

**LAPORAN KECERDASAN KOMPUTASIONAL – IF184503**

**Klasifikasi data prediksi tanggapan asuransi kendaraan**

**Zakiya Azizah C 05111840000080**

**M. Farras Pangestu 05111840000134**

**Excel Deo C 05111840000117**

# 

# ABSTRAK

Dataset yang digunakan berasal dari website Kaggle tentang penawaran asuransi kendaraan sebuah perusahaan di India. Dataset berisi data responsi pelanggan sebuah perusahaan asuransi terhadap tawaran asuransi mobil. Beberapa metode *pre-processing* dilakukan untuk menyiapkan data. Digunakan beberapa model dalam uji coba dengan skenario yang berbeda antara lain support vector machine (SVM), K nearest neighbor (KNN), decision tree, dan jaringan saraf buatan berupa multi layer perceptron (MLP). Hasil dari setiap uji coba akan dievaluasi dengan metrics yang ditentukan yaitu F1. Hasil akhir menunjukkan bahwa model terbaik adalah decision tree dengan F1 score 82.37% dalam waktu latih 0.03 detik.

**Kata Kunci**: ANN, Dataset, Klasifikasi, Machine Learning

# 

# DAFTAR ISI

[ABSTRAK](#_heading=h.30j0zll) 2

[DAFTAR ISI](#_heading=h.3znysh7) 3

[PENDAHULUAN](#_heading=h.3dy6vkm) 1

[Latar Belakang](#_heading=h.4d34og8) 1

[Rumusan Masalah](#_heading=h.2s8eyo1) 1

[DESAIN DAN IMPLEMENTASI](#_heading=h.3rdcrjn) 2

[Persiapan Data](#_heading=h.26in1rg) 2

[Skenario Uji Coba](#_heading=h.tvpum1ht9kce) 4

[HASIL UJI COBA DAN DISKUSI](#_heading=h.1ksv4uv) 5

[Eksplorasi Data](#_heading=h.44sinio) 5

[Hasil Uji Coba Skenario](#_heading=h.ih7s89qsa294) 9

[Pembahasan dan Kesimpulan](#_heading=h.2jxsxqh) 14

[DAFTAR PUSTAKA](#_heading=h.z337ya) 16

# 

# 

# PENDAHULUAN

## Latar Belakang

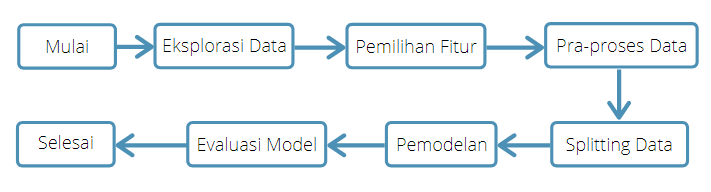
Laporan ini dibuat untuk memenuhi tugas akhir mata kuliah Kecerdasan Komputasional. Dataset yang digunakan berasal dari website Kaggle tentang penawaran asuransi kendaraan sebuah perusahaan di India. Output yang ingin dicapai adalah prediksi apakah seseorang akan menerima tawaran asuransi kendaraan. Selain itu, eksplorasi data akan dilakukan untuk membantu menganalisis data guna mengoptimalkan hasil dari klasifikasi.

## Rumusan Masalah

1. Fitur apakah yang kurang mempengaruhi keputusan?
2. Apakah jumlah data pada setiap kelas target merata? Jika tidak, bagaimana mengatasinya?
3. Bagaimana karakteristik dari masing-masing target?
4. Metrics evaluasi apa yang cocok untuk kasus klasifikasi ini?

