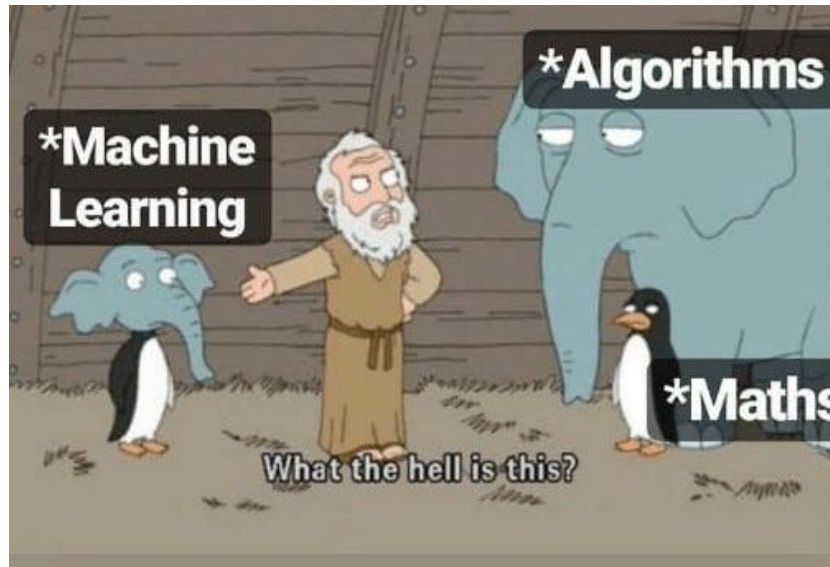


# Tugas Besar 2 IF3170 Inteligensi Artifisial Implementasi Algoritma Pembelajaran Mesin

Dipersiapkan Oleh Tim Asisten Lab AI '22

Versi: 1.0 13/11/2025



Deadline: Jumat, 5 Desember 2025 22.22 WIB

## Tujuan

Tugas Besar 2 pada kuliah IF3170 Inteligensi Buatan bertujuan untuk memberikan pengalaman langsung kepada peserta kuliah dalam menerapkan algoritma pembelajaran mesin pada permasalahan nyata.

## Spesifikasi

**Pembelajaran mesin** merupakan salah satu cabang dari kecerdasan buatan yang memungkinkan sistem untuk belajar dari data dan membuat prediksi atau keputusan tanpa diprogram secara eksplisit.

Dataset yang digunakan dalam tugas besar ini adalah kumpulan data tentang mahasiswa yang terdaftar dalam berbagai program sarjana yang ditawarkan oleh perguruan tinggi. Pada tugas

ini, Anda diminta untuk mengimplementasikan algoritma pembelajaran mesin yang telah kalian pelajari di kuliah, yaitu **DTL, Logistic Regression, dan SVM** pada dataset **UNSW-NB15**. Rincian spesifikasi untuk tugas besar 2 dapat dilihat sebagai berikut:

1. Implementasikan Decision Tree Learning (DTL) *from scratch*. Pilih **salah satu** algoritma DTL berikut.
  - Iterative Dichotomiser 3 (ID3)
  - C4.5
  - Classification and Regression Tree (CART)

Algoritma DTL yang diimplementasikan seminimal mungkin harus dapat menangani tipe data numerik dan *categorical* serta dapat menangani *null values*. Algoritma optimasi model selain yang diajarkan di kelas akan dihitung sebagai bonus.

2. Implementasikan Logistic Regression *from scratch*. Gunakan algoritma yang sama dengan algoritma yang telah diajarkan di kelas. Algoritma optimasi model selain yang diajarkan di kelas akan dihitung sebagai bonus.
3. Implementasikan SVM *from scratch*. Implementasikan juga algoritma untuk menangani *multiclass prediction* dengan memilih **salah satu** dari algoritma berikut.
  - One-against all
  - One-against one
  - DAGSVM

Algoritma optimasi model selain yang diajarkan di kelas akan dihitung sebagai bonus.

