**Laporan Tugas Besar Machine Learning Tahap II**

Diajukan untuk Memenuhi Syarat Kelulusan Mata Kuliah Pembelajaran Mesin Pada Program Studi Informatika Jenjang Pendidikan S-1 Universitas Telkom

Logo

Description automatically generated

Disusun Oleh

|  |  |
| --- | --- |
| 1. | Muhammad Raihan Muhith (1301184245) |
| 2. | Mohammad Dwiantara Mahardhika (1301184467) |

**Fakultas Informatika**

**Universitas Telkom**

**2021**

1. **Formulasi Masalah**

Klasifikasi yang akan dilakukan memiliki tujuan untuk memprediksi apakah pelanggan tertarik untuk membeli kendaraan baru atau tidak berdasarkan data pelanggan di dealer. Dataset yang digunakan merupakan Kendaraan\_train.csv dan Kendaraan\_test.csv yang merupakan seluruh data pelanggan dan kendaraan pelanggan yang tersimpan dalam database dealer. Dataset ini diberikan oleh Dosen dari Mata Kuliah Machine Learning sebagai acuan mahasiswa dalam mengerjakan tugas clustering ini.

Pada bagian Pemodelan, Dataset yang digunakan adalah Kendaran\_train.csv yang telah di split menjadi 2 bagian yaitu data train dan data validation. Kendaraan\_test.csv digunakan untuk menguji ketepatan model dari data train dan data validation yang sudah dilakukan pemodelan.

Masalah lain yang dapat diselesaikan menggunakan klasifikasi pada data ini salah satunya adalah mengkategorikan pelanggan yang berhak diberikan pemotongan biaya baik pemotongan biaya beli kendaraan maupun pemotongan biaya pemeliharaan kendaraan dari dealer. Hal ini didasari oleh data dengan kolom lama berlangganan, perusahaan sebaiknya memberikan pemotongan biaya kepada pelanggan dengan lama berlangganan tertentu dan umur tertentu, atau bisa juga kepada pelanggan dengan lama berlangganan tertentu dan jumlah pembayaran premi tertentu. Hasil klasifikasi data tentu saja akan memberikan dampak positif kepada perusahaan, contohnya perusahaan akan mengetahui pelanggan mana yang kira-kira akan membeli kendaraan baru, sehingga perusahaan cukup memberikan rekomendasi kepada pelanggan-pelanggan tertentu saja.

1. **Eksplorasi Data Train**

Pada bagian ini, mahasiswa diwajibkan untuk mengetahui informasi-informasi dari dataset. Memahami data merupakan hal yang penting untuk dilakukan sebelum melakukan clustering. Terdapat beberapa tahapan yang saya lakukan pada bagian ini.

* 1. Data Eksploration and Understanding

Berikut merupakan gambaran dari dataset yang akan kita lakukan eksplorasi.

Table

Description automatically generated

Data tersebut memiliki keterangan kolom sebagai berikut.

|  |  |
| --- | --- |
| Nama Kolom | Keterangan |
| id | Id dari pelanggan |
| Jenis\_Kelamin | Jenis kelamin dari pelanggan |
| Umur | Umur Pelanggan |
| SIM | 1. : Tidak punya SIM, 2. : Punya SIM |
| Kode\_Daerah | Kode area tempat tinggal pelanggan |
| Sudah\_Asuransi | 1 : Pelanggan sudah memiliki asuransi kendaraan,  0 : Pelanggan belum memiliki  asuransi kendaraan |
| Umur\_Kendaraan | Umur kendaraan pelanggan |
| Kendaraan\_Rusak | Apakah Kendaraan pelanggan  pernah rusak atau tidak |
| Premi | Jumlah premi yang harus dibayarkan  per tahun. |
| Kanal\_Penjualan | Kode kanal untuk menghubungi  pelanggan (email, telpon, dll) |
| Lama\_Berlangganan | Sudah berapa lama pelanggan  menjadi klien perusahaan |
| Tertarik (diisi pada Tahap II tugas besar machine learning) | 1 : Pelanggan tertarik untuk membil kendaraan baru,  0 : Pelanggan tidak tertarik untuk  membeli kendaraan baru |

Table 1. Keterangan Kolom

Selanjutnya kita cek ada berapa baris dan kolom dari dataset yang kita punya.

