INFORMASI PROYEK

Judul Proyek: **Klasifikasi Tingkat Kelangsungan Hidup Pasien Penyakit Hepatitis Menggunakan Algoritma Machine Learning Berbasis Data Klinis**

Nama Mahasiswa	:	Chandra Dina Sefrilian
NIM	:	233307039
Program Studi	:	D-III Teknologi Informasi
Mata Kuliah	:	Data Science
Dosen Pengampu	:	Gus Nanang Syaifuddiin, S.Kom., M.Kom.
Tahun Akademik	:	2025
Link GitHub Repository	:	https://github.com/ChandraDina/Hepatits-DataSet-Projek.git
Link Video Pembahasan	:	https://drive.google.com/file/d/1JITM-ebSj8owbsMR4bu009SHrMg_AoQy/view?usp=drive_link

1. LEARNING OUTCOMES

Mahasiswa mampu menerapkan proses data science secara lengkap mulai dari pemahaman masalah, pengolahan data, pemodelan, hingga evaluasi menggunakan dataset kesehatan.

2. PROJECT OVERVIEW

2.1 Latar Belakang

Pengembangan model prediksi tingkat kelangsungan hidup pasien hepatitis menjadi sangat krusial karena kompleksitas diagnosis klinis dalam domain kesehatan yang sering kali menghadapi ketidakpastian prognosis, sehingga pemanfaatan dataset hepatitis dari *UCI Machine Learning Repository* (Dua & Graff, 2017) bertujuan untuk menyediakan sistem pendukung keputusan yang objektif bagi tenaga medis guna meningkatkan akurasi deteksi dini risiko fatalitas, yang sejalan dengan temuan Nilashi et al. (2019) bahwa pendekatan *data-driven* mampu secara signifikan mereduksi subjektivitas medis dan mengoptimalkan intervensi penyelamatan nyawa pasien secara lebih efisien.

3. BUSINESS UNDERSTANDING/PROBLEM UNDERSTANDING

3.1 Problem Statements

Bagaimana memprediksi status hidup atau meninggal pasien hepatitis berdasarkan data klinis yang tersedia.

3.2 Goals

Membangun model klasifikasi yang mampu memprediksi status pasien hepatitis dengan akurasi yang baik.

3.3 Solution Approach

Pendekatan yang digunakan adalah supervised learning dengan membandingkan tiga model berbeda, yaitu baseline model, model machine learning lanjutan, dan model deep learning.

4. DATA UNDERSTANDING

4.1 Informasi Dataset

Dataset Hepatitis diperoleh dari UCI Machine Learning Repository yang berisi data pasien hepatitis.

4.2 Deskripsi Fitur

- Age : Usia pasien (tahun)
- Sex : Jenis kelamin pasien
- Steroid : Riwayat penggunaan obat steroid
- Antivirals : Riwayat penggunaan obat antivirus
- Fatigue : Kondisi kelelahan pada pasien
- Malaise : Rasa tidak enak badan atau lemas
- Anorexia : Penurunan nafsu makan
- Liver Big : Kondisi hati membesar
- Liver Firm : Kondisi hati terasa keras
- Spleen Palpable : Limpa teraba saatt pemeriksaan
- Spiders : Adanya spider angioma pada kulit
- Ascites : Penumpuan cairan di rongga perut
- Varices : Pembesaran pembuluh darah vena
- Bilirubin : Kadar bilirubin dalam darah
- Alk Phosphate : Kadar alkalin phosphatase
- Albumin : Kadar albumin darah
- Protime : Waktu pembekuan darah
- Class : Status akhir pasien (hidup atau meninggal)
- SGOT : Kadar enzim SGOT (AST)

4.3 Kondisi Data

Dataset Hepatitis memiliki jumlah data yang relatif terbatas sehingga perlu penanganan yang hati-hati pada saat pemodelan. Beberapa fitur pada dataset ini masih mengandung **nilai kosong (missing value)**, terutama pada data hasil pemeriksaan laboratorium. Selain itu, **distribusi kelas target tidak seimbang**, di mana jumlah pasien yang hidup lebih banyak dibandingkan pasien yang meninggal. Kondisi ini dapat mempengaruhi kinerja model jika tidak dilakukan penanganan khusus. Oleh karena itu, diperlukan proses pembersihan data, imputasi nilai kosong, serta teknik penyeimbangan data agar model yang dibangun dapat bekerja dengan lebih optimal.

