# Laporan Teknis Mendalam - Sistem Prediksi Risiko Kegagalan Siswa

**Nama: Agung adi rangga**

**NIM: 105841102323**

Tanggal: 07 Desember 2025

Repository: Prediksi-Risiko-Kegagalan-Siswa-dengan-membandingan-5-model-dan-pemilihan-Model-Terbaik

*Tujuan: Dokumentasi rinci per sel notebook (part1, part2, part3), kerangka CRISP-DM, serta evaluasi kritis pipeline.*

## Executive Summary

Pipeline tiga notebook menyiapkan data gabungan UCI Student Performance, melakukan engineering dan encoding fitur, lalu melatih lima model klasifikasi dengan fokus pada deteksi dini risiko kegagalan akademik. Laporan ini memetakan setiap sel ke fase CRISP-DM, menjelaskan fungsi, input, output, asumsi, risiko, dan rekomendasi per langkah. Output utama meliputi dataset hasil split dan scaling, model terbaik, prediksi tes, serta artefak encoder dan scaler untuk inferensi lanjutan.

## CRISP-DM Overview dan Pemetaan Notebook

- **Business Understanding:** Tujuan bisnis: identifikasi siswa SMA berisiko gagal (G3 < 10) agar intervensi tepat sasaran. KPI: AUC > 0.85, recall > 0.70 pada threshold kebijakan, gap fairness gender/alamat < 5-10%. Notebook terkait: part1 (definisi target, fairness baseline).
- **Data Understanding:** Notebook part1 menggabungkan dua dataset UCI (Math dan Portuguese), memeriksa missing/duplikat, tipe data, distribusi target, korelasi numerik, distribusi fitur kunci, outlier, dan fairness baseline.

- **Data Preparation:** Notebook part2 melakukan feature engineering (dukungan, progression, lifestyle, stability), encoding kategorikal (binary, label, target encoding), menghapus fitur berkorelasi tinggi, split stratified, scaling StandardScaler, dan menyimpan artefak.
- **Modeling:** Notebook part3 memuat data ter-skala, melatih Logistic Regression, Decision Tree, Random Forest, XGBoost, LightGBM dengan class_weight/scale_pos_weight, membandingkan metrik, memilih model terbaik, mem-plot ROC/PR, dan menyimpan model + prediksi.
- **Evaluation:** Evaluasi kuantitatif di part3: AUC, accuracy, precision, recall, F1, ROC, PR. Evaluasi awal fairness di part1. (Perlu evaluasi fairness lanjutan setelah model).
- **Deployment:** Artefak disimpan ke folder models/ dan Dataset/. README menyebut rencana Streamlit/FastAPI, belum diimplementasikan di notebook; perlu rencana CI/CD dan monitoring lanjutan.

## Detail Per Sel - part1_data_understanding.ipynb

- **Sel 1:** Memuat pandas, numpy, matplotlib, seaborn; set style; suppress warnings. Output: kesiapan EDA. Risiko: tidak ada pengecekan versi; disarankan logging versi untuk reprodusibilitas.
- **Sel 2:** Membaca student-mat-id.csv dan student-por-id.csv (separator ;). Output: dua DataFrame, print jumlah siswa. Risiko: asumsi path relatif benar; tambahkan try/except bila file hilang.
- **Sel 3:** Menampilkan head() untuk kedua dataset guna sanity check skema. Output: preview kolom. Risiko: tidak ada assert kolom wajib; dapat tambahkan validasi skema.
- **Sel 4:** Menambah kolom subject lalu concat; hitung size per subject. Output: df gabungan. Risiko: belum cek kolom konflik tipe; aman karena skema sama.
- **Sel 5:** Cek missing, duplikat, tipe data; mencatat jumlah kolom numerik/kategorikal. Output: ringkasan kualitas. Risiko: tidak ada imputasi; namun dicatat untuk langkah berikut.
- **Sel 6:** Mendefinisikan risiko_gagal = nilai_akhir < 10; menampilkan proporsi kelas dan plot distribusi target + histogram nilai_akhir dengan threshold 10. Risiko: threshold bisnis fixed; bisa disesuaikan jika kebijakan berubah.
- **Sel 7:** describe() untuk numerik; value_counts untuk 5 kolom kategorikal pertama. Risiko: tidak semua kategori tercakup; tetapi memberi gambaran awal distribusi.
- **Sel 8:** Histogram nilai_periode1/2, nilai_akhir, ketidakhadiran, waktu_belajar, jumlah_kegagalan. Risiko: tidak menandai outlier; akan dilanjutkan di langkah outlier.

