Nama: Aulia Safira

NIM: 22611144

**UJIAN AKHIR SEMESTER**

**MODERN PREDICTION AND MACHINE LEARNING**

Course: Machine Learning

Topic: Supervised Learning and Model Deployment

Anda ditugaskan untuk melakukan tugas pembelajaran yang diawasi menggunakan kumpulan data dari Kaggle. Itu proyek melibatkan membandingkan beberapa metode pembelajaran yang diawasi dan menerapkan model terbaik menggunakan Heroku.

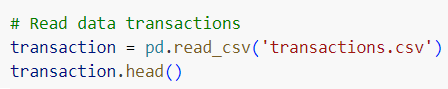
1. **Data Selection & Exploration**

Dari beberapa dataset yang tersedia, data yang akan digunakan pada proyek ini adalah UPI Payment Transactions Dataset. Alasan memilih dataset ini adalah untuk memprediksi status transaksi (sukses atau gagal) menggunakan model Regresi Logistik, KNN dan SVM pada dataset UPI Payment Transactions. Selain itu datanya terstruktur dan mudah untuk dikelola serta di analisis.

Langkah pertama *import* pandas dan numpy terlebih dahulu.



Selanjutnya panggil dataset transaction dan juga melihat 5 baris pertama dari data transaction menggunakan perintah head().

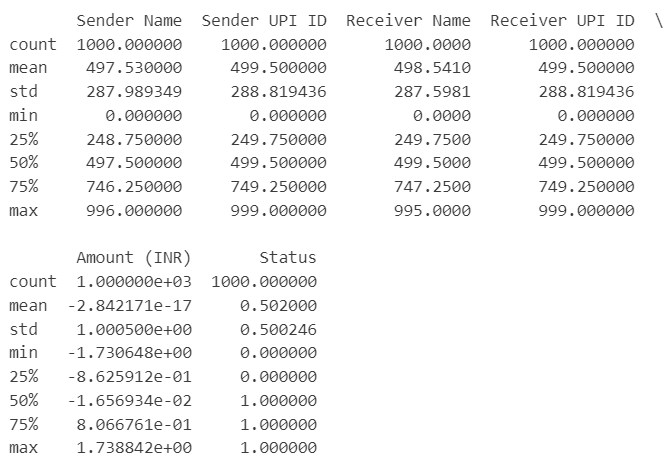




Data transaction memiliki 8 kolom yaitu Transaction ID, Timestamp, Sender Name, Sender UPI ID, Receiver Name, Receiver UPI ID, Amount, dan Status.

Berikutnya kita akan melihat statistic deskriptik dengan menggunakan perintah .describe().

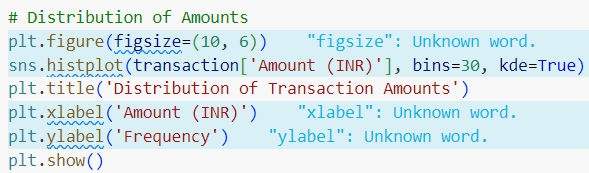




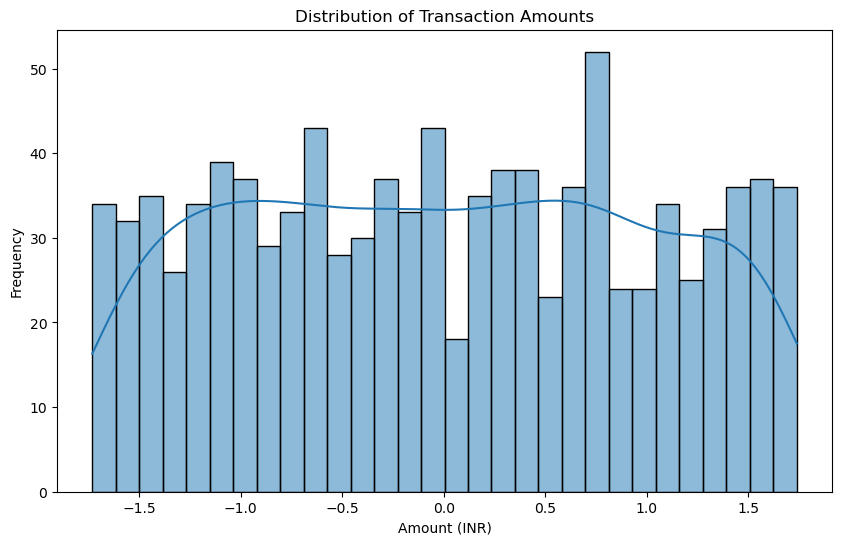
Dapat dilihat hasil statistik deskriptik yang terdiri dari count, mean, standar deviasi, nilai minimum, Q1, Q2, Q3, dan nilai maksimum untuk semua variabel.