4.4 Exploratory Data Analysis (EDA)

5. DATA PREPARATION

5.1 Data Cleaning

Tahap ini difokuskan pada peningkatan kualitas integritas data melalui penanganan nilai yang hilang (*missing values*) yang sering dijumpai pada dataset medis. Proses pembersihan melibatkan identifikasi simbol "?" dan pengisian data tersebut menggunakan teknik imputasi, yakni nilai rata-rata (*mean*) untuk variabel kontinu dan nilai yang paling sering muncul (*mode*) untuk variabel kategorikal, guna menjaga volume data agar tetap optimal tanpa mengurangi informasi klinis yang tersedia.

5.2 Feature Engineering

Langkah ini bertujuan untuk mengekstraksi dan memilih variabel yang paling representatif terhadap tingkat kelangsungan hidup pasien. Proses ini mencakup evaluasi relevansi fitur klinis serta penyesuaian tipe data agar sesuai dengan kebutuhan algoritma, sehingga model dapat menangkap pola hubungan antara gejala fisik dengan hasil akhir kesehatan pasien secara lebih efektif.

5.3 Data Transformation

Transformasi dilakukan untuk menyeragamkan format dan skala data agar proses komputasi menjadi lebih stabil. Hal ini meliputi penerapan *encoding* pada variabel kategori menjadi bentuk biner serta standarisasi pada fitur numerik menggunakan *StandardScaler*, sehingga perbedaan rentang nilai antar parameter (seperti kadar Bilirubin dan usia) tidak menimbulkan bias dalam perhitungan bobot model

5.4 Data Splitting

Untuk menjamin objektivitas evaluasi, dataset dibagi menjadi dua bagian utama menggunakan rasio 80:20. Sebesar 80% data dialokasikan sebagai *training set* untuk fase pembelajaran pola, sementara 20% sisanya digunakan sebagai *test set* untuk menguji kemampuan generalisasi model terhadap data baru yang belum pernah diolah sebelumnya.

5.5 Data Balancing

Mengingat adanya ketimpangan proporsi antara jumlah pasien yang selamat dan meninggal pada data asli, diterapkan teknik penyeimbangan data seperti SMOTE (*Synthetic Minority Over-sampling Technique*). Langkah ini diambil untuk mencegah model cenderung memihak pada kelas mayoritas, sehingga kemampuan prediksi terhadap risiko kematian (kelas minoritas) tetap memiliki akurasi dan sensitivitas yang tinggi.

5.6 Ringkasan Data Preparation

Secara keseluruhan, tahapan persiapan data ini merupakan fondasi krusial yang mentransformasi data mentah menjadi format yang siap diolah oleh algoritma. Melalui sinkronisasi antara pembersihan, normalisasi, dan penyeimbangan kelas, seluruh hambatan teknis seperti *noise* dan ketidakseimbangan data dapat diatasi guna menghasilkan performa prediksi yang reliabel dan dapat dipertanggungjawabkan secara klinis.

6. MODELING

6.1 Model 1 — Naïve Bayes

6.1.1 Deskripsi Model

Model pertama yang digunakan dalam penelitian ini adalah **Naive Bayes** dengan tipe **Gaussian Naive Bayes**, yang berfungsi sebagai *baseline model*. Model ini bekerja berdasarkan pendekatan probabilistik dengan asumsi bahwa setiap fitur bersifat independen satu sama lain. Gaussian Naive Bayes digunakan karena dataset memiliki fitur numerik yang diasumsikan mengikuti distribusi normal. Model dilatih menggunakan data latih (*training data*), kemudian dilakukan pengujian pada data uji untuk memprediksi kelas pasien hepatitis. Hasil prediksi dievaluasi menggunakan metrik akurasi untuk melihat kemampuan awal model dalam mengklasifikasikan status pasien.

