disusun untuk menjembatani dua dunia yang semakin sulit dipisahkan dalam riset modern: kimia sebagai ilmu yang berangkat dari struktur, sifat, reaktivitas, pengukuran, serta mekanisme; dan machine learning sebagai seperangkat metode komputasi untuk menemukan pola, membangun prediksi, melakukan klasifikasi, serta memetakan ruang data. Titik tolak penyusunan mengikuti alur pembelajaran yang lazim digunakan dalam pelatihan Machine Learning with Python—mulai dari fundamental Python, preparasi data, pipeline, evaluasi, regresi, klasifikasi, clustering, PCA, hingga peningkatan model—namun seluruh konteks, studi kasus, variabel, dan interpretasinya dialihkan secara khusus ke kebutuhan riset kimia.

Tujuan utama buku bukan sekadar membuat pembaca mampu memanggil fungsi dari scikit-learn. Dalam penelitian kimia, model yang mempunyai nilai R^2 tinggi belum tentu memiliki makna ilmiah apabila data berasal dari assay yang tidak sebanding, satuan bercampur, struktur duplikat tersebar di train dan test, atau fitur dibentuk dengan informasi yang sebenarnya baru tersedia setelah eksperimen. Karena itu, setiap bab menempatkan kualitas data, provenance, validasi, keterbatasan, serta interpretasi kimia sebagai bagian dari metodologi, bukan sebagai catatan tambahan setelah model selesai.

Ruang lingkup buku mencakup pemrograman Python dasar untuk data kimia; penggunaan NumPy dan Pandas; representasi molekul melalui SMILES, descriptor dan fingerprint; akses data PubChem dan ChEMBL; pembersihan struktur; pemisahan data; standarisasi; supervised learning untuk regresi dan klasifikasi; unsupervised learning menggunakan K-Means dan PCA; QSAR; contoh data kimia fisik, farmasi, dan kimia kuantum; hyperparameter tuning; data leakage; applicability domain; reproducibility; serta gambaran deployment. Dataset publik seperti ESOL/Delaney, BBBP, Lipophilicity, FreeSolv, dan QM9 digunakan karena memungkinkan pembaca mengulang eksperimen komputasi dengan sumber yang dapat ditelusuri.

Sasaran pembaca adalah mahasiswa kimia, farmasi, biokimia, teknik kimia, material, dan bidang terkait; peneliti laboratorium yang ingin mengolah data secara lebih sistematis; dosen yang memerlukan contoh praktikum komputasi; serta programmer yang ingin memahami karakter khusus data kimia. Penjelasan dimulai dari konsep yang relatif mendasar, tetapi secara bertahap bergerak menuju isu yang lebih dekat dengan praktik publikasi ilmiah, seperti scaffold split, ketidakseimbangan kelas, variasi assay, standarisasi aktivitas, pIC50, validasi silang, dan penyimpanan pipeline model.

Pendekatan penyusunan bersifat hands-on. Kode program ditulis sebagai skrip Python yang dapat dieksekusi, bukan hanya pseudocode. Sebagian skrip

mengunduh dataset atau memanggil API publik; karena data daring dan antarmuka layanan dapat berubah, pembaca perlu mencatat tanggal akses, menyimpan salinan data penelitian, dan memeriksa dokumentasi sumber sebelum menjadikan hasil sebagai dasar publikasi. Nilai metrik yang diperoleh pembaca dapat berbeda bila versi data, versi library, split, atau random seed berubah; perbedaan tersebut harus dipandang sebagai bagian dari reproducibility yang perlu dilaporkan.

Buku ini tidak memuat soal latihan maupun pembahasan soal. Sebagai gantinya, penekanan diberikan pada contoh kode, pembacaan output, alasan metodologis, dan alur riset yang dapat langsung dimodifikasi untuk data pengguna sendiri. Kode pendamping disediakan terpisah agar pembaca dapat menjalankan contoh tanpa harus menyalin ulang seluruh listing dari dokumen.

Keterbatasan buku perlu dinyatakan dengan jelas. Machine learning tidak menggantikan eksperimen, teori kimia, atau perhitungan kuantum; ia menyediakan pendekatan prediktif dan eksploratif yang kualitasnya sangat bergantung pada representativitas data dan validitas desain penelitian. Model pada contoh-contoh buku terutama ditujukan untuk pembelajaran metodologi. Sebelum digunakan untuk keputusan eksperimen berbiaya tinggi, keamanan, klinis, regulatori, atau klaim mekanistik, diperlukan validasi yang lebih kuat dan sesuai domain.

Penulis menyampaikan penghargaan kepada pengembang komunitas Python ilmiah, pengelola basis data kimia terbuka, serta peneliti yang mempublikasikan dataset dan perangkat lunak sehingga pembelajaran dapat direproduksi. Semoga buku ini membantu pembaca membangun kebiasaan riset komputasi yang tidak hanya mampu menghasilkan prediksi, tetapi juga dapat ditelusuri, diperiksa ulang, dan dipertanggungjawabkan secara ilmiah.

Semarang, Oktober 2026

Penyusun,



Kasmui

PETUNJUK PENGGUNAAN BUKU

Buku dapat dipelajari berurutan dari Bab 1 sampai Bab 14, tetapi pembaca yang telah menguasai Python dapat langsung menuju bab preparasi data dan studi kasus. Setiap listing kode utama juga tersedia sebagai file .py pada paket kode pendamping. Gunakan satu environment khusus agar versi library penelitian tidak tercampur dengan proyek lain.

1. Siapkan Python 3.11 atau 3.12, Jupyter/VS Code/Google Colab, dan environment terpisah.
2. Instal paket pada requirements.txt. Bila RDKit sulit dipasang melalui pip, gunakan conda-forge.
3. Jalankan 00_cek_environment.py sebelum studi kasus lain.
4. Simpan dataset mentah, dataset hasil pembersihan, kode, seed, versi library, model, dan ringkasan hasil dalam folder proyek yang terstruktur.
5. Untuk riset publikasi, jangan hanya mengandalkan random split; pertimbangkan scaffold split, time split, group split, atau external test set sesuai pertanyaan kimia.

Prinsip penting: Kode yang berjalan tanpa error belum otomatis menjadi metodologi penelitian yang valid. Selalu periksa asal data, satuan, duplikasi, definisi target, dan independensi data uji.

DAFTAR ISI

KATA PENGANTAR	3
PETUNJUK PENGGUNAAN BUKU	5
DAFTAR ISI.....	6
PETA STUDI KASUS BUKU	18
BAB 1 MACHINE LEARNING DALAM PARADIGMA RISET KIMIA	19
1.1 Dari Data Eksperimen ke Model Prediktif	19
1.2 Supervised, Unsupervised, dan Tipe Pertanyaan Kimia	19
1.3 Machine Learning Bukan Pengganti Kimia	20
1.4 Ekosistem Python untuk Riset Kimia.....	21
BAB 2 FONDASI PYTHON DAN LINGKUNGAN KOMPUTASI UNTUK KIMIA.....	22
2.1 Memilih Environment yang Reprodusibel	22
2.2 Variabel, List, Dictionary, Loop, dan Function dalam Konteks Kimia	22
2.3 Struktur Folder Proyek	23
BAB 3 DATA KIMIA, REPRESENTASI MOLEKUL, DAN SUMBER TERBUKA.....	24
3.1 Mengapa Representasi Menentukan Apa yang Dapat Dipelajari Model	24
3.2 SMILES, InChI, dan Identitas Kimia	25
3.3 Menghitung Descriptor dengan RDKit	25
3.4 Mengambil Data dari PubChem	25
3.5 ChEMBL untuk Data Bioaktivitas	26
BAB 4 DATA PREPARATION DAN QUALITY CONTROL KIMIA.....	27
4.1 Audit Sebelum Modeling	27
4.2 Encoding, Scaling, dan Standardisasi.....	27
4.3 Train, Validation, Test: Random Split Tidak Selalu Cukup	27
4.4 Data Leakage dalam Riset Kimia.....	28
BAB 5 PIPELINE DAN EVALUASI MODEL	29

5.1 Mengapa Pipeline Penting.....	29
5.2 Metrik Regresi	29
5.3 Metrik Klasifikasi.....	29
5.4 Cross-Validation dan Ketidakpastian	30
BAB 6 STUDI KASUS REGRESI: PREDIKSI KELARUTAN AIR ESOL	31
6.1 Latar Belakang Kimia dan Dataset.....	31
6.2 Implementasi Linear, Ridge, dan Random Forest	31
6.3 Diagnostik Residual	32
6.4 Fingerprint Morgan sebagai Representasi Alternatif	33
BAB 7 STUDI KASUS REGRESI KIMIA FISIK: FREESOLV DAN LIPOFILISITAS	35
7.1 Energi Hidrasi FreeSolv	35
7.2 Lipofilisitas dan LogD.....	36
7.3 Membandingkan Model tanpa Terjebak Leaderboard	36
BAB 8 STUDI KASUS KLASIFIKASI: BLOOD-BRAIN BARRIER PENETRATION	37
8.1 Definisi Masalah Klasifikasi	37
8.2 Logistic Regression dan Random Forest.....	37
8.3 KNN dan Grid Search	38
8.4 Scaffold Split dan Evaluasi yang Lebih Realistik	39
BAB 9 UNSUPERVISED LEARNING: CHEMICAL SPACE, K-MEANS, DAN PCA.....	40
9.1 Chemical Space sebagai Ruang Multidimensi	40
9.2 Menentukan Jumlah Cluster.....	40
9.3 Praktik PCA + K-Means pada Dataset Lipofilisitas.....	40
9.4 Clustering untuk Desain Eksperimen	41
BAB 10 QSAR MODERN DENGAN DESCRIPTOR DAN MOLECULAR FINGERPRINT.....	42
10.1 Dari Struktur ke Fingerprint Morgan	42

10.2 Similarity dan Applicability Domain	42
10.3 Feature Importance dan Interpretasi	42
BAB 11 STUDI KASUS BIOAKTIVITAS: MEMBANGUN DATASET HER2/ERBB2 DARI ChEMBL	43
11.1 Mengapa Data Bioaktivitas Memerlukan Audit Lebih Ketat	43
11.2 Mengambil dan Memodelkan Data HER2/ERBB2	43
11.3 Dari Baseline ke Model Riset	45
BAB 12 MACHINE LEARNING UNTUK DATA KIMIA KUANTUM	46
12.1 Posisi ML terhadap Perhitungan Kuantum	46
12.2 QM9 sebagai Dataset Benchmark	46
12.3 Menghubungkan ML dengan Workflow QM Komputasi	47
12.4 Δ -Learning dan Multi-Fidelity	47
BAB 13 MODEL IMPROVEMENT, VALIDASI, DAN REPRODUCIBILITY	48
13.1 Hyperparameter Tuning yang Tidak Membocorkan Data	48
13.2 Feature Selection dan Multikolinearitas	48
13.3 Calibration, Threshold, dan Probabilitas	48
13.4 Checklist Reproducibility	48
13.5 Penyimpanan Model dan Inference	49
BAB 14 PROYEK TERINTEGRASI: DARI PERTANYAAN RISET KE LAPORAN ILMIAH	50
14.1 Template Alur Riset	50
14.2 Struktur Laporan Metodologi	50
14.3 Deployment: Dari Model ke Aplikasi	51
14.4 Penutup: Kebiasaan Riset yang Perlu Dipertahankan	51
LAMPIRAN A INSTALASI DAN KONFIGURASI	52
A.1 Instalasi dengan venv	52
A.2 Instalasi RDKit dengan Conda	52
A.3 Verifikasi	52
LAMPIRAN B INDEKS KODE PENDAMPING	53

LAMPIRAN C TROUBLESHOOTING PRAKTIK	54
LAMPIRAN D 15 PROYEK RISET KIMIA MENGGUNAKAN MACHINE LEARNING DAN PYTHON	55
D.1 Prinsip Pelaksanaan Proyek Dua Pekan	55
D.2 Instalasi Python dan Seluruh Library dalam Satu Perintah	56
D.3 Standar Folder, Output, dan Pelaporan.....	56
D.4 PROYEK 1: PREDIKSI KELARUTAN AIR SENYAWA ORGANIK MENGGUNAKAN ESOL	57
Judul.....	57
Latar Belakang.....	57
Rumusan Masalah.....	58
Tujuan Riset.....	58
Telaah Pustaka	59
Metode Penelitian	59
Alur Penelitian	60
Rencana Kerja 14 Hari.....	60
Coding Machine Learning Python yang Digunakan	61
Hasil Riset dan Pembahasan	62
Diskusi	63
Kesimpulan	63
Referensi Inti Proyek	64
D.5 PROYEK 2: PREDIKSI KELARUTAN AIR SKALA LEBIH BESAR MENGGUNAKAN AQSOLDB	64
Judul.....	64
Latar Belakang.....	64
Rumusan Masalah.....	65
Tujuan Riset.....	65
Telaah Pustaka	66
Metode Penelitian	66

Alur Penelitian	67
Rencana Kerja 14 Hari.....	67
Coding Machine Learning Python yang Digunakan	68
Hasil Riset dan Pembahasan	69
Diskusi	70
Kesimpulan	70
Referensi Inti Proyek	70
D.6 PROYEK 3: PREDIKSI ENERGI BEBAS HIDRASI MENGGUNAKAN FREESOLV	71
Judul.....	71
Latar Belakang.....	71
Rumusan Masalah.....	72
Tujuan Riset.....	72
Telaah Pustaka	72
Metode Penelitian	73
Alur Penelitian	73
Rencana Kerja 14 Hari.....	74
Coding Machine Learning Python yang Digunakan	74
Hasil Riset dan Pembahasan	75
Diskusi	76
Kesimpulan	76
Referensi Inti Proyek	77
D.7 PROYEK 4: PREDIKSI LIPOFILISITAS LOGD PADA PH 7,4.....	77
Judul.....	77
Latar Belakang.....	77
Rumusan Masalah.....	78
Tujuan Riset.....	78
Telaah Pustaka	78

Metode Penelitian	79
Alur Penelitian	79
Rencana Kerja 14 Hari.....	80
Coding Machine Learning Python yang Digunakan	80
Hasil Riset dan Pembahasan	81
Diskusi	82
Kesimpulan	83
Referensi Inti Proyek	83
D.8 PROYEK 5: KLASIFIKASI PENETRASI BLOOD-BRAIN BARRIER (BBBP)	83
Judul.....	83
Latar Belakang.....	84
Rumusan Masalah.....	84
Tujuan Riset.....	84
Telaah Pustaka	85
Metode Penelitian	85
Alur Penelitian	86
Rencana Kerja 14 Hari.....	86
Coding Machine Learning Python yang Digunakan	87
Hasil Riset dan Pembahasan.....	88
Diskusi	89
Kesimpulan	89
Referensi Inti Proyek	89
D.9 PROYEK 6: KLASIFIKASI AMES MUTAGENICITY DARI STRUKTUR MOLEKUL	89
Judul.....	90
Latar Belakang.....	90
Rumusan Masalah.....	90
Tujuan Riset.....	91

Telaah Pustaka	91
Metode Penelitian	92
Alur Penelitian	92
Rencana Kerja 14 Hari.....	93
Coding Machine Learning Python yang Digunakan	93
Hasil Riset dan Pembahasan	94
Diskusi	95
Kesimpulan	95
Referensi Inti Proyek	96
D.10 PROYEK 7: KLASIFIKASI READY BIODEGRADABILITY BAHAN KIMIA	96
Judul.....	96
Latar Belakang.....	96
Rumusan Masalah.....	97
Tujuan Riset.....	97
Telaah Pustaka	97
Metode Penelitian	98
Alur Penelitian	98
Rencana Kerja 14 Hari.....	99
Coding Machine Learning Python yang Digunakan	99
Hasil Riset dan Pembahasan	100
Diskusi	101
Kesimpulan	102
Referensi Inti Proyek	102
D.11 PROYEK 8: PREDIKSI TOKSISITAS AKUT TERHADAP IKAN FATHEAD MINNOW	102
Judul.....	102
Latar Belakang.....	103
Rumusan Masalah.....	103

Tujuan Riset.....	103
Telaah Pustaka	104
Metode Penelitian	104
Alur Penelitian.....	105
Rencana Kerja 14 Hari.....	105
Coding Machine Learning Python yang Digunakan	106
Hasil Riset dan Pembahasan	106
Diskusi	107
Kesimpulan	108
Referensi Inti Proyek	108
D.12 PROYEK 9: KLASIFIKASI TOKSISITAS TOX21 PADA JALUR STRESS RESPONSE P53.....	108
Judul.....	108
Latar Belakang.....	109
Rumusan Masalah.....	109
Tujuan Riset.....	109
Telaah Pustaka	110
Metode Penelitian	110
Alur Penelitian	111
Rencana Kerja 14 Hari.....	111
Coding Machine Learning Python yang Digunakan	112
Hasil Riset dan Pembahasan	112
Diskusi	113
Kesimpulan	114
Referensi Inti Proyek	114
D.13 PROYEK 10: KLASIFIKASI SENYAWA PENGHAMBAT REPLIKASI HIV	114
Judul.....	115
Latar Belakang.....	115

Rumusan Masalah.....	115
Tujuan Riset.....	115
Telaah Pustaka	116
Metode Penelitian	116
Alur Penelitian	117
Rencana Kerja 14 Hari.....	117
Coding Machine Learning Python yang Digunakan	118
Hasil Riset dan Pembahasan	119
Diskusi	120
Kesimpulan	120
Referensi Inti Proyek	120
D.14 PROYEK 11: QSAR INHIBITOR BACE-1 UNTUK STUDI ALZHEIMER	120
Judul.....	121
Latar Belakang.....	121
Rumusan Masalah.....	121
Tujuan Riset.....	122
Telaah Pustaka	122
Metode Penelitian	123
Alur Penelitian	123
Rencana Kerja 14 Hari.....	124
Coding Machine Learning Python yang Digunakan	124
Hasil Riset dan Pembahasan	125
Diskusi	126
Kesimpulan	126
Referensi Inti Proyek	127
D.15 PROYEK 12: QSAR BIOAKTIVITAS HER2/ERBB2 DARI CHEMBL	127
Judul.....	127

Latar Belakang.....	127
Rumusan Masalah.....	128
Tujuan Riset.....	128
Telaah Pustaka	128
Metode Penelitian	129
Alur Penelitian	129
Rencana Kerja 14 Hari.....	130
Coding Machine Learning Python yang Digunakan	130
Hasil Riset dan Pembahasan	132
Diskusi	132
Kesimpulan	133
Referensi Inti Proyek	133
D.16 PROYEK 13: PREDIKSI HOMO-LUMO GAP MOLEKUL DARI DATASET QM9	133
Judul.....	134
Latar Belakang.....	134
Rumusan Masalah.....	134
Tujuan Riset.....	135
Telaah Pustaka	135
Metode Penelitian	135
Alur Penelitian	136
Rencana Kerja 14 Hari.....	136
Coding Machine Learning Python yang Digunakan	137
Hasil Riset dan Pembahasan	138
Diskusi	139
Kesimpulan	139
Referensi Inti Proyek	139
D.17 PROYEK 14: PREDIKSI YIELD REAKSI BUCHWALD-HARTWIG C-N CROSS-COUPPLING	139

Judul.....	140
Latar Belakang.....	140
Rumusan Masalah.....	141
Tujuan Riset.....	141
Telaah Pustaka	141
Metode Penelitian	142
Alur Penelitian	142
Rencana Kerja 14 Hari.....	143
Coding Machine Learning Python yang Digunakan	143
Hasil Riset dan Pembahasan	144
Diskusi	145
Kesimpulan	145
Referensi Inti Proyek	145
D.18 PROYEK 15: PREDIKSI BAND GAP EKSPERIMENTAL MATERIAL ANORGANIK.....	146
Judul.....	146
Latar Belakang.....	146
Rumusan Masalah.....	147
Tujuan Riset.....	147
Telaah Pustaka	147
Metode Penelitian	148
Alur Penelitian	148
Rencana Kerja 14 Hari.....	149
Coding Machine Learning Python yang Digunakan	149
Hasil Riset dan Pembahasan	150
Diskusi	151
Kesimpulan	151
Referensi Inti Proyek	152

D.19 Rubrik Kendali Mutu 15 Proyek	152
D.20 Daftar Referensi Jurnal Utama Lampiran D.....	152
GLOSARIUM.....	154
DAFTAR PUSTAKA	158
SUMBER DATA DAN DOKUMENTASI DARING.....	160

PETA STUDI KASUS BUKU

Tabel 1. Peta studi kasus riil yang digunakan dalam buku

Kasus	Jenis masalah	Data utama	Model yang dipraktikkan	Output ilmiah
Kelarutan air (ESOL)	Regresi	Delaney/ESOL	Linear, Ridge, Random Forest	MAE, RMSE, R ² , interpretasi descriptor
Blood-brain barrier (BBBP)	Klasifikasi	BBBP MoleculeNet	Logistic Regression, Random Forest, KNN	Accuracy, precision, recall, F1, ROC-AUC
Lipofilisitas	Clustering	Lipophilicity MoleculeNet	K-Means + PCA	Peta chemical space, silhouette
Energi hidrasi	Regresi	FreeSolv	Extra Trees, Gradient Boosting	Error energi dan generalisasi
Bioaktivitas HER2/ERBB2	QSAR	ChEMBL	Random Forest baseline	pIC50, provenance assay, duplikasi
Sifat kuantum	Regresi	QM9	HistGradientBoosting baseline	Prediksi gap dan diskusi representasi

BAB 1

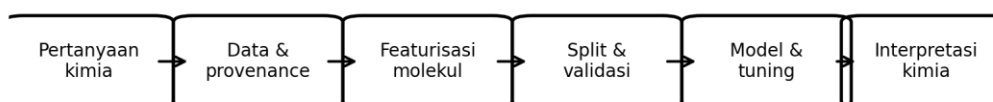
MACHINE LEARNING DALAM PARADIGMA RISET KIMIA

1.1 Dari Data Eksperimen ke Model Prediktif

Machine learning (ML) adalah pendekatan komputasi yang mempelajari hubungan antara masukan dan keluaran dari data. Dalam kimia, masukan dapat berupa komposisi, descriptor molekul, fingerprint, spektrum, kondisi reaksi, komposisi material, parameter proses, atau hasil perhitungan kuantum. Keluaran dapat berupa nilai kontinu seperti kelarutan, logD, energi hidrasi, energi adsorpsi, band gap, dan aktivitas biologis; atau label diskrit seperti aktif/tidak aktif, permeabel/tidak permeabel, fase material, dan kelas spektrum.

Perbedaan penting antara ML generik dan ML untuk kimia adalah struktur data yang sarat makna fisik. Dua baris pada spreadsheet tidak selalu independen: molekul dapat mempunyai scaffold yang hampir identik, senyawa dapat diukur berulang pada assay berbeda, dan descriptor dapat saling berkorelasi karena berasal dari struktur yang sama. Oleh sebab itu, pemilihan split, featurisasi, dan validasi harus mempertimbangkan struktur kimia, bukan hanya aturan statistik umum.

ML sebaiknya ditempatkan sebagai bagian dari siklus penelitian. Pertanyaan ilmiah menentukan target; target menentukan data yang relevan; karakter data menentukan representasi; representasi memengaruhi model; dan model harus diuji pada kondisi yang mendekati penggunaan nyata. Alur ini mencegah kebiasaan memilih algoritma terlebih dahulu kemudian mencari data yang cocok untuk menghasilkan metrik tinggi.



Validasi ilmiah harus mengawal seluruh alur, bukan hanya tahap pemodelan.

Gambar 1.1. Alur kerja machine learning yang terintegrasi dengan validasi ilmiah dalam riset kimia.

1.2 Supervised, Unsupervised, dan Tipe Pertanyaan Kimia

Supervised learning menggunakan pasangan fitur X dan target y . Regresi digunakan bila y berupa nilai kontinu, sedangkan klasifikasi digunakan bila y berupa kategori. Unsupervised learning tidak menggunakan label target dan lebih sesuai untuk

eksplorasi chemical space, pengelompokan sampel, deteksi pola, atau reduksi dimensi. Dalam praktik, satu proyek kimia dapat memakai keduanya: PCA dan clustering untuk memahami ruang data, lalu model supervised untuk prediksi kuantitatif.

Tabel 1.1. Pemetaan jenis machine learning ke pertanyaan riset kimia

Tipe ML	Pertanyaan kimia	Contoh target	Algoritma awal
Regresi	Berapa nilai sifat/aktivitas?	logS, logD, pIC50, ΔG hidrasi, band gap	Linear/Ridge, Random Forest, Gradient Boosting
Klasifikasi	Termasuk kelas apa?	aktif/tidak aktif, BBB+/BBB-, toksik/non-toksik	Logistic Regression, KNN, Decision Tree, Random Forest
Clustering	Sampel mana yang mirip?	cluster chemical space, kelompok spektrum	K-Means, hierarchical clustering
Reduksi dimensi	Bagaimana melihat struktur variasi?	proyeksi descriptor/fingerprint	PCA; untuk visualisasi lanjutan dapat dipertimbangkan metode nonlinier

1.3 Machine Learning Bukan Pengganti Kimia

Model dapat menemukan korelasi yang kuat tanpa mengetahui mekanisme. Sebagai contoh, descriptor lipofilisitas dapat sangat informatif untuk permeabilitas membran, tetapi koefisien korelasi atau feature importance tidak otomatis membuktikan sebab-akibat. Kesimpulan mekanistik harus tetap ditopang teori, eksperimen, atau perhitungan yang relevan.

Dalam domain kimia, extrapolation adalah risiko utama. Model yang dilatih pada molekul kecil netral dapat gagal pada ion, kompleks logam, makromolekul, atau scaffold yang jauh berbeda. Karena itu, applicability domain perlu dilaporkan bersama metrik. Prediksi yang berada di luar domain seharusnya diberi tanda ketidakpastian atau tidak digunakan untuk keputusan.

Kualitas data sering lebih menentukan daripada kerumitan algoritma. MoleculeNet menekankan bahwa representasi yang sesuai domain dan cara evaluasi yang tepat sangat penting, khususnya pada data yang terbatas atau tidak seimbang. Prinsip ini menjadi dasar seluruh studi kasus di buku.

1.4 Ekosistem Python untuk Riset Kimia

Tabel 1.2. Ekosistem perangkat utama yang digunakan

Library/layanan	Peran	Contoh penggunaan
NumPy	Komputasi array numerik	matriks fitur, transformasi numerik
Pandas	Data tabular	CSV, cleaning, agregasi data assay
Matplotlib	Visualisasi	distribusi, scatter, residual, PCA
scikit-learn	ML klasik	pipeline, model, metrik, CV, tuning
RDKit	Cheminformatika	SMILES, descriptor, fingerprint, validasi struktur
PubChem PUG REST	Akses data kimia publik	sifat molekul, identifier, struktur
ChEMBL Web Services	Bioaktivitas terkurasi	IC50, Ki, target, assay

BAB 2

FONDASI PYTHON DAN LINGKUNGAN KOMPUTASI UNTUK KIMIA

2.1 Memilih Environment yang Reproducibel

Kesalahan paling umum pada proyek Python ilmiah adalah mencampur seluruh paket dalam satu environment. Ketika satu proyek membutuhkan RDKit versi tertentu dan proyek lain membutuhkan versi berbeda, ketergantungan dapat saling merusak. Solusinya adalah membuat environment per proyek dan mencatat versi paket. Pendekatan ini juga memudahkan reproduksi komputasi beberapa bulan kemudian.

Google Colab cocok untuk pembelajaran dan demonstrasi singkat, sedangkan Jupyter Notebook atau VS Code di komputer lokal lebih nyaman untuk proyek yang memerlukan file besar, environment stabil, atau akses ke perangkat lunak kimia lain. Untuk penelitian yang harus diulang, simpan kode sebagai .py atau notebook bersama requirements.txt atau environment.yml.

Random seed bukan jaminan bahwa semua hasil identik pada setiap platform, namun ia merupakan bagian penting dari reproducibility. Catat seed untuk split, inisialisasi model, dan prosedur sampling. Catat pula versi data karena basis data kimia daring dapat diperbarui.

Kode 2.1. Pemeriksaan environment Python dan library utama

```
import platform
import importlib

packages = [
    "numpy", "pandas", "matplotlib", "sklearn", "requests", "joblib", "rdkit"
]
print("Python:", platform.python_version())
for name in packages:
    try:
        mod = importlib.import_module(name)
        print(f"{name:12s} OK   versi={getattr(mod, '__version__', 'n/a')}")
    except Exception as exc:
        print(f"{name:12s} GAGAL: {exc}")
```

2.2 Variabel, List, Dictionary, Loop, dan Function dalam Konteks Kimia

Konsep fundamental Python sebaiknya dipelajari melalui objek yang dekat dengan riset. Dictionary cocok untuk menyimpan satu molekul dengan pasangan nama–nilai, list cocok untuk kumpulan molekul, loop untuk memproses banyak sampel, dan function untuk memastikan prosedur perhitungan dipakai konsisten.

