# BAB IV

# HASIL DAN PEMBAHASAN

## 4.1 Hasil

Berdasarkan metodologi yang telah dirancang pada kasus prediksi turnover karyawan menggunakan metode *Decision Tree, Random Forest, K-Nearest Neighbors (KNN).* Yang terdiri dari beberapa proses pada gogle colaboratory diantaranya sebagai berikut **https://colab.research.google.com/drive/1FCOIW-AHJ2gdIcTMVnJ_C76HXlelhdUG?usp=sharing**

### 4.1.1 Data Collection

**1. Menyiapkan dataset**

Dataset yang digunakan dalam penelitian ini bebrbentuk file CSV yang bersumber dari repositori *Kaggle.* Data yang akan diolah dalam penelitian ini terdiri dari 4653 baris dan 9 column, Adapun dataset tesebut sebagai sebagai berikut

| A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|
| Education | JoiningYear | City | PaymentTier | Age | Gender | EverBenched | ExperienceInCurrentDomain | LeaveOrNot |
| Bachelors | 2017 | Bangalore | 3 | 34 | Male | No | 0 | 0 |
| Bachelors | 2013 | Pune | 1 | 28 | Female | No | 3 | 1 |
| Bachelors | 2014 | New Delhi | 3 | 38 | Female | No | 2 | 0 |
| Masters | 2016 | Bangalore | 3 | 27 | Male | No | 5 | 1 |
| Masters | 2017 | Pune | 3 | 24 | Male | Yes | 2 | 1 |
| Bachelors | 2016 | Bangalore | 3 | 22 | Male | No | 0 | 0 |
| Bachelors | 2015 | New Delhi | 3 | 38 | Male | No | 0 | 0 |
| Bachelors | 2016 | Bangalore | 3 | 34 | Female | No | 2 | 1 |
| Bachelors | 2016 | Pune | 3 | 23 | Male | No | 1 | 0 |
| Masters | 2017 | New Delhi | 2 | 37 | Male | No | 2 | 0 |
| Masters | 2012 | Bangalore | 3 | 27 | Male | No | 5 | 1 |
| Bachelors | 2016 | Pune | 3 | 34 | Male | No | 3 | 0 |
| Bachelors | 2018 | Pune | 3 | 32 | Male | Yes | 5 | 1 |
| Bachelors | 2016 | Bangalore | 3 | 39 | Male | No | 2 | 0 |
| Bachelors | 2012 | Bangalore | 3 | 37 | Male | No | 4 | 0 |
| Bachelors | 2017 | Bangalore | 1 | 29 | Male | No | 3 | 0 |
| Bachelors | 2014 | Bangalore | 3 | 34 | Female | No | 2 | 0 |
| Bachelors | 2014 | Pune | 3 | 34 | Male | No | 4 | 0 |
| Bachelors | 2015 | Pune | 2 | 30 | Female | No | 0 | 1 |
| Bachelors | 2016 | New Delhi | 2 | 22 | Female | No | 0 | 1 |
| Bachelors | 2012 | Bangalore | 3 | 37 | Male | No | 0 | 0 |
| Masters | 2017 | New Delhi | 2 | 28 | Male | No | 4 | 0 |
| Bachelors | 2017 | New Delhi | 2 | 36 | Male | No | 3 | 0 |
| Bachelors | 2015 | Bangalore | 3 | 27 | Male | Yes | 5 | 0 |
| Bachelors | 2017 | Bangalore | 3 | 29 | Male | No | 4 | 0 |
| Bachelors | 2013 | Bangalore | 3 | 22 | Female | Yes | 0 | 0 |

**Gambar 4.1. 1 Dataset**

**2. Menyiapkan Data dan Library**

Tahap selanjutnya, jika dataset yang dibutuhkan sudah tersedia maka menyiapkan library pada google colab untuk memudahkan dalam proses pengolahan data.

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, f1_score, precision_score,
recall_score, confusion_matrix, classification_report
import matplotlib.pyplot as plt
from sklearn.model_selection import RandomizedSearchCV
from scipy.stats import randint
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
```

**Gambar 4.1. 2 Import library**

**3. Mengimpor data ke google colab**

Pada tahapan ini penulisan mengimpor dataset yang telah didapat ke google colab untuk mempermudah pengolahan data.