# 

# DESAIN DAN IMPLEMENTASI



Gambar 1. Alur Metodologi

Percobaan diawali dengan eksplorasi data untuk menganalisis data yang dimiliki. Dari eksplorasi data tersebut, akan disimpulkan fitur mana saja yang dianggap berpengaruh terhadap target. Setelahnya, data perlu di pra-proses sebelum digunakan sebagai data latih. Hal ini berguna untuk membersihkan dan menyiapkan data agar siap dipelajari oleh model. Pre-proses meliputi pengkodean data non-numerik, normalisasi data, dan balancing data. Setelah data disiapkan melalui pre-proses, data akan dibagi menjadi data latih dan data uji. Data latih adalah data yang digunakan untuk dipelajari oleh model, sedangkan data uji merupakan data berbeda yang digunakan untuk menguji model. Dalam pemodelan, beberapa model diuji coba dengan beberapa skenario berbeda. Hasil dari setiap uji coba akan dievaluasi dengan metrics yang ditentukan. Detail tentang data dan skenario uji coba akan dibahas pada sub-bab setelah ini.

## Persiapan Data

Data yang digunakan merupakan data responsi pelanggan sebuah perusahaan asuransi terhadap tawaran asuransi mobil. Data ini digabungkan dengan data pelanggan dari produk asuransi kesehatan perusahaan tersebut. Data terdiri dari 12 kolom dengan detail seperti pada Tabel 1. Data ini memiliki target pada kolom Response yang akan diprediksi berdasarkan berdasarkan kolom-kolom lainnya.

Total baris pada data mentah berjumlah 381,109 baris tanpa *missing value*. Sehingga, dalam percobaan ini tidak perlu dilakukan penanganan *missing value*. Terdapat 3 kolom bertipe Object/String yang akan dikodekan menjadi angka yaitu Gender, Vehicle\_Age, dan Vehicle\_Damage. Data memiliki 334,339 baris dengan nilai Response 0 dan 46710 baris dengan nilai Response 1. Ketidakseimbangan data dapat menyebabkan model cenderung lebih mempelajari data yang lebih banyak. Sehingga, perlu dilakukan penyeimbangan pada jumlah *record* kedua kelas tersebut. Selain itu, *range* nilai yang berbeda antar kolom, seperti contoh antara Age dan Annual\_Premium, perlu diatasi dengan melakukan normalisasi nilai. Normalisasi yang dilakukan merubah *range* nilai setiap kolom menjadi antara 0 sampai 1.

Tabel 1. Deskripsi kolom pada data

|  |  |  |
| --- | --- | --- |
| Nama Kolom | Tipe Data | Deskripsi |
| Id | Integer | Nomor ID unik pelanggan |
| Gender | Object/String | Jenis kelamin pelanggan |
| Age | Integer | Umur pelanggan |
| Driving\_License | Integer | Kepemilikan SIM pelanggan, 0: belum memiliki SIM, 1: memiliki SIM |
| Region\_Code | Float | Kode unik daerah pelanggan |
| Previously\_Insured | Integer | Kepemilikan asuransi kendaraan pelanggan, 0: pelanggan memiliki asuransi kendaraan, 1: pelanggan tidak memiliki asuransi kendaraan |
| Vehicle\_Age | Object/String | Umur kendaraan pelanggan. Terbagi menjadi 3 kelas: < 1 Year, 1-2 Year, > 2 Years |
| Vehicle\_Damage | Object/String | Pernah atau tidaknya kendaraan pelanggan mengalami kerusakan sebelumnya, 0: belum, 1: pernah |
| Annual\_Premium | Float | Premi tahunan asuransi yang harus dibayar pelanggan |
| Policy\_Sales\_Channel | Float | Kode media untuk menghubungi pelanggan yang disamarkan. Contoh: melalui email, melalui telepon, dll |
| Vintage | Integer | Jumlah hari pelanggan berhubungan dengan perusahaan |
| Response | Integer | Ketertarikan pelanggan terhadap asuransi kendaraan, 0: tidak tertarik, 1: tertarik |

## Skenario Uji Coba

## Dilakukan beberapa skenario uji coba untuk mengoptimasi hasil dari klasifikasi ini. Proporsi pembagian data menjadi data latih dan uji dicoba dengan beberapa skenario yaitu 30%, 40%, dan 50% data uji dari keseluruhan data. Data lalu akan dipelajari oleh beberapa model yaitu support vector machine (SVM), K nearest neighbor (KNN), decision tree, dan multi layer perceptron (MLP). Selain itu, setiap model akan dicoba dengan kombinasi *hyperparameter* yang berbeda-beda. Hasil dari setiap skenario uji coba akan dibahas lebih lanjut pada