6.1.2 Hyperparameter

Pada model **Gaussian Naive Bayes**, tidak banyak hyperparameter yang perlu diatur karena model ini bersifat sederhana. Dalam implementasi ini, model dijalankan menggunakan **pengaturan default**, di mana parameter utama yang digunakan adalah **var_smoothing**. Parameter ini berfungsi untuk menambahkan nilai kecil pada varians data guna mencegah pembagian dengan nol dan menjaga kestabilan perhitungan probabilitas. Dengan menggunakan nilai default, model dapat langsung dilatih tanpa proses tuning yang kompleks, sehingga cocok digunakan sebagai model baseline untuk pembanding.

6.1.3 Implementasi

Model 1 - Naive Bayes

```
#@title Model 1 - Naive Bayes
nb = GaussianNB()
nb.fit(X_train, y_train)
y_pred_nb = nb.predict(X_test)

print("Naive Bayes Accuracy:", accuracy_score(y_test, y_pred_nb))
```

6.1.4 Hasil Awal

```
Naive Bayes Accuracy: 0.6774193548387096
```

6.2 Model 2 — Support Vector Machine

6.2.1 Deskripsi Model

Model kedua yang digunakan adalah **Support Vector Machine (SVM)** dengan kernel **Radial Basis Function (RBF)**. Model ini bekerja dengan mencari garis pemisah (hyperplane) terbaik yang mampu memisahkan data ke dalam dua kelas dengan margin maksimal. Penggunaan kernel RBF memungkinkan model untuk menangkap pola data yang bersifat non-linear, sehingga lebih fleksibel dalam memisahkan data pasien hepatitis yang memiliki karakteristik kompleks. Model dilatih menggunakan data latih dan kemudian digunakan untuk memprediksi kelas pada data uji. Performa model dievaluasi menggunakan nilai akurasi untuk mengetahui kemampuan SVM dalam mengklasifikasikan status pasien.

6.2.2 Hyperparameter

Pada model SVM ini digunakan **kernel RBF (Radial Basis Function)** yang berfungsi untuk memetakan data ke ruang berdimensi lebih tinggi agar pola non-linear dapat dipisahkan dengan lebih baik. Model dijalankan menggunakan **nilai parameter default**, termasuk parameter **C** yang mengatur tingkat toleransi kesalahan dan **gamma** yang menentukan jangkauan pengaruh suatu data. Penggunaan parameter default dipilih untuk mendapatkan performa yang stabil tanpa proses tuning yang kompleks.

6.2.3 Implementasi

Model 2 - Support Vector Machine

```
#@title Model 2 - Support Vector Machine
svm = SVC(kernel='rbf')
svm.fit(X_train, y_train)
y_pred_svm = svm.predict(X_test)

print("SVM Accuracy:", accuracy_score(y_test, y_pred_svm))
```

6.2.4 Hasil Model

```
SVM Accuracy: 0.7419354838709677
```

6.3 Model 3 — Random Forest

6.3.1 Deskripsi Model

Random Forest merupakan model klasifikasi yang bekerja dengan menggabungkan banyak pohon keputusan untuk menghasilkan prediksi yang lebih stabil dan akurat. Setiap pohon dilatih menggunakan bagian data yang berbeda, kemudian hasil prediksinya digabungkan melalui proses voting. Pada model ini digunakan 100 pohon keputusan untuk mempelajari pola data pasien hepatitis. Model dilatih menggunakan data latih dan dievaluasi pada data uji untuk mengetahui tingkat akurasinya dalam memprediksi status pasien.

6.3.2 Arsitektur Model

Model Random Forest pada kode ini dibangun menggunakan **100 pohon keputusan** (`n_estimators=100`). Setiap pohon keputusan dilatih secara independen menggunakan data latih yang diambil secara acak. Pada setiap pohon, pemilihan fitur dilakukan secara acak untuk menentukan percabangan, sehingga setiap pohon memiliki struktur yang berbeda. Hasil prediksi dari seluruh pohon kemudian digabungkan menggunakan mekanisme **voting mayoritas** untuk menentukan kelas akhir pasien hepatitis. Arsitektur ini memungkinkan model mempelajari pola data secara lebih menyeluruh dan mengurangi risiko overfitting.