```
From google.colab import drive
Drive.mount('/content/drive')
File_path = "/content/drive/MyDrive/SKRIPSI AULIA MAHARANI/Employee (1).csv"
```

**Gambar 4.1. 3 Import Dataset**

### 4.1.2 Exploratory Data Analysis

*a) Describe*

Pada tahapan pertama exploratory data analysis yaitu describe untuk menampilkan ringkasan statistik data numerik, seperti jumlah nilai yang tersedia **(count),** rata-rata **(mean),** standar deviasi **(std),** nilai minimum **(min),** kuartil pertama, kuartil kedua **(median),** kuartil ketiga, dan nilai maksimum **(max).**

| | JoiningYear | PaymentTier | Age | ExperienceInCurrentDomain | LeaveOrNot |
|---|---|---|---|---|---|
| **count** | 4653.000000 | 4653.000000 | 4653.000000 | 4653.000000 | 4653.000000 |
| **mean** | 2015.062970 | 2.698259 | 29.393295 | 2.905652 | 0.343864 |
| **std** | 1.863377 | 0.561435 | 4.826087 | 1.558240 | 0.475047 |
| **min** | 2012.000000 | 1.000000 | 22.000000 | 0.000000 | 0.000000 |
| **25%** | 2013.000000 | 3.000000 | 26.000000 | 2.000000 | 0.000000 |
| **50%** | 2015.000000 | 3.000000 | 28.000000 | 3.000000 | 0.000000 |