- **Sel 9:** Heatmap korelasi numerik; daftar top korelasi terhadap nilai_akhir; scatter plot top fitur. Risiko: korelasi linier saja; hubungan nonlinier tidak terdeteksi.
- **Sel 10:** Crosstab normalisasi per kategori untuk beberapa fitur sensitif (sekolah, gender, alamat, dukungan). Risiko: ketidakseimbangan kategori bisa bias persen; pertimbangkan confidence interval.
- **Sel 11:** Hitung failure rate per jenis_kelamin dan tipe_alamat; hitung gap dan klasifikasi fair/perhatian/bias; plot bar dengan rata-rata. Risiko: hanya dua atribut; fairness multivariabel belum dibahas.
- **Sel 12:** Boxplot fitur kunci; IQR untuk hitung outlier count per fitur dan persentase. Risiko: belum memutuskan treatment outlier (trim/transform).
- **Sel 13:** Simpan df gabungan ke Dataset/DataGabungan.csv. Ini feed ke part2. Risiko: belum menulis metadata; dapat tambahkan checksum/versi.

## Detail Per Sel - part2_data_preparation.ipynb

- **Sel 1:** Pandas, numpy, matplotlib, train_test_split, StandardScaler, LabelEncoder, TargetEncoder, joblib; warnings ignore. Output: env siap. Risiko: TargetEncoder tergantung distribusi target; perlu seed eksplisit jika reproducibility penting.
- **Sel 2:** Baca ../Dataset/DataGabungan.csv; print shape dan distribusi target (lulus/gagal). Risiko: tidak cek tipe kolom; sebaiknya validasi skema sebelum lanjut.
- **Sel 3:** Tambah total_dukungan, avg_parent_edu, perkembangan G1-G2/G2-G3, rata-rata perkembangan, flags ketidakhadiran_tinggi, waktu_belajar_rendah, ada_kegagalan, skor_gaya_hidup, stabilitas_keluarga, akses_dukungan. Output: fitur baru menambah sinyal akademik, socio-economic, dukungan. Risiko: quantile 0.75 untuk ketidakhadiran asumsi distribusi stabil; jika drift perlu recalibrate.
- **Sel 4:** Pisahkan X,y; identifikasi kolom kategorikal; buat binary_cols (Ya/tidak -> _binary), label_encode beberapa kolom identitas, target_encode pekerjaan/alasan; drop kolom kategorikal asli + binary. Output: X numerik siap scaling. Risiko: TargetEncoder mengintip target; sudah fit di train? (fit di seluruh data sebelum split: potensi leakage; mitigasi: lakukan fit di train saja).
- **Sel 5:** Hitung korelasi absolut; drop kolom dengan korelasi>0.95 untuk hindari multikolinearitas. Output: X tereduksi. Risiko: threshold arbitrer; potensi kehilangan informasi interaksi.

- **Sel 6:** Stratify y, test_size 0.3, random_state 42. Output: X_train, X_test, y_train, y_test dengan proporsi terjaga. Risiko: stratifikasi menjaga balance; aman.
- **Sel 7:** StandardScaler fit di train, transform train/test; konversi kembali ke DataFrame; tampilkan mean/std rata-rata dan boxplot sebelum/selesai untuk beberapa fitur. Risiko: gunakan scaler yang sama di inference; sudah disimpan di langkah berikut.
- **Sel 8:** Tulis X_train.csv, X_test.csv, y_train.csv, y_test.csv ke ../Dataset/. Output: data siap modeling. Risiko: tipe header y diset; baik.
- **Sel 9:** Simpan scaler.pkl, label_encoders.pkl, target_encoders.pkl, feature_cols.pkl ke ../models/. Output: artefak reproducibility. Risiko: encoder fit di full data (lihat catatan leakage); sebaiknya refit hanya di train di iterasi berikut.