Kode 2.2. Fundamental Python menggunakan data molekul sederhana

```

from statistics import mean

molekul = [
    {"nama": "air", "Mr": 18.015, "dipol_D": 1.85},
    {"nama": "etanol", "Mr": 46.069, "dipol_D": 1.69},
    {"nama": "aseton", "Mr": 58.080, "dipol_D": 2.88},
]

def ringkas(data):
    mr = [x["Mr"] for x in data]
    dipol = [x["dipol_D"] for x in data]
    return {
        "jumlah_molekul": len(data),
        "rata_rata_Mr": mean(mr),
        "rata_rata_dipol_D": mean(dipol),
    }

for item in molekul:
    print(f"{item['nama']}:8s} Mr={item['Mr']}:7.3f} dipol={item['dipol_D']}:.2f}
D")
print(ringkas(molekul))

```

Pada riset nyata, pola yang sama berkembang menjadi fungsi untuk membersihkan SMILES, menghitung descriptor, mengonversi satuan, atau mengagregasi pengukuran. Memecah workflow menjadi fungsi kecil membuat kode lebih mudah diuji dan mengurangi copy-paste yang berisiko menghasilkan inkonsistensi.

2.3 Struktur Folder Proyek

Kode 2.3. Struktur folder yang disarankan untuk proyek penelitian

```

project_ml_kimia/
├── data_raw/
├── data_processed/
├── notebooks/
├── src/
├── models/
├── figures/
├── reports/
├── requirements.txt
└── README.md

```

Folder `data_raw` sebaiknya bersifat read-only secara konseptual: jangan mengedit file asli langsung. Setiap transformasi menghasilkan file baru di `data_processed`. Dengan demikian, peneliti dapat menelusuri bagaimana suatu tabel analisis dihasilkan dari sumber awal.

BAB 3

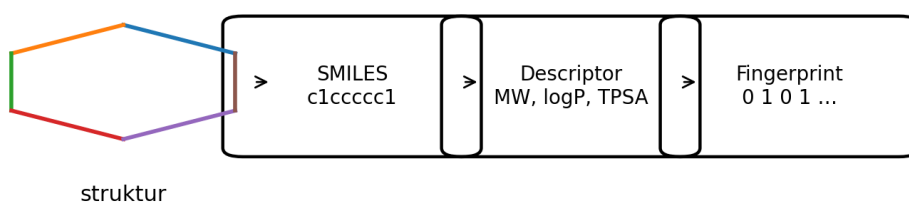
DATA KIMIA, REPRESENTASI MOLEKUL, DAN SUMBER TERBUKA

3.1 Mengapa Representasi Menentukan Apa yang Dapat Dipelajari Model

Algoritma scikit-learn pada umumnya menerima matriks numerik. Struktur molekul tidak dapat langsung dimasukkan sebagai gambar rumus bangun tanpa proses representasi. Dalam cheminformatika, SMILES adalah notasi teks untuk struktur, descriptor adalah ringkasan numerik sifat atau topologi, sedangkan fingerprint mengkodekan keberadaan lingkungan atom atau substruktur menjadi vektor panjang tetap.

Descriptor seperti massa molekul, logP, TPSA, jumlah donor dan akseptor ikatan hidrogen, jumlah ikatan rotatable, jumlah cincin, dan fraction Csp3 mudah diinterpretasikan. Namun descriptor yang sedikit dapat kehilangan detail struktur. Fingerprint Morgan menyimpan informasi lingkungan atom yang lebih kaya, tetapi interpretasi setiap bit lebih sulit. Pilihan representasi harus sesuai tujuan riset dan ukuran dataset.

Untuk data kuantum, representasi 2D sederhana sering tidak cukup karena target dapat bergantung kuat pada geometri 3D, muatan, atau lingkungan atom. Pada tahap awal pembelajaran, descriptor 2D tetap berguna sebagai baseline. Bila baseline sudah tervalidasi, peneliti dapat membandingkan dengan representasi graph neural network, message passing, atau descriptor berbasis fisika.



Representasi mengubah struktur kimia menjadi fitur numerik yang dapat dipelajari model.

Gambar 3.1. Jalur konseptual dari struktur molekul ke representasi numerik.

3.2 SMILES, InChI, dan Identitas Kimia

SMILES ringkas dan sangat berguna untuk pipeline, tetapi satu struktur dapat memiliki lebih dari satu SMILES non-kanonik. Untuk deduplikasi, gunakan standardisasi dan canonical SMILES atau InChIKey dengan hati-hati. Garam, tautomer, protonasi, stereokimia, dan campuran dapat membuat dua record tampak berbeda walaupun secara eksperimen merujuk entitas yang berkaitan.

Audit identitas: Jangan menganggap string SMILES sebagai identitas absolut. Untuk proyek bioaktivitas, tentukan kebijakan terhadap salt stripping, neutralization, stereokimia, tautomer, dan multiple components sebelum deduplikasi.

3.3 Menghitung Descriptor dengan RDKit

Kode 3.1. Menghitung descriptor molekul dengan RDKit

```
from rdkit import Chem
from rdkit.Chem import Descriptors

smi = "CC(=O)OC1=CC=CC=C1C(=O)O" # aspirin
mol = Chem.MolFromSmiles(smi)

print("MW   =", Descriptors.MolWt(mol))
print("logP =", Descriptors.MolLogP(mol))
print("TPSA =", Descriptors.TPSA(mol))
print("HBD  =", Descriptors.NumHDonors(mol))
print("HBA  =", Descriptors.NumHAcceptors(mol))
```

RDKit juga menyediakan Morgan fingerprint. Fingerprint sangat efektif untuk model QSAR klasik karena mempertahankan informasi substruktur dalam bentuk vektor. Namun jumlah bit, radius, count vs bit vector, dan collision perlu dicatat sebagai bagian metode.

3.4 Mengambil Data dari PubChem

PubChem menyediakan PUG REST, antarmuka REST yang memungkinkan skrip mengambil properti, identifier, struktur, assay summary, dan tipe data lain melalui URL. Untuk penelitian berskala kecil sampai menengah, akses terprogram memudahkan pembuatan dataset dengan provenance yang jelas. Untuk pengambilan besar, gunakan fasilitas bulk download dan hormati kebijakan layanan.

Kode 3.2. Mengambil properti sejumlah senyawa melalui PubChem PUG REST

```
import time
import requests
import pandas as pd
from urllib.parse import quote

NAMES = [
    "aspirin", "caffeine", "acetaminophen", "ibuprofen",
```

```

    "naproxen", "diclofenac", "benzene", "toluene"
]
PROPS = (
    "MolecularFormula,MolecularWeight,XLogP,TPSA,"
    "HBondDonorCount,HBondAcceptorCount,RotatableBondCount,CanonicalSMILES"
)

def get_props(name):
    enc = quote(name, safe='')
    url = (
        "https://pubchem.ncbi.nlm.nih.gov/rest/pug/compound/name/"
        f"{enc}/property/{PROPS}/JSON"
    )
    r = requests.get(url, timeout=30)
    r.raise_for_status()
    row = r.json()["PropertyTable"]["Properties"][0]
    row["query_name"] = name
    return row

rows = []
for name in NAMES:
    try:
        rows.append(get_props(name))
        time.sleep(0.2)
    except Exception as exc:
        print("Gagal:", name, exc)

df = pd.DataFrame(rows)
df.to_csv("pubchem_molecular_properties.csv", index=False)
print(df)

```

Setelah file CSV terbentuk, peneliti dapat melakukan eksplorasi distribusi MW, XLogP, TPSA, donor/akseptor, dan menggabungkan data dengan hasil eksperimen sendiri. Simpan juga query_name dan identifier PubChem agar data dapat ditelusuri kembali.

3.5 ChEMBL untuk Data Bioaktivitas

ChEMBL adalah basis data bioaktivitas molekul yang menghubungkan senyawa dengan target dan assay. Data ChEMBL kaya tetapi tidak boleh diperlakukan sebagai satu tabel homogen. IC50 dari assay seluler, binding assay, dan format eksperimen berbeda dapat memiliki konteks biologis berbeda. Sebelum pemodelan, lakukan filter target, tipe aktivitas, satuan, relation, organism, assay, dan quality flag yang relevan.

BAB 4

DATA PREPARATION DAN QUALITY CONTROL KIMIA

4.1 Audit Sebelum Modeling

Data preparation bukan tahap administratif; ia menentukan validitas hasil. Audit awal minimal mencakup jumlah record, kolom, tipe data, nilai hilang, rentang target, satuan, duplikasi, distribusi kelas, struktur invalid, outlier, dan provenance. Pada data kimia, audit tambahan mencakup multiple components, salt, formal charge, stereokimia, dan kemungkinan pengukuran berulang untuk senyawa yang sama.

Missing value dapat muncul karena descriptor gagal dihitung, eksperimen tidak dilakukan, atau data tidak dilaporkan. Penyebabnya perlu dibedakan. Mengisi missing value dengan median dapat diterima untuk beberapa fitur numerik, tetapi tidak boleh digunakan untuk 'menciptakan' target eksperimen yang tidak ada. Target yang hilang biasanya dikeluarkan dari task supervised tertentu.

Duplikasi harus diperiksa setelah standardisasi struktur. Bila satu molekul memiliki beberapa pengukuran target, pilihan agregasi—median, mean, nilai terbaru, assay tertentu, atau mempertahankan semua record dengan group-aware split—harus sesuai tujuan penelitian dan dilaporkan.

4.2 Encoding, Scaling, dan Standardisasi

StandardScaler mentransformasikan fitur menggunakan $z = (x - \mu) / \sigma$, dengan μ dan σ dihitung hanya dari data training. Ini penting pada model sensitif skala seperti KNN, logistic regression dengan regularisasi, SVM, dan banyak metode berbasis jarak. Tree-based model umumnya tidak membutuhkan scaling untuk alasan yang sama, walaupun preprocessing lain tetap dapat diperlukan.

Data kategorikal seperti jenis pelarut, kelas katalis, atau tipe assay dapat di-encode menggunakan one-hot encoding bila kategori tidak memiliki urutan intrinsik. Jika fitur kategorikal mempunyai banyak level, pertimbangkan apakah kategori tersebut sebenarnya proxy untuk kelompok eksperimen yang seharusnya ditangani lewat group split atau stratifikasi desain.

4.3 Train, Validation, Test: Random Split Tidak Selalu Cukup

train_test_split menghasilkan subset acak dan sangat berguna untuk baseline. Namun pada data molekuler, random split dapat menempatkan analog dekat pada train dan test sehingga performa tampak lebih tinggi daripada kemampuan model menghadapi scaffold baru. Untuk QSAR, scaffold split sering lebih informatif apabila skenario penggunaan adalah memprediksi seri kimia baru.

Time split sesuai bila penelitian ingin mensimulasikan prediksi prospektif: data lama untuk training dan data lebih baru untuk testing. Group split berguna ketika beberapa baris berasal dari batch, pasien, eksperimen, atau molekul yang sama. Prinsip umumnya adalah mencegah 'kerabat dekat' bocor ke kedua sisi batas evaluasi.

Validation set digunakan untuk memilih model atau hyperparameter. Test set seharusnya disentuh sesedikit mungkin dan hanya digunakan untuk estimasi akhir. Bila test set dipakai berulang untuk memutuskan fitur atau parameter, ia berubah fungsi menjadi validation set.

4.4 Data Leakage dalam Riset Kimia

Tabel 4.1. Bentuk data leakage yang umum dalam proyek kimia

Sumber leakage	Contoh kimia	Pencegahan
Preprocessing sebelum split	scaler dihitung dari seluruh dataset	fit transformer hanya pada training melalui Pipeline
Duplikasi struktur	molekul sama di train dan test	canonicalize, deduplicate, atau group split
Analog terlalu dekat	seri scaffold tersebar acak	scaffold/group split
Fitur pasca-eksperimen	fitur memakai informasi yang baru diketahui setelah target diukur	gunakan hanya informasi tersedia saat prediksi
Target proxy	descriptor/kolom mengandung transformasi langsung dari target	audit definisi setiap fitur

BAB 5

PIPELINE DAN EVALUASI MODEL

5.1 Mengapa Pipeline Penting

Pipeline mengikat preprocessing dan model menjadi satu objek. Saat cross-validation, setiap fold menjalankan fit scaler atau imputer hanya pada subset training fold tersebut. Ini mengurangi risiko leakage dan membuat model lebih mudah disimpan sebagai satu artefak. Pipeline juga membuat GridSearchCV dapat menala parameter preprocessing dan model secara konsisten.

Kode 5.1. Pipeline scaling dan regresi Ridge

```
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Ridge

pipe = Pipeline([
    ("scale", StandardScaler()),
    ("model", Ridge(alpha=1.0)),
])
pipe.fit(X_train, y_train)
pred = pipe.predict(X_test)
```

5.2 Metrik Regresi

$MAE = (1/n) \sum |y_i - \hat{y}_i|$ menyatakan rata-rata besar error absolut dalam satuan target. $RMSE = \sqrt{(1/n) \sum (y_i - \hat{y}_i)^2}$ memberi penalti lebih besar pada error ekstrem. $R^2 = 1 - [\sum (y_i - \hat{y}_i)^2 / \sum (y_i - \bar{y})^2]$ menyatakan proporsi variasi relatif terhadap baseline mean, tetapi tidak memberi informasi langsung tentang besarnya error dalam satuan kimia.

Untuk target logS atau pIC50, MAE 0,5 berarti error tipikal sekitar setengah log unit, yang secara skala konsentrasi dapat bermakna beberapa kali lipat. Karena itu, selalu terjemahkan metrik ke konsekuensi kimia dan tampilkan residual. Nilai R^2 tinggi pada rentang target luas dapat tetap menyembunyikan error besar pada daerah yang paling penting.

5.3 Metrik Klasifikasi

Accuracy berguna bila kelas relatif seimbang dan biaya salah klasifikasi serupa. $Precision = TP/(TP+FP)$ menjawab: dari prediksi positif, berapa yang benar. $Recall = TP/(TP+FN)$ menjawab: dari seluruh positif nyata, berapa yang tertangkap. F1 adalah harmonic mean precision dan recall. ROC-AUC menilai kemampuan ranking lintas threshold, sedangkan precision-recall curve sering lebih informatif pada kelas positif yang jarang.

Confusion matrix perlu dibaca dalam konteks penelitian. Pada screening senyawa aktif, false negative dapat berarti kandidat potensial terlewat; pada penyaringan toksisitas, false negative dapat memiliki konsekuensi berbeda. Karena itu, threshold 0,5 tidak selalu optimal dan harus dipilih berdasarkan kebutuhan aplikasi serta validasi yang terpisah.

5.4 Cross-Validation dan Ketidakpastian

Satu split dapat kebetulan mudah atau sulit. Cross-validation memberi distribusi performa pada beberapa partisi. Laporkan mean dan standard deviation atau interval yang relevan. Untuk data berkelompok, gunakan GroupKFold atau strategi split yang menghormati kelompok. Untuk chemical scaffold, gunakan splitter yang sesuai struktur, atau buat scaffold group terlebih dahulu.

BAB 6

STUDI KASUS REGRESI: PREDIKSI KELARUTAN AIR ESOL

6.1 Latar Belakang Kimia dan Dataset

Kelarutan air merupakan sifat penting dalam kimia obat, formulasi, lingkungan, dan pemisahan. Dataset ESOL yang diasosiasikan dengan karya Delaney memuat struktur molekul dan measured log solubility. Karena ukurannya relatif kecil dan formatnya sederhana, dataset ini menjadi benchmark yang baik untuk mempelajari QSAR regresi.

Tujuan studi kasus bukan mereplikasi persis model ESOL asli, melainkan membangun baseline ML modern menggunakan descriptor RDKit. Fitur meliputi massa molekul, MolLogP, TPSA, donor/akseptor ikatan hidrogen, ikatan rotatable, jumlah cincin, dan fraction Csp3. Target adalah measured log solubility in mols per litre.

Baseline descriptor memiliki kelebihan interpretasi: peneliti dapat mengaitkan performa dan feature importance dengan konsep ukuran molekul, lipofilisitas, polaritas, dan fleksibilitas. Kekurangannya adalah struktur detail yang membedakan isomer atau motif tertentu dapat hilang.

6.2 Implementasi Linear, Ridge, dan Random Forest

Kode 6.1. Regresi ESOL menggunakan descriptor RDKit

```
import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression, Ridge
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/delaney-processed.csv"
TARGET = "measured log solubility in mols per litre"

df = pd.read_csv(URL)

def desc(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol),
        Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol),
        Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol),
```

```

        Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol),
        Descriptors.FractionCSP3(mol),
    ]

    names = ["MolWt", "MolLogP", "TPSA", "HBD", "HBA", "RotB", "Rings", "FracCSP3"]
    rows = []
    for smi in df["smiles"]:
        rows.append(desc(smi))
    X = pd.DataFrame(rows, columns=names)
    y = df[TARGET].astype(float)
    valid = X.notna().all(axis=1) & y.notna()
    X, y = X.loc[valid], y.loc[valid]

    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=0.20, random_state=42
    )

    models = {
        "Linear": Pipeline([
            ("imputer", SimpleImputer(strategy="median")),
            ("scale", StandardScaler()),
            ("model", LinearRegression()),
        ]),
        "Ridge": Pipeline([
            ("imputer", SimpleImputer(strategy="median")),
            ("scale", StandardScaler()),
            ("model", Ridge(alpha=1.0)),
        ]),
        "RandomForest": RandomForestRegressor(
            n_estimators=500, random_state=42, n_jobs=-1
        ),
    }

    for name, model in models.items():
        model.fit(X_train, y_train)
        pred = model.predict(X_test)
        mae = mean_absolute_error(y_test, pred)
        rmse = mean_squared_error(y_test, pred) ** 0.5
        r2 = r2_score(y_test, pred)
        print(f"{name:12s} MAE={mae:.3f} RMSE={rmse:.3f} R2={r2:.3f}")

```

Jalankan skrip dan catat MAE, RMSE, serta R^2 untuk setiap model. Jangan menyalin nilai contoh dari sumber lain ke laporan Anda; metrik harus berasal dari run lokal dengan versi data, library, dan seed yang dicatat. Model linear menyediakan baseline sederhana; Ridge menambahkan regularisasi; Random Forest menangkap hubungan nonlinier dan interaksi antar-descriptor.

6.3 Diagnostik Residual

Kode 6.2. Plot residual untuk mengidentifikasi bias model

```

import matplotlib.pyplot as plt

residual = y_test - pred
plt.figure(figsize=(6, 5))
plt.scatter(pred, residual, s=20)
plt.axhline(0, linewidth=1)
plt.xlabel("Prediksi logS")
plt.ylabel("Residual (aktual - prediksi)")
plt.tight_layout()
plt.show()

```

Residual yang membentuk pola melengkung menandakan hubungan belum tertangkap; sebaran yang meningkat pada rentang tertentu menandakan heteroskedastisitas; beberapa titik ekstrem perlu diperiksa apakah merupakan domain kimia berbeda, kesalahan data, atau molekul memang sulit diprediksi. Outlier tidak boleh dihapus hanya karena memperburuk metrik tanpa alasan ilmiah.

6.4 Fingerprint Morgan sebagai Representasi Alternatif

Kode 6.3. Regresi ESOL menggunakan Morgan fingerprint 2048 bit

```
import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import rdFingerprintGenerator
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/delaney-processed.csv"
TARGET = "measured log solubility in mols per litre"
BIT_SIZE = 2048

df = pd.read_csv(URL)
gen = rdFingerprintGenerator.GetMorganGenerator(radius=2, fpSize=BIT_SIZE)

def fp(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    arr = np.zeros((BIT_SIZE,), dtype=np.uint8)
    arr[:] = gen.GetFingerprintAsNumPy(mol)
    return arr

fps, ys = [], []
for smi, y in zip(df["smiles"], df[TARGET]):
    v = fp(smi)
    if v is not None and pd.notna(y):
        fps.append(v)
        ys.append(float(y))
X = np.asarray(fps)
y = np.asarray(ys)

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.20, random_state=42
)
model = RandomForestRegressor(
    n_estimators=500, max_features="sqrt", random_state=42, n_jobs=-1
)
model.fit(X_train, y_train)
pred = model.predict(X_test)
print("MAE:", mean_absolute_error(y_test, pred))
print("R2 :", r2_score(y_test, pred))
```

Membandingkan descriptor dengan Morgan fingerprint menunjukkan trade-off antara interpretabilitas dan representasi struktur. Untuk evaluasi yang adil, gunakan split dan seed yang sama. Pada penelitian yang lebih serius, lakukan scaffold split

dan cross-validation. Jika Morgan unggul pada random split namun turun tajam pada scaffold split, model mungkin terutama memanfaatkan kemiripan lokal antar-analog.

BAB 7

STUDI KASUS REGRESI KIMIA FISIK: FREESOLV DAN LIPOFILISITAS

7.1 Energi Hidrasi FreeSolv

FreeSolv adalah basis data energi bebas hidrasi eksperimental dan hasil perhitungan untuk molekul kecil netral. Dataset ini berguna untuk menunjukkan bahwa target energi memiliki makna fisik dan satuan yang harus dijaga. Error model perlu dibaca dalam satuan energi, bukan hanya skor statistik.

Baseline berbasis descriptor 2D tidak mengandung geometri solvent atau detail interaksi eksplisit, sehingga performanya memiliki batas. Justru keterbatasan ini mendidik: model sederhana dapat menjadi pembanding sebelum menggunakan descriptor 3D, metode graph, atau model berbasis simulasi.

Ketika target fisik berasal dari kombinasi eksperimen dan kalkulasi, peneliti harus memastikan kolom yang dipilih benar. Jangan mencampur nilai calculated dan experimental sebagai satu label tanpa tujuan yang jelas.

Kode 7.1. Regresi energi hidrasi FreeSolv

```
import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import ExtraTreesRegressor, GradientBoostingRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = "https://deepchemdata.s3.us-west-1.amazonaws.com/datasets/freesolv.csv.gz"
df = pd.read_csv(URL)
SMILES_COL = "smiles" if "smiles" in df.columns else "mol"
TARGET = "y"

def desc(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol), Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol), Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol), Descriptors.FractionCSP3(mol),
    ]

X, y = [], []
for smi, val in zip(df[SMILES_COL], df[TARGET]):
    v = desc(smi)
    if v is not None and pd.notna(val):
        X.append(v); y.append(float(val))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=42)

for model in [
    ExtraTreesRegressor(n_estimators=500, random_state=42, n_jobs=-1),
```

```

GradientBoostingRegressor(random_state=42),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(type(model).__name__)
    print("MAE =", mean_absolute_error(yte, pred))
    print("RMSE=", mean_squared_error(yte, pred) ** 0.5)
    print("R2 =", r2_score(yte, pred))

```

7.2 Lipofilisitas dan LogD

Dataset Lipophilicity MoleculeNet menyediakan nilai logD oktanol/air pada pH 7,4 untuk ribuan senyawa. LogD berbeda dari logP karena logD bergantung pada pH dan mencakup distribusi semua bentuk ionisasi pada kondisi tersebut. Perbedaan ini penting agar model tidak diberi label atau interpretasi yang salah.

Untuk regresi logD, workflow sama dengan ESOL: validasi struktur, hitung fitur, split, model, dan evaluasi. Untuk eksplorasi, dataset yang sama dapat dipakai pada Bab 9 untuk memetakan chemical space dengan PCA dan K-Means.

7.3 Membandingkan Model tanpa Terjebak Leaderboard

Model terbaik secara statistik belum tentu paling berguna secara ilmiah. Pertimbangkan interpretabilitas, stabilitas, biaya komputasi, ketahanan pada scaffold baru, dan kemampuan menghasilkan uncertainty estimate. Gunakan baseline sederhana dan jelaskan peningkatan model secara incremental. Tanpa baseline, sulit mengetahui apakah model kompleks benar-benar belajar hal yang bermakna.

BAB 8

STUDI KASUS KLASIFIKASI: BLOOD-BRAIN BARRIER PENETRATION

8.1 Definisi Masalah Klasifikasi

Dataset BBBP berisi label biner penetrasi blood-brain barrier untuk lebih dari dua ribu senyawa. Studi kasus ini mengajarkan bahwa target biner memerlukan analisis distribusi kelas dan metrik selain accuracy. Dalam konteks kimia obat, sifat lipofilisitas, polar surface area, hydrogen bonding, ukuran, dan fleksibilitas sering terkait dengan permeabilitas, tetapi hubungan tidak cukup disederhanakan menjadi satu aturan.

8.2 Logistic Regression dan Random Forest

Kode 8.1. Klasifikasi BBBP dengan descriptor RDKit

```
import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import (
    accuracy_score, precision_score, recall_score, f1_score,
    roc_auc_score, confusion_matrix, classification_report
)

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/BBBP.csv"
df = pd.read_csv(URL)

def desc(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol), Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol), Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol), Descriptors.FractionCSP3(mol),
    ]

features, target = [], []
for smi, y in zip(df["smiles"], df["p_np"]):
    v = desc(smi)
    if v is not None and pd.notna(y):
        features.append(v)
        target.append(int(y))
X = np.asarray(features, dtype=float)
y = np.asarray(target, dtype=int)

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.20, random_state=42, stratify=y
)
```

```

models = {
    "Logistic": Pipeline([
        ("scale", StandardScaler()),
        ("model", LogisticRegression(max_iter=3000, class_weight="balanced")),
    ]),
    "RandomForest": RandomForestClassifier(
        n_estimators=500, random_state=42, class_weight="balanced", n_jobs=-1
    ),
}

for name, model in models.items():
    model.fit(X_train, y_train)
    pred = model.predict(X_test)
    probab = model.predict_proba(X_test)[:, 1]
    print("\n", name)
    print("accuracy =", accuracy_score(y_test, pred))
    print("precision =", precision_score(y_test, pred, zero_division=0))
    print("recall   =", recall_score(y_test, pred, zero_division=0))
    print("f1       =", f1_score(y_test, pred, zero_division=0))
    print("ROC-AUC  =", roc_auc_score(y_test, probab))
    print("confusion matrix:\n", confusion_matrix(y_test, pred))
    print(classification_report(y_test, pred, zero_division=0))

```

Logistic regression memberikan probabilitas kelas melalui kombinasi linear fitur setelah scaling. Random Forest dapat menangkap hubungan nonlinier. `class_weight='balanced'` adalah baseline sederhana bila kelas tidak seimbang, tetapi bukan pengganti evaluasi yang tepat. Periksa confusion matrix dan ROC-AUC, lalu tentukan threshold sesuai tujuan.

8.3 KNN dan Grid Search

Kode 8.2. GridSearchCV untuk KNN pada BBBP

```

import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split, GridSearchCV,
StratifiedKFold
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report, roc_auc_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/BBBP.csv"
df = pd.read_csv(URL)

def desc(smi):
    mol = Chem.MolFromSmiles(smi)
    if mol is None: return None
    return [Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
            Descriptors.TPSA(mol),
            Descriptors.NumHDonors(mol), Descriptors.NumHAcceptors(mol),
            Descriptors.NumRotatableBonds(mol), Descriptors.RingCount(mol)]

X, y = [], []
for smi, label in zip(df["smiles"], df["p_np"]):
    v = desc(smi)
    if v is not None and pd.notna(label):
        X.append(v); y.append(int(label))
X, y = np.asarray(X), np.asarray(y)

```

```

Xtr, Xte, ytr, yte = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)
pipe = Pipeline([
    ("scale", StandardScaler()),
    ("knn", KNeighborsClassifier()),
])
param_grid = {
    "knn__n_neighbors": [3, 5, 7, 11, 15, 21],
    "knn__weights": ["uniform", "distance"],
    "knn__p": [1, 2],
}
cv = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
search = GridSearchCV(pipe, param_grid, scoring="roc_auc", cv=cv, n_jobs=-1)
search.fit(Xtr, ytr)
print("Best:", search.best_params_, search.best_score_)
pred = search.predict(Xte)
prob = search.predict_proba(Xte)[: , 1]
print("Test ROC-AUC:", roc_auc_score(yte, prob))
print(classification_report(yte, pred, zero_division=0))

```

KNN bergantung pada jarak sehingga scaling wajib. GridSearchCV di dalam Pipeline mencegah scaler mempelajari statistik dari fold validation. Parameter `n_neighbors` mengontrol bias-variance: `k` terlalu kecil sensitif noise, sedangkan `k` terlalu besar dapat mengaburkan struktur lokal chemical space.

8.4 Scaffold Split dan Evaluasi yang Lebih Realistis

Random split pada BBBP berguna untuk demonstrasi, tetapi publikasi QSAR sebaiknya menguji split yang memisahkan scaffold. Tujuan evaluasi bukan membuat angka setinggi mungkin, tetapi memperkirakan kinerja pada molekul yang belum dikenal model. Jika performa scaffold split turun, laporkan secara transparan dan gunakan hasil tersebut untuk mendefinisikan domain model.

BAB 9

UNSUPERVISED LEARNING: CHEMICAL SPACE, K-MEANS, DAN PCA

9.1 Chemical Space sebagai Ruang Multidimensi

Setiap molecule descriptor menambah satu dimensi. Delapan descriptor berarti setiap senyawa adalah titik dalam ruang delapan dimensi. Clustering berusaha menemukan kelompok berdasarkan kedekatan, sedangkan PCA mencari kombinasi linear fitur yang menjelaskan variasi terbesar. Kedua metode membantu eksplorasi sebelum modeling supervised.