Selanjutnya akan dilakukan visualisasi variabel Amount menggunakan histogram untuk melihat distribusi jumlah transaksi (Amount).



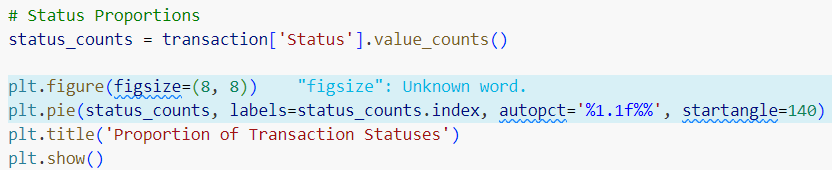


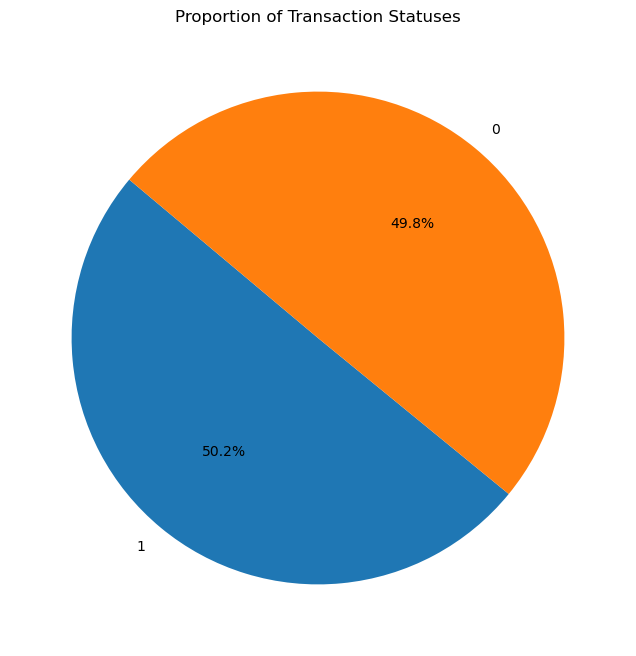
Diperoleh *output* sebagai berikut,



Distribusi jumlah transaksi terlihat cukup merata, dengan beberapa puncak frekuensi yang lebih tinggi pada interval tertentu. Dapat dilihat juga bahwa tidak ada transaksi tertentu yang mendominasi, berarti transaksi UPI memiliki variasi yang luas.

Kita juga akan melihat proporsi dari variabel Status menggunakan pie-cart.





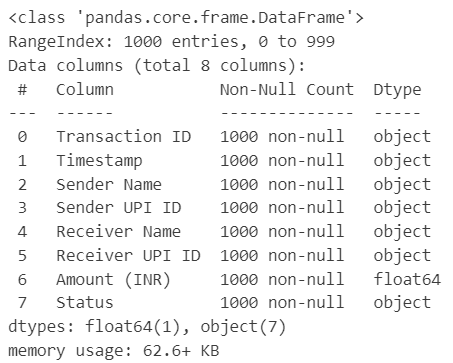
Proporsi transaksi yang berhasil (1) ada sebanyak 50.2%, sedangkan proporsi transaksi yang gagal (0) ada 49.8%. Proporsi antara transaksi berhasil dan gagal hampir sama.

1. **Data Preprocessing**

Pada proses preprocessing data akan dilakukan pengecekan missing values & pengecekan missing data, encoding categorical variabel dan scaling numerical features.

Pertama akan dilakukan pengecekan missing values dan untuk mendapatkan informasi umum data transaction menggunakan perintah .info().

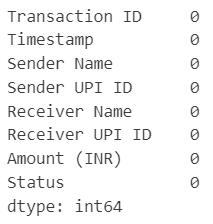




Dapat dilihat bahwa data transaction tidak memiliki missing value di semua kolomnya, serta kita dapat melihat juga tipe-tipe data dari masing-masing kolomnya.

Kemudian kita akan melakukan pengecekan missing data atau data hilang pada data transaction yang dapat menggunakan perintah .isnull() atau .isna() dan .sum() untuk menghitung jumlah missing values di setiap kolom.