# 

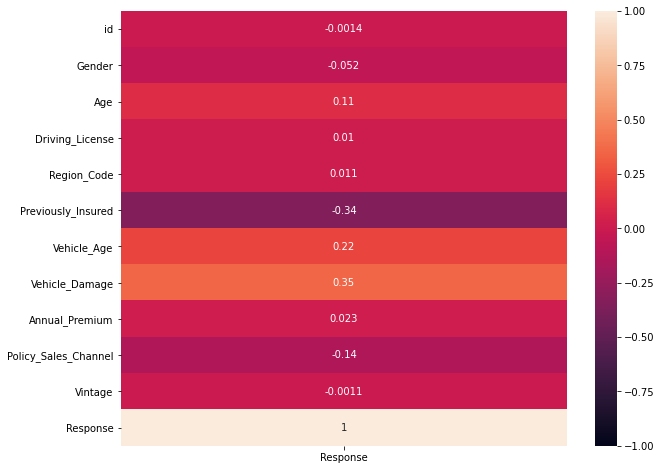
# HASIL UJI COBA DAN DISKUSI

## Eksplorasi Data

Eksplorasi data dilakukan untuk menganalisis data. Analisis dibantu dengan pertanyaan yang telah dirumuskan pada sub-bab Rumusan Masalah. Sub-bab ini akan mengeksplorasi data berdasarkan dan digunakan untuk menjawab pertanyaan yang telah dirumuskan sebelumnya.

1. *Fitur apa yang kurang mempengaruhi pengambilan keputusan?*

Dalam menjawab ini, analisis dilakukan menggunakan perhitungan korelasi Pearson. Korelasi pearson menghitung korelasi linear antara 2 variabel dengan nilai +1 berarti korelasi positif yang kuat dan -1 korelasi negatif yang kuat. Korelasi positif berarti ketika satu variabel mengalami kenaikan nilai, maka variabel lainnya juga mengalami kenaikan nilai, dan sebaliknya. Korelasi setiap kolom dengan Response dapat dilihat pada Gambar 2.

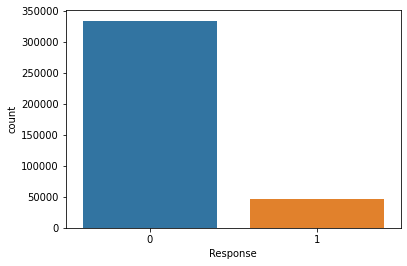


Gambar 2. Heatmap Korelasi Setiap Kolom dengan Kolom Response

Dari korelasi pada Gambar 2 terlihat bahwa beberapa kolom memiliki korelasi yang sangat rendah terhadap Response. Dari rendahnya korelasi ini, dapat disimpulkan bahwa fitur tersebut kurang mempengaruhi target Response. Dengan menaruh threshold (batas) senilai 0.1 dari nilai absolute korelasi, beberapa kolom fitur yang dianggap kurang mempengaruhi adalah Gender, Driving\_License, Region\_Code, Annual\_Premium, dan Vintage. Dengan catatan bahwa kolom Id diabaikan.

1. *Apakah jumlah data pada setiap kelas target merata? Jika tidak, bagaimana mengatasinya?*

Seperti yang sudah dibahas sebelumnya, jumlah data pada kedua target tidak merata dengan 334,339 data pada kelas 0 atau No dan 46,710 data pada kelas 1 atau Yes. Gambar 3 menggambarkan ketidakseimbangan tersebut. Dari gambar tersebut terlihat jelas bahwa data 0 jauh lebih banyak dari data 1, sehingga data perlu diseimbangkan jumlahnya.