9.2 Menentukan Jumlah Cluster

Elbow method menilai penurunan within-cluster sum of squares ketika jumlah cluster k bertambah. Silhouette score menggabungkan kekompakan internal dan pemisahan antar-cluster. Tidak ada angka k yang otomatis 'benar' secara kimia; cluster perlu diinterpretasikan dengan melihat scaffold, descriptor, dan target eksperimen.

9.3 Praktik PCA + K-Means pada Dataset Lipofilisitas

Kode 9.1. Pemetaan chemical space dengan PCA dan K-Means

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/Lipophilicity.csv"
df = pd.read_csv(URL)

def desc(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol), Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol), Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol), Descriptors.FractionCSP3(mol),
    ]

rows, keep = [], []
for i, smi in enumerate(df["smiles"]):
    v = desc(smi)
    if v is not None:
        rows.append(v)
```

```

        keep.append(i)
X = np.asarray(rows, dtype=float)
scaler = StandardScaler()
Z = scaler.fit_transform(X)

scores = {}
for k in range(2, 9):
    labels = KMeans(n_clusters=k, random_state=42, n_init="auto").fit_predict(Z)
    scores[k] = silhouette_score(Z, labels)
best_k = max(scores, key=scores.get)
print("Silhouette:", scores)
print("best_k:", best_k)

kmeans = KMeans(n_clusters=best_k, random_state=42, n_init="auto")
labels = kmeans.fit_predict(Z)
pca = PCA(n_components=2, random_state=42)
P = pca.fit_transform(Z)

plt.figure(figsize=(8, 6))
plt.scatter(P[:, 0], P[:, 1], c=labels, s=12)
plt.xlabel("PC1")
plt.ylabel("PC2")
plt.title("Chemical Space: PCA + K-Means")
plt.tight_layout()
plt.savefig("pca_kmeans_lipophilicity.png", dpi=180)
print("Explained variance ratio:", pca.explained_variance_ratio_)

```

Kode mencari k dengan silhouette score, kemudian memproyeksikan descriptor terstandarisasi ke dua principal components. Plot PCA adalah proyeksi, bukan keseluruhan ruang. Dua titik yang tampak dekat pada PC1–PC2 tidak selalu sangat mirip di seluruh dimensi, terutama bila explained variance dua komponen rendah.

9.4 Clustering untuk Desain Eksperimen

Clustering dapat digunakan untuk memilih perwakilan dari chemical space. Misalnya, daripada menguji 1.000 senyawa yang sangat mirip, peneliti dapat mengambil kandidat dari cluster berbeda untuk meningkatkan cakupan struktur. Strategi ini sebaiknya dikombinasikan dengan kriteria kimia, ketersediaan material, dan keselamatan; clustering hanya salah satu sumber informasi.

BAB 10

QSAR MODERN DENGAN DESCRIPTOR DAN MOLECULAR FINGERPRINT

10.1 Dari Struktur ke Fingerprint Morgan

Morgan fingerprint membangun identifier lingkungan atom secara iteratif hingga radius tertentu dan kemudian memetakan lingkungan tersebut ke vektor. Dalam bentuk bit vector, sebuah bit menunjukkan keberadaan fitur hasil hashing. Ukuran 2048 bit dan radius 2 adalah konfigurasi populer sebagai baseline, tetapi bukan aturan universal.

Fingerprint cocok untuk chemical similarity dan QSAR, namun collision dapat terjadi ketika fitur berbeda masuk bit sama. Count fingerprint menyimpan frekuensi dan dapat memberi informasi tambahan. Dalam laporan penelitian, catat jenis fingerprint, radius, fpSize, useChirality, count/bit, serta cara standardisasi struktur.

10.2 Similarity dan Applicability Domain

Tanimoto similarity untuk bit fingerprint dapat ditulis $T = c/(a+b-c)$, dengan a dan b jumlah bit aktif pada dua molekul dan c jumlah bit aktif bersama. Salah satu pendekatan applicability domain adalah menghitung kemiripan kandidat terhadap molekul training terdekat. Kandidat dengan maximum similarity sangat rendah sebaiknya dianggap extrapolation.

Kode 10.1. Menghitung similarity Tanimoto dua molekul

```
from rdkit import Chem, DataStructs
from rdkit.Chem import rdFingerprintGenerator

gen = rdFingerprintGenerator.GetMorganGenerator(radius=2, fpSize=2048)
a = Chem.MolFromSmiles("CC(=O)OC1=CC=CC=C1C(=O)O")
b = Chem.MolFromSmiles("CC(C)CC1=CC=C(C=C1)C(C)C(=O)O")
fa = gen.GetFingerprint(a)
fb = gen.GetFingerprint(b)
print(DataStructs.TanimotoSimilarity(fa, fb))
```

10.3 Feature Importance dan Interpretasi

Feature importance dari tree-based model menunjukkan seberapa besar fitur membantu pemisahan pada model, tetapi bukan kausalitas. Permutation importance lebih dekat dengan pertanyaan 'seberapa banyak performa turun ketika fitur diacak', namun tetap dapat bias pada fitur berkorelasi. SHAP dan metode interpretabilitas lain dapat digunakan, tetapi harus tetap dibaca bersama pengetahuan kimia.

BAB 11

STUDI KASUS BIOAKTIVITAS: MEMBANGUN DATASET HER2/ERBB2 DARI ChEMBL

11.1 Mengapa Data Bioaktivitas Memerlukan Audit Lebih Ketat

Bioaktivitas adalah domain yang kaya namun heterogen. Satu target dapat mempunyai IC₅₀, K_i, K_d, EC₅₀, inhibition percentage, dan jenis endpoint lain. Menyatukan semuanya tanpa transformasi dan justifikasi akan menghasilkan target yang tidak homogen. Bahkan untuk IC₅₀, assay format, construct protein, substrat, kondisi buffer, dan sistem biologis dapat berbeda.

Workflow yang aman dimulai dengan memilih target manusia secara eksplisit, menyaring standard_type dan standard_units, menangani relation, menghapus nilai nonpositif, memeriksa canonical_smiles, dan menyimpan assay_chembl_id. Transformasi pIC₅₀ hanya dilakukan setelah nilai berada dalam satuan yang konsisten.

Untuk IC₅₀ dalam nM, $pIC_{50} = 9 - \log_{10}(IC_{50_nM})$. Rumus ini berasal dari $pIC_{50} = -\log_{10}(IC_{50} \text{ dalam mol/L})$. Nilai pIC₅₀ lebih besar berarti konsentrasi IC₅₀ lebih kecil. Namun pIC₅₀ dari assay berbeda tetap belum otomatis sebanding secara biologis.

11.2 Mengambil dan Memodelkan Data HER2/ERBB2

Kode 11.1. Mengambil IC₅₀ target ERBB2/HER2 dari ChEMBL dan membangun baseline QSAR

```
import numpy as np
import pandas as pd
from chembl_webresource_client.new_client import new_client
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, r2_score

# HER2/ERBB2: cari target secara programatik, jangan hard-code ID tanpa
verifikasi.
target_api = new_client.target
hits = list(target_api.search("ERBB2"))
for h in hits[:10]:
    print(h.get("target_chembl_id"), h.get("pref_name"), h.get("organism"))

target = next(
    h for h in hits
    if h.get("organism") == "Homo sapiens" and
    (h.get("pref_name") or "").lower().find("erbb2") >= 0
)
target_id = target["target_chembl_id"]
print("Target dipilih:", target_id, target.get("pref_name"))

activities = new_client.activity.filter(
```

```

        target_chembl_id=target_id,
        standard_type="IC50",
        standard_units="nM",
        standard_relation="="
    ).only([
        "molecule_chembl_id", "canonical_smiles", "standard_value",
        "standard_type", "standard_units", "assay_chembl_id"
    ])

    rows = []
    for a in activities:
        smi = a.get("canonical_smiles")
        val = a.get("standard_value")
        try:
            nm = float(val)
        except (TypeError, ValueError):
            continue
        if not smi or nm <= 0:
            continue
        mol = Chem.MolFromSmiles(smi)
        if mol is None:
            continue
        pIC50 = 9.0 - np.log10(nm) # IC50 nM -> mol/L lalu -log10
        rows.append({
            "chembl_id": a.get("molecule_chembl_id"),
            "smiles": smi,
            "assay": a.get("assay_chembl_id"),
            "pIC50": pIC50,
            "MolWt": Descriptors.MolWt(mol),
            "MolLogP": Descriptors.MolLogP(mol),
            "TPSA": Descriptors.TPSA(mol),
            "HBD": Descriptors.NumHDonors(mol),
            "HBA": Descriptors.NumHAcceptors(mol),
            "RotB": Descriptors.NumRotatableBonds(mol),
            "Rings": Descriptors.RingCount(mol),
        })

    df = pd.DataFrame(rows)
    # Agregasi sederhana per senyawa mengurangi duplikasi pengukuran.
    df = df.groupby(["chembl_id", "smiles"],
        as_index=False).median(numeric_only=True)
    print("Jumlah senyawa unik:", len(df))

    def cols():
        return ["MolWt", "MolLogP", "TPSA", "HBD", "HBA", "RotB", "Rings"]

    X = df[cols()]
    y = df["pIC50"]
    Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=42)
    model = RandomForestRegressor(n_estimators=500, random_state=42, n_jobs=-1)
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print("MAE pIC50 =", mean_absolute_error(yte, pred))
    print("R2          =", r2_score(yte, pred))
    print("Untuk publikasi, audit assay, duplikasi, struktur, censoring, dan split scaffold.")

```

Scrip sengaja mencari target ERBB2 terlebih dahulu, kemudian memilih record Homo sapiens, sehingga ID target tidak diasumsikan tetap. Pengambilan data diakhiri dengan agregasi median per senyawa sebagai baseline. Untuk riset publikasi, tahap ini perlu diperkuat dengan audit assay, confidence score, target

type, variant, measurement relation, censoring, salt/tautomer handling, dan scaffold split.

11.3 Dari Baseline ke Model Riset

1. Simpan snapshot data mentah ChEMBL dalam CSV dengan tanggal akses dan versi/release bila tersedia.
2. Lakukan standardisasi struktur yang terdokumentasi; pertahankan mapping ke ID asli.
3. Definisikan aturan agregasi pengukuran dan ukur variasi intra-molekul sebelum agregasi.
4. Gunakan scaffold split atau time split agar evaluasi lebih dekat dengan penemuan senyawa baru.
5. Laporkan applicability domain dan contoh false positive/false negative atau residual ekstrem.
6. Bandingkan descriptor baseline dengan fingerprint dan model yang lebih kompleks hanya setelah baseline tervalidasi.

BAB 12

MACHINE LEARNING UNTUK DATA KIMIA KUANTUM

12.1 Posisi ML terhadap Perhitungan Kuantum

Metode kuantum seperti DFT menyediakan informasi elektronik berdasarkan model fisik, tetapi biayanya meningkat dengan ukuran sistem dan tingkat teori. ML dapat dipakai sebagai surrogate model untuk memprediksi sifat yang telah dihitung pada banyak molekul, melakukan screening awal, atau mempelajari koreksi terhadap metode yang lebih murah. ML tidak menghapus kebutuhan definisi tingkat teori, geometri, basis set, muatan, spin, dan kondisi perhitungan.

12.2 QM9 sebagai Dataset Benchmark

QM9 memuat molekul kecil organik dan berbagai properti kuantum seperti dipole moment, polarizability, HOMO, LUMO, gap, dan energi. Dataset ini cocok untuk belajar supervised regression pada properti kuantum. Namun unit dan preprocessing harus diverifikasi terhadap dokumentasi versi dataset yang benar sebelum membandingkan error dengan literatur.

Kode 12.1. Baseline prediksi HOMO-LUMO gap pada subset QM9

```
import pandas as pd
import numpy as np
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import HistGradientBoostingRegressor
from sklearn.metrics import mean_absolute_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/qm9.csv"
NROWS = 25000 # subset agar contoh ringan; ubah None untuk seluruh data

df = pd.read_csv(URL, nrows=NROWS)
smiles_col = "smiles" if "smiles" in df.columns else "mol"
TARGET = "gap"

def desc(smiles):
    mol = Chem.MolFromSmiles(smiles)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol), Descriptors.TPSA(mol),
        Descriptors.NumHDonors(mol), Descriptors.NumHAcceptors(mol),
        Descriptors.NumRotatableBonds(mol), Descriptors.RingCount(mol),
        Descriptors.FractionCSP3(mol), Descriptors.HeavyAtomCount(mol),
    ]

X, y = [], []
for smi, val in zip(df[smiles_col], df[TARGET]):
    v = desc(smi)
    if v is not None and pd.notna(val):
        X.append(v); y.append(float(val))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=42)
model = HistGradientBoostingRegressor(random_state=42)
```

```

model.fit(Xtr, ytr)
pred = model.predict(Xte)
print("MAE =", mean_absolute_error(yte, pred))
print("R2 =", r2_score(yte, pred))
print("Catatan: verifikasi unit target terhadap versi dataset yang digunakan.")

```

Contoh sengaja memakai subset agar dapat dijalankan di komputer biasa. Descriptor 2D hanya baseline; gap elektron bergantung pada struktur elektronik dan geometri sehingga model graph 3D atau representasi physics-aware berpotensi lebih sesuai. Penurunan error dari model kompleks harus dibandingkan pada split dan unit yang sama.

12.3 Menghubungkan ML dengan Workflow QM Komputasi

Pada proyek laboratorium komputasi, output dari banyak run QM dapat digabung menjadi dataset: struktur, metode, basis, energi, HOMO, LUMO, gap, dipole, frekuensi, descriptor geometri, dan status konvergensi. Sebelum ML, filter perhitungan gagal, tandai imaginary frequencies bila relevan, serta pastikan target berasal dari prosedur yang konsisten. Mencampur metode/basis berbeda dapat dilakukan hanya bila metode tersebut dimodelkan sebagai fitur atau tujuan harmonisasi didefinisikan dengan jelas.

12.4 Δ -Learning dan Multi-Fidelity

Salah satu strategi adalah Δ -learning: model memprediksi selisih antara metode murah dan metode referensi. Jika E_{low} adalah energi metode murah dan E_{high} energi referensi, target $\Delta E = E_{\text{high}} - E_{\text{low}}$. Prediksi akhir adalah $E_{\text{high,pred}} = E_{\text{low}} + \Delta E_{\text{pred}}$. Pendekatan ini dapat lebih mudah daripada memprediksi E_{high} langsung karena model mempelajari koreksi. Validitasnya bergantung pada konsistensi pasangan data dan domain struktur.

BAB 13

MODEL IMPROVEMENT, VALIDASI, DAN REPRODUCIBILITY

13.1 Hyperparameter Tuning yang Tidak Membocorkan Data

Hyperparameter tuning harus dilakukan hanya pada data training melalui cross-validation. Test set tidak digunakan untuk memilih `n_estimators`, `max_depth`, `C`, `k`, atau parameter lain. Pipeline memastikan transformer termasuk dalam proses CV. `GridSearchCV` cocok untuk ruang parameter kecil; `RandomizedSearchCV` lebih efisien untuk ruang besar.

13.2 Feature Selection dan Multikolinearitas

Descriptor kimia sering berkorelasi: MW, heavy atom count, dan volume misalnya membawa informasi ukuran yang tumpang tindih. Pada model linear, multikolinearitas dapat membuat koefisien tidak stabil. Regularisasi Ridge atau Elastic Net dapat membantu, tetapi seleksi fitur sebaiknya dilakukan di dalam cross-validation untuk menghindari optimistic bias.

13.3 Calibration, Threshold, dan Probabilitas

Probabilitas 0,8 dari classifier tidak selalu berarti 80% peluang nyata. Calibration curve dapat menguji apakah probabilitas sesuai frekuensi empiris. Dalam screening, threshold dapat disesuaikan untuk menekankan recall atau precision. Threshold dipilih pada validation data, bukan test set.

13.4 Checklist Reproducibility

Tabel 13.1. Checklist minimum reproducibility proyek ML kimia

Komponen	Yang harus dicatat
Data	sumber, DOI/URL, tanggal akses, release, hash file bila memungkinkan
Cleaning	aturan missing, duplikasi, standardisasi struktur, unit
Features	descriptor/fingerprint, parameter, versi RDKit
Split	metode, seed, scaffold/group/time logic
Model	algoritma, hyperparameter, versi scikit-learn
Evaluation	metric, fold, CI/SD, threshold klasifikasi
Artefak	model serial, scaler/pipeline, code commit, figures
Domain	cakupan kimia, out-of-domain rule, keterbatasan

13.5 Penyimpanan Model dan Inference

Kode 13.1. Menyimpan keseluruhan pipeline model

```
import joblib

joblib.dump(pipe, "model_pipeline.joblib")
loaded = joblib.load("model_pipeline.joblib")
y_new = loaded.predict(X_new)
```

Simpan pipeline, bukan hanya estimator, agar scaling dan preprocessing tetap identik saat inference. Untuk penelitian jangka panjang, serialisasi Python dapat sensitif versi; simpan requirements, kode pembentuk model, dan data training reference selain file model.

BAB 14

PROYEK TERINTEGRASI: DARI PERTANYAAN RISET KE LAPORAN ILMIAH

14.1 Template Alur Riset

1. Rumuskan pertanyaan: sifat apa yang diprediksi dan untuk populasi molekul apa?
2. Tentukan sumber data, kriteria inklusi, satuan, dan provenance.
3. Standardisasi struktur dan audit kualitas sebelum menghitung fitur.
4. Tetapkan representasi baseline yang interpretable dan representasi alternatif.
5. Tentukan split sesuai penggunaan nyata: random, scaffold, group, time, atau external.
6. Bangun baseline sederhana, lalu model nonlinier bila diperlukan.
7. Evaluasi dengan metrik yang relevan dan analisis error per domain kimia.
8. Lakukan tuning hanya pada training/validation; kunci test set sampai akhir.
9. Definisikan applicability domain dan cara menangani prediksi out-of-domain.
10. Simpan pipeline, seed, versi paket, data snapshot, dan laporan hasil.

14.2 Struktur Laporan Metodologi

Tabel 14.1. Struktur minimum laporan machine learning untuk riset kimia

Bagian	Isi minimum
Data	asal, jumlah record, target, satuan, periode/release, kriteria inklusi
Preprocessing	missing, duplicate, structure standardization, agregasi assay
Featurization	descriptor/fingerprint dan semua parameter
Split	metode, rasio, seed, aturan grouping/scaffold
Model	algoritma dan hyperparameter
Validasi	cross-validation, metric, uncertainty
Hasil	test performance, residual/confusion matrix, error cases
Interpretasi	hubungan dengan kimia, bukan hanya angka
Limitasi	domain, bias data, endpoint, extrapolation
Reproducibility	kode, versi, data snapshot, model artefact

14.3 Deployment: Dari Model ke Aplikasi

Deployment bukan sekadar membuat antarmuka. Aplikasi harus memvalidasi input, menjalankan preprocessing yang sama, menolak struktur invalid, menampilkan domain model, dan membedakan prediksi dari data eksperimen. Streamlit dapat digunakan untuk prototipe, sedangkan API dapat digunakan untuk integrasi laboratorium. Untuk model kimia, tampilkan identifier, struktur yang telah distandardisasi, nilai prediksi, satuan, model version, serta warning out-of-domain.

Kode 14.1. Kerangka deployment Streamlit yang menjaga alur preprocessing

```
# kerangka sangat ringkas Streamlit
import streamlit as st
import joblib

model = joblib.load("model_pipeline.joblib")
st.title("Prediksi Sifat Molekul")
smiles = st.text_input("SMILES")
if st.button("Prediksi"):
    # 1) validasi & standardisasi SMILES
    # 2) hitung descriptor/fingerprint sesuai training
    # 3) cek applicability domain
    # 4) prediksi dan tampilkan satuan + warning
    st.write("Integrasikan fungsi preprocessing proyek Anda di sini.")
```

14.4 Penutup: Kebiasaan Riset yang Perlu Dipertahankan

Machine learning yang baik dalam kimia bukan kompetisi memilih model paling rumit. Nilai ilmiahnya terletak pada kesesuaian pertanyaan, kualitas data, transparansi preprocessing, validasi yang realistis, interpretasi yang tidak melampaui bukti, dan kemampuan peneliti lain untuk mereproduksi alur.

Mulailah dari baseline sederhana yang dapat dijelaskan. Jika model kompleks memberi peningkatan, ukur peningkatan tersebut pada test yang independen dan domain yang relevan. Jika tidak ada peningkatan, baseline yang lebih sederhana sering lebih mudah dipertahankan dan dijelaskan.

Akhirnya, gunakan prediksi sebagai alat untuk memprioritaskan eksperimen, bukan sebagai pengganti observasi. Siklus paling produktif adalah data → model → hipotesis/prioritas → eksperimen → data baru → model yang diperbarui. Dengan siklus tersebut, ML menjadi bagian dari metode ilmiah yang mempercepat pembelajaran, bukan sekadar mesin penghasil angka.

LAMPIRAN A

INSTALASI DAN KONFIGURASI

A.1 Instalasi dengan venv

Kode A.1. Environment berbasis venv

```
python -m venv .venv
# Windows
.venv\Scripts\activate
# Linux/macOS
source .venv/bin/activate

pip install --upgrade pip
pip install -r requirements.txt
```

A.2 Instalasi RDKit dengan Conda

Kode A.2. Environment conda yang direkomendasikan bila RDKit bermasalah di pip

```
conda create -n mlkimia python=3.12 -y
conda activate mlkimia
conda install -c conda-forge rdkit pandas numpy scikit-learn matplotlib requests
joblib -y
pip install chembl_webresource_client
```

A.3 Verifikasi

Jalankan `00_cek_environment.py`. Simpan output versi package di log proyek. Untuk publikasi, tuliskan versi utama pada metode dan simpan file `requirements/environment` agar environment dapat direkonstruksi.

LAMPIRAN B

INDEKS KODE PENDAMPING

Tabel B.1. Isi paket kode pendamping

File	Fungsi
requirements.txt	Dependensi atau dokumentasi
README.md	Dependensi atau dokumentasi
00_cek_environment.py	Memeriksa Python dan package utama
01_python_fundamental_kimia.py	Fundamental Python dengan data kimia
02_pubchem_dataset.py	Mengambil properti melalui PubChem PUG REST
03_esol_regression_descriptors.py	Regresi ESOL dengan descriptor
04_esol_morgan_random_forest.py	Regresi ESOL dengan Morgan fingerprint
05_bbbp_classification.py	Klasifikasi BBBP
06_kmeans_pca_lipophilicity.py	Clustering dan PCA chemical space
07_freesolv_regression.py	Regresi energi hidrasi FreeSolv
08_qm9_gap_regression.py	Baseline prediksi gap QM9
09_chembl_her2_qsar.py	Pengambilan data HER2/ERBB2 dari ChEMBL dan QSAR
10_gridsearch_pipeline_bbbp.py	Grid search KNN BBBP

LAMPIRAN C

TROUBLESHOOTING PRAKTIK

Tabel C.1. Masalah umum dan strategi penanganannya

Masalah	Penyebab umum	Tindakan
ModuleNotFoundError: rdkit	environment berbeda atau RDKit belum terpasang	aktifkan environment yang benar; gunakan conda-forge bila perlu
SMILES gagal diparse	struktur invalid / format tidak didukung	log record, jangan diam-diam mengganti struktur
HTTP error PubChem/ChEMBL	jaringan, rate limit, endpoint berubah	retry wajar, cache data, cek dokumentasi resmi
Metrik terlalu bagus	leakage/duplikasi/analog dekat	audit split, duplicate, target proxy, scaffold
Metrik sangat berubah antar-run	dataset kecil atau split tidak stabil	cross-validation, repeated split, laporkan variasi
Grid search lambat	ruang parameter terlalu besar	perkecil grid, gunakan RandomizedSearchCV, parallel n_jobs
PCA didominasi satu fitur	fitur belum distandardisasi	StandardScaler sebelum PCA
KNN buruk	skala fitur berbeda atau dimensi tinggi	scaling, feature selection/PCA, tuning k
Model tidak bisa dimuat	perbedaan versi library	simpan requirements dan rebuild bila perlu

LAMPIRAN D | 15 PROYEK RISET KIMIA MENGGUNAKAN MACHINE LEARNING DAN PYTHON

Lampiran ini dirancang sebagai jembatan dari pembelajaran machine learning menuju proyek mini riset yang dapat dilaksanakan mahasiswa S1 peserta mata kuliah Informatika Kimia dalam waktu dua pekan. Semua topik menggunakan kasus kimia nyata yang memiliki landasan jurnal ilmiah, dataset terbuka atau dapat ditelusuri, serta workflow yang realistis untuk laptop mahasiswa. Tujuannya bukan menghasilkan klaim publikasi secara otomatis, melainkan melatih desain riset, provenance data, pemodelan, evaluasi, interpretasi kimia, dan pelaporan yang dapat direproduksi.

Setiap proyek ditulis dengan format proposal dan laporan: Judul, Latar Belakang, Rumusan Masalah, Tujuan Riset, Telaah Pustaka, Metode Penelitian (Desain Penelitian, Alur Penelitian, Coding Machine Learning Python), Hasil Riset dan Pembahasan, Diskusi, dan Kesimpulan. Bagian hasil tidak mengarang angka. Mahasiswa wajib menjalankan skrip, menyimpan output aktual, lalu mengisi tabel hasil dengan metrik yang benar-benar diperoleh pada environment dan versi data yang digunakan.

D.1 Prinsip Pelaksanaan Proyek Dua Pekan

- Gunakan pertanyaan riset yang sempit dan terukur. Dalam 14 hari, fokus utama adalah menghasilkan satu dataset yang auditable, satu baseline yang benar, satu atau dua model pembanding, dan evaluasi yang dapat direproduksi.
- Gunakan data publik atau supporting information jurnal. Jangan memasukkan hasil eksperimen baru yang tidak sempat diverifikasi. Catat DOI, URL sumber, tanggal akses, versi file, dan aturan cleaning.
- Kunci random seed, simpan data mentah tanpa diedit, dan lakukan seluruh transformasi melalui kode. Output minimum adalah tabel metrik, satu visual diagnostik, satu tabel error analysis, serta file model atau pipeline bila relevan.
- Mahasiswa tidak diwajibkan mengalahkan performa paper sumber. Nilai akademik proyek terletak pada desain yang valid, reproduksibilitas, interpretasi kimia yang proporsional, dan kemampuan menjelaskan keterbatasan.
- Untuk dataset molekuler, random split dipakai sebagai baseline praktikum. Jika waktu memungkinkan, tambahkan scaffold/group split sebagai analisis sensitivitas. Jelaskan mengapa dua jenis split dapat memberikan estimasi generalisasi yang berbeda.

D.2 Instalasi Python dan Seluruh Library dalam Satu Perintah

Perintah berikut ditujukan untuk Windows PowerShell dengan winget. Seluruh instalasi Python 3.12 dan library yang dipakai oleh 15 proyek dijalankan sebagai satu baris perintah. Jalankan PowerShell dengan akses internet. Apabila organisasi kampus memblokir winget atau PyPI, gunakan komputer/lab yang diizinkan dan dokumentasikan perubahan environment.

Kode D.1. Satu perintah instalasi Python dan seluruh library proyek (Windows PowerShell)

```
winget install -e --id Python.Python.3.12 --accept-package-agreements --accept-source-agreements; $py="$env:LOCALAPPDATA\Programs\Python\Python312\python.exe"; if(-not (Test-Path $py)){ $py="py"}; & $py -m pip install --upgrade pip; & $py -m pip install numpy pandas scipy matplotlib scikit-learn jupyterlab requests joblib rdkit chembl_webresource_client ucimlrepo xgboost openpyxl pyarrow tqdm matminer pymatgen
```

Setelah perintah selesai, tutup dan buka kembali terminal bila launcher Python belum terbaca. Verifikasi environment dengan menjalankan `python --version` dan mengimpor library yang diperlukan. Perintah instalasi sengaja mencakup `matminer` dan `pymatgen` karena Proyek 15 menggunakan materials informatics, sedangkan `chembl_webresource_client` digunakan pada Proyek 12.

Alternatif Ubuntu/WSL: apabila praktikum dijalankan pada Ubuntu atau WSL, gunakan satu baris perintah berikut. Perintah ini memasang Python, membuat environment `.venv`, mengaktifkannya pada shell yang sama, memperbarui `pip`, dan memasang seluruh library yang diperlukan oleh kelima belas proyek.