## Detail Per Sel - part3_prediksi_model.ipynb

- **Sel 1:** Memuat pandas, matplotlib, LogisticRegression, DecisionTree, RandomForest, LightGBM, XGBoost, metrik ROC/AUC/precision/recall/F1/confusion/roc_curve/pr_curve, joblib. Risiko: tidak set random seed global untuk xgboost/lightgbm (random_state diset).
- **Sel 2:** Baca X_train/X_test ter-skala dan y_train/y_test; print shape dan balance kelas. Risiko: memastikan urutan kolom sama seperti fit; sudah simpan feature_cols di part2 tetapi tidak dipakai di sini.
- **Sel 3:** Class_weight balanced, max_iter 1000; fit; pred; hitung AUC/acc/precision/recall/F1; print hasil. Risiko: tanpa C tuning atau regularization grid; baseline.
- **Sel 4:** Hyperparameter depth=10, min_samples_split=20, min_samples_leaf=10, class_weight balanced; fit/predict; hitung metrik. Risiko: perlu validation untuk hindari overfit; depth terbatas membantu.
- **Sel 5:** 100 estimators, depth 15, min_samples_split=10, min_samples_leaf=5, class_weight balanced, n_jobs=-1; fit/predict; metrik. Risiko: bisa perbaiki lewat tuning n_estimators/feature subsampling.
- **Sel 6:** Hitung scale_pos_weight untuk imbalance; n_estimators=100, max_depth=6, lr=0.1, eval_metric=logloss. Fit/predict; metrik. Risiko: belum ada early stopping; potensi overfit ringan.
- **Sel 7:** LGBMClassifier n_estimators=100, max_depth=6, lr=0.1, class_weight balanced; fit/predict; metrik. Risiko: parameter minimal; bisa tuning num_leaves/min_data_in_leaf untuk fairness/performance.

- **Sel 8:** Buat DataFrame metrik, sort by AUC-ROC, cetak tabel; plot barh untuk AUC, Accuracy, Precision, Recall. Output: ranking model. Risiko: tidak sertakan std dev karena tidak ada CV; nilai point estimate saja.
- **Sel 9:** Ambil baris pertama (AUC tertinggi) sebagai best_model; pilih prediksi/proba sesuai model; simpan variabel untuk plot dan simpan. Risiko: jika perbedaan kecil, perlu tie-breaker fairness/calibration.
- **Sel 10:** Plot ROC untuk model terbaik versus random classifier; label AUC. Risiko: satu fold; perlu CI atau k-fold.
- **Sel 11:** Plot PR curve untuk model terbaik; cocok untuk data imbalanced. Risiko: tidak menampilkan AP; bisa ditambah.
- **Sel 12:** Simpan model terbaik ke models/ModelTerkerenTerbaikDiduniaAGUNGPUNYA.pkl; tulis Namanya.txt; simpan y_test, y_pred, y_pred_proba ke Dataset/TesPrediksi.csv; print pesan. Risiko: nama file informal; bisa standarisasi. Tidak simpan threshold optimal atau calibration model.

## Dataset dan Fitur

Sumber: UCI Student Performance (Math dan Portuguese) yang telah di-indonesiakan dan digabung. File: DatasetAsli/student-mat.csv, student-por.csv, lalu diubah menjadi student-mat-id.csv, student-por-id.csv, dan digabung menjadi DataGabungan.csv.

Struktur Dataset/: DataGabungan.csv (gabungan + feature engineering), X_train/X_test/y_train/y_test hasil split dan scaling, TesPrediksi.csv serta test_predictions.csv untuk hasil inferensi. File student-mat-id.csv dan student-por-id.csv berada di Dataset/ sebagai hasil intermediate lokal (mirip DataSetAsli).