Graphical user interface, text, application

Description automatically generated

Dari gambar diatas, kita tahu bahwa terdapat 28531 baris dan 12 kolom.

Setelah itu kami ingin mengetahui data statistika sederhana dari dataset yang kita punya.

Graphical user interface, application, table

Description automatically generated

Dari gambar diatas, kami mengetahui bahwa terdapat kolom-kolom yang hilang, hal ini disebabkan oleh tipe data yang dimiliki oleh kolom tersebut, tipe data yang bukan numerik tidak dapat dilakukan perhitungan statistic. Maka dari itu kita cek tipe data dari seluruh kolom yang ada pada dataset.

Text

Description automatically generated

Setelah itu kita identifikasi variable-variabel yang ada pada dataset

Table

Description automatically generated

Setelah mengetahui bahwa kolom Jenis\_Kelamin, Umur\_Kendaraan, dan Kendaraan\_Rusak memiliki tipe data non-numerik, kita lakukan transformasi data(mapping) terhadap variable tersebut. Disini kami mapping isi variabel tersebut berdasarkan kategori sebagai berikut.

|  |  |  |
| --- | --- | --- |
| Nama Kolom | Awal Isi Kolom | Setelah di ubah |
| Jenis Kelamin | Pria | 0 |
| Wanita | 1 |
| Umur\_Kendaraan | < 1 Tahun | 0 |
| 1-2 Tahun | 1 |
| > 2 Tahun | 2 |
| Kendaraan\_Rusak | Tidak | 0 |
| Pernah | 1 |

Table 2. Kategori Mapping

Maka hasil dari mapping adalah sebagai berikut.

Table

Description automatically generated

Sebelum masuk ke tahap selanjutnya, disini kami sepakat untuk menghapus kolom id dari dataset, karena kolom tersebut hanya bertugas sebagai index.

* 1. Data Cleansing

Data yang baik merupakan data yang bersih dari missing value dan outliers, Missing Value merupakan data yang hilang atau tidak diisi, umumnya missing value pada dataset berbentuk nan, namun ada juga missing value yang berbentuk karakter “-“. Sedangkan Outliers atau data pencilan merupakan data yang ganjil atau data yang aneh. Data yang memiliki Missing value dan outliers dapat berpengaruh terhadap pemodelan, sehingga perlu dilakukan handling terhadap dua kasus tersebut.

* Identifikasi dan Handling Missing value

Missing value dapat dilihat dengan menggunakan sintaks berikut.

Table

Description automatically generated

Dapat kita ketahui bahwa seluruh kolom memiliki missing value, hal ini perlu kita atasi. Kami memilih untuk mereplace missing value tersebut dengan nilai rata-rata dan modus variable tersebut. Untuk varibel kategorical, missing value direplace dengan modus, sedangkan variable numerik, missing value direplace dengan rata-rata(mean). Hal ini dilakukan untuk menjaga kestabilan distribusi data dari setiap variable.

Graphical user interface, text, application

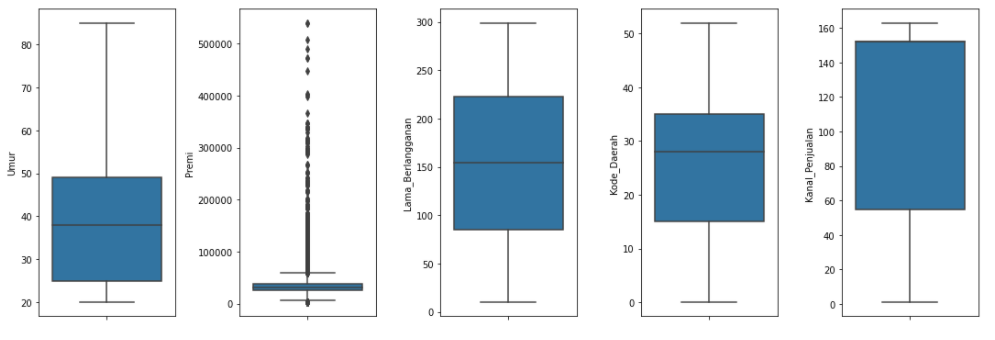
Description automatically generated

* Identifikasi dan Handling Outliers

Data pencilan dapat kita identifikasi menggunakan boxplot untuk data numerik dan ketegorik yang memiliki nilai kontinu, dan menggunakan countplot untuk data kategorik nominal.