```
sudo apt update && sudo apt install -y python3 python3-venv python3-pip build-essential && python3 -m venv .venv && . .venv/bin/activate && python -m pip install --upgrade pip && python -m pip install numpy pandas scipy matplotlib scikit-learn jupyterlab requests joblib rdkit chembl_webresource_client ucimlrepo xgboost openpyxl pyarrow tqdm matminer pymatgen
```

Pilih salah satu perintah instalasi sesuai sistem operasi; masing-masing sudah merupakan satu perintah utuh. Sesudah instalasi, simpan hasil `python --version` dan `pip freeze` pada folder environment sebagai bukti reproduksibilitas.

D.3 Standar Folder, Output, dan Pelaporan

Kode D.2. Struktur folder minimum setiap proyek

```
project_XX/
  data_raw/
  data_processed/
  src/
  figures/
  models/
  reports/
  environment/
  README.md
```

Tabel D.1. Artefak wajib yang dikumpulkan pada akhir proyek

Artefak	Isi minimum	Tujuan audit
data_raw	salinan data asli atau catatan query/API	menjaga provenance dan mencegah edit diam-diam
data_processed	dataset hasil cleaning dengan script pembentuk	mereproduksi transformasi
src	script Python final	memastikan analisis dapat dijalankan ulang
figures	grafik target, predicted-vs-actual, confusion matrix/residual	mendukung interpretasi
models	pipeline/joblib bila relevan	menghindari preprocessing berbeda saat inferensi
reports	proposal, laporan, tabel hasil	menjaga jejak pelaporan
environment	pip freeze atau daftar versi	reproducibility environment

D.4 PROYEK 1: PREDIKSI KELARUTAN AIR SENYAWA ORGANIK MENGGUNAKAN ESOL

Tabel D.2. Ringkasan rancangan Proyek 1

Komponen	Rancangan proyek
Bidang	kimia fisik, kimia obat, dan QSAR
Basis jurnal	Delaney, J. S. (2004). ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. Journal of Chemical Information and Computer Sciences, 44(3), 1000-1005. DOI: 10.1021/ci034243x.
Dataset	Dataset ESOL/Delaney yang didistribusikan sebagai delaney-processed.csv melalui koleksi MoleculeNet/DeepChem. Dataset pembelajaran berisi sekitar seribu struktur dengan target measured log solubility in mols per litre.
Target	measured log solubility in mols per litre (logS)
Fitur	descriptor RDKit: MW, MolLogP, TPSA, HBD, HBA, rotatable bonds, ring count, FractionCSP3; dibandingkan dengan Morgan fingerprint
Model	Linear Regression, Ridge, Random Forest Regressor
Evaluasi	MAE, RMSE, R2, residual plot, dan analisis error pada senyawa ber-logS ekstrem
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Kelarutan Air Senyawa Organik Menggunakan ESOL

Latar Belakang

Artikel ESOL membangun model kelarutan dari data terukur dan menunjukkan bahwa sifat seperti logP, massa molekul, aromatisitas, dan ikatan rotatable berhubungan dengan kelarutan. Studi ini merupakan kasus klasik yang sangat sesuai untuk memperkenalkan regresi QSAR pada mahasiswa S1.

Proyek ini ditempatkan pada konteks kimia fisik, kimia obat, dan QSAR. Target yang dianalisis adalah measured log solubility in mols per litre (logS). Pertanyaan

inti penelitian adalah: Seberapa baik descriptor molekul 2D sederhana memprediksi logS, dan apakah Random Forest memberi peningkatan yang konsisten dibanding model linear? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Interpretasi difokuskan pada hubungan ukuran molekul, lipofilisitas, polaritas, fleksibilitas, dan aromatisitas terhadap kemampuan molekul berinteraksi dengan air. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa baik descriptor molekul 2D sederhana memprediksi logS, dan apakah Random Forest memberi peningkatan yang konsisten dibanding model linear?
2. Bagaimana kualitas, distribusi, dan potensi bias pada Dataset ESOL/Delaney yang didistribusikan sebagai delaney-processed.csv melalui koleksi MoleculeNet/DeepChem. Dataset pembelajaran berisi sekitar seribu struktur dengan target measured log solubility in mols per litre.?
3. Bagaimana performa Linear Regression, Ridge, Random Forest Regressor jika dinilai menggunakan MAE, RMSE, R2, residual plot, dan analisis error pada senyawa ber-logS ekstrem?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Linear Regression, Ridge, Random Forest Regressor.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2, residual plot, dan analisis error pada senyawa ber-logS ekstrem.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Delaney, J. S. (2004). ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *Journal of Chemical Information and Computer Sciences*, 44(3), 1000-1005. DOI: 10.1021/ci034243x.

Artikel ESOL membangun model kelarutan dari data terukur dan menunjukkan bahwa sifat seperti logP, massa molekul, aromatisitas, dan ikatan rotatable berhubungan dengan kelarutan. Studi ini merupakan kasus klasik yang sangat sesuai untuk memperkenalkan regresi QSAR pada mahasiswa S1. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah Dataset ESOL/Delaney yang didistribusikan sebagai `delaney-processed.csv` melalui koleksi MoleculeNet/DeepChem. Dataset pembelajaran berisi sekitar seribu struktur dengan target measured log solubility in mols per litre. Fitur yang dipakai adalah descriptor RDKit: MW, MolLogP, TPSA, HBD, HBA, rotatable bonds, ring count, FractionCSP3; dibandingkan dengan Morgan fingerprint. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Nilai logS berasal dari kompilasi eksperimen dengan kondisi yang tidak selalu identik. Model 2D tidak secara eksplisit memuat bentuk kristal, pH, suhu, ionisasi, atau polymorphism sehingga klaim harus dibatasi pada prediksi statistik dalam domain dataset. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. Dataset ESOL/Delaney yang didistribusikan sebagai `delaney-processed.csv` melalui koleksi MoleculeNet/DeepChem. Dataset pembelajaran berisi sekitar seribu struktur dengan target measured log solubility in mols per litre. Variabel target: measured log solubility in mols per litre (logS). Variabel prediktor: descriptor RDKit: MW, MolLogP, TPSA, HBD, HBA,

rotatable bonds, ring count, FractionCSP3; dibandingkan dengan Morgan fingerprint.

Quality Control Data. Periksa SMILES invalid, nilai target hilang, duplikasi canonical SMILES, distribusi logS, dan outlier. Split data sebelum fitting scaler atau pemilihan fitur.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor RDKit: MW, MolLogP, TPSA, HBD, HBA, rotatable bonds, ring count, FractionCSP3; dibandingkan dengan Morgan fingerprint.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Linear Regression, Ridge, Random Forest Regressor.
8. Evaluasi menggunakan MAE, RMSE, R2, residual plot, dan analisis error pada senyawa ber-logS ekstrem.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.17. Jadwal kerja 14 hari Proyek 1

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: Dataset ESOL/Delaney yang didistribusikan sebagai delaney-processed.csv melalui koleksi MoleculeNet/DeepChem. Dataset pembelajaran berisi sekitar seribu struktur dengan target measured log solubility in mols per litre.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Linear Regression, Ridge, Random Forest Regressor.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.

Hari	Kegiatan dan output
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Interpretasi difokuskan pada hubungan ukuran molekul, lipofilisitas, polaritas, fleksibilitas, dan aromatisitas terhadap kemampuan molekul berinteraksi dengan air.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.3. Script Python lengkap Proyek 1

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.linear_model import LinearRegression, Ridge
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = (
    "https://deepchemdata.s3-us-west-1.amazonaws.com/"
    "datasets/delaney-processed.csv"
)
SEED = 42

df = pd.read_csv(URL)
smiles_col = "smiles"
target_col = "measured log solubility in mols per litre"

def desc(smi):
    mol = Chem.MolFromSmiles(smi)
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol), Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol), Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol), Descriptors.FractionCSP3(mol),
    ]

rows, y = [], []
for smi, target in zip(df[smiles_col], df[target_col]):
    d = desc(smi)
    if d is not None and pd.notna(target):
        rows.append(d)
        y.append(float(target))

X = np.asarray(rows, dtype=float)
y = np.asarray(y, dtype=float)
```

```

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.20, random_state=SEED
)

models = {
    "Linear": Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("scale", StandardScaler()),
        ("model", LinearRegression()),
    ]),
    "Ridge": Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("scale", StandardScaler()),
        ("model", Ridge(alpha=1.0)),
    ]),
    "RandomForest": RandomForestRegressor(
        n_estimators=500, random_state=SEED, n_jobs=-1,
        min_samples_leaf=2
    ),
}

result = []
best_pred = None
for name, model in models.items():
    model.fit(X_train, y_train)
    pred = model.predict(X_test)
    mae = mean_absolute_error(y_test, pred)
    rmse = mean_squared_error(y_test, pred) ** 0.5
    r2 = r2_score(y_test, pred)
    result.append([name, mae, rmse, r2])
    if name == "RandomForest":
        best_pred = pred

print(pd.DataFrame(result, columns=["model", "MAE", "RMSE", "R2"]))
plt.scatter(y_test, best_pred, alpha=0.7)
lo = min(y_test.min(), best_pred.min())
hi = max(y_test.max(), best_pred.max())
plt.plot([lo, hi], [lo, hi], "--")
plt.xlabel("logS eksperimen")
plt.ylabel("logS prediksi")
plt.title("ESOL: eksperimen vs prediksi Random Forest")
plt.tight_layout()
plt.savefig("proyek01_esol_prediksi.png", dpi=200)
plt.show()

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.32. Template tabel hasil aktual Proyek 1

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R², residual plot, dan analisis error pada senyawa ber-logS ekstrem. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Interpretasi difokuskan pada hubungan ukuran molekul, lipofilisitas, polaritas, fleksibilitas, dan aromatisitas terhadap kemampuan molekul berinteraksi dengan air. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Nilai logS berasal dari kompilasi eksperimen dengan kondisi yang tidak selalu identik. Model 2D tidak secara eksplisit memuat bentuk kristal, pH, suhu, ionisasi, atau polymorphism sehingga klaim harus dibatasi pada prediksi statistik dalam domain dataset.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar

ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Nilai logS berasal dari kompilasi eksperimen dengan kondisi yang tidak selalu identik. Model 2D tidak secara eksplisit memuat bentuk kristal, pH, suhu, ionisasi, atau polymorphism sehingga klaim harus dibatasi pada prediksi statistik dalam domain dataset. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Delaney (2004), DOI 10.1021/ci034243x
- Wu et al. (2018), DOI 10.1039/C7SC02664A

D.5 PROYEK 2: PREDIKSI KELARUTAN AIR SKALA LEBIH BESAR MENGGUNAKAN AQSOLDB

Tabel D.3. Ringkasan rancangan Proyek 2

Komponen	Rancangan proyek
Bidang	kurasi data kimia, QSAR, dan chemical data quality
Basis jurnal	Sorkun, M. C., Khetan, A., & Er, S. (2019). AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. <i>Scientific Data</i> , 6, 143. DOI: 10.1038/s41597-019-0151-1.
Dataset	File resmi data_curated.csv dari repositori AqSolDB. Kolom utama yang digunakan adalah SMILES dan Solubility; bila nama kolom berbeda karena pembaruan, skrip melakukan pencarian nama kolom secara case-insensitive.
Target	Solubility/logS yang telah dikurasi oleh penyusun AqSolDB
Fitur	descriptor RDKit 2D dan, sebagai pengembangan, fingerprint Morgan
Model	Ridge, Random Forest Regressor, Gradient Boosting Regressor
Evaluasi	MAE, RMSE, R2, distribusi residual, dan perbandingan performa pada kuartil kelarutan
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Kelarutan Air Skala Lebih Besar Menggunakan AqSolDB

Latar Belakang

AqSolDB menghimpun dan mengkurasi nilai kelarutan dari berbagai sumber menjadi basis data terbuka berukuran hampir sepuluh ribu senyawa. Kasus ini tepat untuk mengajarkan bahwa performa machine learning sangat dipengaruhi provenance, duplikasi, harmonisasi satuan, dan konsistensi struktur.

Proyek ini ditempatkan pada konteks kurasi data kimia, QSAR, dan chemical data quality. Target yang dianalisis adalah Solubility/logS yang telah dikurasi oleh penyusun AqSolDB. Pertanyaan inti penelitian adalah: Apakah peningkatan ukuran dan keragaman dataset AqSolDB mengubah hubungan antara descriptor 2D dan kelarutan dibanding benchmark ESOL yang lebih kecil? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Analisis diarahkan pada heterogenitas senyawa, rentang logS, lipofilisitas, ukuran molekul, jumlah heteroatom, serta kemungkinan pengaruh garam atau struktur multikomponen. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah peningkatan ukuran dan keragaman dataset AqSolDB mengubah hubungan antara descriptor 2D dan kelarutan dibanding benchmark ESOL yang lebih kecil?
2. Bagaimana kualitas, distribusi, dan potensi bias pada File resmi data_curated.csv dari repositori AqSolDB. Kolom utama yang digunakan adalah SMILES dan Solubility; bila nama kolom berbeda karena pembaruan, skrip melakukan pencarian nama kolom secara case-insensitive.?
3. Bagaimana performa Ridge, Random Forest Regressor, Gradient Boosting Regressor jika dinilai menggunakan MAE, RMSE, R2, distribusi residual, dan perbandingan performa pada kuartil kelarutan?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Ridge, Random Forest Regressor, Gradient Boosting Regressor.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2, distribusi residual, dan perbandingan performa pada kuartil kelarutan.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Sorkun, M. C., Khetan, A., & Er, S. (2019). AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Scientific Data*, 6, 143. DOI: 10.1038/s41597-019-0151-1.

AqSolDB menghimpun dan mengkurasi nilai kelarutan dari berbagai sumber menjadi basis data terbuka berukuran hampir sepuluh ribu senyawa. Kasus ini tepat untuk mengajarkan bahwa performa machine learning sangat dipengaruhi provenance, duplikasi, harmonisasi satuan, dan konsistensi struktur. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah File resmi data_curated.csv dari repositori AqSolDB. Kolom utama yang digunakan adalah SMILES dan Solubility; bila nama kolom berbeda karena pembaruan, skrip melakukan pencarian nama kolom secara case-insensitive. Fitur yang dipakai adalah descriptor RDKit 2D dan, sebagai pengembangan, fingerprint Morgan. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. AqSolDB merupakan kompilasi multi-sumber; walaupun telah dikurasi, kondisi eksperimen asal tetap beragam. Mahasiswa wajib membedakan kualitas prediksi statistik dari klaim mekanistik dan tidak menafsirkan korelasi descriptor sebagai sebab tunggal kelarutan. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. File resmi data_curated.csv dari repositori AqSolDB. Kolom utama yang digunakan adalah SMILES dan Solubility; bila nama kolom berbeda karena pembaruan, skrip melakukan pencarian nama kolom secara case-insensitive. Variabel target: Solubility/logS yang telah dikurasi oleh penyusun AqSolDB. Variabel prediktor: descriptor RDKit 2D dan, sebagai pengembangan, fingerprint Morgan.

Quality Control Data. Audit provenance, canonical SMILES, multiple fragments, missing values, duplicate InChIKey, dan reliability label bila tersedia. Laporkan jumlah baris sebelum dan sesudah cleaning.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor RDKit 2D dan, sebagai pengembangan, fingerprint Morgan.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Ridge, Random Forest Regressor, Gradient Boosting Regressor.
8. Evaluasi menggunakan MAE, RMSE, R2, distribusi residual, dan perbandingan performa pada kuartil kelarutan.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.18. Jadwal kerja 14 hari Proyek 2

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: File resmi data_curated.csv dari repositori AqSolDB. Kolom utama yang digunakan adalah SMILES dan Solubility; bila nama kolom berbeda karena pembaruan, skrip melakukan pencarian nama kolom secara case-insensitive.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Ridge, Random Forest Regressor, Gradient Boosting Regressor.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Analisis diarahkan pada heterogenitas senyawa, rentang logS, lipofilisitas, ukuran molekul, jumlah heteroatom, serta kemungkinan pengaruh garam atau struktur multikomponen.

Hari	Kegiatan dan output
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.4. Script Python lengkap Proyek 2

```
import numpy as np
import pandas as pd
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.linear_model import Ridge
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = (
    "https://raw.githubusercontent.com/mcsorkun/AqSolDB/"
    "master/results/data_curated.csv"
)
SEED = 42

df = pd.read_csv(URL, low_memory=False)
cols = {c.lower(): c for c in df.columns}
smiles_col = cols.get("smiles")
target_col = cols.get("solubility")
if smiles_col is None or target_col is None:
    raise ValueError(f"Kolom tidak ditemukan. Kolom tersedia: {list(df.columns)}")

def desc(smi):
    mol = Chem.MolFromSmiles(str(smi))
    if mol is None:
        return None
    return [
        Descriptors.MolWt(mol), Descriptors.MolLogP(mol),
        Descriptors.TPSA(mol), Descriptors.NumHDonors(mol),
        Descriptors.NumHAcceptors(mol), Descriptors.NumRotatableBonds(mol),
        Descriptors.RingCount(mol), Descriptors.FractionCSP3(mol),
    ]

records = []
for smi, target in zip(df[smiles_col], pd.to_numeric(df[target_col],
errors="coerce")):
    d = desc(smi)
    if d is not None and pd.notna(target):
        records.append(d + [float(target)])

data = np.asarray(records, dtype=float)
X, y = data[:, :-1], data[:, -1]
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.20, random_state=SEED
```

```

)

models = {
    "Ridge": Pipeline([("scale", StandardScaler()), ("model",
Ridge(alpha=2.0))]),
    "RandomForest": RandomForestRegressor(
        n_estimators=400, random_state=SEED, n_jobs=-1, min_samples_leaf=2
    ),
    "GradientBoosting": GradientBoostingRegressor(random_state=SEED),
}

for name, model in models.items():
    model.fit(X_train, y_train)
    pred = model.predict(X_test)
    print(
        name,
        "MAE=", round(mean_absolute_error(y_test, pred), 4),
        "RMSE=", round(mean_squared_error(y_test, pred) ** 0.5, 4),
        "R2=", round(r2_score(y_test, pred), 4),
    )

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.33. Template tabel hasil aktual Proyek 2

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R2, distribusi residual, dan perbandingan performa pada kuartil kelarutan. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Analisis diarahkan pada heterogenitas senyawa, rentang logS, lipofilisitas, ukuran molekul, jumlah heteroatom, serta kemungkinan pengaruh garam atau struktur multikomponen. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: AqSolDB merupakan kompilasi multi-sumber; walaupun telah dikurasi, kondisi eksperimen asal tetap beragam. Mahasiswa wajib membedakan kualitas prediksi statistik dari klaim mekanistik dan tidak menafsirkan korelasi descriptor sebagai sebab tunggal kelarutan.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: AqSolDB merupakan kompilasi multi-sumber; walaupun telah dikurasi, kondisi eksperimen asal tetap beragam. Mahasiswa wajib membedakan kualitas prediksi statistik dari klaim mekanistik dan tidak menafsirkan korelasi descriptor sebagai sebab tunggal kelarutan. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Sorkun, Khetan, & Er (2019), DOI 10.1038/s41597-019-0151-1

D.6 PROYEK 3: PREDIKSI ENERGI BEBAS HIDRASI MENGGUNAKAN FREESOLV

Tabel D.4. Ringkasan rancangan Proyek 3

Komponen	Rancangan proyek
Bidang	kimia fisik, termodinamika molekuler, dan surrogate modeling
Basis jurnal	Mobley, D. L., & Guthrie, J. P. (2014). FreeSolv: a database of experimental and calculated hydration free energies, with input files. <i>Journal of Computer-Aided Molecular Design</i> , 28, 711-720. DOI: 10.1007/s10822-014-9747-x.
Dataset	SAMPL/FreeSolv CSV dari DeepChem dengan SMILES dan target expt. Dataset relatif kecil sehingga seluruh analisis dapat dijalankan di laptop tanpa GPU.
Target	energi bebas hidrasi eksperimental, umumnya dalam kcal/mol pada file benchmark
Fitur	descriptor molekul 2D yang menggambarkan ukuran, polaritas, donor/akseptor, lipofilisitas, dan fleksibilitas
Model	Ridge, Extra Trees Regressor, Random Forest Regressor
Evaluasi	MAE dan RMSE dalam satuan energi, R ² , residual terhadap target, serta error pada molekul polar vs nonpolar
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Energi Bebas Hidrasi Menggunakan FreeSolv

Latar Belakang

FreeSolv menyediakan energi bebas hidrasi eksperimental untuk molekul kecil dan banyak digunakan sebagai benchmark metode komputasi. Untuk kelas S1, target ini memberikan jembatan langsung antara konsep termodinamika dan regresi berbasis struktur.

Proyek ini ditempatkan pada konteks kimia fisik, termodinamika molekuler, dan surrogate modeling. Target yang dianalisis adalah energi bebas hidrasi eksperimental, umumnya dalam kcal/mol pada file benchmark. Pertanyaan inti penelitian adalah: Sejauh mana descriptor 2D sederhana dapat menjadi surrogate cepat untuk energi bebas hidrasi eksperimental pada molekul kecil? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Pembahasan menghubungkan tanda dan magnitudo energi hidrasi dengan polaritas, kemampuan ikatan hidrogen, hidrofobisitas, dan ukuran permukaan molekul. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat

membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Sejauh mana descriptor 2D sederhana dapat menjadi surrogate cepat untuk energi bebas hidrasi eksperimental pada molekul kecil?
2. Bagaimana kualitas, distribusi, dan potensi bias pada SAMPL/FreeSolv CSV dari DeepChem dengan SMILES dan target expt. Dataset relatif kecil sehingga seluruh analisis dapat dijalankan di laptop tanpa GPU.?
3. Bagaimana performa Ridge, Extra Trees Regressor, Random Forest Regressor jika dinilai menggunakan MAE dan RMSE dalam satuan energi, R², residual terhadap target, serta error pada molekul polar vs nonpolar?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Ridge, Extra Trees Regressor, Random Forest Regressor.
3. Mengevaluasi performa menggunakan MAE dan RMSE dalam satuan energi, R², residual terhadap target, serta error pada molekul polar vs nonpolar.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Mobley, D. L., & Guthrie, J. P. (2014). FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design*, 28, 711-720. DOI: 10.1007/s10822-014-9747-x.

FreeSolv menyediakan energi bebas hidrasi eksperimental untuk molekul kecil dan banyak digunakan sebagai benchmark metode komputasi. Untuk kelas S1, target ini memberikan jembatan langsung antara konsep termodinamika dan regresi berbasis struktur. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah SAMPL/FreeSolv CSV dari DeepChem dengan SMILES dan target expt. Dataset relatif kecil sehingga seluruh analisis dapat dijalankan di laptop tanpa GPU. Fitur yang dipakai adalah descriptor molekul 2D yang menggambarkan ukuran, polaritas, donor/akseptor, lipofilisitas, dan fleksibilitas. Representasi dipilih karena dapat dijalankan dengan library open-

source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. FreeSolv berfokus pada molekul kecil dan tidak mewakili ion, makromolekul, kompleks logam, atau semua kelas fungsional. Model tidak menggantikan simulasi free-energy atau eksperimen. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. SAMPL/FreeSolv CSV dari DeepChem dengan SMILES dan target expt. Dataset relatif kecil sehingga seluruh analisis dapat dijalankan di laptop tanpa GPU. Variabel target: energi bebas hidrasi eksperimental, umumnya dalam kcal/mol pada file benchmark. Variabel prediktor: descriptor molekul 2D yang menggambarkan ukuran, polaritas, donor/akseptor, lipofilisitas, dan fleksibilitas.

Quality Control Data. Periksa unit target, validitas SMILES, duplikasi, dan sebaran nilai energi. Karena dataset kecil, gunakan cross-validation tambahan dan laporkan variasi fold.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor molekul 2D yang menggambarkan ukuran, polaritas, donor/akseptor, lipofilisitas, dan fleksibilitas.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Ridge, Extra Trees Regressor, Random Forest Regressor.
8. Evaluasi menggunakan MAE dan RMSE dalam satuan energi, R2, residual terhadap target, serta error pada molekul polar vs nonpolar.

9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.19. Jadwal kerja 14 hari Proyek 3

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: SAMPL/FreeSolv CSV dari DeepChem dengan SMILES dan target expt. Dataset relatif kecil sehingga seluruh analisis dapat dijalankan di laptop tanpa GPU.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standarisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Ridge, Extra Trees Regressor, Random Forest Regressor.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Pembahasan menghubungkan tanda dan magnitudo energi hidrasi dengan polaritas, kemampuan ikatan hidrogen, hidrofobisitas, dan ukuran permukaan molekul.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.5. Script Python lengkap Proyek 3

```
import numpy as np
import pandas as pd
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split, KFold, cross_val_score
from sklearn.ensemble import ExtraTreesRegressor, RandomForestRegressor
from sklearn.linear_model import Ridge
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/SAMPL.csv"
SEED = 42
```

```

df = pd.read_csv(URL)

def featurize(smi):
    m = Chem.MolFromSmiles(smi)
    if m is None:
        return None
    return [
        Descriptors.MolWt(m), Descriptors.MolLogP(m), Descriptors.TPSA(m),
        Descriptors.NumHDonors(m), Descriptors.NumHAcceptors(m),
        Descriptors.NumRotatableBonds(m), Descriptors.RingCount(m),
        Descriptors.FractionCSP3(m)
    ]

rows = []
for smi, y in zip(df["smiles"], pd.to_numeric(df["expt"], errors="coerce")):
    x = featurize(smi)
    if x is not None and pd.notna(y):
        rows.append(x + [float(y)])
arr = np.asarray(rows, dtype=float)
X, y = arr[:, :-1], arr[:, -1]
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)

models = {
    "Ridge": Pipeline([("scale", StandardScaler()), ("model",
Ridge(alpha=1.0))]),
    "ExtraTrees": ExtraTreesRegressor(n_estimators=500, random_state=SEED,
n_jobs=-1),
    "RandomForest": RandomForestRegressor(n_estimators=500, random_state=SEED,
n_jobs=-1),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(name,
          "MAE", mean_absolute_error(yte, pred),
          "RMSE", mean_squared_error(yte, pred) ** 0.5,
          "R2", r2_score(yte, pred))

cv = KFold(n_splits=5, shuffle=True, random_state=SEED)
rf = ExtraTreesRegressor(n_estimators=400, random_state=SEED, n_jobs=-1)
score = -cross_val_score(rf, X, y, cv=cv, scoring="neg_mean_absolute_error")
print("CV MAE mean/std:", score.mean(), score.std())

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.34. Template tabel hasil aktual Proyek 3

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE dan RMSE dalam satuan energi, R², residual terhadap target, serta error pada molekul polar vs nonpolar. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Pembahasan menghubungkan tanda dan magnitudo energi hidrasi dengan polaritas, kemampuan ikatan hidrogen, hidrofobisitas, dan ukuran permukaan molekul. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: FreeSolv berfokus pada molekul kecil dan tidak mewakili ion, makromolekul, kompleks logam, atau semua kelas fungsional. Model tidak menggantikan simulasi free-energy atau eksperimen.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar

ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: FreeSolv berfokus pada molekul kecil dan tidak mewakili ion, makromolekul, kompleks logam, atau semua kelas fungsional. Model tidak menggantikan simulasi free-energy atau eksperimen. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Mobley & Guthrie (2014), DOI 10.1007/s10822-014-9747-x

D.7 PROYEK 4: PREDIKSI LIPOFILISITAS LOGD PADA pH 7,4

Tabel D.5. Ringkasan rancangan Proyek 4

Komponen	Rancangan proyek
Bidang	kimia obat, sifat fisikokimia, dan QSAR
Basis jurnal	Wu, Z., Ramsundar, B., Feinberg, E. N., et al. (2018). MoleculeNet: a benchmark for molecular machine learning. <i>Chemical Science</i> , 9, 513-530. DOI: 10.1039/C7SC02664A. Dataset lipophilicity di dalam benchmark dikurasi dari ChEMBL dan memuat sekitar 4200 pengukuran logD pada pH 7,4.
Dataset	Lipophilicity.csv dari DeepChem/MoleculeNet dengan kolom smiles dan exp.
Target	experimental octanol/water distribution coefficient logD pada pH 7,4
Fitur	RDKit descriptor dan Morgan fingerprint sebagai dua representasi alternatif
Model	Ridge, Random Forest, HistGradientBoosting atau Extra Trees
Evaluasi	MAE, RMSE, R2, residual terhadap logD, serta perbandingan random split dengan scaffold-aware discussion
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Lipofilisitas logD pada pH 7,4

Latar Belakang

Lipofilisitas memengaruhi partisi fase, permeabilitas membran, kelarutan, dan distribusi obat. Dataset MoleculeNet memungkinkan mahasiswa menguji model regresi pada target eksperimen yang secara kimia mudah diinterpretasikan.

Proyek ini ditempatkan pada konteks kimia obat, sifat fisikokimia, dan QSAR. Target yang dianalisis adalah experimental octanol/water distribution coefficient logD pada pH 7,4. Pertanyaan inti penelitian adalah: Apakah descriptor interpretable cukup untuk memodelkan logD, dan fitur mana yang paling konsisten berkaitan dengan lipofilisitas? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa

mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Bahas peran logP terhitung, TPSA, jumlah heteroatom, donor/akseptor, ukuran molekul, serta ionisasi yang tidak tertangkap penuh oleh descriptor netral 2D. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah descriptor interpretable cukup untuk memodelkan logD, dan fitur mana yang paling konsisten berkaitan dengan lipofilisitas?
2. Bagaimana kualitas, distribusi, dan potensi bias pada Lipophilicity.csv dari DeepChem/MoleculeNet dengan kolom smiles dan exp.?
3. Bagaimana performa Ridge, Random Forest, HistGradientBoosting atau Extra Trees jika dinilai menggunakan MAE, RMSE, R2, residual terhadap logD, serta perbandingan random split dengan scaffold-aware discussion?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Ridge, Random Forest, HistGradientBoosting atau Extra Trees.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2, residual terhadap logD, serta perbandingan random split dengan scaffold-aware discussion.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Wu, Z., Ramsundar, B., Feinberg, E. N., et al. (2018). MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9, 513-530. DOI: 10.1039/C7SC02664A. Dataset lipophilicity di dalam benchmark dikurasi dari ChEMBL dan memuat sekitar 4200 pengukuran logD pada pH 7,4.

Lipofilisitas memengaruhi partisi fase, permeabilitas membran, kelarutan, dan distribusi obat. Dataset MoleculeNet memungkinkan mahasiswa menguji model regresi pada target eksperimen yang secara kimia mudah diinterpretasikan. Dalam

proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah Lipophilicity.csv dari DeepChem/MoleculeNet dengan kolom smiles dan exp. Fitur yang dipakai adalah RDKit descriptor dan Morgan fingerprint sebagai dua representasi alternatif. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. logD bergantung pH dan keadaan ionisasi. SMILES yang tidak mencerminkan protomer dominan dapat membatasi interpretasi. Random split berpotensi optimistik untuk analog dekat. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. Lipophilicity.csv dari DeepChem/MoleculeNet dengan kolom smiles dan exp. Variabel target: experimental octanol/water distribution coefficient logD pada pH 7,4. Variabel prediktor: RDKit descriptor dan Morgan fingerprint sebagai dua representasi alternatif.

Quality Control Data. Pastikan target numeric, SMILES valid, duplikasi terdeteksi, dan rentang logD diperiksa. Catat bahwa data berasal dari ChEMBL dengan curation benchmark.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.

5. Bangun fitur: RDKit descriptor dan Morgan fingerprint sebagai dua representasi alternatif.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Ridge, Random Forest, HistGradientBoosting atau Extra Trees.
8. Evaluasi menggunakan MAE, RMSE, R2, residual terhadap logD, serta perbandingan random split dengan scaffold-aware discussion.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.20. Jadwal kerja 14 hari Proyek 4

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: Lipophilicity.csv dari DeepChem/MoleculeNet dengan kolom smiles dan exp.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Ridge, Random Forest, HistGradientBoosting atau Extra Trees.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Bahas peran logP terhitung, TPSA, jumlah heteroatom, donor/akseptor, ukuran molekul, serta ionisasi yang tidak tertangkap penuh oleh descriptor netral 2D.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.6. Script Python lengkap Proyek 4