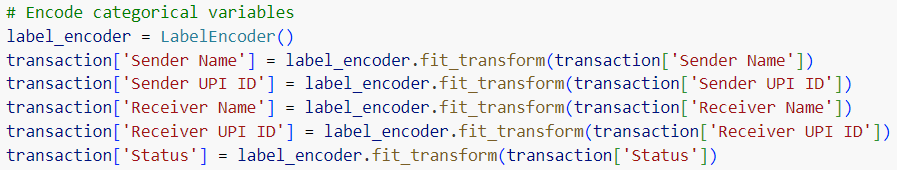


Dapat dilihat bahwa kedelapan kolom data transaction tidak memiliki data hilang atau NA.

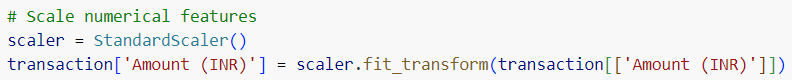
Langkah berikutnya akan dilakukan *encoding categorical* variabel dan scaling numerical features, tetapi sebelumnya *import* train\_test\_split, StandardScaler, dan LabelEncoder terlebih dahulu.



Pertama akan dilakukan *encoding categorical* pada variabel yang memiliki tipe data objek dengan menggunakan LabelEncoder angka integer. Variabel yang akan diubah yaitu: Sender Name, Sender UPI ID, receiver Name, Receiver UPI ID, dan Status.



Kemudian melakukan *scaling numeric features* pada variabel Amount (INR), agar variabel Amount (INR) berada dalam rentang tertentu, biasanya 0 dan 1.



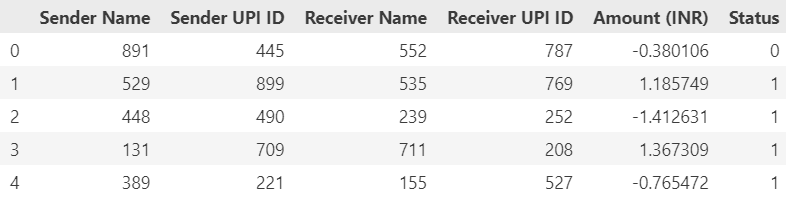
Selanjutnya, kita akan menghapus variabel Timestamp dan Transaction ID karena kita tidak akan menggunakan variabel tersebut menggunakan perintah .drop().



Panggil kembali data transaction untuk melihat hasil dari *encoding categorical variable* dan *scaling numeric features*.



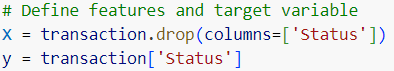
Diperoleh *output* sebagai berikut,



Pada data transaction dapat dilihat bahwa semua variabel telah berubah sesuai dengan encoding categorical variabel dan scaling numerical features. Pada variabel Status, angka 0 menunjukkan transaksi gagal dan angka 1 menunjukkan transaksi sukses.

1. **Model Training and Comparison**

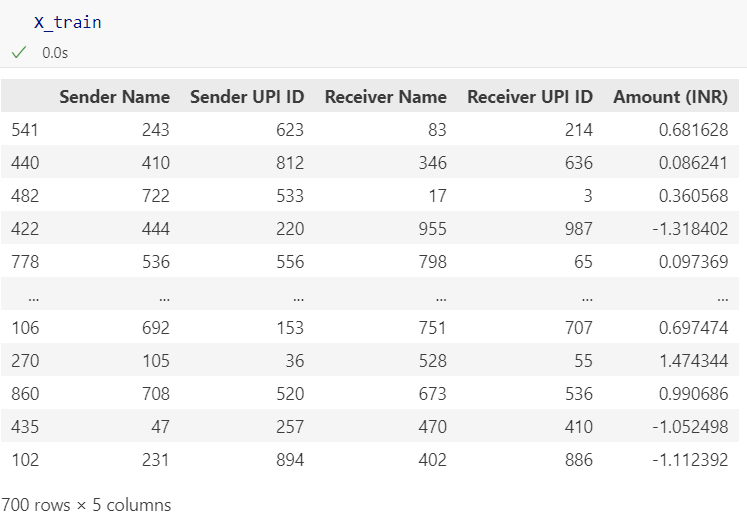
Langkah pertama adalah menentukan X dan y terlebih dahulu. Tujuan dari analisis ini adalah membangun model untuk memprediksi apakah sebuah transaksi akan sukses atau gagal. Oleh karena itu, variabel Status akan berperan sebagai y (variabel dependen/target).