4. Implementasi algoritma poin 1-3 menggunakan *scikit-learn*. Bandingkan hasil dari algoritma *from scratch* dan algoritma *scikit-learn*. Untuk ID3 di *scikit-learn*, gunakan `DecisionTreeClassifier` dengan parameter `criterion='entropy'`. Untuk C4.5 di *scikit-learn*, gunakan `DecisionTreeClassifier` dengan parameter `criterion='entropy'` dan tambahkan `ccp_alpha`. Untuk CART di *scikit-learn*, gunakan `DecisionTreeClassifier` dengan parameter `criterion='gini'`.
5. Model harus bisa di-save dan di-load. Implementasinya dibebaskan (misal menggunakan .txt, .pkl, dll).
6. Kaggle Submission pada link [ini](#) minimal 1 *submission*.

Implementasi DTL, Logistic Regression, dan SVM *from scratch* bisa dalam bentuk kelas-kelas (class KNN, dst.) yang nantinya akan di-import ke notebook pengerjaan. Untuk implementasi *from scratch*, *library* yang boleh digunakan adalah untuk perhitungan matematika saja seperti numpy dan sejenisnya.

Asisten telah menyediakan notebook [berikut](#) untuk Anda lengkapi, dan deskripsi lengkap mengenai dataset dapat dilihat sebagai berikut:

## Deskripsi Dataset

[Dataset](#) yang digunakan merupakan gambaran tentang mahasiswa yang terdaftar dalam berbagai program sarjana yang ditawarkan oleh perguruan tinggi. Data ini mencakup data demografis, faktor sosial-ekonomi, dan informasi kinerja akademik yang dapat digunakan untuk menganalisis faktor-faktor yang mungkin memprediksi tingkat kesuksesan akademik mahasiswa. Deskripsi fitur dan target dataset juga dapat dilihat [di sini](#). **Tidak ada perubahan dataset, silahkan handle semua ketidaksesuaian pada kode masing-masing.**

Untuk menghasilkan prediksi yang berkualitas, Anda diharuskan untuk melakukan beberapa tahap berikut ini (tahap lebih lengkap dapat dilihat di template notebook):

## Data Cleaning

Tahap ini bertujuan untuk membersihkan dataset dari nilai yang hilang (missing values), data duplikat, atau data yang tidak valid sehingga dataset siap digunakan untuk analisis.

## Data Transformation

Transformasi data melibatkan langkah-langkah seperti encoding variabel kategori, normalisasi atau standarisasi fitur numerik, serta penanganan ketidakseimbangan data (imbalanced data) untuk memastikan data berada dalam format yang sesuai dengan algoritma pembelajaran mesin.

## Feature Selection

Pemilihan fitur yang relevan bertujuan untuk mengurangi kompleksitas model, menghindari overfitting, serta meningkatkan kinerja model. Langkah ini melibatkan identifikasi fitur yang memiliki pengaruh signifikan terhadap variabel target.

## Dimensionality Reduction

Jika dataset memiliki jumlah fitur yang besar, reduksi dimensi dapat digunakan untuk mengurangi dimensi tanpa kehilangan informasi penting. Teknik seperti Principal Component Analysis (PCA) sering digunakan pada tahap ini.

## Modeling dan Validation

Pada tahap ini, algoritma pembelajaran mesin seperti DTL, Logistic Regression, dan SVM diterapkan pada dataset. Anda akan melatih model pembelajaran mesin yang akan **mengklasifikasi fitur 'Target'** berdasarkan fitur-fitur lain yang telah diberikan. Model yang

telah dibuat divalidasi menggunakan metode seperti **train-test split** atau **k-fold cross-validation** untuk memastikan kinerja yang optimal.

## Bonus

### 1. Bonus Kaggle

Untuk bonus, nilai diberikan berdasarkan ranking *leaderboard Kaggle* yang dirincikan sebagai berikut:

- Rank 1-3 = 10 poin
- Rank 4-5 = 5 poin
- Rank 6-10 = 3 poin

Dalam leaderboard, gunakan nama kelompok. Identifikasi dilakukan berdasarkan nama kelompok, jadi cukup 1 orang saja yang berada dalam tim Kaggle. Hasil prediksi di kaggle harus **reproducible**, sehingga notebook yang dikumpulkan harus bisa menghasilkan nilai akhir yang sama dengan submisi kaggle. Jika tidak sama, maka akan **didiskualifikasi** dari leaderboard. **Model yang boleh digunakan hanya DTL, Logistic Regression, dan SVM yang diimplementasikan from scratch.**