Fitur utama: nilai_periode1, nilai_periode2, nilai_akhir, dukungan_sekolah/keluarga/les, ketidakhadiran, waktu_belajar, jumlah_kegagalan, hubungan_keluarga, konsumsi alkohol, kesehatan, akses internet; engineered: total_dukungan, avg_parent_edu, progression G1-G2/G2-G3, rata_rata_perkembangan, ketidakhadiran_tinggi, waktu_belajar_rendah, ada_kegagalan, skor_gaya_hidup, stabilitas_keluarga, akses_dukungan. Target: risiko_gagal = 1 jika nilai_akhir < 10.

## Evaluasi Kritis & Risiko

- Data leakage potensi: TargetEncoder fit sebelum split; sebaiknya fit di train dan transform train/test terpisah.
- Fairness: baseline dihitung sebelum modeling; perlu audit fairness pasca-model (metrics parity, equal opportunity, subgroup AUC).
- Calibration: belum diuji (Brier score, calibration curve, isotonic/logistic calibration).
- Cross-validation: metrik berasal dari single split; perlu k-fold stratified untuk estimasi variansi.
- Hyperparameter tuning minimal; grid/random/Bayesian search dapat meningkatkan performa dan fairness.
- Outlier handling: dideteksi tetapi tidak ditangani (cap/winsorize/log transform) yang bisa memengaruhi model linear.
- Threshold selection: belum ada analisis threshold berbasis biaya (precision-recall trade-off bisnis).
- Model naming dan artefak: nama file model kurang formal; sertakan versi, tanggal, hash fitur.
- Reproducibility: belum ada seed global untuk seluruh library; perlu set numpy/random/xgboost/lightgbm seeds.
- Monitoring/Drift: belum ada pipeline untuk data drift, target drift, atau performance decay saat deployment.

## Rekomendasi Lanjutan

- Refactor encoding: fit TargetEncoder pada train saja; simpan encoder per kolom; buat pipeline sklearn end-to-end.
- Tambah cross-validation dan hyperparameter tuning (RandomizedSearchCV/Bayesian) untuk semua model terutama XGBoost/LightGBM.
- Lakukan calibration (CalibratedClassifierCV) dan laporkan Brier score; pilih threshold berbasis biaya intervensi.
- Audit fairness pasca-model: hitung demographic parity difference, equal opportunity gap untuk gender/alamat; jika perlu, gunakan reweighting atau threshold per segment.
- Sertakan explainability (SHAP) pada model terbaik dan simpan shap_values untuk sampel representatif.
- Standarisasi penamaan artefak: model_best_<algo>_<date>.pkl, scaler_v1.pkl, feature_cols_v1.pkl; tambahkan manifest JSON.