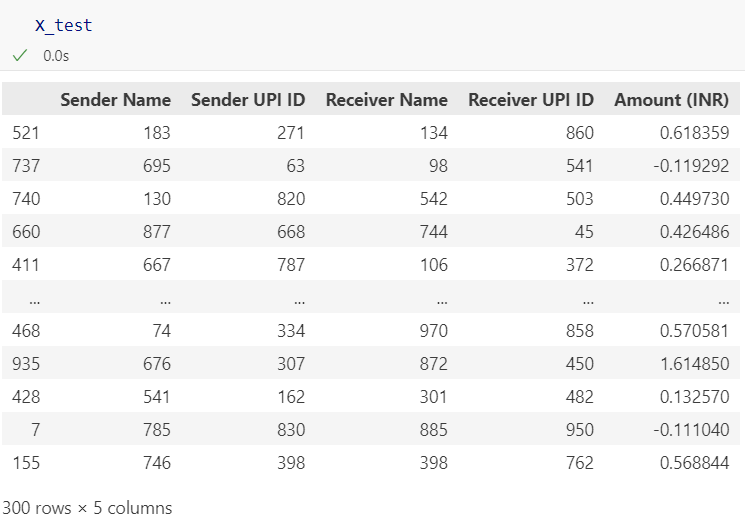
Selanjutnya akan kita bagi data transaction menjadi data train dan data test, dimana 70% sebagai data train (pelatihan) dan 30% untuk data test (pengujian).



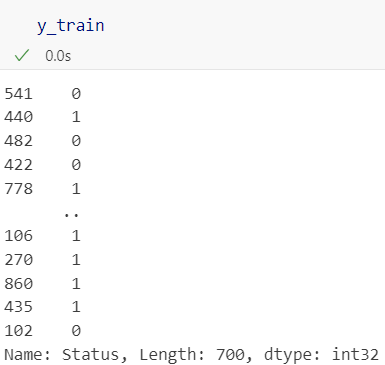
Diperoleh *output* X\_train sebagai berikut,



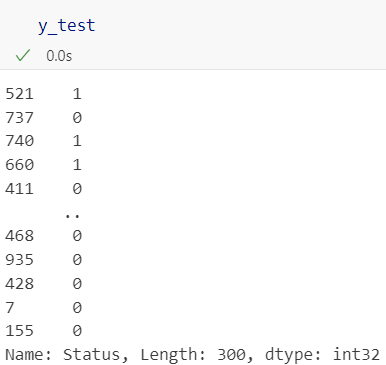
Diperoleh *output* X\_test sebagai berikut,



Diperoleh *output* y\_train sebagai berikut,



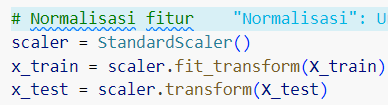
Diperoleh *output* y\_test sebagai berikut,



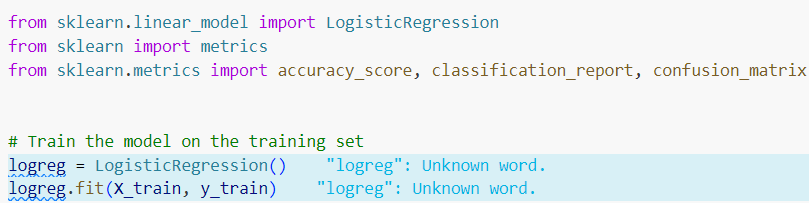
3 Model yang akan digunakan untuk memprediksi status transaksi UPI sukses atau gagal, yaitu Regresi Logistik, KNN dan SVM.

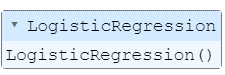
Model pertama yang akan kita gunakan adalah **Regresi Logistik**. Alasan memilih model ini adalah karena model Logreg umum digunakan, sederhana, dan sangat mudah dipahami. Regularisasinya dapat membantu mengurangi overfitting dan meningkatkan generalisasi model. Selain itu, model ini cocok untuk masalah klasifikasi biner seperti prediksi status transaksi yang sukses atau gagal.

Sebelum membangun model akan dilakukan normalisasi fitur pada data X\_train dan X\_test menggunakan *standard scaler*.

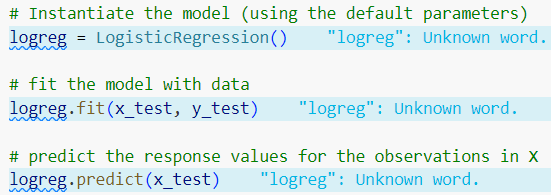


Untuk membangun model Logreg kita perlu meng*import* LogisticRegression, metrics, accuracy\_score, classification\_report, dan confusion\_matrix. Lalu melakukan pelatihan model menggunakan data train untuk X dan y.