Chart, box and whisker chart

Description automatically generated



Chart, bar chart

Description automatically generated

Pada countplot, dapat dilihat seluruh variable tidak memiliki nilai sumbu x yang ganjil/aneh, sehingga semua variable yang dilakukan countplot tidak memiliki data pencilan. Sedangkan untuk boxplot bulatan-bulatan disekitar boxplot merupakan data-data yang kurang dari batas bawah dan lebih dari batas atas pada variabel tersebut, hal ini dapat menjadi acuan dalam mendeteksi outliers.

Namun data-data tersebut belum dapat kita identifikasi sebagai outliers, karena bisa saja data-data tersebut memang kredibel. Mengingat bahwa kami belum memiliki kemampuan dalam mengidentifikasi bulatan-bulatan hitam disekitaran box itu merupakan data pencilan atau bukan, maka kami sepakat untuk membuang bulatan-bulatan hitam tersebut. Kami membuangnya berdasarkan interquartile range variable tersebut.

Text

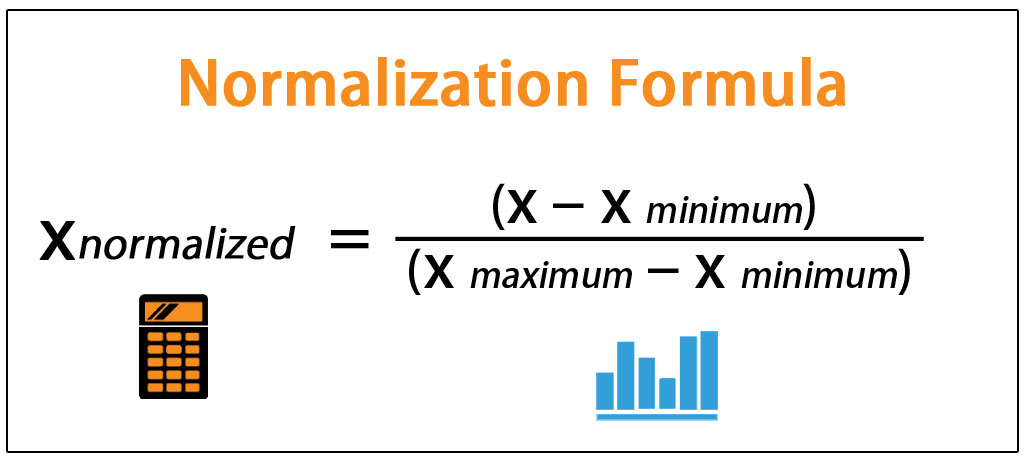
Description automatically generated Text

Description automatically generated

Kedua gambar diatas merupakan fungsi yang kami definisikan untuk hanya mengambil data-data yang dalam range ≥ batas bawah dan ≤ batas atas.

* 1. Feature Engineering

Pada bagian ini, seluruh data dari semua kolom akan dilakukan scalling. Scalling merupakan sebuah metode untuk mentranformasikan nilai data menjadi nilai dalam range tertentu. Disini kami menggunakan scalling menggunakan cara MinMax Normalization. MinMax Normalization akan mentransformasikan data kedalam range 0 – 1. Berikut merupakan cara perhitungan MinMax Normalization.



Berikut merupakan data dari seluruh kolom kecuali yang telah di scale.

A screenshot of a computer

Description automatically generated with low confidence

1. **Eksplorasi Data Test**

Tahapan yang dilakukan pada bagian ini sama seperti pada bagian sebelumnya, berikut tahapan-tahapan yang kami lakukan.

* 1. Data Eksploration and Understanding

Berikut gambaran dari dataset yang akan kita lakukan eksplorasi.

A screenshot of a computer

Description automatically generated with medium confidence

Keterangan kolom dapat dilihat pada Table 1 dibagian Eksplorasi Data Train. Selanjutnya kita cek ada berapa baris dan kolom dari dataset yang kita punya.

Graphical user interface, text, application, Word, chat or text message, website

Description automatically generated

Dari gambar diatas kita tahu bahwa terdapat 47639 baris dan 11 kolom. Setelah itu kami ingin mengetahui data statistika sederhana dari dataset yang kita punya.