```
import numpy as np
import pandas as pd
from rdkit import Chem
```

```

from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import ExtraTreesRegressor, RandomForestRegressor
from sklearn.linear_model import Ridge
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/Lipophilicity.csv"
SEED = 42

df = pd.read_csv(URL)

def desc(s):
    m = Chem.MolFromSmiles(s)
    if m is None:
        return None
    return [
        Descriptors.MolWt(m), Descriptors.MolLogP(m), Descriptors.TPSA(m),
        Descriptors.NumHDonors(m), Descriptors.NumHAcceptors(m),
        Descriptors.NumRotatableBonds(m), Descriptors.RingCount(m),
        Descriptors.FractionCSP3(m)
    ]

rows = []
for smi, val in zip(df["smiles"], pd.to_numeric(df["exp"], errors="coerce")):
    x = desc(smi)
    if x is not None and pd.notna(val):
        rows.append(x + [float(val)])
a = np.asarray(rows, dtype=float)
X, y = a[:, :-1], a[:, -1]
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)

models = {
    "Ridge": Pipeline([("scale", StandardScaler()), ("model",
Ridge(alpha=1.0))]),
    "RandomForest": RandomForestRegressor(n_estimators=400, random_state=SEED,
n_jobs=-1),
    "ExtraTrees": ExtraTreesRegressor(n_estimators=400, random_state=SEED,
n_jobs=-1),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(name,
          "MAE", round(mean_absolute_error(yte, pred), 4),
          "RMSE", round(mean_squared_error(yte, pred) ** 0.5, 4),
          "R2", round(r2_score(yte, pred), 4))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.35. Template tabel hasil aktual Proyek 4

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R2, residual terhadap logD, serta perbandingan random split dengan scaffold-aware discussion. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Bahas peran logP terhitung, TPSA, jumlah heteroatom, donor/akseptor, ukuran molekul, serta ionisasi yang tidak tertangkap penuh oleh descriptor netral 2D. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: logD bergantung pH dan keadaan ionisasi. SMILES yang tidak mencerminkan protomer dominan dapat membatasi interpretasi. Random split berpotensi optimistik untuk analog dekat.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: logD bergantung pH dan keadaan ionisasi. SMILES yang tidak mencerminkan protomer dominan dapat membatasi interpretasi. Random split berpotensi optimistik untuk analog dekat. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Wu et al. (2018), DOI 10.1039/C7SC02664A
- Gaulton et al. (2017), DOI 10.1093/nar/gkw1074

D.8 PROYEK 5: KLASIFIKASI PENETRASI BLOOD-BRAIN BARRIER (BBBP)

Tabel D.6. Ringkasan rancangan Proyek 5

Komponen	Rancangan proyek
Bidang	kimia obat, ADME, dan klasifikasi molekuler
Basis jurnal	Martins, I. F., Teixeira, A. L., Pinheiro, L., & Falcao, A. O. (2012). A Bayesian Approach to in Silico Blood-Brain Barrier Penetration Modeling. <i>Journal of Chemical Information and Modeling</i> , 52(6), 1686-1697. DOI: 10.1021/ci300124c.
Dataset	BBBP.csv dari DeepChem/MoleculeNet, berisi SMILES dan label p_np untuk lebih dari dua ribu senyawa.
Target	label permeabel/tidak permeabel terhadap blood-brain barrier
Fitur	descriptor 2D dan Morgan fingerprint; kelas dibandingkan dengan logistic regression dan random forest
Model	Logistic Regression dengan class_weight, Random Forest Classifier, KNN sebagai baseline tambahan
Evaluasi	balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, dan pembahasan threshold
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Klasifikasi Penetrasi Blood-Brain Barrier (BBBP)

Latar Belakang

Penetrasi blood-brain barrier merupakan endpoint penting dalam desain obat sistem saraf pusat. Studi Martins dan kolega menyediakan basis ilmiah dataset BBBP yang kemudian menjadi benchmark MoleculeNet.

Proyek ini ditempatkan pada konteks kimia obat, ADME, dan klasifikasi molekuler. Target yang dianalisis adalah label permeabel/tidak permeabel terhadap blood-brain barrier. Pertanyaan inti penelitian adalah: Seberapa baik fitur struktur 2D memisahkan molekul BBB+ dan BBB-, serta bagaimana perubahan threshold memengaruhi false positive dan false negative? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Diskusi mengaitkan permeabilitas dengan ukuran, lipofilisitas, TPSA, donor ikatan hidrogen, muatan, dan fleksibilitas sambil menegaskan bahwa hubungan tersebut bersifat multivariat. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa baik fitur struktur 2D memisahkan molekul BBB+ dan BBB-, serta bagaimana perubahan threshold memengaruhi false positive dan false negative?
2. Bagaimana kualitas, distribusi, dan potensi bias pada BBBP.csv dari DeepChem/MoleculeNet, berisi SMILES dan label p_np untuk lebih dari dua ribu senyawa.?
3. Bagaimana performa Logistic Regression dengan class_weight, Random Forest Classifier, KNN sebagai baseline tambahan jika dinilai menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, dan pembahasan threshold?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Logistic Regression dengan class_weight, Random Forest Classifier, KNN sebagai baseline tambahan.

3. Mengevaluasi performa menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, dan pembahasan threshold.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Martins, I. F., Teixeira, A. L., Pinheiro, L., & Falcao, A. O. (2012). A Bayesian Approach to in Silico Blood-Brain Barrier Penetration Modeling. *Journal of Chemical Information and Modeling*, 52(6), 1686-1697. DOI: 10.1021/ci300124c.

Penetrasi blood-brain barrier merupakan endpoint penting dalam desain obat sistem saraf pusat. Studi Martins dan kolega menyediakan basis ilmiah dataset BBBP yang kemudian menjadi benchmark MoleculeNet. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah BBBP.csv dari DeepChem/MoleculeNet, berisi SMILES dan label p_np untuk lebih dari dua ribu senyawa. Fitur yang dipakai adalah descriptor 2D dan Morgan fingerprint; kelas dibandingkan dengan logistic regression dan random forest. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Dataset benchmark tidak sepenuhnya mewakili prevalensi dunia nyata dan random split dapat menempatkan analog dekat di train dan test. Hasil tidak boleh dipakai untuk keputusan klinis. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. BBBP.csv dari DeepChem/MoleculeNet, berisi SMILES dan label p_np untuk lebih dari dua ribu senyawa. Variabel target: label permeabel/tidak permeabel terhadap blood-brain barrier. Variabel prediktor: descriptor 2D dan Morgan fingerprint; kelas dibandingkan dengan logistic regression dan random forest.

Quality Control Data. Audit imbalance kelas, SMILES invalid, duplikasi, dan scaffold. Gunakan stratify pada random split dan bandingkan dengan scaffold split sebagai diskusi metodologis.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor 2D dan Morgan fingerprint; kelas dibandingkan dengan logistic regression dan random forest.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Logistic Regression dengan `class_weight`, Random Forest Classifier, KNN sebagai baseline tambahan.
8. Evaluasi menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, dan pembahasan threshold.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.21. Jadwal kerja 14 hari Proyek 5

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan <code>data_raw</code> , catat DOI/URL/tanggal akses. Sumber proyek: BBBP.csv dari DeepChem/MoleculeNet, berisi SMILES dan label <code>p_np</code> untuk lebih dari dua ribu senyawa.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan <code>data_processed</code> serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Logistic Regression dengan <code>class_weight</code> , Random Forest Classifier, KNN sebagai baseline tambahan.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koefisien bila relevan. Fokus kimia: Diskusi mengaitkan permeabilitas dengan ukuran, lipofilisitas, TPSA, donor ikatan hidrogen,

Hari	Kegiatan dan output
	muatan, dan fleksibilitas sambil menegaskan bahwa hubungan tersebut bersifat multivariat.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.7. Script Python lengkap Proyek 5

```
import numpy as np
import pandas as pd
from rdkit import Chem, DataStructs
from rdkit.Chem import AllChem
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score,
confusion_matrix

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/BBBP.csv"
SEED = 42
N_BITS = 1024

df = pd.read_csv(URL)

def fp(smi):
    m = Chem.MolFromSmiles(str(smi))
    if m is None:
        return None
    bitvect = AllChem.GetMorganGenerator(radius=2,
fpSize=N_BITS).GetFingerprint(m)
    arr = np.zeros((N_BITS,), dtype=np.int8)
    DataStructs.ConvertToNumpyArray(bitvect, arr)
    return arr

X, y = [], []
for smi, label in zip(df["smiles"], pd.to_numeric(df["p_np"], errors="coerce")):
    v = fp(smi)
    if v is not None and pd.notna(label):
        X.append(v); y.append(int(label))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(
    X, y, test_size=0.20, random_state=SEED, stratify=y
)

models = {
    "LogReg": LogisticRegression(max_iter=2000, class_weight="balanced"),
    "RandomForest": RandomForestClassifier(
        n_estimators=500, random_state=SEED, n_jobs=-1,
        class_weight="balanced_subsample"
    ),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
```

```

pred = model.predict(Xte)
prob = model.predict_proba(Xte)[: , 1]
print("\n", name)
print(classification_report(yte, pred, digits=3))
print("ROC-AUC:", roc_auc_score(yte, prob))
print("Confusion matrix:\n", confusion_matrix(yte, pred))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.36. Template tabel hasil aktual Proyek 5

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, dan pembahasan threshold. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Diskusi mengaitkan permeabilitas dengan ukuran, lipofilisitas, TPSA, donor ikatan hidrogen, muatan, dan fleksibilitas sambil menegaskan bahwa hubungan tersebut bersifat multivariat. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Dataset benchmark tidak sepenuhnya mewakili prevalensi dunia nyata dan random split dapat menempatkan analog dekat di train dan test. Hasil tidak boleh dipakai untuk keputusan klinis.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembanding.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Dataset benchmark tidak sepenuhnya mewakili prevalensi dunia nyata dan random split dapat menempatkan analog dekat di train dan test. Hasil tidak boleh dipakai untuk keputusan klinis. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Martins et al. (2012), DOI 10.1021/ci300124c
- Wu et al. (2018), DOI 10.1039/C7SC02664A

D.9 PROYEK 6: KLASIFIKASI AMES MUTAGENICITY DARI STRUKTUR MOLEKUL

Tabel D.7. Ringkasan rancangan Proyek 6

Komponen	Rancangan proyek
Bidang	toksikologi komputasi dan QSAR klasifikasi
Basis jurnal	Hansen, K., Mika, S., Schroeter, T., et al. (2009). Benchmark Data Set for in Silico Prediction of Ames Mutagenicity. <i>Journal of Chemical Information and Modeling</i> , 49(9), 2077-2081. DOI: 10.1021/ci900161g.
Dataset	Supporting information jurnal berisi canonical SMILES dan hasil Ames. Agar praktikum stabil, mahasiswa mengunduh file supporting information sekali,

Komponen	Rancangan proyek
	mengeksrak file SMILES/CSV, lalu menjalankan skrip yang menerima dua kolom: smiles dan label.
Target	mutagenik vs non-mutagenik menurut benchmark Ames
Fitur	Morgan fingerprint dan descriptor fisikokimia sederhana
Model	Logistic Regression, Random Forest Classifier, KNN atau SVM linear sebagai pembanding
Evaluasi	ROC-AUC, precision, recall, F1, balanced accuracy, confusion matrix; utamakan interpretasi false negative
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Klasifikasi Ames Mutagenicity dari Struktur Molekul

Latar Belakang

Hansen dan kolega mempublikasikan benchmark publik sekitar 6500 senyawa dengan label Ames mutagenicity, disertai perbandingan beberapa metode machine learning. Dataset ini merupakan contoh kuat penggunaan QSAR untuk endpoint toksikologi eksperimental.

Proyek ini ditempatkan pada konteks toksikologi komputasi dan QSAR klasifikasi. Target yang dianalisis adalah mutagenik vs non-mutagenik menurut benchmark Ames. Pertanyaan inti penelitian adalah: Apakah Morgan fingerprint mampu menangkap pola struktur yang berhubungan dengan mutagenisitas, dan bagaimana class weighting memengaruhi sensitivitas model? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Pembahasan diarahkan pada kemungkinan toxicophore dan keterbatasan menghubungkan bit fingerprint dengan mekanisme reaktivitas DNA tanpa analisis struktural lanjutan. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah Morgan fingerprint mampu menangkap pola struktur yang berhubungan dengan mutagenisitas, dan bagaimana class weighting memengaruhi sensitivitas model?

2. Bagaimana kualitas, distribusi, dan potensi bias pada Supporting information jurnal berisi canonical SMILES dan hasil Ames. Agar praktikum stabil, mahasiswa mengunduh file supporting information sekali, mengekstrak file SMILES/CSV, lalu menjalankan skrip yang menerima dua kolom: smiles dan label.?
3. Bagaimana performa Logistic Regression, Random Forest Classifier, KNN atau SVM linear sebagai pembanding jika dinilai menggunakan ROC-AUC, precision, recall, F1, balanced accuracy, confusion matrix; utamakan interpretasi false negative?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Logistic Regression, Random Forest Classifier, KNN atau SVM linear sebagai pembanding.
3. Mengevaluasi performa menggunakan ROC-AUC, precision, recall, F1, balanced accuracy, confusion matrix; utamakan interpretasi false negative.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Hansen, K., Mika, S., Schroeter, T., et al. (2009). Benchmark Data Set for in Silico Prediction of Ames Mutagenicity. *Journal of Chemical Information and Modeling*, 49(9), 2077-2081. DOI: 10.1021/ci900161g.

Hansen dan kolega mempublikasikan benchmark publik sekitar 6500 senyawa dengan label Ames mutagenicity, disertai perbandingan beberapa metode machine learning. Dataset ini merupakan contoh kuat penggunaan QSAR untuk endpoint toksikologi eksperimental. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah Supporting information jurnal berisi canonical SMILES dan hasil Ames. Agar praktikum stabil, mahasiswa mengunduh file supporting information sekali, mengekstrak file SMILES/CSV, lalu menjalankan skrip yang menerima dua kolom: smiles dan label. Fitur yang dipakai adalah Morgan fingerprint dan descriptor fisikokimia sederhana. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Ames adalah assay biologis spesifik dan tidak identik dengan karsinogenisitas manusia. Dataset menggabungkan literatur; domain, protokol, dan strain dapat bervariasi. Model hanya untuk pendidikan dan prioritas hipotesis. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. Supporting information jurnal berisi canonical SMILES dan hasil Ames. Agar praktikum stabil, mahasiswa mengunduh file supporting information sekali, mengekstrak file SMILES/CSV, lalu menjalankan skrip yang menerima dua kolom: smiles dan label. Variabel target: mutagenik vs non-mutagenik menurut benchmark Ames. Variabel prediktor: Morgan fingerprint dan descriptor fisikokimia sederhana.

Quality Control Data. Simpan sumber supporting information, hilangkan struktur invalid, canonicalize, deduplicate, dan bila satu struktur mempunyai label konflik, laporkan aturan penanganannya.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: Morgan fingerprint dan descriptor fisikokimia sederhana.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Logistic Regression, Random Forest Classifier, KNN atau SVM linear sebagai pembanding.
8. Evaluasi menggunakan ROC-AUC, precision, recall, F1, balanced accuracy, confusion matrix; utamakan interpretasi false negative.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.

10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.22. Jadwal kerja 14 hari Proyek 6

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: Supporting information jurnal berisi canonical SMILES dan hasil Ames. Agar praktikum stabil, mahasiswa mengunduh file supporting information sekali, mengekstrak file SMILES/CSV, lalu menjalankan skrip yang menerima dua kolom: smiles dan label.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembandingan utama. Model utama: Logistic Regression, Random Forest Classifier, KNN atau SVM linear sebagai pembandingan.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Pembahasan diarahkan pada kemungkinan toxicophore dan keterbatasan menghubungkan bit fingerprint dengan mekanisme reaktivitas DNA tanpa analisis struktural lanjutan.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.8. Script Python lengkap Proyek 6

```
# Unduh Supporting Information Hansen et al. (2009), lalu siapkan ames.csv
# dengan kolom: smiles,label. Label harus 0/1.
import numpy as np
import pandas as pd
from rdkit import Chem, DataStructs
from rdkit.Chem import AllChem
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score

SEED = 42
N_BITS = 1024
```

```

df = pd.read_csv("ames.csv")

def fp(smi):
    mol = Chem.MolFromSmiles(str(smi))
    if mol is None:
        return None
    bv = AllChem.GetMorganGenerator(radius=2, fpSize=N_BITS).GetFingerprint(mol)
    arr = np.zeros(N_BITS, dtype=np.int8)
    DataStructs.ConvertToNumpyArray(bv, arr)
    return arr

X, y = [], []
for smi, lab in zip(df["smiles"], pd.to_numeric(df["label"], errors="coerce")):
    v = fp(smi)
    if v is not None and pd.notna(lab):
        X.append(v); y.append(int(lab))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(
    X, y, test_size=0.2, random_state=SEED, stratify=y
)

for model in [
    LogisticRegression(max_iter=2500, class_weight="balanced"),
    RandomForestClassifier(
        n_estimators=500, random_state=SEED, n_jobs=-1,
        class_weight="balanced_subsample"
    ),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    prob = model.predict_proba(Xte)[: , 1]
    print("\n", model.__class__.__name__)
    print(classification_report(yte, pred, digits=3))
    print("ROC-AUC:", roc_auc_score(yte, prob))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.37. Template tabel hasil aktual Proyek 6

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembandingan, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada ROC-AUC, precision, recall, F1, balanced accuracy, confusion matrix; utamakan interpretasi false negative. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Pembahasan diarahkan pada kemungkinan toxicophore dan keterbatasan menghubungkan bit fingerprint dengan mekanisme reaktivitas DNA tanpa analisis struktural lanjutan. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Ames adalah assay biologis spesifik dan tidak identik dengan karsinogenisitas manusia. Dataset menggabungkan literatur; domain, protokol, dan strain dapat bervariasi. Model hanya untuk pendidikan dan prioritas hipotesis.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Ames adalah assay biologis spesifik dan tidak identik dengan karsinogenisitas manusia. Dataset menggabungkan literatur; domain, protokol, dan strain dapat bervariasi.

Model hanya untuk pendidikan dan prioritas hipotesis. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Hansen et al. (2009), DOI 10.1021/ci900161g

D.10 PROYEK 7: KLASIFIKASI READY BIODEGRADABILITY BAHAN KIMIA

Tabel D.8. Ringkasan rancangan Proyek 7

Komponen	Rancangan proyek
Bidang	kimia lingkungan, green chemistry, dan QSAR
Basis jurnal	Mansouri, K., Ringsted, T., Ballabio, D., Todeschini, R., & Consonni, V. (2013). Quantitative Structure-Activity Relationship Models for Ready Biodegradability of Chemicals. <i>Journal of Chemical Information and Modeling</i> , 53(4), 867-878. DOI: 10.1021/ci4000213.
Dataset	UCI Machine Learning Repository, QSAR biodegradation, id 254. Data dapat dipanggil langsung melalui paket ucimlrepo.
Target	kelas ready biodegradable (RB) atau not ready biodegradable (NRB)
Fitur	41 descriptor molekul yang sudah dihitung pada dataset asli
Model	Logistic Regression, KNN, Random Forest, Support Vector Machine opsional
Evaluasi	balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, feature importance/permutation importance
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Klasifikasi Ready Biodegradability Bahan Kimia

Latar Belakang

Studi Mansouri menggunakan descriptor molekul untuk membedakan bahan kimia ready biodegradable dan not ready biodegradable. Dataset UCI berisi 1055 senyawa dan 41 descriptor sehingga sangat cocok untuk proyek dua minggu tanpa langkah featurisasi struktur yang berat.

Proyek ini ditempatkan pada konteks kimia lingkungan, green chemistry, dan QSAR. Target yang dianalisis adalah kelas ready biodegradable (RB) atau not ready biodegradable (NRB). Pertanyaan inti penelitian adalah: Descriptor mana yang paling informatif untuk membedakan ready biodegradability, dan apakah model nonlinear memberi keuntungan dibanding model linear? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak

perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Interpretasi menghubungkan ukuran, struktur karbon, heteroatom, topologi, dan lipofilisitas dengan kecenderungan biodegradasi, tanpa menyamakan feature importance dengan jalur biodegradasi mekanistik. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Descriptor mana yang paling informatif untuk membedakan ready biodegradability, dan apakah model nonlinear memberi keuntungan dibanding model linear?
2. Bagaimana kualitas, distribusi, dan potensi bias pada UCI Machine Learning Repository, QSAR biodegradation, id 254. Data dapat dipanggil langsung melalui paket ucimlrepo.?
3. Bagaimana performa Logistic Regression, KNN, Random Forest, Support Vector Machine opsional jika dinilai menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, feature importance/permutation importance?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Logistic Regression, KNN, Random Forest, Support Vector Machine opsional.
3. Mengevaluasi performa menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, feature importance/permutation importance.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Mansouri, K., Ringsted, T., Ballabio, D., Todeschini, R., & Consonni, V. (2013). Quantitative Structure-Activity Relationship Models for Ready Biodegradability of Chemicals. *Journal of Chemical Information and Modeling*, 53(4), 867-878. DOI: 10.1021/ci4000213.

Studi Mansouri menggunakan descriptor molekul untuk membedakan bahan kimia ready biodegradable dan not ready biodegradable. Dataset UCI berisi 1055 senyawa dan 41 descriptor sehingga sangat cocok untuk proyek dua minggu tanpa langkah featurisasi struktur yang berat. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah UCI Machine Learning Repository, QSAR biodegradation, id 254. Data dapat dipanggil langsung melalui paket ucimlrepo. Fitur yang dipakai adalah 41 descriptor molekul yang sudah dihitung pada dataset asli. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Label ready biodegradability mengikuti protokol tertentu dan tidak menjamin nasib lingkungan pada seluruh kondisi. Descriptor telah dipilih dari pipeline sumber sehingga proyek ini tidak menguji keseluruhan ruang descriptor. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. UCI Machine Learning Repository, QSAR biodegradation, id 254. Data dapat dipanggil langsung melalui paket ucimlrepo. Variabel target: kelas ready biodegradable (RB) atau not ready biodegradable (NRB). Variabel prediktor: 41 descriptor molekul yang sudah dihitung pada dataset asli.

Quality Control Data. Periksa tipe kolom, encoding target, class imbalance, korelasi antar-descriptor, dan scaling untuk KNN/logistic regression. Gunakan Pipeline agar preprocessing tidak bocor.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.

3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: 41 descriptor molekul yang sudah dihitung pada dataset asli.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Logistic Regression, KNN, Random Forest, Support Vector Machine opsional.
8. Evaluasi menggunakan balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, feature importance/permutation importance.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.23. Jadwal kerja 14 hari Proyek 7

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: UCI Machine Learning Repository, QSAR biodegradation, id 254. Data dapat dipanggil langsung melalui paket ucimlrepo.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Logistic Regression, KNN, Random Forest, Support Vector Machine opsional.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Interpretasi menghubungkan ukuran, struktur karbon, heteroatom, topologi, dan lipofilisitas dengan kecenderungan biodegradasi, tanpa menyamakan feature importance dengan jalur biodegradasi mekanistik.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan

skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.9. Script Python lengkap Proyek 7

```
import pandas as pd
from ucimlrepo import fetch_ucirepo
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score

SEED = 42
repo = fetch_ucirepo(id=254)
X = repo.data.features.copy()
y = repo.data.targets.iloc[:, 0].astype(str)
# UCI umumnya memakai RB dan NRB; petakan secara eksplisit.
y = y.str.upper().map({"RB": 1, "NRB": 0})
mask = y.notna()
X, y = X.loc[mask], y.loc[mask].astype(int)

Xtr, Xte, ytr, yte = train_test_split(
    X, y, test_size=0.20, random_state=SEED, stratify=y
)
models = {
    "LogReg": Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("scale", StandardScaler()),
        ("model", LogisticRegression(max_iter=2000, class_weight="balanced")),
    ]),
    "RandomForest": Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("model", RandomForestClassifier(
            n_estimators=500, random_state=SEED, n_jobs=-1,
            class_weight="balanced_subsample"
        )),
    ]),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    prob = model.predict_proba(Xte)[:, 1]
    print("\n", name)
    print(classification_report(yte, pred, digits=3))
    print("ROC-AUC:", roc_auc_score(yte, prob))
```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.38. Template tabel hasil aktual Proyek 7

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembandingan, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada balanced accuracy, precision, recall, F1, ROC-AUC, confusion matrix, feature importance/permutation importance. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Interpretasi menghubungkan ukuran, struktur karbon, heteroatom, topologi, dan lipofilisitas dengan kecenderungan biodegradasi, tanpa menyamakan feature importance dengan jalur biodegradasi mekanistik. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Label ready biodegradability mengikuti protokol tertentu dan tidak menjamin nasib lingkungan pada seluruh kondisi. Descriptor telah dipilih dari pipeline sumber sehingga proyek ini tidak menguji keseluruhan ruang descriptor.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.

- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Label ready biodegradability mengikuti protokol tertentu dan tidak menjamin nasib lingkungan pada seluruh kondisi. Descriptor telah dipilih dari pipeline sumber sehingga proyek ini tidak menguji keseluruhan ruang descriptor. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Mansouri et al. (2013), DOI 10.1021/ci4000213
- UCI dataset DOI 10.24432/C5H60M

D.11 PROYEK 8: PREDIKSI TOKSISITAS AKUT TERHADAP IKAN FATHEAD MINNOW

Tabel D.9. Ringkasan rancangan Proyek 8

Komponen	Rancangan proyek
Bidang	ekotoksikologi, kimia lingkungan, dan QSAR regresi
Basis jurnal	Cassotti, M., Ballabio, D., Todeschini, R., & Consonni, V. (2015). A similarity-based QSAR model for predicting acute toxicity towards the fathead minnow (<i>Pimephales promelas</i>). SAR and QSAR in Environmental Research, 26(3), 217-243. DOI: 10.1080/1062936X.2015.1018938.
Dataset	UCI QSAR fish toxicity, id 504, dengan enam descriptor numerik dan satu target kuantitatif LC50 yang dinyatakan sebagai $-\log(\text{mol/L})$.
Target	LC50 96 jam pada <i>Pimephales promelas</i> dalam skala $-\log(\text{mol/L})$
Fitur	CIC0, SM1_Dz(Z), GATSIi, NdsCH, NdssC, dan MLOGP sesuai dataset
Model	KNN Regressor, Random Forest Regressor, Gradient Boosting Regressor
Evaluasi	MAE, RMSE, R2; analisis residual dan error pada wilayah target rendah/tinggi
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Toksisitas Akut terhadap Ikan Fathead Minnow

Latar Belakang

Kasus menggunakan 908 bahan kimia dengan enam descriptor dan target LC50 96 jam. Karena fitur sudah tersedia, mahasiswa dapat fokus pada desain evaluasi, applicability domain, dan interpretasi ekotoksikologi.