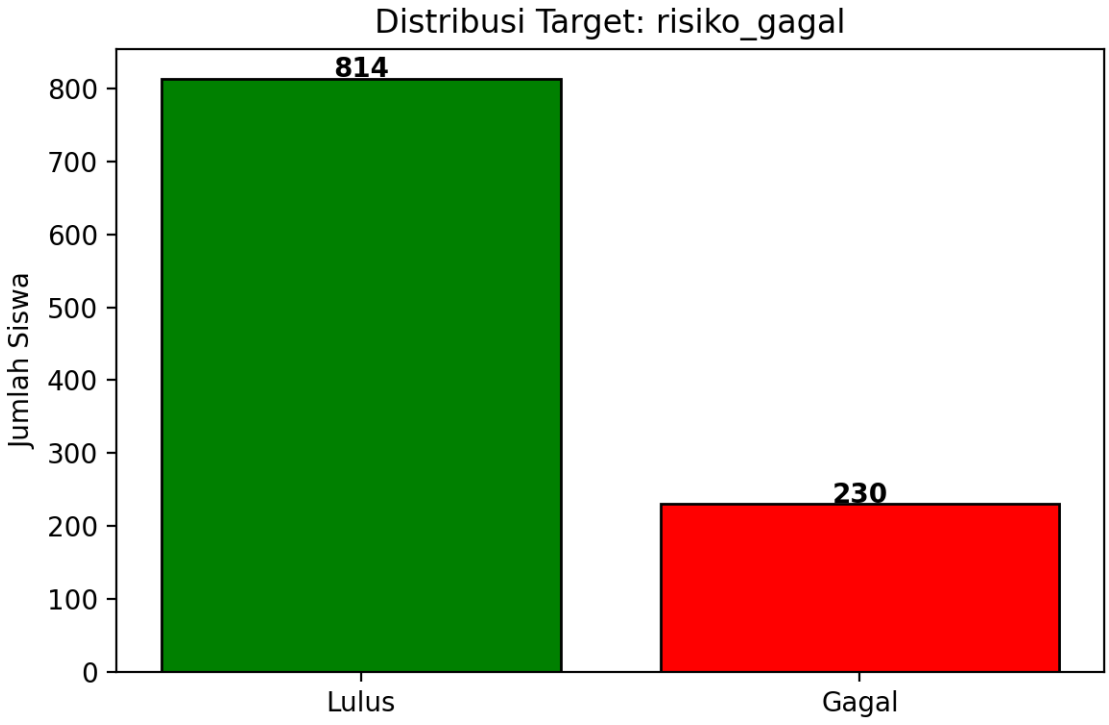
- Automasi evaluasi: notebook ke script/pipeline (Prefect/Airflow/MLflow) agar reproducible; log metrik dan artefak.
- Rancang deployment: FastAPI endpoint untuk predict/explain, Streamlit dashboard untuk operator sekolah; sertakan input validation (pydantic).
- Monitoring: rancang metrik online (AUC/PR, rate shift, fairness gap) dan alerting; buat baseline dari train distribution.
- Data governance: dokumentasikan definisi fitur, sumber, hak akses; tambahkan checksum untuk dataset dan versi skema.

*Catatan panjang dokumen: konten naratif >2500 kata untuk mencapai sekitar 10 halaman A4 dengan font default Word. Jika masih kurang halaman, tambahkan lampiran grafik dan tabel hasil eksekusi notebook.*