Gambar 3. Plot Batang Jumlah Data pada Kelas Target

Dalam menyeimbangkan jumlah data, metode undersampling dilakukan pada data. Undersampling adalah metode penyeimbangan data dengan membuang sejumlah data pada kelas yang berlebih. Dalam kasus ini, beberapa data pada kelas 0 akan dibuang secara acak. Keputusan menggunakan undersampling didasari oleh jumlah data yang banyak. Sehingga, walaupun dibuang beberapa, jumlah data masih tetap cukup untuk dipelajari. Selain itu, jika oversampling dilakukan, maka data akan berjumlah dua kali data mayoritas. Sehingga, jika oversampling dilakukan, akan menghasilkan lebih dari 600 ribu data. Jumlah data yang besar ini akan memberatkan dan memperlambat waktu belajar model klasifikasi. Mengingat keterbatasan *resource* dan waktu, maka undersampling dipilih dan menghasilkan total 93,420 data dengan 46,710 pada masing-masing kelas.

1. *Bagaimana karakteristik dari masing-masing target?*

Eksplorasi terhadap target bernilai 1 atau Yes dilakukan untuk menganalisis karakteristiknya dan membantu menjawab pertanyaan ini. Hal ini dikarenakan data Yes merupakan minoritas. Sehingga, untuk data berkelas 0 atau No, memiliki kecenderungan karakteristik di luar karakteristik data Yes. Analisis dilakukan berdasarkan beberapa kolom fitur dengan mengesampingkan kolom-kolom yang dianggap kurang berpengaruh pada pertanyaan A dan sebelum dilakukan undersampling.

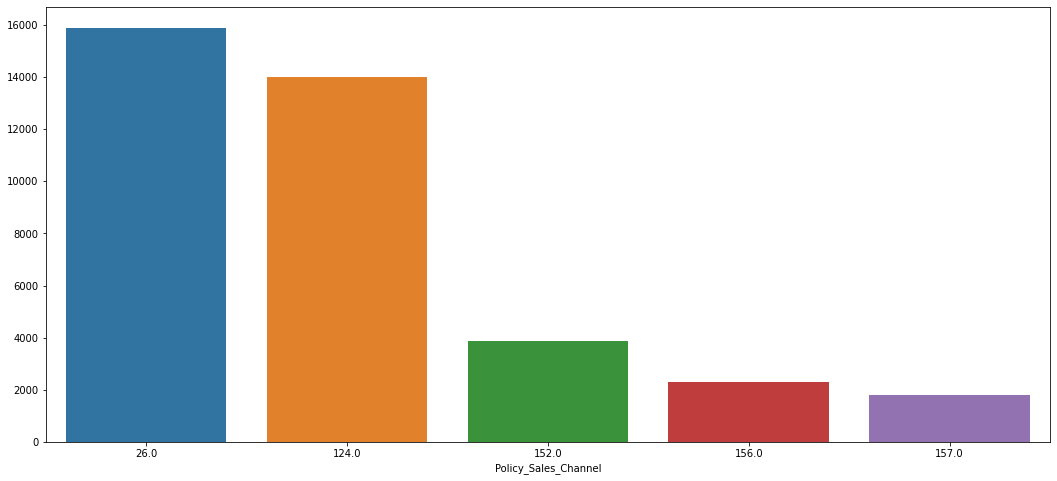
|  |  |
| --- | --- |
| 1. Kelas 0 (No) | 1. Kelas 1 (Yes) |

Gambar 4. Distribusi Umur setiap Kelas Target

Gambar 4.a. menunjukkan distribusi umur pada kelas No yang condong ke kiri pada umur antara 20 - 30 tahun. Di sisi lain, Gambar 4.b. yang menunjukkan distribusi umur pada kelas Yes cenderung banyak pada umur 40 - 50 tahun. Rata-rata umur yang berbeda ini kemungkinan disebabkan oleh harga asuransi kendaraan yang cenderung mahal dan rata-rata income di India [ref: <https://www.averagesalarysurvey.com/india> ]. Berdasarkan survei rata-rata pendapatan di India yang dikelompokkan sesuai umur, terlihat bahwa golongan umur 40 - 50 tahun memiliki pendapatan yang lebih tinggi dibandingkan dengan yang berumur 20 - 30 tahun.