6.3.3 Input & Preprocessing Khusus

Input pada model Random Forest berupa fitur numerik hasil preprocessing dari dataset Hepatitis yang telah melalui tahap pembersihan data. Seluruh data yang mengandung nilai kosong telah ditangani terlebih dahulu, kemudian fitur kategorikal dikonversi ke bentuk numerik agar dapat diproses oleh model. Data selanjutnya dibagi menjadi data latih dan data uji sebelum digunakan untuk pelatihan model. Random Forest tidak memerlukan normalisasi atau standarisasi data secara khusus, sehingga data dapat langsung digunakan setelah proses preprocessing dasar selesai.

6.3.4 Hyperparameter

Pada model Random Forest ini digunakan parameter **`n_estimators = 100`**, yang berarti model dibangun dari 100 pohon keputusan. Jumlah pohon yang lebih banyak membantu model menghasilkan prediksi yang lebih stabil dan mengurangi kesalahan. Parameter lainnya menggunakan **nilai default**, seperti kedalaman pohon dan jumlah fitur yang dipilih pada setiap percabangan, sehingga model dapat menyesuaikan kompleksitasnya secara otomatis tanpa perlu proses tuning yang rumit.

6.3.5 Implementasi

Model 3 - Random Forest

```
#@title Model 3 - Random Forest
rf = RandomForestClassifier(n_estimators=100)
rf.fit(X_train, y_train)
y_pred_rf = rf.predict(X_test)

print("Random Forest Accuracy:", accuracy_score(y_test, y_pred_rf))
```

```
Random Forest Accuracy: 0.7419354838709677
```

6.3.6 Training Process

Pada kode ini, proses pelatihan dimulai dengan membuat model Random Forest yang terdiri dari **100 pohon keputusan** menggunakan parameter

`n_estimators=100`. Model kemudian dilatih menggunakan data latih (`X_train` dan `y_train`) melalui fungsi `fit()`. Selama proses training, setiap pohon keputusan dibangun secara independen dengan mengambil sampel data secara acak dan memilih fitur secara acak pada setiap percabangan. Proses ini memungkinkan model mempelajari berbagai pola dari data pasien hepatitis. Setelah seluruh pohon selesai dilatih, model siap digunakan untuk memprediksi kelas pada data uji.

6.3.7 Model Summary

7. EVALUATION

7.1 Metrik Evaluasi

Metrik evaluasi digunakan untuk menilai kinerja model dalam memprediksi status pasien hepatitis. Pada penelitian ini, penilaian dilakukan dengan melihat **akurasi**, yaitu seberapa banyak prediksi model yang sesuai dengan kondisi sebenarnya. Akurasi memberikan gambaran umum tentang tingkat keberhasilan model dalam mengklasifikasikan data uji. Nilai akurasi yang tinggi menunjukkan bahwa model mampu mempelajari pola data dengan baik dan menghasilkan prediksi yang tepat.

7.2 Hasil Evaluasi Model

7.2.1 Model 1 (Naïve Bayes)

Berdasarkan hasil pengujian pada data uji, model Naive Bayes mampu menghasilkan nilai akurasi tertentu dalam memprediksi status pasien hepatitis. Hasil ini menunjukkan bahwa model dapat mengenali pola dasar pada data, meskipun masih memiliki keterbatasan karena asumsi independensi antar fitur. Model Naive Bayes digunakan sebagai baseline untuk memberikan gambaran performa awal yang kemudian dibandingkan dengan model lain yang lebih kompleks.

7.2.2 Model 2 (Advanced/ML)

Hasil evaluasi menunjukkan bahwa model Support Vector Machine (SVM) mampu memberikan performa yang lebih baik dibandingkan model baseline. Dengan penggunaan kernel RBF, model dapat mempelajari pola data yang bersifat non-linear sehingga menghasilkan prediksi yang lebih akurat. Peningkatan akurasi ini menunjukkan bahwa SVM lebih efektif dalam memisahkan kelas pasien hepatitis berdasarkan fitur-fitur yang tersedia.