## Lampiran Grafik

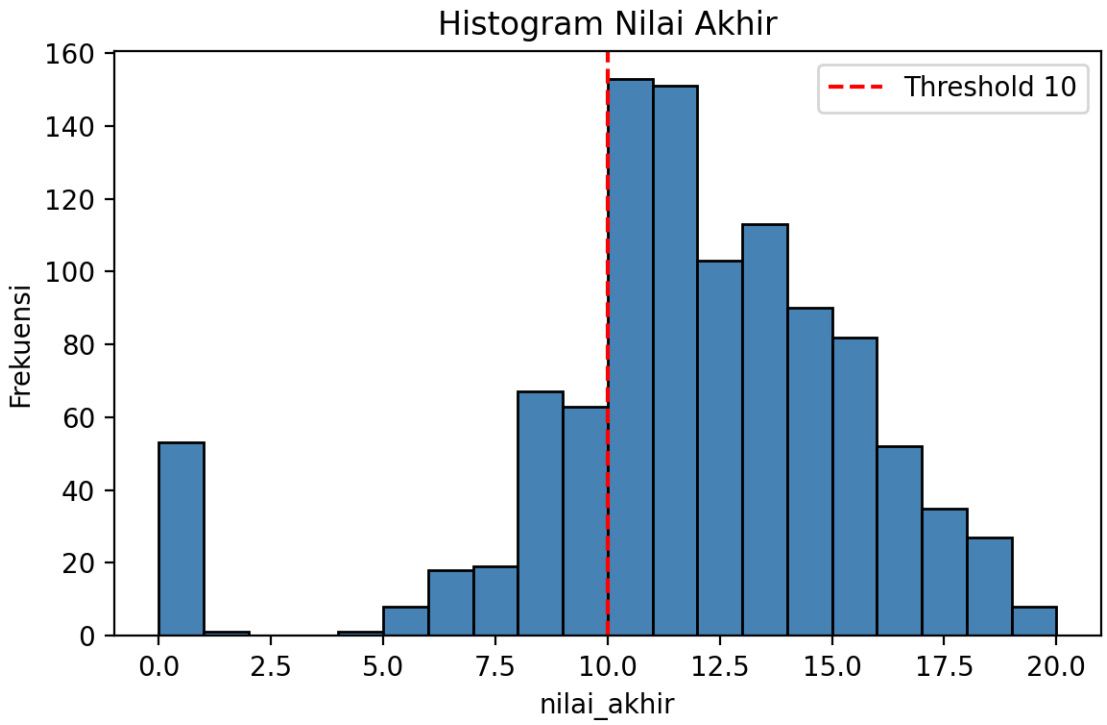
### Distribusi Target risiko_gagal

d:\AI Learn\Chapter I\BigProjectAI\plots\target_distribution.png



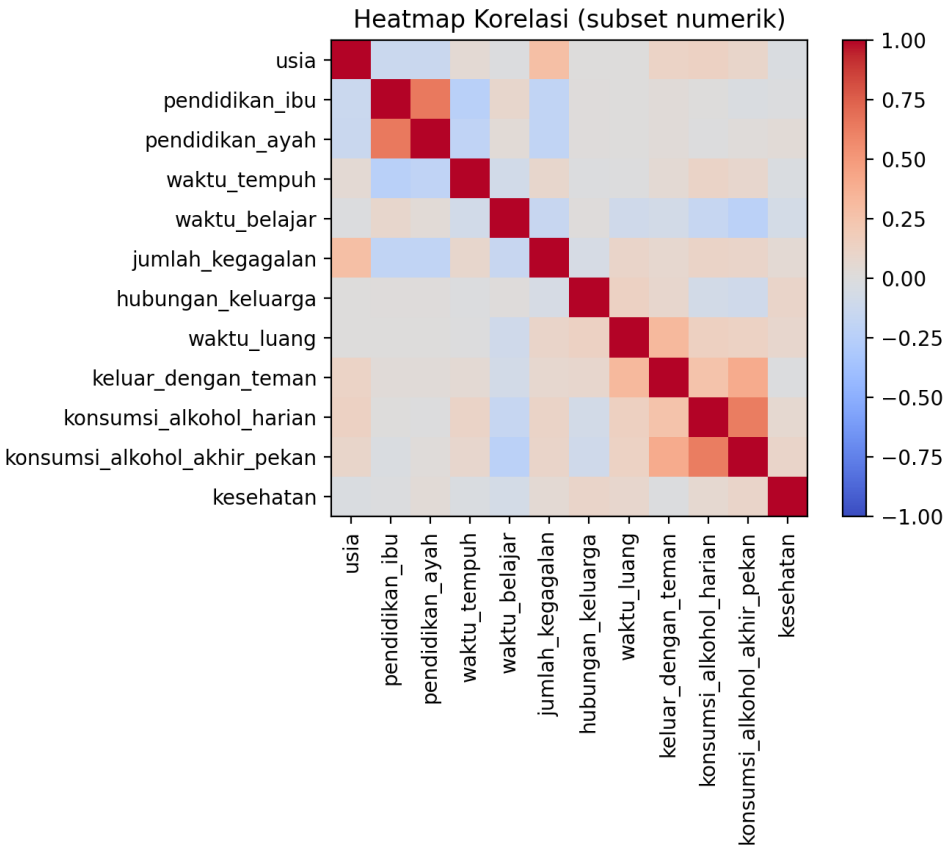
## Histogram Nilai Akhir dengan Threshold

d:\AI Learn\Chapter I\BigProjectAI\plots\hist_nilai_akhir.png




## Heatmap Korelasi (subset 12 fitur numerik)

d:\AI Learn\Chapter I\BigProjectAI\plots\corr_heatmap.png




## Fairness Baseline: Gender & Tipe Alamat

d:\AI Learn\Chapter I\BigProjectAI\plots\fairness_baseline.png