Selain umur, plotting dilakukan pada beberapa fitur seperti Previously\_Insured (Gambar 5.a.), Vehicle\_Damage (Gambar 5.b.), dan Policy\_Sales\_Channel (Gambar 5.c.) ketika kelas target bernilai 1 atau Yes. Dari Gambar 5.a dan 5.b terlihat bahwa yang menerima tawaran asuransi kendaraan banyak yang belum memiliki asuransi atau pernah mengalami kerusakan. Gambar 5.c. menunjukkan bahwa banyak dari yang menerima asuransi dihubungi melalui media dengan kode 26.

|  |  |
| --- | --- |
| 1. Previously\_Insured | 1. Vehicle\_Damage |



(c) 5 Terbanyak Policy\_Sales\_Channel

Gambar 5. Jumlah Data setiap Kelas pada suatu Fitur dengan Target Yes

Menggunakan gabungan dari kondisi-kondisi pada Gambar 5, dilakukan pengecekan terhadap fitur Vehicle\_Age yang ditampilkan pada Gambar 6. Dari eksplorasi ini, disimpulkan bahwa pelanggan yang menerima tawaran asuransi cenderung memiliki karakteristik sebagai berikut:

* Berumur antara 40 - 50 Tahun
* Dihubungi melalui media berkode 26
* Belum memiliki asuransi kendaraan
* Pernah mengalami kerusakan pada kendaraaan
* Umur kendaraan kurang dari 1 tahun

|  |  |  |
| --- | --- | --- |
| 1. < 1 Year | 1. 1-2 Year | (c) > 2 Years |

Gambar 6. Jumlah Data Kelas Target pada Vehicle\_Age Tertentu

1. *Metrics evaluasi apa yang cocok untuk kasus klasifikasi ini?*

Marketing merupakan aspek penting dalam penjualan produk. Prediksi yang tepat dalam memperkirakan apakah seorang pelanggan akan menerima asuransi atau tidak dapat mengoptimasi pendapatan perusahaan. Dengan demikian, perusahaan hanya perlu me-*reach out* pelanggan yang diperkirakan akan menerima asuransi saja sehingga dapat mengurangi pengeluaran untuk menghubungi pelanggan. Maka dari itu, *precision* dan *recall* terhadap prediksi pelanggan yang akan menerima asuransi perlu diperhatikan. F1 score menjadi metrics yang cocok untuk evaluasi kasus ini karena dihitung berdasarkan *precision* dan *recall* dari prediksi.

## Hasil Uji Coba Skenario

**3.2.1 Model SVM**

Dengan menggunakan combinasi dari parameter C dengan nilai 1, 10, 0.1 dan gamma dengan nilai 0.1, 0.01, 1

Tabel 2. Hasil Uji Coba SVM dengan Data Uji 30% dan Data Latih 70%

|  |  |  |  |
| --- | --- | --- | --- |
| C | Gamma | | |
| 0.1 | 0.01 | 1.0 |
| 1 | 78,61% | 78,61% | 79,21% |
| 10 | 78,61% | 78,61% | 79,50% |
| 0.1 | 78,61% | 78,61% | 79,21% |

Tabel 3. Hasil Uji Coba SVM dengan Data Uji 50% dan Data Latih 50%

|  |  |  |  |
| --- | --- | --- | --- |
| C/Kernel | Gamma | | |
| 0.1 | 0.01 | 1.0 |
| 1 | 82.05% | 82.05% | 81.84% |
| 10 | 82.05% | 82.05% | 81.85% |
| 0.1 | 82.05% | 82.05% | 82.05% |

Tabel 4. Hasil Uji Coba SVM dengan Data Uji 40% dan Data Latih 60%

|  |  |  |  |
| --- | --- | --- | --- |
| C/Kernel | Gamma | | |
| 0.1 | 0.01 | 1.0 |
| 1 | 82.05% | 82.05% | 81.84% |
| 10 | 82.05% | 82.05% | 81.85% |
| 0.1 | 82.05% | 82.05% | 82.05% |

**3.2.2 Model KNN**

Menggunakan kombinasi dari parameter n dengan nilai 3, 5, 7, 11, 31, kemudian parameter weight dengan nilai uniform & distance dan parameter metric dengan nilai euclidean, manhattan, dan chebyshev.