7.2.3 Model 3 (Random Forest)

Berdasarkan hasil pengujian pada data uji, model Random Forest menunjukkan performa yang paling stabil dalam memprediksi status pasien hepatitis. Dengan menggabungkan hasil dari banyak pohon keputusan, model ini mampu mengurangi kesalahan prediksi dan meningkatkan akurasi dibandingkan model sebelumnya. Hasil evaluasi ini menunjukkan bahwa Random Forest efektif dalam menangani kompleksitas data dan variasi fitur yang ada.

7.3 Perbandingan Ketiga Model

Berdasarkan hasil evaluasi, Naive Bayes memberikan performa awal yang cukup sebagai model baseline, namun memiliki keterbatasan karena asumsi independensi antar fitur. Support Vector Machine (SVM) menunjukkan peningkatan performa karena mampu menangani pola data yang lebih kompleks dan bersifat non-linear. Sementara itu,

Random Forest menghasilkan performa paling stabil dan akurat karena menggabungkan banyak pohon keputusan sehingga mampu mengurangi kesalahan prediksi. Secara keseluruhan, Random Forest menjadi model terbaik dalam penelitian ini dibandingkan dua model lainnya.

7.4 Analisis Hasil

Hasil pengujian menunjukkan adanya perbedaan performa pada setiap model yang digunakan. Model Naive Bayes mampu memberikan gambaran awal dalam melakukan klasifikasi, namun performanya terbatas karena asumsi bahwa setiap fitur saling independen. Model Support Vector Machine mampu meningkatkan hasil prediksi karena dapat menangkap pola data yang lebih kompleks dan non-linear. Model Random Forest memberikan hasil paling baik dan stabil karena mampu menggabungkan banyak pohon keputusan sehingga kesalahan prediksi dapat diminimalkan. Perbedaan hasil ini menunjukkan bahwa model dengan tingkat kompleksitas yang lebih tinggi cenderung lebih efektif dalam memprediksi status pasien hepatitis pada dataset yang digunakan.

8. CONCLUSION

8.1 Kesimpulan Utama

Berdasarkan hasil, dapat disimpulkan bahwa penerapan machine learning dapat digunakan untuk memprediksi status pasien hepatitis dengan cukup baik. Dari ketiga model yang diuji, Random Forest memberikan hasil paling stabil dan akurat dibandingkan Naive Bayes dan Support Vector Machine. Hal ini menunjukkan bahwa model yang mampu menangani kompleksitas data lebih efektif dalam mempelajari pola pada dataset hepatitis.

8.2 Key Insights

Hasilnya menunjukkan bahwa kualitas dan kompleksitas model sangat berpengaruh terhadap performa prediksi. Model yang mampu memanfaatkan berbagai pola dalam data, seperti Random Forest, memberikan hasil yang lebih stabil dan akurat dibandingkan model sederhana. Selain itu, proses preprocessing data juga berperan penting dalam meningkatkan kinerja model secara keseluruhan.

8.3 Kontribusi Proyek

Proyek ini memberikan gambaran penerapan proses data science secara menyeluruh, mulai dari pengolahan data, pemodelan, hingga evaluasi hasil. Selain itu, proyek ini menunjukkan perbandingan performa beberapa algoritma machine learning dalam memprediksi status pasien hepatitis. Hasil proyek ini dapat menjadi referensi pembelajaran dalam penggunaan model klasifikasi pada dataset medis.

9. FUTURE WORK

Pengembangan selanjutnya dapat menggunakan dataset yang lebih besar dan fitur klinis tambahan.

10. REPRODUCIBILITY

10.1 GitHub Repository

https://github.com/ChandraDina/Hepatits-DataSet-Projek.git

10.2 Environment & Dependencies

Python, NumPy, Pandas, Scikit-learn, TensorFlow