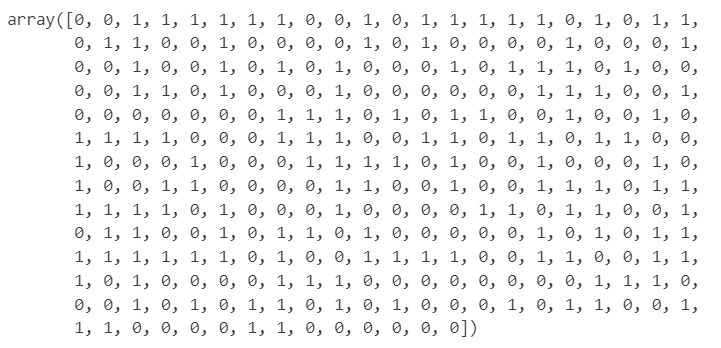




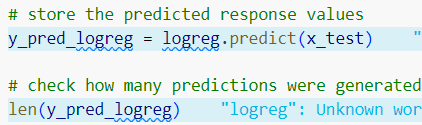
Kemudian kita melakukan instansiasi model, melatih model dan prediksi model Logreg.



Diperoleh hasil prediksi model Logreg sebagai berikut,

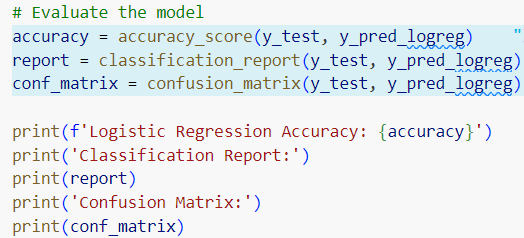


Kita akan melihat jumlah prediksi yang dihasilkan.

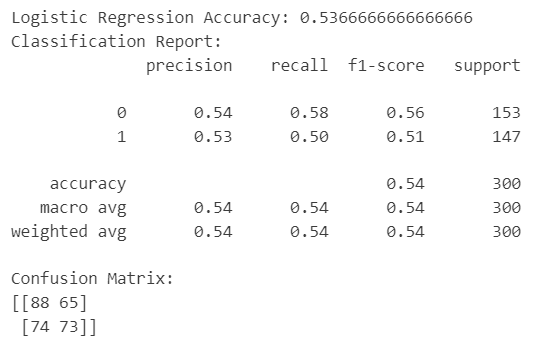


Jumlah prediksi yang dihasilkan ada sebanyak 300.

Kemudian kita mengevaluasi model Logreg dengan menggunakan *classification task* yang memuat akurasi, presisi, recall dan f1-score.

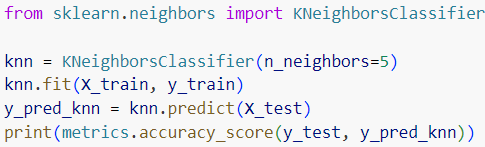


Diperoleh *output* sebagai berikut,

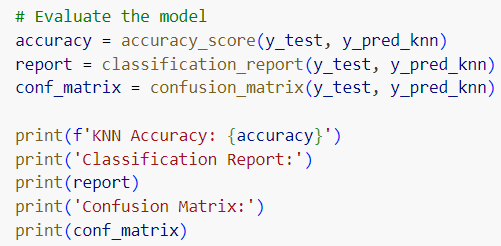


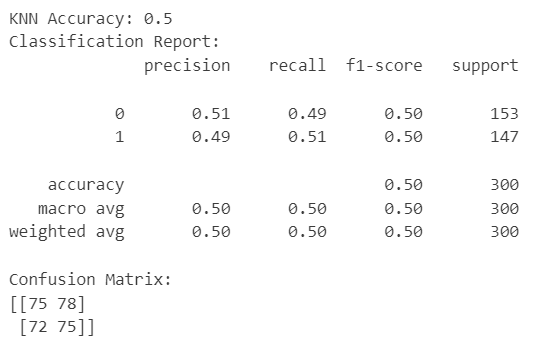
Diperoleh akurasi dari model Regresi Logistik adalah 53.66%, akurasi ini mengindikasi bahwa model belum cukup baik dan perlu dioptimalkan lagi.

Model kedua yang dipilih adalah KNN (K-Nearest Neighbors) dengan . Alasannya KNN dapat memberikan prediksi berdasarkan tetangga terdekatnya data transaction. Misalnya, transaksi yang mirip dalam hal pengirim, penerima, dan jumlah mungkin memiliki status yang sama.