Tabel 5. Hasil Uji Coba KNN dengan Data Uji 30% dan Data Latih 70%

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Perhitungan Jarak | Jumlah tetangga | | | | | | | | | |
| 3 | | 5 | | 7 | | 11 | | 31 | |
| uniform | distance | uniform | distance | uniform | distance | uniform | distance | uniform | distance |
| euclidean | 72,41% | 72,71% | 73,46% | 73,99% | 73,80% | 74,47% | 74,82% | 75,70% | 76,26% | 77,37% |
| manhattan | 72,41% | 72,71% | 73,45% | 73,99% | 73,77% | 74,47% | 74,83% | 75,70% | 76,28% | 77,37% |
| chebyshev | 72,40% | 72,71% | 73,43% | 73,99% | 73,78% | 74,47% | 74,77% | 75,70% | 76,24% | 73,71% |

Tabel 6. Hasil Uji Coba KNN dengan Data Uji 50% dan Data Latih 50%

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Perhitungan Jarak | Jumlah tetangga | | | | | | | | | |
| 3 | | 5 | | 7 | | 11 | | 31 | |
| uniform | distance | uniform | distance | uniform | distance | uniform | distance | uniform | distance |
| euclidean | **75.50%** | **75.25%** | **78.70%** | **78.25%** | **79.20%** | **78.51%** | **80.40%** | **79.53%** | **81.61%** | **80.55%** |
| manhattan | **75.48%** | **75.21%** | **78.71%** | **78.25%** | **79.21%** | **78.51%** | **80.40%** | **79.53%** | **81.70%** | **80.60%** |
| chebyshev | **75.47%** | **75.21%** | **78.69%** | **78.24%** | **79.19%** | **78.50%** | **80.36%** | **79.55%** | **81.62%** | **80.57%** |

Tabel 7. Hasil Uji Coba KNN dengan Data Uji 40% dan Data Latih 60%

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Perhitungan Jarak | Jumlah tetangga | | | | | | | | | |
| 3 | | 5 | | 7 | | 11 | | 31 | |
| uniform | distance | uniform | distance | uniform | distance | uniform | distance | uniform | distance |
| euclidean | **76.06%** | **75.73%** | **78.69%** | **78.21%** | **79.47%** | **78.78%** | **80.02%** | **79.09%** | **81.56%** | **80.48%** |
| manhattan | **76.06%** | **75.73%** | **78.72%** | **78.23%** | **79.51%** | **78.77%** | **80.03%** | **79.11%** | **81.55%** | **80.50%** |
| chebyshev | **76.07%** | **75.74%** | **78.70%** | **78.22%** | **79.45%** | **78.77%** | **79.91%** | **79.09%** | **81.52%** | **80.48%** |

**3.2.3 Model Decision Tree**

Dengan menggunakan kombinasi dari parameter kedalaman maksimal dengan nilai tidak ada, 5, 7, 10, 20 dan kriteria dengan nilai gini, entropy

Tabel 8. Hasil Uji Coba Decision Tree dengan Data Uji 30% dan Data Latih 70%

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Param grid** | | | | | | | | | |
| **none** | | **5** | | **7** | | **10** | | **20** | |
| **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** |
| 81,11% | 81,11% | 79,34% | 79,34% | 79,54% | 79,34% | 80,11% | 79,34% | 81,05% | 79,34% |

Tabel 9. Hasil Uji Coba Decision Tree dengan Data Uji 50% dan Data Latih 50%

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Param grid** | | | | | | | | | |
| **none** | | **5** | | **7** | | **10** | | **20** | |
| **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** |
| **80.10%** | **80.08%** | **81.67%** | **82.37%** | **82.34%** | **82.19%** | **82.05%** | **81.82%** | **80.35%** | **80.25%** |

Tabel 10. Hasil Uji Coba Decision Tree dengan Data Uji 40% dan Data Latih 60%

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Param grid** | | | | | | | | | |
| **none** | | **5** | | **7** | | **10** | | **20** | |
| **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** | **gini** | **entropy** |
| **80.23%** | **80.18%** | **81.44%** | **81.04%** | **82.34%** | **82.13%** | **82.08%** | **81.76%** | **80.44%** | **80.46%** |