Proyek ini ditempatkan pada konteks ekotoksikologi, kimia lingkungan, dan QSAR regresi. Target yang dianalisis adalah LC50 96 jam pada *Pimephales promelas* dalam skala $-\log(\text{mol/L})$. Pertanyaan inti penelitian adalah: Apakah model ensemble modern meningkatkan prediksi LC50 dibanding pendekatan berbasis tetangga pada enam descriptor yang sama? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Bahas MLOGP sebagai indikator partisi/hidrofobisitas serta descriptor topologi sebagai ringkasan struktur. Tekankan bahwa toksisitas juga dipengaruhi mekanisme aksi yang tidak seluruhnya tertangkap oleh enam fitur. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah model ensemble modern meningkatkan prediksi LC50 dibanding pendekatan berbasis tetangga pada enam descriptor yang sama?
2. Bagaimana kualitas, distribusi, dan potensi bias pada UCI QSAR fish toxicity, id 504, dengan enam descriptor numerik dan satu target kuantitatif LC50 yang dinyatakan sebagai $-\log(\text{mol/L})$?
3. Bagaimana performa KNN Regressor, Random Forest Regressor, Gradient Boosting Regressor jika dinilai menggunakan MAE, RMSE, R2; analisis residual dan error pada wilayah target rendah/tinggi?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan KNN Regressor, Random Forest Regressor, Gradient Boosting Regressor.

3. Mengevaluasi performa menggunakan MAE, RMSE, R²; analisis residual dan error pada wilayah target rendah/tinggi.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Cassotti, M., Ballabio, D., Todeschini, R., & Consonni, V. (2015). A similarity-based QSAR model for predicting acute toxicity towards the fathead minnow (*Pimephales promelas*). *SAR and QSAR in Environmental Research*, 26(3), 217-243. DOI: 10.1080/1062936X.2015.1018938.

Kasus menggunakan 908 bahan kimia dengan enam descriptor dan target LC50 96 jam. Karena fitur sudah tersedia, mahasiswa dapat fokus pada desain evaluasi, applicability domain, dan interpretasi ekotoksikologi. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah UCI QSAR fish toxicity, id 504, dengan enam descriptor numerik dan satu target kuantitatif LC50 yang dinyatakan sebagai $-\log(\text{mol/L})$. Fitur yang dipakai adalah CIC0, SM1_Dz(Z), GATS1i, NdsCH, NdssC, dan MLOGP sesuai dataset. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Data spesifik spesies, endpoint, dan durasi. Hasil tidak dapat digeneralisasikan ke manusia atau organisme lain tanpa validasi. Applicability domain penting ketika descriptor sampel baru jauh dari data training. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. UCI QSAR fish toxicity, id 504, dengan enam descriptor numerik dan satu target kuantitatif LC50 yang dinyatakan sebagai $-\log(\text{mol/L})$. Variabel target: LC50 96 jam pada *Pimephales promelas* dalam skala -

log(mol/L). Variabel prediktor: CIC0, SM1_Dz(Z), GATS1i, NdsCH, NdssC, dan MLOGP sesuai dataset.

Quality Control Data. Tidak ada missing value menurut dokumentasi UCI, tetapi tipe numerik dan target tetap diaudit. Standardisasi wajib untuk KNN; tree ensemble tidak memerlukannya secara prinsip.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: CIC0, SM1_Dz(Z), GATS1i, NdsCH, NdssC, dan MLOGP sesuai dataset.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: KNN Regressor, Random Forest Regressor, Gradient Boosting Regressor.
8. Evaluasi menggunakan MAE, RMSE, R²; analisis residual dan error pada wilayah target rendah/tinggi.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.24. Jadwal kerja 14 hari Proyek 8

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: UCI QSAR fish toxicity, id 504, dengan enam descriptor numerik dan satu target kuantitatif LC50 yang dinyatakan sebagai -log(mol/L).
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: KNN Regressor, Random Forest Regressor, Gradient Boosting Regressor.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.

Hari	Kegiatan dan output
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Bahas MLOGP sebagai indikator partisi/hidrofobisitas serta descriptor topologi sebagai ringkasan struktur. Tekankan bahwa toksisitas juga dipengaruhi mekanisme aksi yang tidak seluruhnya tertangkap oleh enam fitur.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.10. Script Python lengkap Proyek 8

```
from ucimlrepo import fetch_ucirepo
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

SEED = 42
repo = fetch_ucirepo(id=504)
X = repo.data.features.copy()
y = repo.data.targets.iloc[:, 0].astype(float)
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)

models = {
    "KNN": Pipeline([("scale", StandardScaler()), ("model",
KNeighborsRegressor(n_neighbors=7))]),
    "RandomForest": RandomForestRegressor(n_estimators=500, random_state=SEED,
n_jobs=-1),
    "GradientBoosting": GradientBoostingRegressor(random_state=SEED),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(name,
          "MAE", round(mean_absolute_error(yte, pred), 4),
          "RMSE", round(mean_squared_error(yte, pred) ** 0.5, 4),
          "R2", round(r2_score(yte, pred), 4))
```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.39. Template tabel hasil aktual Proyek 8

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R²; analisis residual dan error pada wilayah target rendah/tinggi. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Bahas MLOGP sebagai indikator partisi/hidrofobisitas serta descriptor topologi sebagai ringkasan struktur. Tekankan bahwa toksisitas juga dipengaruhi mekanisme aksi yang tidak seluruhnya tertangkap oleh enam fitur. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Data spesifik spesies, endpoint, dan durasi. Hasil tidak dapat digeneralisasikan ke manusia atau organisme lain tanpa validasi. Applicability domain penting ketika descriptor sampel baru jauh dari data training.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.

- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Data spesifik spesies, endpoint, dan durasi. Hasil tidak dapat digeneralisasikan ke manusia atau organisme lain tanpa validasi. Applicability domain penting ketika descriptor sampel baru jauh dari data training. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Cassotti et al. (2015), DOI 10.1080/1062936X.2015.1018938
- UCI dataset DOI 10.24432/C5JG7B

D.12 PROYEK 9: KLASIFIKASI TOKSISITAS TOX21 PADA JALUR STRESS RESPONSE P53

Tabel D.10. Ringkasan rancangan Proyek 9

Komponen	Rancangan proyek
Bidang	toksikologi komputasi, high-throughput screening, dan klasifikasi tidak seimbang
Basis jurnal	Mayr, A., Klambauer, G., Unterthiner, T., & Hochreiter, S. (2016). DeepTox: Toxicity Prediction using Deep Learning. <i>Frontiers in Environmental Science</i> , 3, 80. DOI: 10.3389/fenvs.2015.00080.
Dataset	tox21.csv.gz dari DeepChem/MoleculeNet. Target SR-p53 memiliki nilai hilang untuk sebagian struktur; hanya baris dengan label tersedia yang digunakan.
Target	aktivitas biner pada assay SR-p53
Fitur	Morgan fingerprint 1024 bit
Model	Logistic Regression balanced dan Random Forest balanced
Evaluasi	ROC-AUC, PR-AUC, precision, recall, F1, confusion matrix; PR-AUC penting bila kelas positif jarang
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Klasifikasi Toksisitas Tox21 pada Jalur Stress Response p53

Latar Belakang

Tox21 Data Challenge mengukur ribuan bahan kimia pada beberapa jalur nuclear receptor dan stress response. Proyek S1 dibatasi pada satu endpoint SR-p53 agar fokus dan komputasi tetap realistis dalam dua minggu.

Proyek ini ditempatkan pada konteks toksikologi komputasi, high-throughput screening, dan klasifikasi tidak seimbang. Target yang dianalisis adalah aktivitas biner pada assay SR-p53. Pertanyaan inti penelitian adalah: Seberapa baik fingerprint struktur memprediksi respons SR-p53 dan bagaimana class weighting mengubah trade-off sensitivitas-spesifisitas? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Diskusi harus berhati-hati: label assay adalah respons jalur seluler, bukan diagnosis toksisitas manusia. Bit fingerprint dapat menjadi marker statistik struktur, bukan bukti mekanisme toksik. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa baik fingerprint struktur memprediksi respons SR-p53 dan bagaimana class weighting mengubah trade-off sensitivitas-spesifisitas?
2. Bagaimana kualitas, distribusi, dan potensi bias pada tox21.csv.gz dari DeepChem/MoleculeNet. Target SR-p53 memiliki nilai hilang untuk sebagian struktur; hanya baris dengan label tersedia yang digunakan.?
3. Bagaimana performa Logistic Regression balanced dan Random Forest balanced jika dinilai menggunakan ROC-AUC, PR-AUC, precision, recall, F1, confusion matrix; PR-AUC penting bila kelas positif jarang?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Logistic Regression balanced dan Random Forest balanced.
3. Mengevaluasi performa menggunakan ROC-AUC, PR-AUC, precision, recall, F1, confusion matrix; PR-AUC penting bila kelas positif jarang.

4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Mayr, A., Klambauer, G., Unterthiner, T., & Hochreiter, S. (2016). DeepTox: Toxicity Prediction using Deep Learning. *Frontiers in Environmental Science*, 3, 80. DOI: 10.3389/fenvs.2015.00080.

Tox21 Data Challenge mengukur ribuan bahan kimia pada beberapa jalur nuclear receptor dan stress response. Proyek S1 dibatasi pada satu endpoint SR-p53 agar fokus dan komputasi tetap realistis dalam dua minggu. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah tox21.csv.gz dari DeepChem/MoleculeNet. Target SR-p53 memiliki nilai hilang untuk sebagian struktur; hanya baris dengan label tersedia yang digunakan. Fitur yang dipakai adalah Morgan fingerprint 1024 bit. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Tox21 multi-assay dan missing labels bersifat umum. Proyek ini hanya single-task baseline, tidak memanfaatkan transfer antar-task seperti model DeepTox asli. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. tox21.csv.gz dari DeepChem/MoleculeNet. Target SR-p53 memiliki nilai hilang untuk sebagian struktur; hanya baris dengan label tersedia yang digunakan. Variabel target: aktivitas biner pada assay SR-p53. Variabel prediktor: Morgan fingerprint 1024 bit.

Quality Control Data. Hitung prevalensi positif, buang target missing, cek struktur invalid, gunakan stratified split, dan laporkan PR-AUC selain ROC-AUC.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: Morgan fingerprint 1024 bit.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Logistic Regression balanced dan Random Forest balanced.
8. Evaluasi menggunakan ROC-AUC, PR-AUC, precision, recall, F1, confusion matrix; PR-AUC penting bila kelas positif jarang.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.25. Jadwal kerja 14 hari Proyek 9

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: tox21.csv.gz dari DeepChem/MoleculeNet. Target SR-p53 memiliki nilai hilang untuk sebagian struktur; hanya baris dengan label tersedia yang digunakan.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembandingan utama. Model utama: Logistic Regression balanced dan Random Forest balanced.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Diskusi harus berhati-hati: label assay adalah respons jalur seluler, bukan diagnosis toksisitas manusia. Bit fingerprint dapat menjadi marker statistik struktur, bukan bukti mekanisme toksik.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.11. Script Python lengkap Proyek 9

```
import numpy as np
import pandas as pd
from rdkit import Chem, DataStructs
from rdkit.Chem import AllChem
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score,
average_precision_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/tox21.csv.gz"
TARGET = "SR-p53"
SEED = 42
N_BITS = 1024

df = pd.read_csv(URL)

def fp(smi):
    m = Chem.MolFromSmiles(str(smi))
    if m is None:
        return None
    bv = AllChem.GetMorganGenerator(radius=2, fpSize=N_BITS).GetFingerprint(m)
    a = np.zeros(N_BITS, dtype=np.int8)
    DataStructs.ConvertToNumpyArray(bv, a)
    return a

X, y = [], []
for smi, lab in zip(df["smiles"], pd.to_numeric(df[TARGET], errors="coerce")):
    v = fp(smi)
    if v is not None and pd.notna(lab):
        X.append(v); y.append(int(lab))
X, y = np.asarray(X), np.asarray(y)
print("N=", len(y), "prevalensi positif=", y.mean())
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED,
stratify=y)

models = {
    "LogReg": LogisticRegression(max_iter=2000, class_weight="balanced"),
    "RF": RandomForestClassifier(n_estimators=500, random_state=SEED, n_jobs=-1,
                                class_weight="balanced_subsample"),
}
for name, model in models.items():
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    prob = model.predict_proba(Xte)[:, 1]
    print("\n", name)
    print(classification_report(yte, pred, digits=3))
    print("ROC-AUC", roc_auc_score(yte, prob))
    print("PR-AUC", average_precision_score(yte, prob))
```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output

terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.40. Template tabel hasil aktual Proyek 9

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembandingan, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada ROC-AUC, PR-AUC, precision, recall, F1, confusion matrix; PR-AUC penting bila kelas positif jarang. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Diskusi harus berhati-hati: label assay adalah respons jalur seluler, bukan diagnosis toksisitas manusia. Bit fingerprint dapat menjadi marker statistik struktur, bukan bukti mekanisme toksik. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Tox21 multi-assay dan missing labels bersifat umum. Proyek ini hanya single-task baseline, tidak memanfaatkan transfer antar-task seperti model DeepTox asli.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.

- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Tox21 multi-assay dan missing labels bersifat umum. Proyek ini hanya single-task baseline, tidak memanfaatkan transfer antar-task seperti model DeepTox asli. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Mayr et al. (2016), DOI 10.3389/fenvs.2015.00080
- Wu et al. (2018), DOI 10.1039/C7SC02664A

D.13 PROYEK 10: KLASIFIKASI SENYAWA PENGHAMBAT REPLIKASI HIV

Tabel D.11. Ringkasan rancangan Proyek 10

Komponen	Rancangan proyek
Bidang	kimia obat, virtual screening, dan imbalanced classification
Basis jurnal	Wu, Z., Ramsundar, B., Feinberg, E. N., et al. (2018). MoleculeNet: a benchmark for molecular machine learning. Chemical Science, 9, 513-530. DOI: 10.1039/C7SC02664A. MoleculeNet menjelaskan bahwa dataset HIV berasal dari DTP AIDS Antiviral Screen dan memuat lebih dari 40 ribu senyawa.
Dataset	HIV.csv dari DeepChem/MoleculeNet dengan kolom smiles dan HIV_active.
Target	aktif vs tidak aktif menghambat replikasi HIV pada screen sumber
Fitur	Morgan fingerprint 1024 bit
Model	Logistic Regression balanced dan Random Forest; subset stratified 12.000 senyawa untuk laptop standar
Evaluasi	ROC-AUC, PR-AUC, recall kelas aktif, F1, confusion matrix
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Klasifikasi Senyawa Penghambat Replikasi HIV

Latar Belakang

Dataset HIV merupakan contoh realistis virtual screening dengan class imbalance kuat. Untuk proyek 14 hari, mahasiswa menggunakan subset stratified yang tetap cukup besar untuk membandingkan model dan memahami pentingnya PR-AUC.

Proyek ini ditempatkan pada konteks kimia obat, virtual screening, dan imbalanced classification. Target yang dianalisis adalah aktif vs tidak aktif menghambat replikasi HIV pada screen sumber. Pertanyaan inti penelitian adalah: Bagaimana class imbalance memengaruhi accuracy dan mengapa PR-AUC serta recall kelas aktif lebih informatif untuk screening senyawa? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Diskusi menekankan bahwa fingerprint mengukur kemiripan struktur; prediksi aktif tidak mengidentifikasi target molekuler atau mekanisme antiviral tertentu. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Bagaimana class imbalance memengaruhi accuracy dan mengapa PR-AUC serta recall kelas aktif lebih informatif untuk screening senyawa?
2. Bagaimana kualitas, distribusi, dan potensi bias pada HIV.csv dari DeepChem/MoleculeNet dengan kolom smiles dan HIV_active.?
3. Bagaimana performa Logistic Regression balanced dan Random Forest; subset stratified 12.000 senyawa untuk laptop standar jika dinilai menggunakan ROC-AUC, PR-AUC, recall kelas aktif, F1, confusion matrix?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.

2. Membangun dan membandingkan Logistic Regression balanced dan Random Forest; subset stratified 12.000 senyawa untuk laptop standar.
3. Mengevaluasi performa menggunakan ROC-AUC, PR-AUC, recall kelas aktif, F1, confusion matrix.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Wu, Z., Ramsundar, B., Feinberg, E. N., et al. (2018). MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9, 513-530. DOI: 10.1039/C7SC02664A. MoleculeNet menjelaskan bahwa dataset HIV berasal dari DTP AIDS Antiviral Screen dan memuat lebih dari 40 ribu senyawa.

Dataset HIV merupakan contoh realistis virtual screening dengan class imbalance kuat. Untuk proyek 14 hari, mahasiswa menggunakan subset stratified yang tetap cukup besar untuk membandingkan model dan memahami pentingnya PR-AUC. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah HIV.csv dari DeepChem/MoleculeNet dengan kolom smiles dan HIV_active. Fitur yang dipakai adalah Morgan fingerprint 1024 bit. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Dataset screening mengandung label yang bergantung protokol biologis. Subsampling dipakai untuk batas waktu praktikum dan harus dilaporkan; model bukan alat keputusan terapi. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. HIV.csv dari DeepChem/MoleculeNet dengan kolom smiles dan HIV_active. Variabel target: aktif vs tidak aktif menghambat replikasi HIV pada screen sumber. Variabel prediktor: Morgan fingerprint 1024 bit.

Quality Control Data. Gunakan stratified subsampling dengan seed, cek prevalensi kelas, struktur invalid, dan duplikasi. Jangan menilai model hanya dengan accuracy.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: Morgan fingerprint 1024 bit.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Logistic Regression balanced dan Random Forest; subset stratified 12.000 senyawa untuk laptop standar.
8. Evaluasi menggunakan ROC-AUC, PR-AUC, recall kelas aktif, F1, confusion matrix.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.26. Jadwal kerja 14 hari Proyek 10

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: HIV.csv dari DeepChem/MoleculeNet dengan kolom smiles dan HIV_active.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standarisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Logistic Regression balanced dan Random Forest; subset stratified 12.000 senyawa untuk laptop standar.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.

Hari	Kegiatan dan output
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Diskusi menekankan bahwa fingerprint mengukur kemiripan struktur; prediksi aktif tidak mengidentifikasi target molekuler atau mekanisme antiviral tertentu.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.12. Script Python lengkap Proyek 10

```
import numpy as np
import pandas as pd
from rdkit import Chem, DataStructs
from rdkit.Chem import AllChem
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import roc_auc_score, average_precision_score,
classification_report

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/HIV.csv"
SEED = 42
N_BITS = 1024
MAX_N = 12000

df = pd.read_csv(URL).dropna(subset=["smiles", "HIV_active"])
if len(df) > MAX_N:
    a, _ = train_test_split(
        df, train_size=MAX_N, random_state=SEED, stratify=df["HIV_active"]
    )
    df = a.reset_index(drop=True)

def fp(smi):
    m = Chem.MolFromSmiles(str(smi))
    if m is None:
        return None
    bv = AllChem.GetMorganGenerator(radius=2, fpSize=N_BITS).GetFingerprint(m)
    x = np.zeros(N_BITS, dtype=np.int8)
    DataStructs.ConvertToNumpyArray(bv, x)
    return x

X, y = [], []
for smi, lab in zip(df["smiles"], df["HIV_active"]):
    v = fp(smi)
    if v is not None:
        X.append(v); y.append(int(lab))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED,
stratify=y)
print("Prevalensi aktif:", y.mean())

for model in [
    LogisticRegression(max_iter=2500, class_weight="balanced"),
    RandomForestClassifier(n_estimators=400, random_state=SEED, n_jobs=-1,
```

```

class_weight="balanced_subsample"),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    prob = model.predict_proba(Xte)[: , 1]
    print("\n", model.__class__.__name__)
    print(classification_report(yte, pred, digits=3))
    print("ROC-AUC", roc_auc_score(yte, prob))
    print("PR-AUC", average_precision_score(yte, prob))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.41. Template tabel hasil aktual Proyek 10

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada ROC-AUC, PR-AUC, recall kelas aktif, F1, confusion matrix. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Diskusi menekankan bahwa fingerprint mengukur kemiripan struktur; prediksi aktif tidak mengidentifikasi target molekuler atau mekanisme antiviral tertentu. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Dataset screening mengandung label yang bergantung protokol biologis. Subsampling dipakai untuk batas waktu praktikum dan harus dilaporkan; model bukan alat keputusan terapi.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembanding.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Dataset screening mengandung label yang bergantung protokol biologis. Subsampling dipakai untuk batas waktu praktikum dan harus dilaporkan; model bukan alat keputusan terapi. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Wu et al. (2018), DOI 10.1039/C7SC02664A

D.14 PROYEK 11: QSAR INHIBITOR BACE-1 UNTUK STUDI ALZHEIMER

Tabel D.12. Ringkasan rancangan Proyek 11

Komponen	Rancangan proyek
Bidang	kimia medisn, QSAR, dan protein-target activity modeling
Basis jurnal	Subramanian, G., Ramsundar, B., Pande, V., & Denny, R. A. (2016). Computational Modeling of beta-Secretase 1 (BACE-1) Inhibitors Using Ligand Based Approaches. <i>Journal of Chemical Information and Modeling</i> , 56(10), 1936-1949. DOI: 10.1021/acs.jcim.6b00290.
Dataset	bace.csv dari DeepChem/MoleculeNet. Proyek utama memakai Class untuk klasifikasi; pengembangan dapat mengganti target menjadi pIC50 untuk regresi.

Komponen	Rancangan proyek
Target	Class aktif/tidak aktif terhadap BACE-1; opsi pengembangan pIC50
Fitur	Morgan fingerprint 1024 bit dan descriptor RDKit untuk interpretasi tambahan
Model	Random Forest Classifier, Logistic Regression; opsi Random Forest Regressor untuk pIC50
Evaluasi	ROC-AUC, F1, precision, recall, confusion matrix; untuk regresi MAE/RMSE/R2
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

QSAR Inhibitor BACE-1 untuk Studi Alzheimer

Latar Belakang

Paper BACE-1 menghimpun binding affinities eksperimental dari beragam kelas inhibitor dan membandingkan pendekatan ligand-based termasuk machine learning. Dataset benchmark menyediakan label kelas dan pIC50 sehingga cocok untuk klasifikasi atau regresi.

Proyek ini ditempatkan pada konteks kimia medisn, QSAR, dan protein-target activity modeling. Target yang dianalisis adalah Class aktif/tidak aktif terhadap BACE-1; opsi pengembangan pIC50. Pertanyaan inti penelitian adalah: Apakah fingerprint ligand cukup untuk memisahkan inhibitor BACE-1 aktif dan tidak aktif, serta bagaimana scaffold similarity memengaruhi generalisasi? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Pembahasan mencakup hubungan struktur ligand dan aktivitas, tetapi model tidak memasukkan struktur protein atau mode binding sehingga tidak boleh dianggap sebagai docking surrogate penuh. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah fingerprint ligand cukup untuk memisahkan inhibitor BACE-1 aktif dan tidak aktif, serta bagaimana scaffold similarity memengaruhi generalisasi?
2. Bagaimana kualitas, distribusi, dan potensi bias pada bace.csv dari DeepChem/MoleculeNet. Proyek utama memakai Class untuk klasifikasi; pengembangan dapat mengganti target menjadi pIC50 untuk regresi.?

3. Bagaimana performa Random Forest Classifier, Logistic Regression; opsi Random Forest Regressor untuk pIC50 jika dinilai menggunakan ROC-AUC, F1, precision, recall, confusion matrix; untuk regresi MAE/RMSE/R2?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Random Forest Classifier, Logistic Regression; opsi Random Forest Regressor untuk pIC50.
3. Mengevaluasi performa menggunakan ROC-AUC, F1, precision, recall, confusion matrix; untuk regresi MAE/RMSE/R2.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Subramanian, G., Ramsundar, B., Pande, V., & Denny, R. A. (2016). Computational Modeling of beta-Secretase 1 (BACE-1) Inhibitors Using Ligand Based Approaches. *Journal of Chemical Information and Modeling*, 56(10), 1936-1949. DOI: 10.1021/acs.jcim.6b00290.

Paper BACE-1 menghimpun binding affinities eksperimental dari beragam kelas inhibitor dan membandingkan pendekatan ligand-based termasuk machine learning. Dataset benchmark menyediakan label kelas dan pIC50 sehingga cocok untuk klasifikasi atau regresi. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah bace.csv dari DeepChem/MoleculeNet. Proyek utama memakai Class untuk klasifikasi; pengembangan dapat mengganti target menjadi pIC50 untuk regresi. Fitur yang dipakai adalah Morgan fingerprint 1024 bit dan descriptor RDKit untuk interpretasi tambahan. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. BACE dataset terdiri dari seri analog yang dapat membuat random split optimistik. Scaffold split atau group split seharusnya dibahas sebagai evaluasi yang lebih menantang. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper

sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. bace.csv dari DeepChem/MoleculeNet. Proyek utama memakai Class untuk klasifikasi; pengembangan dapat mengganti target menjadi pIC50 untuk regresi. Variabel target: Class aktif/tidak aktif terhadap BACE-1; opsi pengembangan pIC50. Variabel prediktor: Morgan fingerprint 1024 bit dan descriptor RDKit untuk interpretasi tambahan.

Quality Control Data. Periksa nama kolom mol, Class, pIC50; canonicalize struktur; deduplicate; laporkan proporsi kelas dan penggunaan random/scaffold split.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: Morgan fingerprint 1024 bit dan descriptor RDKit untuk interpretasi tambahan.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Random Forest Classifier, Logistic Regression; opsi Random Forest Regressor untuk pIC50.
8. Evaluasi menggunakan ROC-AUC, F1, precision, recall, confusion matrix; untuk regresi MAE/RMSE/R2.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.27. Jadwal kerja 14 hari Proyek 11

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: bace.csv dari DeepChem/MoleculeNet. Proyek utama memakai Class untuk klasifikasi; pengembangan dapat mengganti target menjadi pIC50 untuk regresi.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembandingan utama. Model utama: Random Forest Classifier, Logistic Regression; opsi Random Forest Regressor untuk pIC50.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Pembahasan mencakup hubungan struktur ligand dan aktivitas, tetapi model tidak memasukkan struktur protein atau mode binding sehingga tidak boleh dianggap sebagai docking surrogate penuh.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.13. Script Python lengkap Proyek 11

```
import numpy as np
import pandas as pd
from rdkit import Chem, DataStructs
from rdkit.Chem import AllChem
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/bace.csv"
SEED = 42
N_BITS = 1024

df = pd.read_csv(URL)
smiles_col = "mol"
target_col = "Class"

def fp(smi):
    m = Chem.MolFromSmiles(str(smi))
```

```

if m is None:
    return None
bv = AllChem.GetMorganGenerator(radius=2, fpSize=N_BITS).GetFingerprint(m)
a = np.zeros(N_BITS, dtype=np.int8)
DataStructs.ConvertToNumpyArray(bv, a)
return a

X, y = [], []
for smi, lab in zip(df[smiles_col], pd.to_numeric(df[target_col],
errors="coerce")):
    v = fp(smi)
    if v is not None and pd.notna(lab):
        X.append(v); y.append(int(lab))
X, y = np.asarray(X), np.asarray(y)
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED,
stratify=y)

for model in [
    LogisticRegression(max_iter=2500, class_weight="balanced"),
    RandomForestClassifier(n_estimators=500, random_state=SEED, n_jobs=-1,
class_weight="balanced_subsample"),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    prob = model.predict_proba(Xte)[: , 1]
    print("\n", model.__class__.__name__)
    print(classification_report(yte, pred, digits=3))
    print("ROC-AUC", roc_auc_score(yte, prob))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.42. Template tabel hasil aktual Proyek 11

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada ROC-AUC, F1, precision, recall, confusion matrix; untuk regresi MAE/RMSE/R2. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Pembahasan mencakup hubungan struktur ligand dan aktivitas, tetapi model tidak memasukkan struktur protein atau mode binding sehingga tidak boleh dianggap sebagai docking surrogate penuh. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: BACE dataset terdiri dari seri analog yang dapat membuat random split optimistik. Scaffold split atau group split seharusnya dibahas sebagai evaluasi yang lebih menantang.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: BACE dataset terdiri dari seri analog yang dapat membuat random split optimistik. Scaffold split atau group split seharusnya dibahas sebagai evaluasi yang lebih menantang. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Subramanian et al. (2016), DOI 10.1021/acs.jcim.6b00290
- Wu et al. (2018), DOI 10.1039/C7SC02664A

D.15 PROYEK 12: QSAR BIOAKTIVITAS HER2/ERBB2 DARI CHEMBL

Tabel D.13. Ringkasan rancangan Proyek 12

Komponen	Rancangan proyek
Bidang	kimia medisn, database bioaktivitas, dan QSAR regresi
Basis jurnal	Zdrazil, B., Felix, E., Hunter, F., et al. (2024). The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. <i>Nucleic Acids Research</i> , 52(D1), D1180-D1192. DOI: 10.1093/nar/gkad1004.
Dataset	Aktivitas ChEMBL terhadap target manusia ERBB2/HER2. Skrip mencari target, mengambil IC ₅₀ dengan unit nM dan relation =, lalu mengubahnya menjadi pIC ₅₀ = 9 - log ₁₀ (IC ₅₀ nM).
Target	pIC ₅₀ terhadap HER2/ERBB2 berdasarkan record IC ₅₀ ChEMBL yang lolos filter
Fitur	descriptor RDKit dan Morgan fingerprint; baseline di bawah menggunakan descriptor 2D agar interpretasi mudah
Model	Random Forest Regressor, Extra Trees Regressor, Ridge
Evaluasi	MAE, RMSE, R ² , scatter experimental-predicted, audit jumlah assay dan jumlah senyawa unik
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

QSAR Bioaktivitas HER2/ERBB2 dari ChEMBL

Latar Belakang

ChEMBL mengkurasi data struktur-aktivitas dari literatur dan sumber bioaktivitas. Proyek ini mengajarkan pembangunan dataset riset secara terprogram, bukan hanya memakai CSV benchmark yang sudah bersih.