Graphical user interface, application, table

Description automatically generated

Seperti halnya pada statistika sederhana dibagian Eksplorasi data train, informasi yang ditampilkan hanya dari kolom-kolom yang memiliki tipedata numerik saja. Maka dari itu kita perlu cek tipe data dari semua kolom pada dataset.

Text

Description automatically generated

Kami tidak perlu lagi mengidentifikasi variable-variabel yang ada pada dataset ini, karena variable-variabel yang terdapat pada data test sama dengan yang ada pada data train, bedanya data test sudah tidak memiliki id. Setelah mengetahui bahwa kolom Jenis\_Kelamin, Umur\_Kendaraan, dan Kendaraan\_Rusak memiliki tipe data non-numerik, kita lakukan transformasi data(mapping) terhadap variable tersebut. Kategori mapping dapat dilihat di Tabel 2 pada bagian Eksplorasi Data Train.

* 1. Data Cleansing

Sama halnya seperti pada data train, data test juga perlu kami lakukan data cleansing, karena data yang tidak bersih (memiliki null values dan outliers) akan mempengaruhi pemodelan.

* Identifikasi dan Handling missing value

Missing value dapat dilihat dengan menggunakan sintaks berikut.

Text

Description automatically generated

Dari gambar diatas, ternyata data test sudah bersih dari missing value, sehingga tidak ada yang perlu kami atasi.

* Identifikasi dan Handling outliers

Data pencilan dapat kita identifikasi menggunakan boxplot untuk data numerik dan ketegorik yang memiliki nilai kontinu, dan menggunakan countplot untuk data kategorik nominal.

Chart, box and whisker chart

Description automatically generated

Chart, box and whisker chart

Description automatically generated

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

Pada countplot, dapat dilihat seluruh variable tidak memiliki nilai sumbu x yang ganjil/aneh, sehingga semua variable yang dilakukan countplot tidak memiliki data pencilan. Sedangkan untuk boxplot bulatan-bulatan disekitar boxplot merupakan data-data yang kurang dari batas bawah dan lebih dari batas atas pada variabel tersebut, hal ini dapat menjadi acuan dalam mendeteksi outliers.

Namun data-data tersebut belum dapat kita identifikasi sebagai outliers, karena bisa saja data-data tersebut memang kredibel. Mengingat bahwa kami belum memiliki kemampuan dalam mengidentifikasi bulatan-bulatan hitam disekitaran box itu merupakan data pencilan atau bukan, maka kami sepakat untuk membuang bulatan-bulatan hitam tersebut. Metode handling outliers pada dataset ini sama seperti pada data train, yaitu membuangnya berdasarkan interquartile range variable tersebut.

* 1. Feature Engineering

Pada bagian ini, sama halnya pada Feature Engineering data train, seluruh data dari semua kolom akan dilakukan scalling. Metode yang kami gunakan juga sama seperti sebelumnya yaitu scaling dengan menggunakan rumus MinMax Normalization.

Berikut merupakan data dari seluruh kolom yang telah di scale.

Table

Description automatically generated

* 1. Data Splitting

Bagian ini merupakan metode pembagian seluruh dataset menjadi data train, data test, dan data validation. Berhubung data train dan data test sudah dipisah, maka splitting yang kami lakukan adalah membagi data train dan data validation. Rasio yang kami pilih dalam membagi data train dan data validation adalah 80% dan 20%.

Graphical user interface, text, application, Word, website

Description automatically generated Graphical user interface, application

Description automatically generated

Dilihat dari gambar diatas, kita tahu bahwa jumlah baris pada data train yang sudah mencapai Feature Engineering adalah sebanyak 230690 baris dan jumlah baris pada data test yang sudah mencapai Feature Engineering adalah sebanyak 46368 baris. Sehingga jumlah baris dari data validation ingin kami samakan dengan jumlah baris data test.