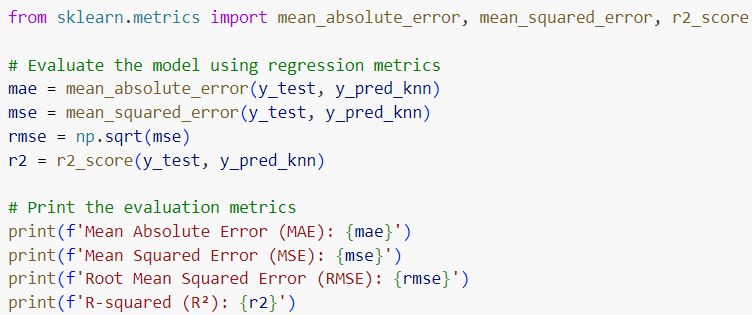
Selanjutnya kita akan melakukan evaluasi model KNN menggunakan *classification task*.

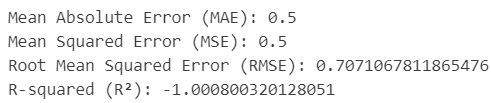




Diperoleh akurasi model KNN sebesar 50%, akurasi ini mengindikasi bahwa model belum cukup baik dan perlu dioptimalkan lagi.

Lalu kita juga melakukan evaluasi model dengan regression tasks untuk melihat nilai Mean Absolute Error, MSE, RMSE, dan R-squarednya.



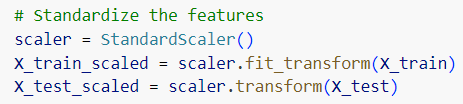


Berdasarkan metrik yang diberikan, nilai R² yang negatif menunjukkan bahwa model tidak mampu menangkap pola dalam data dengan baik. MAE dan RMSE menunjukkan tingkat kesalahan prediksi yang cukup tinggi, mengindikasikan bahwa model mungkin belum dioptimalkan.

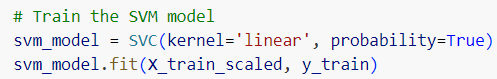
Model ketiga menggunakan model SVM. Alasannya SVM bisa bekerja dengan baik di dimensi yang lebih tinggi, fungsi kernelnya dapat menangani data yang tidak linear dengan memetakan data ke ruang dimensi yang lebih tinggi. Selain itu SVM memiliki regularisasi sehingga dapat membantu mengurangi overfitting.

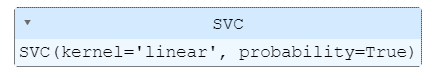


Sebelum membangun model akan dilakukan normalisasi fitur pada data X\_train dan X\_test menggunakan *standard scaler*.



Kita melatih model SVM menggunakan kernel linear.



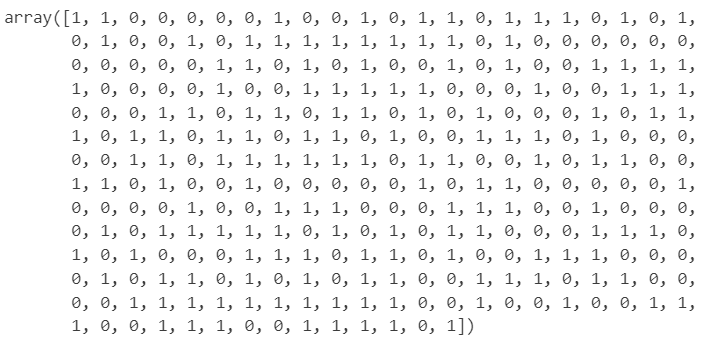


Setelah itu kita melakukan prediksi untuk model SVM.

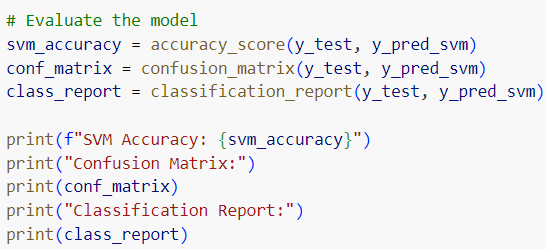


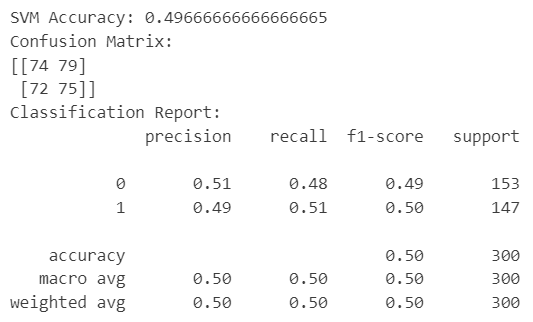


Diperoleh hasil prediskinya sebagai berikut,

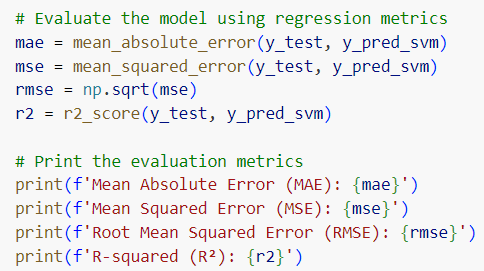


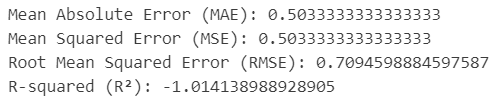
Selanjutnya mengevaluasi model dengan *classification tasks* dan *regression tasks*.