**3.2.4 Model MLP**

Tabel 11. Hasil Uji Coba MLP dengan Data Uji 30% dan Data Latih 70%

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Param grid** | | | | | **F1 score** |
| **Hidden layer 1** | **Hidden layer 2** | **Hidden layer 3** | **Hidden layer 4** | **Learning rate** |
| 256 | 128 | 64 | - | 0.1 | **82,04%** |
| 256 | 128 | 64 | - | 0.01 | **81,81%** |
| 256 | 128 | 64 | 8 | 0.1 | **79,76%** |
| 256 | 128 | 64 | 16 | 0.1 | **66,66%** |
| 256 | 128 | 128 | - | 0.1 | **81,54** |

Tabel 12. Hasil Uji Coba MLP dengan Data Uji 50% dan Data Latih 50%

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Param grid** | | | | | | | | | | **F1 score** |
| **Hidden layer 1** | | **Hidden layer 2** | | **Hidden layer 3** | | **Hidden layer 4** | | **Learning rate** | |
| **256** | | **128** | | **64** | | **-** | | **0.1** | | **82.05%** |
| **128** | | **64** | | **32** | | **16** | | **0.1** | | **81.56%** |
| **256** | | **128** | | **128** | | **-** | | **0.1** | | **81.85%** |
| **256** | | **128** | | **64** | | **-** | | **0.01** | | **82.3%** |
| **256** | | **256** | | **16** | | **8** | | **0.1** | | **81.15%** |

Tabel 13. Hasil Uji Coba MLP dengan Data Uji 40% dan Data Latih 60%

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Neuron** | | | | | | | | **F1 score** |
| **Hidden layer 1** | | **Hidden layer 2** | | **Hidden layer 3** | | **Learning rate init** | |
| **256** | | **128** | | **32** | | **0.1** | | **82.11%** |
| **32** | | **64** | | **-** | | **0.1** | | **81.69%** |
| **256** | | **128** | | **64** | | **0.1** | | **81.80%** |
| **256** | | **128** | | **64** | | **1** | | **66.62%** |
| **256** | | **256** | | **256** | | **0.1** | | **82.38%** |

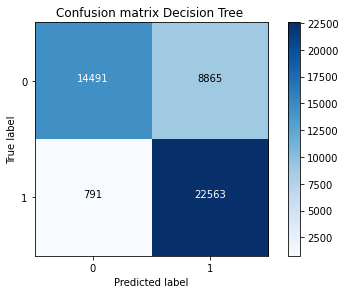
## Pembahasan dan Kesimpulan

Hasil terbaik setiap model dapat terlihat pada Tabel X. F1 tertinggi terdapat pada model MLP dengan data training 60% dan nilai 82.38%. Tetapi, hasil tersebut tidak berbeda jauh dengan decision tree yang senilai 82.37%. Melihat dari waktu latih seperti tertera pada Tabel X, maka model terbaik terpilih adalah decision tree dengan waktu latih 0.03 detik.

Tabel 14. Hasil Terbaik setiap Model

|  |  |  |
| --- | --- | --- |
| Model | F1 | Waktu (detik) |
| SVM | 82.08% | 4.44 |
| KNN | 80.57% | 1.77 |
| Decision Tree | 82.37% | 0.03 |
| MLP | 82.38% | 172.93 |

Dalam memvisualisasikan hasil prediksi, *confusion matrix* (CM) digunakan dengan memplotkan label pada *ground truth* dan yang terprediksi. Gambar 7 menunjukkan CM model terbaik yaitu decision tree. CM yang terlihat pada Gambar 7 menunjukkan bahwa model banyak misklasifikasi pada data 0. Hal ini dapat diartikan model kurang dapat mempelajari pola data pada kelas tersebut.



Gambar 7. *Confusion Matrix* model Terbaik

Link Google Colab:

<https://colab.research.google.com/drive/1Ul1Maw9SB_G8x167CvqIGZdFjkmFJil3?usp=sharing>

Github repository:

<https://github.com/coll-j/Tugas-KK/tree/master/Minggu-15-16>

# DAFTAR PUSTAKA

Average Salary Survey. (2020). India Salary. Retrieved from <https://www.averagesalarysurvey.com/india>. (accessed Jan 1. 2021)