Table

Description automatically generated

Berikut jumlah baris dari ketiga data setelah dilakukan splitting

Graphical user interface, application, Word

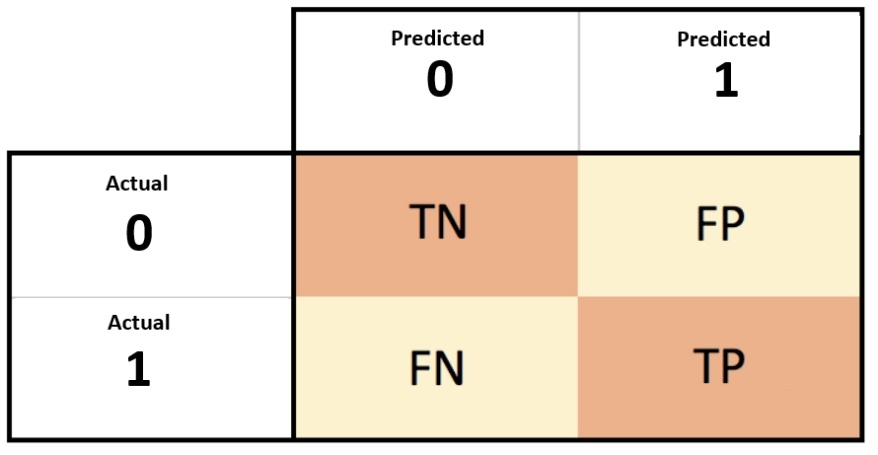
Description automatically generated

184552 baris data train, 46138 baris data validation, dan 46368 baris data test.

1. **Pemodelan**

Pada tahap ini dilakukan penentuan model terbaik dengan membandingkan akurasi prediksi model terhadap data validation. Selain itu confusion matrix serta nilai precision, recall dan F1-score menjadi bahan pertimbangan tambahan dalam menentukan model terbaik.

* **Akurasi** adalah presentase prediksi yang dilakukan dengan benar oleh model yang digunakan.
* **Confusion Matrix** adalah tabel yang sering digunakan untuk mendeskripsikan performa model klasifikasi pada sekumpulan data yang nilai sebenarnya diketahui.



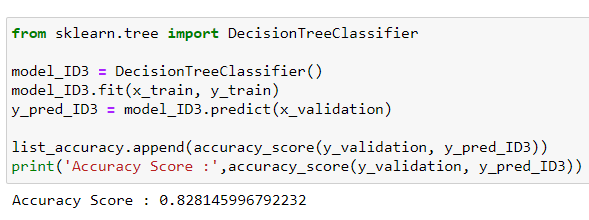
* **Precision** merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positf.
* **Recall** merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.
* **F1-Score** merupakan perbandingan rata-rata presisi dan recall yang dibobotkan.

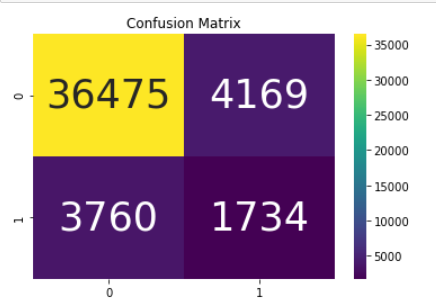


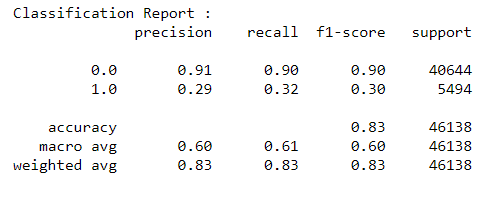
Model klasifikasi yang akan dibandingkan adalah Decision Tree, K-Nearest Neighbors dan Random Forest Classifier. Semua model dibangun menggunakan parameter default dari library masing-masing model klasifikasi.

* **Decision Tree**

Dengan menggunakan model Decision Tree didapatkan akurasi sebesar 82,8%. Berikut merupakah akurasi hasil prediksi, confusion matrix dan classification report dengan model Decision Tree.