2. Untuk DTL, hasilkan **gambar percabangan tree** (format file dibebaskan). Tambahkan parameter untuk mengatur top-N percabangan yang ditampilkan dalam gambar.
3. Untuk Logistic Regression, hasilkan **video** yang menampilkan **garis kontur fungsi loss (log-loss)** dan lintasan parameter ( $\theta_0$ ,  $\theta_1$ ) selama training.
4. Untuk SVM, hasilkan **video** yang menampilkan **proses training model** seperti yang ada di PPT kelas.
5. Untuk seluruh algoritma, implementasikan algoritma untuk **mengoptimasi model** (selain algoritma yang telah diajarkan di kelas). Jelaskan bagaimana algoritma tersebut dapat mengoptimasi model. Sertakan juga referensi/sumbernya (**tanpa referensi, poin nilai dari bonus ini akan dianulir**).

## Kelompok

Pembagian kelompok ditentukan sendiri oleh mahasiswa dengan mengisi [sheets kelompok](#) berikut ini dengan 1 kelompok terdiri dari **5 orang**. Kelompok pada tugas besar 2 tidak dipengaruhi oleh kelompok pada tugas besar 1 (boleh dengan anggota yang sama maupun

berbeda). Batas waktu pengisian kelompok adalah **16 November 2024 pukul 22:22 WIB**. Setelah waktu yang ditentukan, mahasiswa yang belum mengisi sheets kelompok akan diacak.

## QnA

Pertanyaan dapat ditanyakan pada [link QnA](#) berikut. Pastikan pertanyaan yang ditanyakan tidak berulang.

## Aturan

Terdapat beberapa hal yang harus diperhatikan dalam pengerjaan tugas ini, yakni:

1. Jika terdapat hal yang tidak dimengerti, silahkan ajukan pertanyaan kepada asisten melalui **link QnA** yang telah diberikan di atas. Pertanyaan yang diajukan secara personal ke asisten **tidak akan dijawab** untuk menghindari perbedaan informasi yang didapatkan oleh peserta kuliah.
2. Dilarang melakukan **plagiarisme, menggunakan AI dalam bentuk apapun untuk men-generate jawaban Anda, dan melakukan kerjasama antar kelompok**. Pelanggaran pada poin ini akan menyebabkan pemberian **nilai E** pada setiap anggota kelompok.

## Deliverables

- Tugas dikumpulkan dalam bentuk link ke *repository* GitHub yang **minimal** berisi beberapa hal berikut (boleh ditambahkan jika dirasa perlu):
  - Folder **src**, digunakan untuk menyimpan source code
  - Folder **doc**, digunakan untuk menyimpan laporan dalam bentuk **.pdf** yang terdiri atas komponen berikut:
    - Cover
    - Penjelasan singkat implementasi Decision Tree Learning.
    - Penjelasan singkat implementasi Logistic Regression.
    - Penjelasan singkat implementasi SVM.
    - Penjelasan tahap cleaning dan preprocessing yang dilakukan beserta dengan alasannya.
    - Perbandingan hasil prediksi dari algoritma yang diimplementasikan dengan hasil yang didapatkan dengan menggunakan pustaka. Jelaskan insight yang kalian dapatkan dari perbandingan tersebut.
      - Perbandingan hasil dapat menggunakan metrics yang sesuai dengan permasalahan yang ada.

- Kontribusi setiap anggota dalam kelompok.
- Referensi
  - **README.md**, yang berisi deskripsi singkat repository, cara setup dan run program, dan pembagian tugas tiap anggota kelompok.
- Pengumpulan dilakukan melalui **Edunex**. Gunakan form [berikut](#) jika dan hanya jika terdapat kendala pada Edunex seminimalnya J-3 deadline tugas ini (pukul 19.22).
- Batas akhir pengumpulan adalah hari **Jumat, 5 Desember 2025 pukul 22.22**. Tugas yang terlambat dikumpulkan tidak akan diterima.
- Pengumpulan dilakukan oleh NIM terkecil.

## Referensi

- <https://www.geeksforgeeks.org/machine-learning/iterative-dichotomiser-3-id3-algorithm-from-scratch/>
- <https://www.geeksforgeeks.org/machine-learning/cart-classification-and-regression-tree-in-machine-learning/>
- <https://www.geeksforgeeks.org/machine-learning/decision-tree-algorithms/>
- <https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/>
- <https://www.geeksforgeeks.org/machine-learning/understanding-logistic-regression/>