Diperoleh akurasi model SVM sebesar 49.66%, akurasi ini mengindikasi bahwa model belum cukup baik dan perlu dioptimalkan lagi.



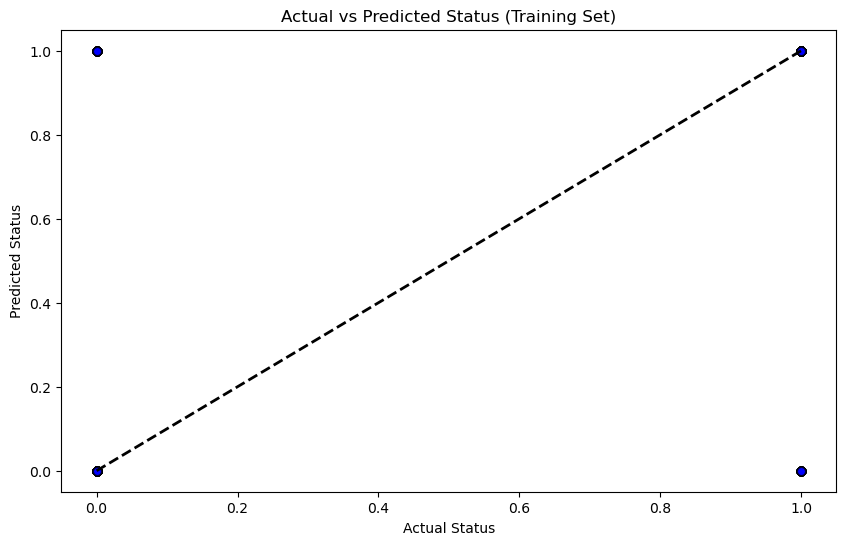


Berdasarkan metrik yang diberikan pada model SVM mengindikasikan bahwa model mungkin belum dioptimalkan.

Berikut ini adalah plot yang menampilkan garis hyperplane yang dihasilkan dari model SVM.



Menghasilkan *output* seperti berikut,

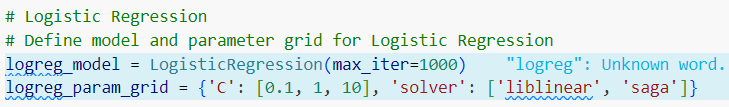


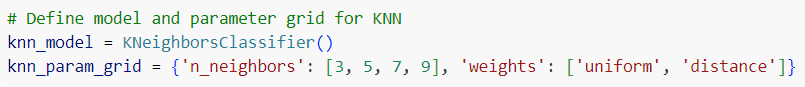
Ternyata hyperplane dapat memisahkan dengan tepat dua kelas yang berbeda.

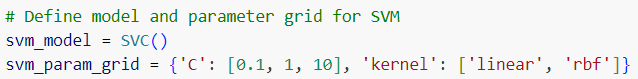
Selanjutnya untuk mengahsilkan model yang optimal kita akan melakukan Hyperparameter Tuning dan Cross-Validation untuk mencari hyperparameter terbaik untuk model dan mengevaluasi model.

Pertama akan didefinisikan model dan hyperparameter untuk ketiga model tersebut.



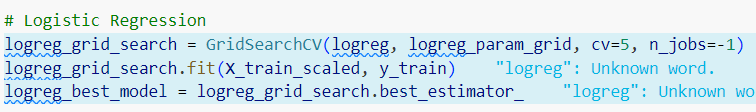


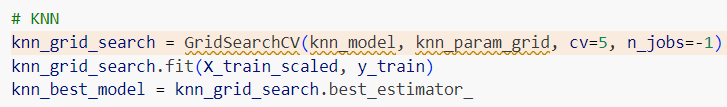


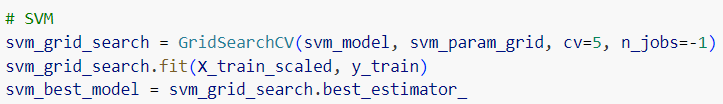


Kemudian melakukan Hyperparameter Tuning and Cross-Validation untuk ketiga model.