Proyek ini ditempatkan pada konteks kimia medisn, database bioaktivitas, dan QSAR regresi. Target yang dianalisis adalah pIC₅₀ terhadap HER2/ERBB2 berdasarkan record IC₅₀ ChEMBL yang lolos filter. Pertanyaan inti penelitian adalah: Seberapa baik descriptor ligand 2D memprediksi pIC₅₀ HER2 setelah data ChEMBL difilter secara ketat untuk tipe aktivitas dan satuan yang seragam? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Diskusi menekankan bahwa IC50 bergantung assay. pIC50 yang sama dari assay berbeda tidak selalu identik secara biologis; provenance assay harus dipertahankan. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa baik descriptor ligand 2D memprediksi pIC50 HER2 setelah data ChEMBL difilter secara ketat untuk tipe aktivitas dan satuan yang seragam?
2. Bagaimana kualitas, distribusi, dan potensi bias pada Aktivitas ChEMBL terhadap target manusia ERBB2/HER2. Skrip mencari target, mengambil IC50 dengan unit nM dan relation =, lalu mengubahnya menjadi $pIC50 = 9 - \log_{10}(IC50 \text{ nM})$?
3. Bagaimana performa Random Forest Regressor, Extra Trees Regressor, Ridge jika dinilai menggunakan MAE, RMSE, R2, scatter experimental-predicted, audit jumlah assay dan jumlah senyawa unik?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Random Forest Regressor, Extra Trees Regressor, Ridge.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2, scatter experimental-predicted, audit jumlah assay dan jumlah senyawa unik.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Zdrazil, B., Felix, E., Hunter, F., et al. (2024). The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1), D1180-D1192. DOI: 10.1093/nar/gkad1004.

ChEMBL mengkurasi data struktur-aktivitas dari literatur dan sumber bioaktivitas. Proyek ini mengajarkan pembangunan dataset riset secara terprogram, bukan hanya memakai CSV benchmark yang sudah bersih. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah Aktivitas ChEMBL terhadap target manusia ERBB2/HER2. Skrip mencari target, mengambil IC50 dengan unit nM dan relation =, lalu mengubahnya menjadi $pIC50 = 9 - \log_{10}(IC50 \text{ nM})$. Fitur yang dipakai adalah descriptor RDKit dan Morgan fingerprint; baseline di bawah menggunakan descriptor 2D agar interpretasi mudah. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Heterogenitas assay, censoring relation, duplikasi pengukuran, garam, stereokimia, dan target construct dapat memengaruhi label. Dataset harus dilaporkan sebagai curated subset, bukan seluruh aktivitas HER2. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. Aktivitas ChEMBL terhadap target manusia ERBB2/HER2. Skrip mencari target, mengambil IC50 dengan unit nM dan relation =, lalu mengubahnya menjadi $pIC50 = 9 - \log_{10}(IC50 \text{ nM})$. Variabel target: pIC50 terhadap HER2/ERBB2 berdasarkan record IC50 ChEMBL yang lolos filter. Variabel prediktor: descriptor RDKit dan Morgan fingerprint; baseline di bawah menggunakan descriptor 2D agar interpretasi mudah.

Quality Control Data. Filter organism Homo sapiens, standard_type IC50, standard_units nM, standard_relation =, standard_value positif; canonicalize SMILES dan agregasi duplikat dengan median pIC50.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor RDKit dan Morgan fingerprint; baseline di bawah menggunakan descriptor 2D agar interpretasi mudah.

6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Random Forest Regressor, Extra Trees Regressor, Ridge.
8. Evaluasi menggunakan MAE, RMSE, R2, scatter experimental-predicted, audit jumlah assay dan jumlah senyawa unik.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.28. Jadwal kerja 14 hari Proyek 12

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: Aktivitas ChEMBL terhadap target manusia ERBB2/HER2. Skrip mencari target, mengambil IC50 dengan unit nM dan relation =, lalu mengubahnya menjadi $pIC50 = 9 - \log_{10}(IC50 \text{ nM})$.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembandingan utama. Model utama: Random Forest Regressor, Extra Trees Regressor, Ridge.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Diskusi menekankan bahwa IC50 bergantung assay. $pIC50$ yang sama dari assay berbeda tidak selalu identik secara biologis; provenance assay harus dipertahankan.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.14. Script Python lengkap Proyek 12

```
import numpy as np
import pandas as pd
from chembl_webresource_client.new_client import new_client
from rdkit import Chem
```

```

from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor, ExtraTreesRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

SEED = 42
# Cari target ERBB2/HER2 manusia secara terprogram.
targets = list(new_client.target.search("ERBB2"))
candidates = [t for t in targets if t.get("organism") == "Homo sapiens"]
if not candidates:
    raise RuntimeError("Target HER2/ERBB2 Homo sapiens tidak ditemukan.")
for i, t in enumerate(candidates[:10]):
    print(i, t.get("target_chembl_id"), t.get("pref_name"))
TARGET_ID = candidates[0]["target_chembl_id"]

acts = new_client.activity.filter(
    target_chembl_id=TARGET_ID,
    standard_type="IC50",
    standard_units="nM",
    standard_relation="=",
).only("canonical_smiles", "standard_value", "assay_chembl_id")

data = pd.DataFrame(list(acts))
data["standard_value"] = pd.to_numeric(data["standard_value"], errors="coerce")
data = data.dropna(subset=["canonical_smiles", "standard_value"])
data = data[data["standard_value"] > 0].copy()
data["pIC50"] = 9.0 - np.log10(data["standard_value"])
# Agregasi pengukuran senyawa yang sama dengan median.
data = data.groupby("canonical_smiles", as_index=False)["pIC50"].median()

def desc(smi):
    m = Chem.MolFromSmiles(smi)
    if m is None:
        return None
    return [
        Descriptors.MolWt(m), Descriptors.MolLogP(m), Descriptors.TPSA(m),
        Descriptors.NumHDonors(m), Descriptors.NumHAcceptors(m),
        Descriptors.NumRotatableBonds(m), Descriptors.RingCount(m),
        Descriptors.FractionCSP3(m),
    ]

rows = []
for smi, y in zip(data["canonical_smiles"], data["pIC50"]):
    x = desc(smi)
    if x is not None:
        rows.append(x + [float(y)])
arr = np.asarray(rows, dtype=float)
X, y = arr[:, :-1], arr[:, -1]
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)

for model in [
    RandomForestRegressor(n_estimators=500, random_state=SEED, n_jobs=-1),
    ExtraTreesRegressor(n_estimators=500, random_state=SEED, n_jobs=-1),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(model.__class__.__name__,
          "MAE", mean_absolute_error(yte, pred),
          "RMSE", mean_squared_error(yte, pred) ** 0.5,
          "R2", r2_score(yte, pred))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.43. Template tabel hasil aktual Proyek 12

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembandingan, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R², scatter experimental-predicted, audit jumlah assay dan jumlah senyawa unik. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Diskusi menekankan bahwa IC₅₀ bergantung assay. pIC₅₀ yang sama dari assay berbeda tidak selalu identik secara biologis; provenance assay harus dipertahankan. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Heterogenitas assay, censoring relation, duplikasi pengukuran, garam, stereokimia, dan target construct dapat memengaruhi label. Dataset harus dilaporkan sebagai curated subset, bukan seluruh aktivitas HER2.

- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Heterogenitas assay, censoring relation, duplikasi pengukuran, garam, stereokimia, dan target construct dapat memengaruhi label. Dataset harus dilaporkan sebagai curated subset, bukan seluruh aktivitas HER2. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Zdrazil et al. (2024), DOI 10.1093/nar/gkad1004

D.16 PROYEK 13: PREDIKSI HOMO-LUMO GAP MOLEKUL DARI DATASET QM9

Tabel D.14. Ringkasan rancangan Proyek 13

Komponen	Rancangan proyek
Bidang	kimia kuantum, surrogate modeling, dan molecular machine learning
Basis jurnal	Ramakrishnan, R., Dral, P. O., Rupp, M., & von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. Scientific Data, 1, 140022. DOI: 10.1038/sdata.2014.22.
Dataset	qm9.csv dari DeepChem. Untuk menjaga waktu komputasi dua minggu dan memori laptop, skrip mengambil sampel acak maksimum 12.000 struktur.
Target	gap HOMO-LUMO pada kolom gap sesuai distribusi dataset QM9
Fitur	descriptor 2D RDKit sebagai baseline surrogate sederhana
Model	Random Forest Regressor, Extra Trees Regressor, HistGradientBoosting opsional
Evaluasi	MAE, RMSE, R2, residual; pembahasan keterbatasan representasi 2D untuk sifat yang sensitif geometri elektronik
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi HOMO-LUMO Gap Molekul dari Dataset QM9

Latar Belakang

QM9 menyediakan struktur dan sifat kuantum untuk sekitar 134 ribu molekul organik kecil. Proyek S1 menggunakan subset dan descriptor 2D sebagai baseline, bukan mencoba menyamai model graph neural network state-of-the-art.

Proyek ini ditempatkan pada konteks kimia kuantum, surrogate modeling, dan molecular machine learning. Target yang dianalisis adalah gap HOMO-LUMO pada kolom gap sesuai distribusi dataset QM9. Pertanyaan inti penelitian adalah: Seberapa jauh descriptor molekul 2D dapat menangkap variasi HOMO-LUMO gap QM9, dan error seperti apa yang muncul ketika informasi geometri 3D diabaikan? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Hubungkan gap dengan conjugation, ukuran sistem pi, heteroatom, struktur cincin, dan keterbatasan descriptor global. Tekankan bahwa korelasi ML tidak menggantikan orbital molecular calculation. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa jauh descriptor molekul 2D dapat menangkap variasi HOMO-LUMO gap QM9, dan error seperti apa yang muncul ketika informasi geometri 3D diabaikan?
2. Bagaimana kualitas, distribusi, dan potensi bias pada qm9.csv dari DeepChem. Untuk menjaga waktu komputasi dua minggu dan memori laptop, skrip mengambil sampel acak maksimum 12.000 struktur.?
3. Bagaimana performa Random Forest Regressor, Extra Trees Regressor, HistGradientBoosting opsional jika dinilai menggunakan MAE, RMSE, R2, residual; pembahasan keterbatasan representasi 2D untuk sifat yang sensitif geometri elektronik?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Random Forest Regressor, Extra Trees Regressor, HistGradientBoosting opsional.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2, residual; pembahasan keterbatasan representasi 2D untuk sifat yang sensitif geometri elektronik.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Ramakrishnan, R., Dral, P. O., Rupp, M., & von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 140022. DOI: 10.1038/sdata.2014.22.

QM9 menyediakan struktur dan sifat kuantum untuk sekitar 134 ribu molekul organik kecil. Proyek S1 menggunakan subset dan descriptor 2D sebagai baseline, bukan mencoba menyamai model graph neural network state-of-the-art. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah qm9.csv dari DeepChem. Untuk menjaga waktu komputasi dua minggu dan memori laptop, skrip mengambil sampel acak maksimum 12.000 struktur. Fitur yang dipakai adalah descriptor 2D RDKit sebagai baseline surrogate sederhana. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. QM9 terbatas pada molekul kecil dengan unsur tertentu dan geometri yang dihitung. Subset praktikum serta fitur 2D sengaja menyederhanakan masalah dan tidak menjadi benchmark state-of-the-art. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility

check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. qm9.csv dari DeepChem. Untuk menjaga waktu komputasi dua minggu dan memori laptop, skrip mengambil sampel acak maksimum 12.000 struktur. Variabel target: gap HOMO-LUMO pada kolom gap sesuai distribusi dataset QM9. Variabel prediktor: descriptor 2D RDKit sebagai baseline surrogate sederhana.

Quality Control Data. Periksa nama target gap, SMILES invalid, sampling seed, target distribution, dan jangan mencampur target turunan yang berkorelasi langsung sebagai fitur.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: descriptor 2D RDKit sebagai baseline surrogate sederhana.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Random Forest Regressor, Extra Trees Regressor, HistGradientBoosting opsional.
8. Evaluasi menggunakan MAE, RMSE, R2, residual; pembahasan keterbatasan representasi 2D untuk sifat yang sensitif geometri elektronik.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.29. Jadwal kerja 14 hari Proyek 13

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: qm9.csv dari DeepChem. Untuk menjaga waktu komputasi dua minggu dan memori laptop, skrip mengambil sampel acak maksimum 12.000 struktur.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standarisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.

Hari	Kegiatan dan output
Hari 7	Bangun model pembandingan utama. Model utama: Random Forest Regressor, Extra Trees Regressor, HistGradientBoosting opsional.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Hubungkan gap dengan conjugation, ukuran sistem pi, heteroatom, struktur cincin, dan keterbatasan descriptor global. Tekankan bahwa korelasi ML tidak menggantikan orbital molecular calculation.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.15. Script Python lengkap Proyek 13

```
import numpy as np
import pandas as pd
from rdkit import Chem
from rdkit.Chem import Descriptors
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor, ExtraTreesRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

URL = "https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/qm9.csv"
SEED = 42
MAX_N = 12000

df = pd.read_csv(URL)
if len(df) > MAX_N:
    df = df.sample(MAX_N, random_state=SEED).reset_index(drop=True)
if "gap" not in df.columns:
    raise ValueError(f"Kolom gap tidak ditemukan: {list(df.columns)}")

def desc(smi):
    m = Chem.MolFromSmiles(str(smi))
    if m is None:
        return None
    return [
        Descriptors.MolWt(m), Descriptors.MolLogP(m), Descriptors.TPSA(m),
        Descriptors.NumHDonors(m), Descriptors.NumHAcceptors(m),
        Descriptors.NumRotatableBonds(m), Descriptors.RingCount(m),
        Descriptors.FractionCSP3(m), Descriptors.NumAromaticRings(m),
    ]

rows = []
for smi, val in zip(df["smiles"], pd.to_numeric(df["gap"], errors="coerce")):
    x = desc(smi)
    if x is not None and pd.notna(val):
        rows.append(x + [float(val)])
a = np.asarray(rows, dtype=float)
```

```

X, y = a[:, :-1], a[:, -1]
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)
for model in [
    RandomForestRegressor(n_estimators=400, random_state=SEED, n_jobs=-1),
    ExtraTreesRegressor(n_estimators=400, random_state=SEED, n_jobs=-1),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(model.__class__.__name__,
          "MAE", mean_absolute_error(yte, pred),
          "RMSE", mean_squared_error(yte, pred) ** 0.5,
          "R2", r2_score(yte, pred))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.44. Template tabel hasil aktual Proyek 13

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R2, residual; pembahasan keterbatasan representasi 2D untuk sifat yang sensitif geometri elektronik. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Hubungkan gap dengan conjugation, ukuran sistem pi, heteroatom, struktur cincin, dan keterbatasan descriptor global. Tekankan bahwa korelasi ML tidak menggantikan orbital molecular calculation. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul

pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: QM9 terbatas pada molekul kecil dengan unsur tertentu dan geometri yang dihitung. Subset praktikum serta fitur 2D sengaja menyederhanakan masalah dan tidak menjadi benchmark state-of-the-art.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembanding.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: QM9 terbatas pada molekul kecil dengan unsur tertentu dan geometri yang dihitung. Subset praktikum serta fitur 2D sengaja menyederhanakan masalah dan tidak menjadi benchmark state-of-the-art. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Ramakrishnan et al. (2014), DOI 10.1038/sdata.2014.22

D.17 PROYEK 14: PREDIKSI YIELD REAKSI BUCHWALD-HARTWIG C-N CROSS-COUPPLING

Tabel D.15. Ringkasan rancangan Proyek 14

Komponen	Rancangan proyek
Bidang	kimia organik sintesis, high-throughput experimentation, dan reaction ML

Komponen	Rancangan proyek
Basis jurnal	Ahneman, D. T., Estrada, J. G., Lin, S., Dreher, S. D., & Doyle, A. G. (2018). Predicting reaction performance in C-N cross-coupling using machine learning. <i>Science</i> , 360(6385), 186-190. DOI: 10.1126/science.aar5169.
Dataset	File OH_trainingset1.csv dan OH_testset1.csv pada repositori resmi doylelab/rxnpredict. Setiap baris berisi one-hot identity komponen reaksi dan kolom yield.
Target	persen yield reaksi Buchwald-Hartwig
Fitur	one-hot additive, aryl halide, base, catalyst/ligand sesuai file publik
Model	Random Forest Regressor dan Ridge sebagai baseline
Evaluasi	MAE, RMSE, R2 pada test set yang sudah dipisahkan; feature importance sebagai petunjuk komponen berpengaruh
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Yield Reaksi Buchwald-Hartwig C-N Cross-Coupling

Latar Belakang

Ahneman dan kolega menunjukkan bahwa machine learning dapat memprediksi yield reaksi Buchwald-Hartwig dari ruang kondisi kimia multidimensi. Repositori penulis menyediakan data/artefak sehingga mahasiswa dapat memakai one-hot reaction conditions tanpa menghitung descriptor kuantum yang berat.

Proyek ini ditempatkan pada konteks kimia organik sintesis, high-throughput experimentation, dan reaction ML. Target yang dianalisis adalah persen yield reaksi Buchwald-Hartwig. Pertanyaan inti penelitian adalah: Apakah model nonlinear mampu menangkap interaksi antara komponen reaksi lebih baik daripada model linear pada one-hot reaction space? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Pembahasan harus menghubungkan pengaruh ligand/base/additive/aryl halide dengan reaktivitas dan poisoning secara kualitatif, tetapi feature importance tidak boleh disamakan dengan mekanisme elementer. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Apakah model nonlinear mampu menangkap interaksi antara komponen reaksi lebih baik daripada model linear pada one-hot reaction space?
2. Bagaimana kualitas, distribusi, dan potensi bias pada File OH_trainingset1.csv dan OH_testset1.csv pada repositori resmi doylelab/rxnpredict. Setiap baris berisi one-hot identity komponen reaksi dan kolom yield.?
3. Bagaimana performa Random Forest Regressor dan Ridge sebagai baseline jika dinilai menggunakan MAE, RMSE, R2 pada test set yang sudah dipisahkan; feature importance sebagai petunjuk komponen berpengaruh?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Random Forest Regressor dan Ridge sebagai baseline.
3. Mengevaluasi performa menggunakan MAE, RMSE, R2 pada test set yang sudah dipisahkan; feature importance sebagai petunjuk komponen berpengaruh.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Ahneman, D. T., Estrada, J. G., Lin, S., Dreher, S. D., & Doyle, A. G. (2018). Predicting reaction performance in C-N cross-coupling using machine learning. *Science*, 360(6385), 186-190. DOI: 10.1126/science.aar5169.

Ahneman dan kolega menunjukkan bahwa machine learning dapat memprediksi yield reaksi Buchwald-Hartwig dari ruang kondisi kimia multidimensi. Repositori penulis menyediakan data/artefak sehingga mahasiswa dapat memakai one-hot reaction conditions tanpa menghitung descriptor kuantum yang berat. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah File OH_trainingset1.csv dan OH_testset1.csv pada repositori resmi doylelab/rxnpredict. Setiap baris berisi one-hot identity komponen reaksi dan kolom yield. Fitur yang dipakai adalah one-hot additive, aryl halide, base, catalyst/ligand sesuai file publik. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. One-hot encoding tidak mengekstrapolasi secara alami ke komponen yang benar-benar baru. Test set sumber hanya salah satu skenario. Paper asli memakai descriptor kimia yang lebih kaya dan evaluasi lebih luas. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. File OH_trainingset1.csv dan OH_testset1.csv pada repositori resmi doylelab/rxnpredict. Setiap baris berisi one-hot identity komponen reaksi dan kolom yield. Variabel target: persen yield reaksi Buchwald-Hartwig. Variabel prediktor: one-hot additive, aryl halide, base, catalyst/ligand sesuai file publik.

Quality Control Data. Pastikan kolom yield numeric, tidak ada NA, kolom train dan test sama, serta tidak memasukkan yield sebagai fitur.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: one-hot additive, aryl halide, base, catalyst/ligand sesuai file publik.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Random Forest Regressor dan Ridge sebagai baseline.
8. Evaluasi menggunakan MAE, RMSE, R2 pada test set yang sudah dipisahkan; feature importance sebagai petunjuk komponen berpengaruh.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.
10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.30. Jadwal kerja 14 hari Proyek 14

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: File OH_trainingset1.csv dan OH_testset1.csv pada repositori resmi doylelab/rxnpredict. Setiap baris berisi one-hot identity komponen reaksi dan kolom yield.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembandingan utama. Model utama: Random Forest Regressor dan Ridge sebagai baseline.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Pembahasan harus menghubungkan pengaruh ligand/base/additive/aryl halide dengan reaktivitas dan poisoning secara kualitatif, tetapi feature importance tidak boleh disamakan dengan mekanisme elementer.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.16. Script Python lengkap Proyek 14

```
import pandas as pd
from sklearn.linear_model import Ridge
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

TRAIN = (
    "https://raw.githubusercontent.com/doylelab/rxnpredict/master/"
    "Response/Activity_Ranking/Onehot/OH_trainingset1.csv"
)
TEST = (
    "https://raw.githubusercontent.com/doylelab/rxnpredict/master/"
    "Response/Activity_Ranking/Onehot/OH_testset1.csv"
)
SEED = 42

train = pd.read_csv(TRAIN)
test = pd.read_csv(TEST)
train.columns = [c.strip() for c in train.columns]
```

```

test.columns = [c.strip() for c in test.columns]

if "yield" not in train.columns or "yield" not in test.columns:
    raise ValueError("Kolom yield tidak ditemukan.")
Xtr = train.drop(columns="yield").apply(pd.to_numeric, errors="coerce").fillna(0)
ytr = pd.to_numeric(train["yield"], errors="coerce")
Xte = test.drop(columns="yield").apply(pd.to_numeric, errors="coerce").fillna(0)
yte = pd.to_numeric(test["yield"], errors="coerce")
Xte = Xte.reindex(columns=Xtr.columns, fill_value=0)

for model in [
    Ridge(alpha=10.0),
    RandomForestRegressor(n_estimators=600, random_state=SEED, n_jobs=-1,
                          min_samples_leaf=2),
]:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(model.__class__.__name__,
          "MAE", mean_absolute_error(yte, pred),
          "RMSE", mean_squared_error(yte, pred) ** 0.5,
          "R2", r2_score(yte, pred))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.45. Template tabel hasil aktual Proyek 14

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembanding, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE, RMSE, R2 pada test set yang sudah dipisahkan; feature importance sebagai petunjuk komponen berpengaruh. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Pembahasan harus menghubungkan pengaruh ligand/base/additive/aryl halide dengan reaktivitas dan poisoning secara kualitatif, tetapi feature importance tidak boleh disamakan dengan mekanisme elementer. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: One-hot encoding tidak mengekstrapolasi secara alami ke komponen yang benar-benar baru. Test set sumber hanya salah satu skenario. Paper asli memakai descriptor kimia yang lebih kaya dan evaluasi lebih luas.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembandingan.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: One-hot encoding tidak mengekstrapolasi secara alami ke komponen yang benar-benar baru. Test set sumber hanya salah satu skenario. Paper asli memakai descriptor kimia yang lebih kaya dan evaluasi lebih luas. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Ahneman et al. (2018), DOI 10.1126/science.aar5169

D.18 PROYEK 15: PREDIKSI BAND GAP EKSPERIMENTAL MATERIAL ANORGANIK

Tabel D.16. Ringkasan rancangan Proyek 15

Komponen	Rancangan proyek
Bidang	kimia material, materials informatics, dan composition-based ML
Basis jurnal	Zhuo, Y., Mansouri Tehrani, A., & Brgoch, J. (2018). Predicting the Band Gaps of Inorganic Solids by Machine Learning. The Journal of Physical Chemistry Letters, 9(7), 1668-1673. DOI: 10.1021/acs.jpclett.8b00124. Dataset juga menjadi bagian dari Matbench yang didokumentasikan oleh Dunn et al. (2020), DOI: 10.1038/s41524-020-00406-3.
Dataset	matbench_expt_gap melalui matminer.datasets.load_dataset. Dataset Matbench berisi ribuan komposisi dan band gap eksperimen dalam eV.
Target	experimental band gap dalam eV
Fitur	Magpie elemental composition statistics melalui matminer.ElementProperty
Model	Random Forest Regressor, Extra Trees Regressor, Gradient Boosting
Evaluasi	MAE (eV), RMSE, R2, residual menurut rentang gap, dan error pada komposisi ekstrem
Batas waktu	14 hari; laptop CPU standar; tidak memerlukan GPU

Judul

Prediksi Band Gap Eksperimental Material Anorganik

Latar Belakang

Studi Zhuo menunjukkan prediksi band gap material anorganik dari komposisi. Matbench menyediakan versi dataset eksperimental yang dibersihkan sehingga proyek dapat dilakukan hanya dengan formula komposisi tanpa simulasi DFT.

Proyek ini ditempatkan pada konteks kimia material, materials informatics, dan composition-based ML. Target yang dianalisis adalah experimental band gap dalam eV. Pertanyaan inti penelitian adalah: Seberapa baik statistik komposisi elemental memprediksi band gap eksperimen tanpa informasi struktur kristal? Pemilihan masalah tersebut sengaja dibatasi agar seluruh siklus data sampai laporan dapat diselesaikan dalam dua pekan tanpa mengorbankan prinsip quality control. Mahasiswa tidak perlu membangun arsitektur deep learning yang mahal; baseline klasik yang diaudit dengan baik lebih tepat untuk tujuan pembelajaran metodologi.

Dari sisi kimia, Diskusi mencakup komposisi, electronegativity, orbital character, unsur berat, dan alasan mengapa struktur kristal serta fase tetap penting meskipun model komposisi memberikan baseline berguna. Machine learning digunakan sebagai alat untuk menguji pola empiris dan membangun prediksi, sedangkan penjelasan mekanistik harus tetap ditopang teori kimia atau eksperimen. Dengan pembatasan ini, mahasiswa dapat membedakan prediksi yang baik secara statistik dari kesimpulan kimia yang benar-benar didukung bukti.

Rumusan Masalah

1. Seberapa baik statistik komposisi elemental memprediksi band gap eksperimen tanpa informasi struktur kristal?
2. Bagaimana kualitas, distribusi, dan potensi bias pada `matbench_expt_gap` melalui `matminer.datasets.load_dataset`. Dataset Matbench berisi ribuan komposisi dan band gap eksperimen dalam eV.?
3. Bagaimana performa Random Forest Regressor, Extra Trees Regressor, Gradient Boosting jika dinilai menggunakan MAE (eV), RMSE, R2, residual menurut rentang gap, dan error pada komposisi ekstrem?
4. Apa batas applicability domain model dan kesalahan apa yang paling relevan secara kimia?

Tujuan Riset

1. Membangun dataset analisis yang dapat ditelusuri dari sumber jurnal/data publik sampai tabel siap model.
2. Membangun dan membandingkan Random Forest Regressor, Extra Trees Regressor, Gradient Boosting.
3. Mengevaluasi performa menggunakan MAE (eV), RMSE, R2, residual menurut rentang gap, dan error pada komposisi ekstrem.
4. Menafsirkan hasil secara kimia, mengidentifikasi keterbatasan, dan menyusun laporan yang dapat direproduksi.

Telaah Pustaka

Zhuo, Y., Mansouri Tehrani, A., & Brgoch, J. (2018). Predicting the Band Gaps of Inorganic Solids by Machine Learning. *The Journal of Physical Chemistry Letters*, 9(7), 1668-1673. DOI: 10.1021/acs.jpcclett.8b00124. Dataset juga menjadi bagian dari Matbench yang didokumentasikan oleh Dunn et al. (2020), DOI: 10.1038/s41524-020-00406-3.

Studi Zhuo menunjukkan prediksi band gap material anorganik dari komposisi. Matbench menyediakan versi dataset eksperimental yang dibersihkan sehingga proyek dapat dilakukan hanya dengan formula komposisi tanpa simulasi DFT. Dalam proposal mahasiswa, artikel tersebut harus dibaca minimal pada bagian abstract, methods/data, results, dan limitations agar definisi endpoint tidak berubah ketika diadaptasi ke kode Python.

Dataset yang digunakan pada proyek ini adalah `matbench_expt_gap` melalui `matminer.datasets.load_dataset`. Dataset Matbench berisi ribuan komposisi dan band gap eksperimen dalam eV. Fitur yang dipakai adalah Magpie elemental composition statistics melalui `matminer.ElementProperty`. Representasi dipilih karena dapat dijalankan dengan library open-source dan cukup untuk membangun

baseline yang interpretable. Model yang lebih kompleks boleh dibahas sebagai pengembangan, tetapi tidak menjadi syarat keberhasilan proyek dua pekan.

Aspek paling penting dalam membaca literatur proyek ini adalah keterbatasannya. Satu formula dapat mempunyai polymorph dengan band gap berbeda. Dataset Matbench telah dideduplicate menurut aturan tertentu. Model berbasis komposisi tidak menangkap struktur, cacat, suhu, dan kondisi pengukuran. Karena itu, telaah pustaka harus membedakan dengan jelas hasil paper sumber, hasil run mahasiswa, dan interpretasi baru yang muncul dari analisis mahasiswa sendiri.

Metode Penelitian

Desain Penelitian. Penelitian bersifat komputasional observasional menggunakan data sekunder publik. Alur utama terdiri dari akuisisi data, audit, cleaning, featurisasi, split, training, evaluasi, error analysis, interpretasi, dan reproducibility check. Seluruh keputusan preprocessing harus ditetapkan sebelum melihat performa test set.

Sumber Data dan Variabel. `matbench_expt_gap` melalui `matminer.datasets.load_dataset`. Dataset Matbench berisi ribuan komposisi dan band gap eksperimen dalam eV. Variabel target: experimental band gap dalam eV. Variabel prediktor: Magpie elemental composition statistics melalui `matminer.ElementProperty`.

Quality Control Data. Gunakan dataset Matbench resmi, buang baris featurization gagal, audit distribusi target, dan dokumentasikan versi `matminer/pymatgen`.

Alur Penelitian

1. Baca artikel utama dan tetapkan definisi endpoint serta satuan.
2. Unduh data atau jalankan query API, lalu simpan salinan mentah dan metadata akses.
3. Audit jumlah record, kolom, missing value, distribusi target/kelas, dan duplikasi.
4. Lakukan cleaning melalui kode; jangan mengedit data mentah secara manual.
5. Bangun fitur: Magpie elemental composition statistics melalui `matminer.ElementProperty`.
6. Pisahkan train dan test sebelum fitting transformer atau melakukan seleksi berbasis target.
7. Latih model: Random Forest Regressor, Extra Trees Regressor, Gradient Boosting.
8. Evaluasi menggunakan MAE (eV), RMSE, R2, residual menurut rentang gap, dan error pada komposisi ekstrem.
9. Lakukan error analysis dan hubungkan pola error dengan ruang kimia atau karakter target.

10. Simpan output, seed, versi library, serta simpulkan hanya hal yang didukung data.

Rencana Kerja 14 Hari

Tabel D.31. Jadwal kerja 14 hari Proyek 15

Hari	Kegiatan dan output
Hari 1	Pahami paper sumber, rumuskan pertanyaan riset, buat folder proyek.
Hari 2	Unduh/ambil data, simpan data_raw, catat DOI/URL/tanggal akses. Sumber proyek: matbench_expt_gap melalui matminer.datasets.load_dataset. Dataset Matbench berisi ribuan komposisi dan band gap eksperimen dalam eV.
Hari 3	Audit struktur data: ukuran, tipe kolom, missing value, duplikasi, unit, label.
Hari 4	Cleaning dan standardisasi; simpan data_processed serta log jumlah record.
Hari 5	Exploratory data analysis dan visualisasi target/kelas.
Hari 6	Bangun baseline model paling sederhana dan pipeline preprocessing.
Hari 7	Bangun model pembanding utama. Model utama: Random Forest Regressor, Extra Trees Regressor, Gradient Boosting.
Hari 8	Evaluasi train/test dan cross-validation yang sesuai.
Hari 9	Analisis error, residual/confusion matrix, dan sampel gagal.
Hari 10	Tuning ringan atau analisis sensitivitas; jangan menyentuh test berulang.
Hari 11	Interpretasi kimia, feature importance/koeffisien bila relevan. Fokus kimia: Diskusi mencakup komposisi, electronegativity, orbital character, unsur berat, dan alasan mengapa struktur kristal serta fase tetap penting meskipun model komposisi memberikan baseline berguna.
Hari 12	Ulang run dari environment bersih; simpan seed dan versi library.
Hari 13	Tulis Hasil, Pembahasan, Diskusi, keterbatasan, dan Kesimpulan.
Hari 14	Finalisasi laporan, periksa sitasi, folder artefak, dan reproducibility checklist.

Coding Machine Learning Python yang Digunakan

Skrip berikut merupakan baseline utuh yang dapat dijalankan atau disesuaikan. Sebelum menjalankan, gunakan satu perintah instalasi pada Bagian D.2. Simpan skrip sebagai file .py sesuai nomor proyek. Jika layanan daring berubah, jangan mengganti sumber secara diam-diam; catat perubahan URL/API pada laporan.

Kode D.17. Script Python lengkap Proyek 15