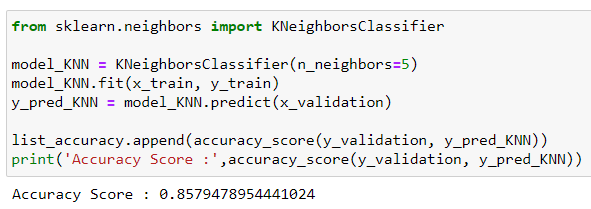


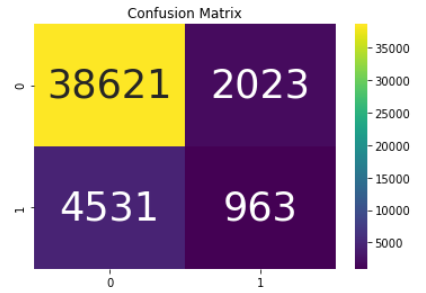


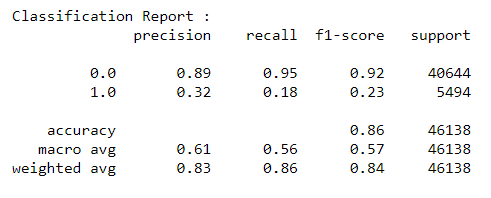


* **K-Nearest Neighbors**

Dengan menggunakan model Decision Tree Classifier didapatkan akurasi sebesar 85,7%. Berikut merupakah akurasi hasil prediksi, confusion matrix dan classification report dengan model KNN.

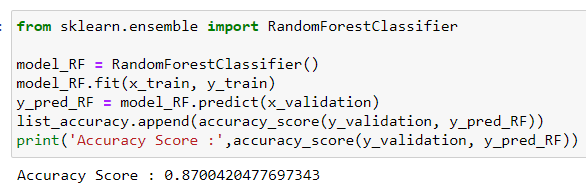


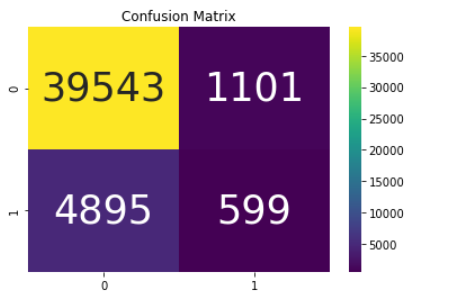


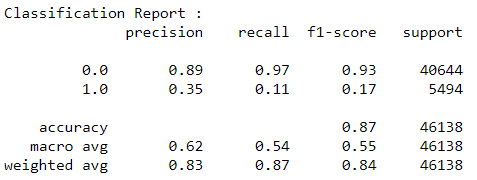


* **Random Forest**

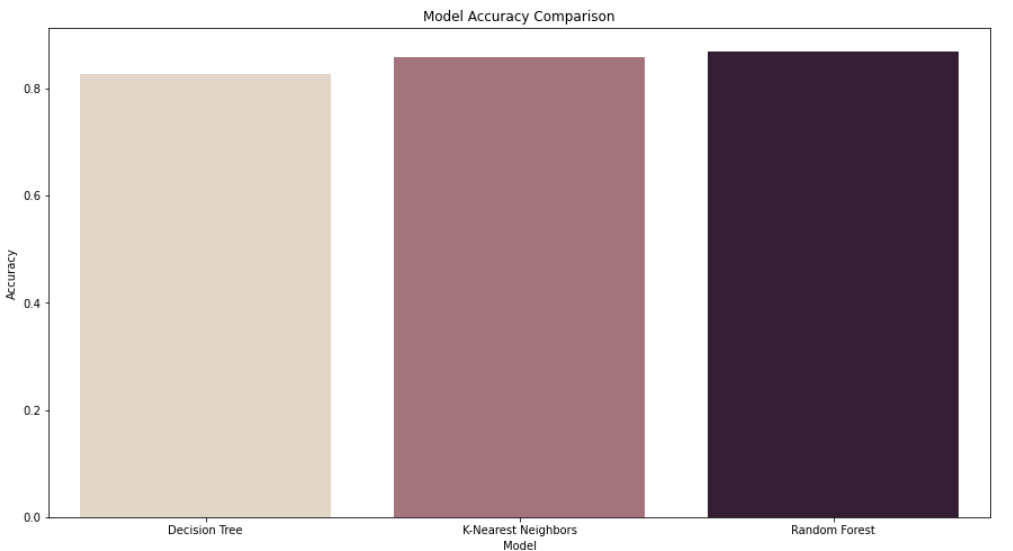
Dengan menggunakan model Decision Tree Classifier didapatkan akurasi sebesar 87%. Berikut merupakah akurasi hasil prediksi, confusion matrix dan classification report dengan model Random Forest.