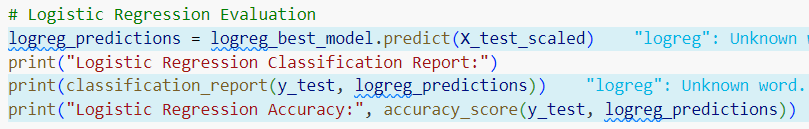


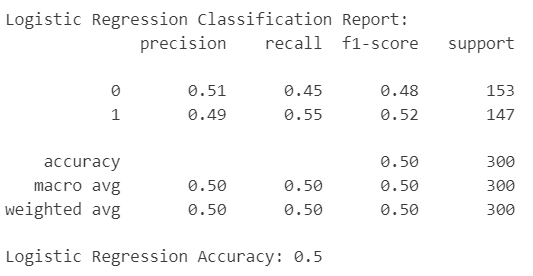




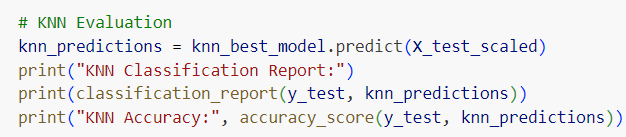


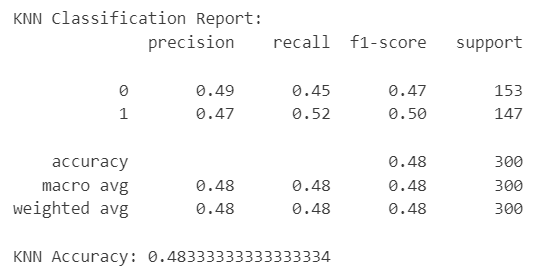
Setelah itu, kita evaluasi model yang telah di Hyperparameter Tuning and Cross-Validation.



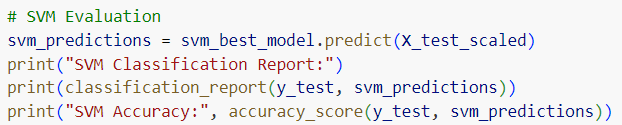


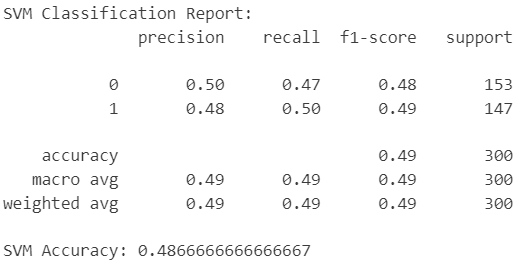
Untuk model Regresi Logistik diperoleh akurasinya 50%.





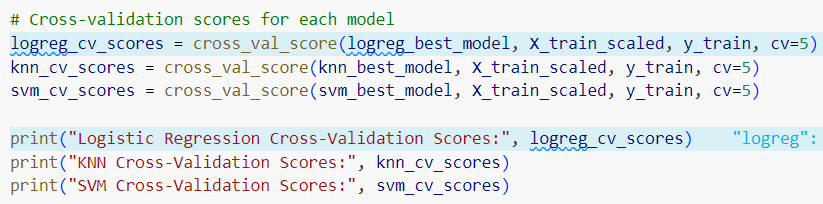
Akurasi dari model KNN adalah 48.33%.



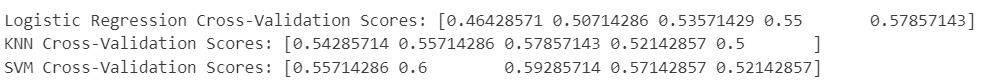


Akurasi dari model SVM sebesar 48.66%.

Lalu kita juga akan melihat skor dari Cross-validation untuk setiap model.



Diperoleh outputnya,



Model SVM memiliki performa terbaik dengan rata-rata cross-validation score tertinggi, yang menunjukkan bahwa hyperplane yang dipelajari SVM cenderung memisahkan kelas dengan lebih baik dibandingkan model lainnya. Model KNN menunjukkan performa yang cukup baik dengan variasi yang lebih sedikit dibandingkan model Logistic Regression.

1. Model Selection and Deployment

Jadi model terbaik yang akan dipilih adalah model SVM. Karena berdasarkan hasil cross-validation score, model SVM memiliki rata-rata tertinggi dibandingkan model KNN dan Regression Logistic. Cross-validation score tertinggi, yang menunjukkan bahwa hyperplane yang dipelajari pada model SVM cenderung memisahkan kelas dengan lebih baik dibandingkan model lainnya.