```
from matminer.datasets import load_dataset
from matminer.featurizers.composition import ElementProperty
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor, ExtraTreesRegressor
from sklearn.impute import SimpleImputer
from sklearn.pipeline import Pipeline
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

SEED = 42

df = load_dataset("matbench_expt_gap")
featurizer = ElementProperty.from_preset("magpie")
# composition pada dataset Matbench sudah berupa objek Composition.
df_feat = featurizer.featurize_dataframe(
    df.copy(), col_id="composition", ignore_errors=True, pbar=True
```

```

)
feature_cols = featurizer.feature_labels()
X = df_feat[feature_cols]
y = df_feat["gap expt"].astype(float)

Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2, random_state=SEED)
models = [
    Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("model", RandomForestRegressor(n_estimators=500, random_state=SEED,
n_jobs=-1))
    ]),
    Pipeline([
        ("impute", SimpleImputer(strategy="median")),
        ("model", ExtraTreesRegressor(n_estimators=500, random_state=SEED,
n_jobs=-1))
    ]),
]
for model in models:
    model.fit(Xtr, ytr)
    pred = model.predict(Xte)
    print(model.named_steps["model"].__class__.__name__,
          "MAE", mean_absolute_error(yte, pred),
          "RMSE", mean_squared_error(yte, pred) ** 0.5,
          "R2", r2_score(yte, pred))

```

Hasil Riset dan Pembahasan

Bagian ini diisi dengan hasil aktual setelah skrip dijalankan. Nilai contoh tidak diberikan agar tidak terjadi fabrikasi hasil. Salin metrik dari output terminal/notebook, sertakan tanggal run, ukuran data akhir, random seed, dan versi library. Bila hasil berbeda dari paper sumber, perbedaan tersebut tidak otomatis berarti salah; evaluasi terlebih dahulu perbedaan data, split, fitur, algoritma, dan versi software.

Tabel D.46. Template tabel hasil aktual Proyek 15

Item hasil	Nilai aktual mahasiswa	Interpretasi yang harus ditulis
Jumlah data final	[isi setelah run]	jelaskan record yang hilang/ditolak dan alasannya
Distribusi target/kelas	[isi setelah run]	nilai apakah distribusi seimbang, skewed, atau memiliki outlier
Model baseline	[isi metrik]	jadikan titik pembandingan, bukan sekadar formalitas
Model utama	[isi metrik]	bandingkan pada test yang sama dan metrik yang sama
Error utama	[isi]	identifikasi kelompok kimia/sampel yang sulit
Reproducibility	[versi/seed]	pastikan run ulang memberi hasil konsisten dalam toleransi wajar

Pembahasan kuantitatif harus mengacu pada MAE (eV), RMSE, R2, residual menurut rentang gap, dan error pada komposisi ekstrem. Untuk regresi, MAE/RMSE perlu diterjemahkan ke satuan atau skala kimia target. Untuk

klasifikasi, confusion matrix dan metrik kelas positif harus dibaca bersama prevalensi. Jangan memilih satu metrik yang paling tinggi lalu mengabaikan metrik lain yang menunjukkan kelemahan model.

Pembahasan kimia proyek ini berpusat pada hal berikut: Diskusi mencakup komposisi, electronegativity, orbital character, unsur berat, dan alasan mengapa struktur kristal serta fase tetap penting meskipun model komposisi memberikan baseline berguna. Gunakan grafik residual, predicted-vs-actual, feature importance, atau contoh true/false prediction sebagai bukti. Bila pola tidak muncul pada data aktual, tuliskan bahwa hipotesis tidak didukung, bukan memaksakan narasi dari literatur.

Diskusi

- Bandingkan hasil mahasiswa dengan ekspektasi dari jurnal sumber tanpa menyalin klaim. Jelaskan perbedaan desain, dataset, split, dan representasi.
- Bahas keterbatasan utama: Satu formula dapat mempunyai polymorph dengan band gap berbeda. Dataset Matbench telah dideduplicate menurut aturan tertentu. Model berbasis komposisi tidak menangkap struktur, cacat, suhu, dan kondisi pengukuran.
- Identifikasi kemungkinan data leakage, domain shift, class imbalance, atau extrapolation yang dapat mengubah kesimpulan.
- Jelaskan eksperimen lanjutan paling realistis: external test set, scaffold/group split, descriptor tambahan, atau validasi eksperimen, tanpa mengklaim hasil yang belum dilakukan.
- Nyatakan dengan eksplisit bagian mana yang merupakan fakta data, hasil model, interpretasi kimia, dan hipotesis.

Kesimpulan

Kesimpulan akhir harus menjawab rumusan masalah dengan angka hasil aktual. Gunakan pola: model terbaik pada split yang ditetapkan menghasilkan metrik utama tertentu; performa tersebut dibandingkan dengan baseline; error paling besar ditemukan pada kelompok tertentu; dan interpretasi kimia dibatasi oleh cakupan data. Hindari frasa "model sangat akurat" tanpa kriteria atau pembanding.

Untuk proyek ini, kesimpulan juga wajib menyatakan batas penggunaan: Satu formula dapat mempunyai polymorph dengan band gap berbeda. Dataset Matbench telah dideduplicate menurut aturan tertentu. Model berbasis komposisi tidak menangkap struktur, cacat, suhu, dan kondisi pengukuran. Dengan demikian laporan tetap proporsional terhadap bukti dan dapat dipakai sebagai portofolio riset komputasi tingkat S1.

Referensi Inti Proyek

- Zhuo et al. (2018), DOI 10.1021/acs.jpcclett.8b00124
- Dunn et al. (2020), DOI 10.1038/s41524-020-00406-3

D.19 Rubrik Kendali Mutu 15 Proyek

Tabel D.47. Rubrik kendali mutu proyek Informatika Kimia

Aspek	Kriteria lulus minimum	Kesalahan yang harus dihindari
Provenance	DOI/URL/tanggal akses dan data_raw tersimpan	dataset tanpa sumber
Cleaning	semua transformasi lewat kode dan jumlah record dilaporkan	menghapus outlier tanpa kriteria
Split	train/test ditentukan sebelum tuning	test dipakai berulang untuk memilih model
Pipeline	scaler/imputer fit hanya pada training	preprocessing seluruh data sebelum split
Metrik	sesuai regresi/klasifikasi dan domain	hanya accuracy atau R2 tanpa error analysis
Interpretasi kimia	berbasis output dan teori yang relevan	feature importance dianggap sebab mekanistik
Reproducibility	seed, versi, script, dan output tersedia	hasil hanya berupa screenshot
Kesimpulan	menjawab pertanyaan dan menyebut keterbatasan	klaim lebih luas daripada domain data

D.20 Daftar Referensi Jurnal Utama Lampiran D

1. Delaney, J. S. (2004). ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *Journal of Chemical Information and Computer Sciences*, 44(3), 1000-1005. <https://doi.org/10.1021/ci034243x>
2. Sorkun, M. C., Khetan, A., & Er, S. (2019). AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Scientific Data*, 6, 143. <https://doi.org/10.1038/s41597-019-0151-1>
3. Mobley, D. L., & Guthrie, J. P. (2014). FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design*, 28, 711-720. <https://doi.org/10.1007/s10822-014-9747-x>
4. Martins, I. F., Teixeira, A. L., Pinheiro, L., & Falcao, A. O. (2012). A Bayesian Approach to in Silico Blood-Brain Barrier Penetration Modeling. *Journal of Chemical Information and Modeling*, 52(6), 1686-1697. <https://doi.org/10.1021/ci300124c>
5. Hansen, K., Mika, S., Schroeter, T., et al. (2009). Benchmark Data Set for in Silico Prediction of Ames Mutagenicity. *Journal of Chemical Information and Modeling*, 49(9), 2077-2081. <https://doi.org/10.1021/ci900161g>

6. Mansouri, K., Ringsted, T., Ballabio, D., Todeschini, R., & Consonni, V. (2013). Quantitative Structure-Activity Relationship Models for Ready Biodegradability of Chemicals. *Journal of Chemical Information and Modeling*, 53(4), 867-878. <https://doi.org/10.1021/ci4000213>
7. Cassotti, M., Ballabio, D., Todeschini, R., & Consonni, V. (2015). A similarity-based QSAR model for predicting acute toxicity towards the fathead minnow (*Pimephales promelas*). *SAR and QSAR in Environmental Research*, 26(3), 217-243. <https://doi.org/10.1080/1062936X.2015.1018938>
8. Mayr, A., Klambauer, G., Unterthiner, T., & Hochreiter, S. (2016). DeepTox: Toxicity Prediction using Deep Learning. *Frontiers in Environmental Science*, 3, 80. <https://doi.org/10.3389/fenvs.2015.00080>
9. Wu, Z., Ramsundar, B., Feinberg, E. N., et al. (2018). MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9, 513-530. <https://doi.org/10.1039/C7SC02664A>
10. Subramanian, G., Ramsundar, B., Pande, V., & Denny, R. A. (2016). Computational Modeling of beta-Secretase 1 (BACE-1) Inhibitors Using Ligand Based Approaches. *Journal of Chemical Information and Modeling*, 56(10), 1936-1949. <https://doi.org/10.1021/acs.jcim.6b00290>
11. Zdrazil, B., Felix, E., Hunter, F., et al. (2024). The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1), D1180-D1192. <https://doi.org/10.1093/nar/gkad1004>
12. Ramakrishnan, R., Dral, P. O., Rupp, M., & von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 140022. <https://doi.org/10.1038/sdata.2014.22>
13. Ahneman, D. T., Estrada, J. G., Lin, S., Dreher, S. D., & Doyle, A. G. (2018). Predicting reaction performance in C-N cross-coupling using machine learning. *Science*, 360(6385), 186-190. <https://doi.org/10.1126/science.aar5169>
14. Zhuo, Y., Mansouri Tehrani, A., & Brgoch, J. (2018). Predicting the Band Gaps of Inorganic Solids by Machine Learning. *The Journal of Physical Chemistry Letters*, 9(7), 1668-1673. <https://doi.org/10.1021/acs.jpcllett.8b00124>
15. Dunn, A., Wang, Q., Ganose, A., Dopp, D., & Jain, A. (2020). Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials*, 6, 138. <https://doi.org/10.1038/s41524-020-00406-3>

GLOSARIUM

A

- Accuracy — proporsi prediksi klasifikasi yang benar dari seluruh sampel.
- Applicability domain — wilayah chemical space tempat model dianggap memiliki dukungan data training yang memadai.
- Assay — prosedur eksperimen untuk mengukur aktivitas atau respons biologis/kimia.

B

- Baseline — model sederhana yang menjadi titik pembanding sebelum meningkatkan kompleksitas.
- BBBP — dataset penetrasi blood-brain barrier yang umum digunakan sebagai benchmark klasifikasi molekuler.

C

- Canonical SMILES — bentuk SMILES yang dipilih secara deterministik oleh suatu toolkit untuk struktur tertentu.
- Chemical space — ruang multidimensi yang merepresentasikan variasi struktur dan/atau sifat molekul.
- Cross-validation — evaluasi berulang dengan membagi data menjadi beberapa fold.

D

- Data leakage — masuknya informasi yang tidak semestinya dari validation/test atau target ke proses training.
- Descriptor — besaran numerik yang merangkum aspek struktur atau sifat molekul.

E

- ESOL — dataset/model klasik untuk estimasi kelarutan air dari struktur molekul.
- External test set — data uji independen yang tidak dipakai dalam pengembangan model.

F

- F1-score — harmonic mean precision dan recall.
- Feature — variabel masukan yang dipakai model.
- Fingerprint — representasi vektor pola/substruktur molekul.

G

- GridSearchCV — pencarian kombinasi hyperparameter dengan cross-validation.
- Group split — split yang memastikan anggota satu kelompok tidak tersebar di train dan test.

H

- HBA — hydrogen-bond acceptor.
- HBD — hydrogen-bond donor.
- Hyperparameter — parameter konfigurasi model yang tidak dipelajari langsung dari fit data.

I

- Imputation — pengisian nilai fitur yang hilang berdasarkan aturan yang ditetapkan.
- Inference — penggunaan model terlatih untuk menghasilkan prediksi baru.

J

- Jupyter Notebook — lingkungan interaktif yang menggabungkan kode, output, dan narasi.

K

- K-Means — algoritma clustering yang meminimalkan jarak sampel terhadap centroid cluster.
- KNN — K-Nearest Neighbors, metode prediksi berdasarkan tetangga terdekat.

L

- Label — target kategori pada klasifikasi.
- Lipofilisitas — kecenderungan relatif suatu zat terhadap lingkungan nonpolar dibandingkan polar; logP/logD adalah ukuran terkait.

M

- MAE — mean absolute error.
- Morgan fingerprint — circular fingerprint yang merepresentasikan lingkungan atom.
- Model — fungsi terparameterisasi yang memetakan fitur ke prediksi.

N

- Normalization — transformasi skala data ke rentang atau norma tertentu; berbeda dari standardization.
- NumPy — library array numerik Python.

O

- Outlier — observasi yang jauh dari pola mayoritas; perlu investigasi sebelum keputusan penghapusan.
- Overfitting — model terlalu menyesuaikan data training dan buruk pada data baru.

P

- Pandas — library Python untuk manipulasi data tabular.
- PCA — principal component analysis.
- Precision — $TP/(TP+FP)$.
- Provenance — rekam jejak asal dan transformasi data.

Q

- QSAR — quantitative structure–activity relationship.
- QM9 — benchmark molekuler untuk berbagai properti kimia kuantum.

R

- R^2 — koefisien determinasi pada regresi.
- Recall — $TP/(TP+FN)$.
- Reproducibility — kemampuan mengulang workflow dengan data, kode, dan konfigurasi yang terdokumentasi.
- RMSE — root mean squared error.

S

- Scaffold split — split berdasarkan kerangka molekul untuk mengurangi kemiripan struktur antara train dan test.
- SMILES — notasi teks untuk struktur molekul.
- StandardScaler — transformasi fitur menjadi mean nol dan standard deviation satu berdasarkan training data.

T

- Tanimoto similarity — ukuran kemiripan yang umum untuk fingerprint molekul.
- Target — variabel keluaran yang diprediksi model.
- Test set — subset independen untuk estimasi akhir performa.

U

- Underfitting — model terlalu sederhana untuk menangkap pola data.
- Unsupervised learning — pembelajaran tanpa label target.

V

- Validation set — data untuk pemilihan model/hyperparameter, bukan estimasi akhir.
- Variance — variasi nilai data; juga terkait sensitivitas model terhadap sampel training.

W

- Workflow — rangkaian terstruktur dari data, preprocessing, modeling, evaluasi, hingga pelaporan.

X

- X — notasi umum untuk matriks fitur dalam machine learning.

Y

- y — notasi umum untuk target/label.

Z

- z-score — hasil standardisasi $z = (x - \mu)/\sigma$.

DAFTAR PUSTAKA

1. Delaney, J. S. (2004). ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *Journal of Chemical Information and Computer Sciences*, 44(3), 1000–1005. <https://doi.org/10.1021/ci034243x>
2. Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., & Pande, V. (2018). MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9, 513–530. <https://doi.org/10.1039/C7SC02664A>
3. Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. <https://www.jmlr.org/papers/v12/pedregosa11a.html>
4. Kim, S., Chen, J., Cheng, T., et al. (2023). PubChem 2023 update. *Nucleic Acids Research*, 51(D1), D1373–D1380. <https://doi.org/10.1093/nar/gkac956>
5. Kim, S., Thiessen, P. A., Cheng, T., Yu, B., & Bolton, E. E. (2018). An update on PUG-REST: RESTful interface for programmatic access to PubChem. *Nucleic Acids Research*, 46(W1), W563–W570. <https://doi.org/10.1093/nar/gky294>
6. Zdrazil, B., Felix, E., Hunter, F., et al. (2024). The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1), D1180–D1192. <https://doi.org/10.1093/nar/gkad1004>
7. Mobley, D. L., & Guthrie, J. P. (2014). FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of Computer-Aided Molecular Design*, 28(7), 711–720. <https://doi.org/10.1007/s10822-014-9747-x>
8. Martins, I. F., Teixeira, A. L., Pinheiro, L., & Falcao, A. O. (2012). A Bayesian Approach to in Silico Blood-Brain Barrier Penetration Modeling. *Journal of Chemical Information and Modeling*, 52(6), 1686–1697. <https://doi.org/10.1021/ci300124c>
9. RDKit. (2026). RDKit Documentation: Getting Started with the RDKit in Python and RDKit Book. <https://www.rdkit.org/docs/>
10. Scikit-learn Developers. (2026). scikit-learn User Guide and API Reference. <https://scikit-learn.org/stable/>
11. PubChem. (2026). PUG REST Documentation and Tutorial. National Center for Biotechnology Information. <https://pubchem.ncbi.nlm.nih.gov/docs/pug-rest>
12. ChEMBL. (2026). ChEMBL Data Web Services Documentation. EMBL-EBI. <https://chembl.gitbook.io/chembl-interface-documentation/web-services/chembl-data-web-services>
13. DeepChem. (2026). MoleculeNet loaders and public datasets (Delaney, BBBP, Lipophilicity, FreeSolv, QM9). <https://github.com/deepchem/deepchem>
14. Nusacodes. (2026). Bootcamp Machine Learning with Python. Materi sumber yang diberikan pengguna, bertanggal 10 Oktober 2026. Digunakan sebagai kerangka pedagogis dan diadaptasi khusus untuk kasus riset kimia.
15. Sorkun, M. C., Khetan, A., & Er, S. (2019). AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Scientific Data*, 6, 143. <https://doi.org/10.1038/s41597-019-0151-1>

16. Hansen, K., Mika, S., Schroeter, T., et al. (2009). Benchmark Data Set for in Silico Prediction of Ames Mutagenicity. *Journal of Chemical Information and Modeling*, 49(9), 2077-2081. <https://doi.org/10.1021/ci900161g>
17. Mansouri, K., Ringsted, T., Ballabio, D., Todeschini, R., & Consonni, V. (2013). Quantitative Structure-Activity Relationship Models for Ready Biodegradability of Chemicals. *Journal of Chemical Information and Modeling*, 53(4), 867-878. <https://doi.org/10.1021/ci4000213>
18. Cassotti, M., Ballabio, D., Todeschini, R., & Consonni, V. (2015). A similarity-based QSAR model for predicting acute toxicity towards the fathead minnow (*Pimephales promelas*). *SAR and QSAR in Environmental Research*, 26(3), 217-243. <https://doi.org/10.1080/1062936X.2015.1018938>
19. Mayr, A., Klambauer, G., Unterthiner, T., & Hochreiter, S. (2016). DeepTox: Toxicity Prediction using Deep Learning. *Frontiers in Environmental Science*, 3, 80. <https://doi.org/10.3389/fenvs.2015.00080>
20. Subramanian, G., Ramsundar, B., Pande, V., & Denny, R. A. (2016). Computational Modeling of beta-Secretase 1 (BACE-1) Inhibitors Using Ligand Based Approaches. *Journal of Chemical Information and Modeling*, 56(10), 1936-1949. <https://doi.org/10.1021/acs.jcim.6b00290>
21. Ramakrishnan, R., Dral, P. O., Rupp, M., & von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 140022. <https://doi.org/10.1038/sdata.2014.22>
22. Ahneman, D. T., Estrada, J. G., Lin, S., Dreher, S. D., & Doyle, A. G. (2018). Predicting reaction performance in C-N cross-coupling using machine learning. *Science*, 360(6385), 186-190. <https://doi.org/10.1126/science.aar5169>
23. Zhuo, Y., Mansouri Tehrani, A., & Brgoch, J. (2018). Predicting the Band Gaps of Inorganic Solids by Machine Learning. *The Journal of Physical Chemistry Letters*, 9(7), 1668-1673. <https://doi.org/10.1021/acs.jpcllett.8b00124>
24. Dunn, A., Wang, Q., Ganose, A., Dopp, D., & Jain, A. (2020). Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials*, 6, 138. <https://doi.org/10.1038/s41524-020-00406-3>

SUMBER DATA DAN DOKUMENTASI DARING

- [PubChem PUG REST](https://pubchem.ncbi.nlm.nih.gov/docs/pug-rest) — <https://pubchem.ncbi.nlm.nih.gov/docs/pug-rest>
- [RDKit Documentation](https://www.rdkit.org/docs/) — <https://www.rdkit.org/docs/>
- [scikit-learn Documentation](https://scikit-learn.org/stable/) — <https://scikit-learn.org/stable/>
- [ChEMBL Web Services](https://chembl.gitbook.io/chembl-interface-documentation/web-services/chembl-data-web-services) — <https://chembl.gitbook.io/chembl-interface-documentation/web-services/chembl-data-web-services>
- [DeepChem Repository](https://github.com/deepchem/deepchem) — <https://github.com/deepchem/deepchem>
- [ESOL dataset loader](#) — https://github.com/deepchem/deepchem/blob/master/deepchem/molnet/load_function/delaney_datasets.py
- [BBBP dataset loader](#) — https://github.com/deepchem/deepchem/blob/master/deepchem/molnet/load_function/bbbp_datasets.py
- [Lipophilicity dataset loader](#) — https://github.com/deepchem/deepchem/blob/master/deepchem/molnet/load_function/lipo_datasets.py
- [FreeSolv dataset loader](#) — https://github.com/deepchem/deepchem/blob/master/deepchem/molnet/load_function/freesolv_dataset.py
- [QM9 dataset loader](#) — https://github.com/deepchem/deepchem/blob/master/deepchem/molnet/load_function/qm9_datasets.py
- AqSolDB official repository - <https://github.com/mcsorkun/AqSolDB>
- UCI Ready Biodegradability dataset - <https://archive.ics.uci.edu/dataset/254/qsar+biodegradation>
- UCI QSAR Aquatic Toxicity dataset - <https://archive.ics.uci.edu/dataset/504/qsar+fish+toxicity>
- Tox21 dataset (DeepChem mirror) - <https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/tox21.csv.gz>
- HIV dataset (DeepChem mirror) - <https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/HIV.csv>
- BACE dataset (DeepChem mirror) - <https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/bace.csv>

- QM9 dataset (DeepChem mirror) - <https://deepchemdata.s3-us-west-1.amazonaws.com/datasets/qm9.csv>
- Buchwald-Hartwig rxnpredict repository - <https://github.com/doylelab/rxnpredict>
- Matminer datasets documentation - https://hackingmaterials.lbl.gov/matminer/dataset_summary.html

- KASMUI -
Dosen Kimia, Komputasi, IT, AI UNNES

<https://hisabmu.com/>
<https://kasmui.cloud/>

Ketua PCM Gunungpati 2	Anggota Majelis Tabligh PDM Kota Semarang & PWM Jawa Tengah	Tim Pengembang Software KHGT MTT PP Muhammadiyah	Praktisi Ilmu Falak
---------------------------	---	--	------------------------

Machine Learning Kasus Riset Kimia Menggunakan Python

SINOPSIS

Machine Learning Kasus Riset Kimia Menggunakan Python adalah buku akademik-praktis yang membahas secara lengkap penerapan machine learning pada berbagai permasalahan nyata di bidang kimia. Buku ini mengintegrasikan konsep dasar machine learning, pemrograman Python, pengolahan data kimia, hingga pembangunan dan evaluasi model dengan studi kasus riil yang dapat langsung dipraktikkan.

Materi disusun secara sistematis mulai dari pengenalan machine learning dalam paradigma riset kimia, Python fundamental, preparasi data, representasi molekul, akses database kimia terbuka (seperti PubChem, ChEMBL, dan MoleculeNet), regresi, klasifikasi, clustering, PCA, hingga hyperparameter tuning dan praktik terbaik. Pembahasan selalu dikaitkan dengan konteks kimia seperti prediksi sifat fisikokimia, bioaktivitas senyawa, toksisitas, kimia material, kimia kuantum, dan prediksi hasil reaksi.

Keunggulan buku ini adalah tersedianya 15 proyek riset kimia yang dirancang berdasarkan jurnal ilmiah kredibel, lengkap dengan format proposal dan laporan penelitian, alur kerja yang jelas, kode Python yang dapat dijalankan, serta rencana kerja 14 hari yang realistis bagi mahasiswa S1. Setiap proyek mencakup latar belakang, rumusan masalah, tujuan, telaah pustaka, metode penelitian, hasil dan pembahasan, diskusi, serta kesimpulan.

Buku ini cocok untuk mahasiswa kimia, farmasi, biokimia, teknik kimia, material, informatika kimia, peneliti, dosen, dan praktisi yang ingin memanfaatkan machine learning secara nyata dalam riset kimia, dengan pendekatan yang ilmiah, aplikatif, dan mudah diikuti.

POIN UTAMA BUKU INI



Konsep lengkap dan mudah dipahami



Contoh kode Python yang dapat dipraktikkan



Studi kasus riil dari jurnal ilmiah



15 proyek riset kimia siap dilaksanakan dalam 14 hari



Aplikasi pada berbagai bidang kimia



Pembahasan metodologi yang sistematis



Cocok untuk mahasiswa, peneliti, dan praktisi



Mendukung riset yang reproducible dan berstandar ilmiah