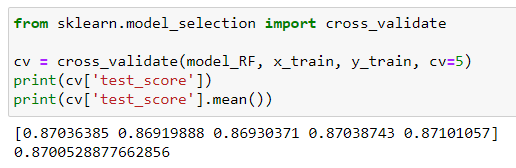


Dari hasil perbandingan model didapatkan bahwa model Random Forest memiliki akurasi tertinggi yakni 87%.



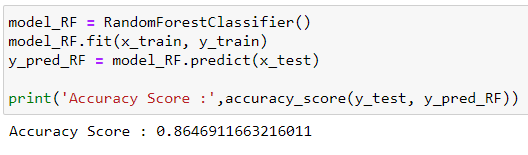
1. **Evaluasi**

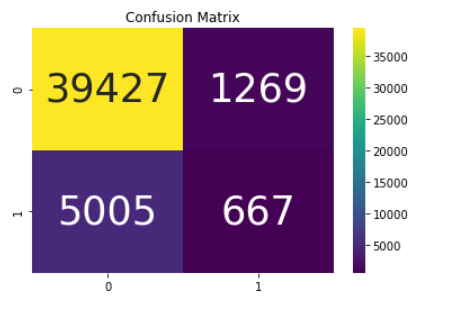
Setelah melakukan pemilihan model terbaik, tahap selanjutnya adalah evaluasi model dengan melihat hasil prediksi model terhadap data test. Sebelum melakukan evaluasi tersebut dilakukan K-Fold cross validation untuk data train menggunakan model Random Forest. Hal ini dilakukan untuk melihat apakah akurasi yang didapatkan dari model stabil untuk berbagai dataset atau akurasi yang didapatkan sebelumnya hanya berlaku untuk data validation. K-Fold cross validation dilakukan menggunakan fungsi cross\_validate dari library sklearn dengan K=5.

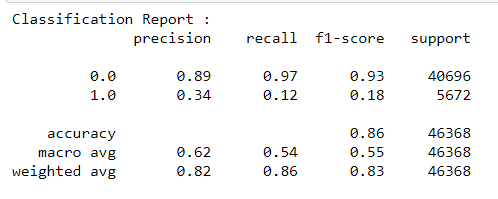


Dari hasil cross validation didapatkan rata-rata akurasi berada pada angka 87%, nilai yang sama didapatkan saat melakukan prediksi terhadap data validation. Berdasarkan hasil tersebut dapat disimpulkan bahwa model yang digunakan stabil.

Selanjutnya dilakukan evaluasi model dengan melakukan prediksi terhadap data test. Berikut merupakan hasil akurasi, confusion matrix dan classification report hasil evaluasi.



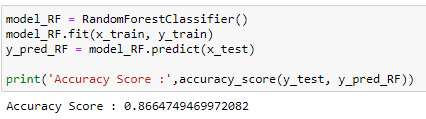


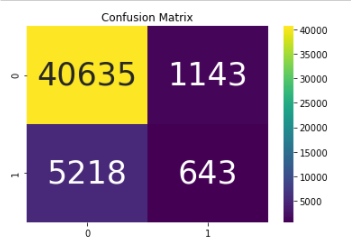


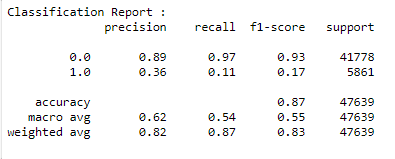
Dari hasil evaluasi yang dilakukan akurasi yang didapat adalah 86.4%, akurasi tersebut tidak terlalu berbeda dengan akurasi saat melakukan validasi model yakni 87%.

1. **Eksperimen**

Setelah mendapatkan hasil prediksi data test, kami melakukan beberapa eksperimen untuk mendapatkan hasil yang berbeda. Salah satunya kami melakukan prediksi data test dengan model random forest tetapi pada tahap preprocessing data tidak dilakukan outlier handling baik pada data train maupun pada data test. Berikut hasil yang didapatkan dari percobaan tersebut.







Dari hasil prediksi didapatkan akurasi 86.6% tidak jauh berbeda dengan prediksi dataset dengan outlier handling yakni 86.4%. Dari hal tersebut dapat disimpulkan bahwa pengaruh outlier pada dataset terhadap akurasi prediksi cukup kecil.

1. **Kesimpulan**