| **75%** | 2017.000000 | 3.000000 | 32.000000 | 4.000000 | 1.000000 |
| **max** | 2018.000000 | 3.000000 | 41.000000 | 7.000000 | 1.000000 |

**Gambar 4.1. 4 Describe**

*b) Missing Value*

Pada tahapan missing value yaitu memeriksa dan menghitung jumlah nilai hilang (missing values) di setiap kolom dalam dataset. agar data menjadi lengkap dan siap dianalisis.

| | 0 |
|---|---|
| Education | 0 |
| JoiningYear | 0 |
| City | 0 |
| PaymentTier | 0 |
| Age | 0 |
| Gender | 0 |
| EverBenched | 0 |
| ExperienceInCurrentDomain | 0 |
| LeaveOrNot | 0 |

dtype: int64

**Gambar 4.1. 5 Missing Value**

*c) Duplicate*

Pada tahapan ini mengecek data yang duplikat, untuk memastikan tidak ada pengaruh ganda dalam analisis data.

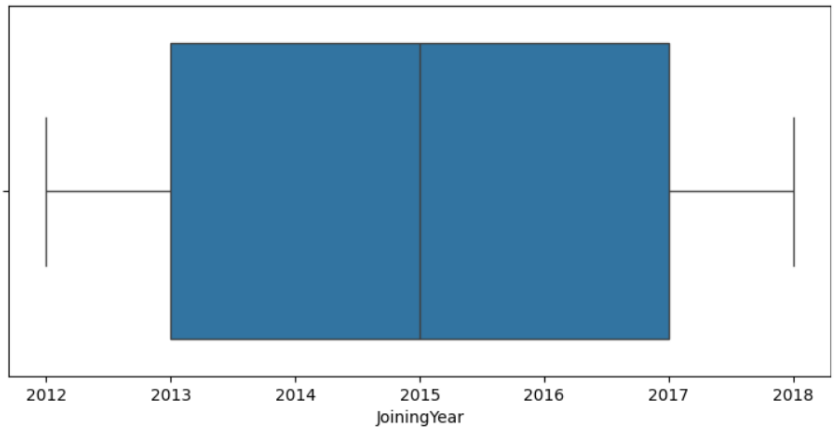
```
data.duplicated()
```

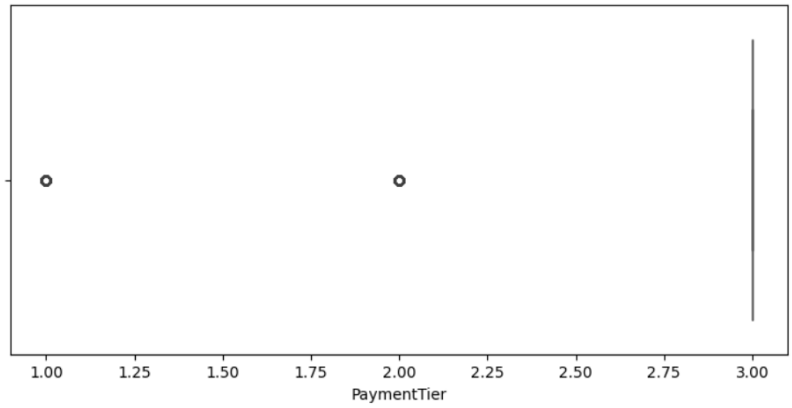
```
0
```

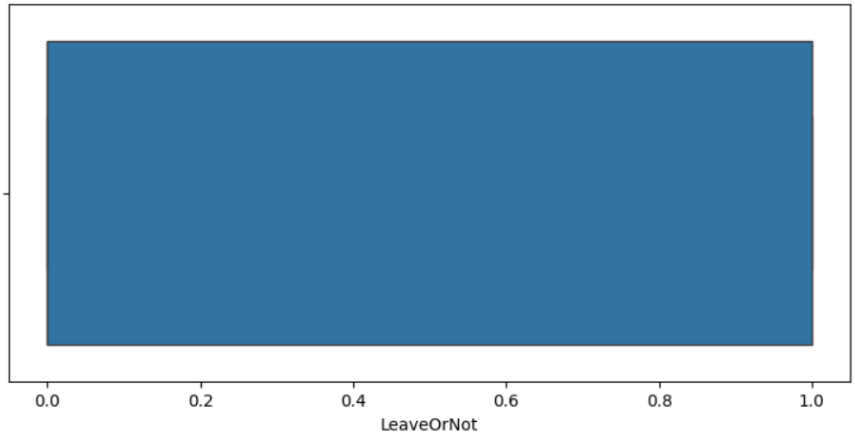
**Gambar 4.1. 6 Duplicate**

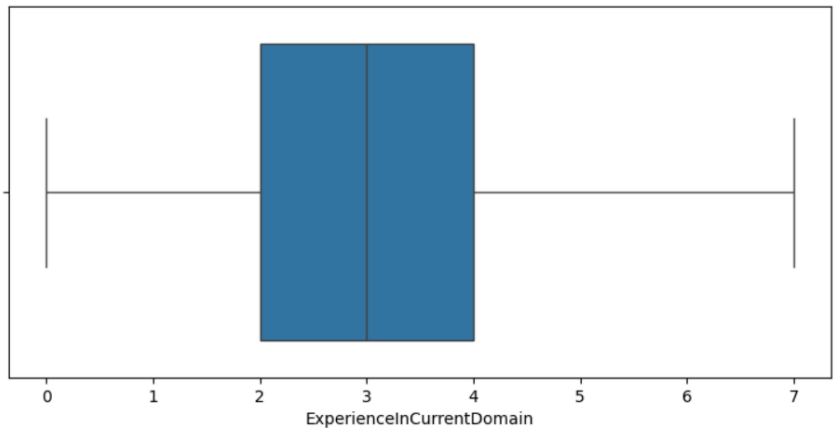
*d) Outlier*

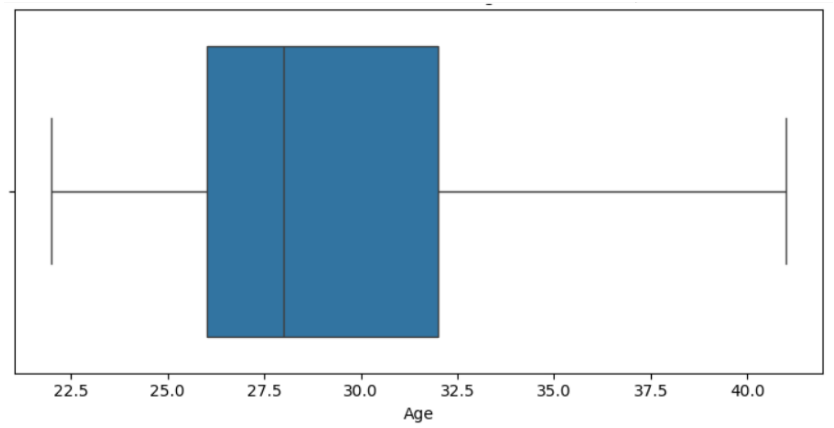
Untuk pengecekan outlier pada dataset menggunakan visualisasi boxplot dan countplot.

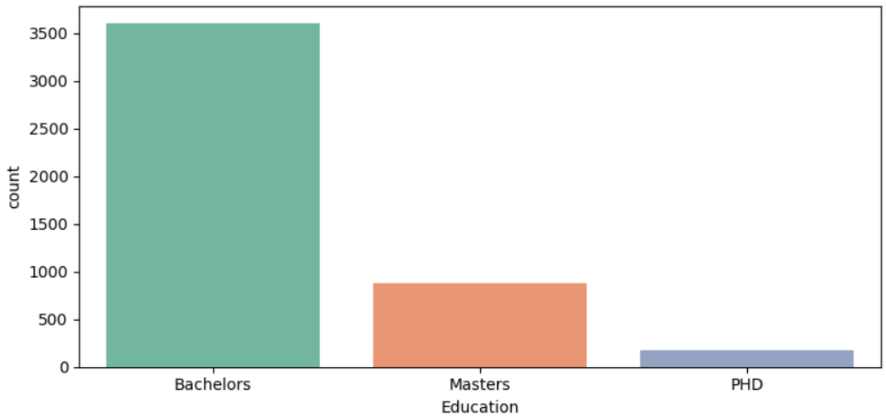


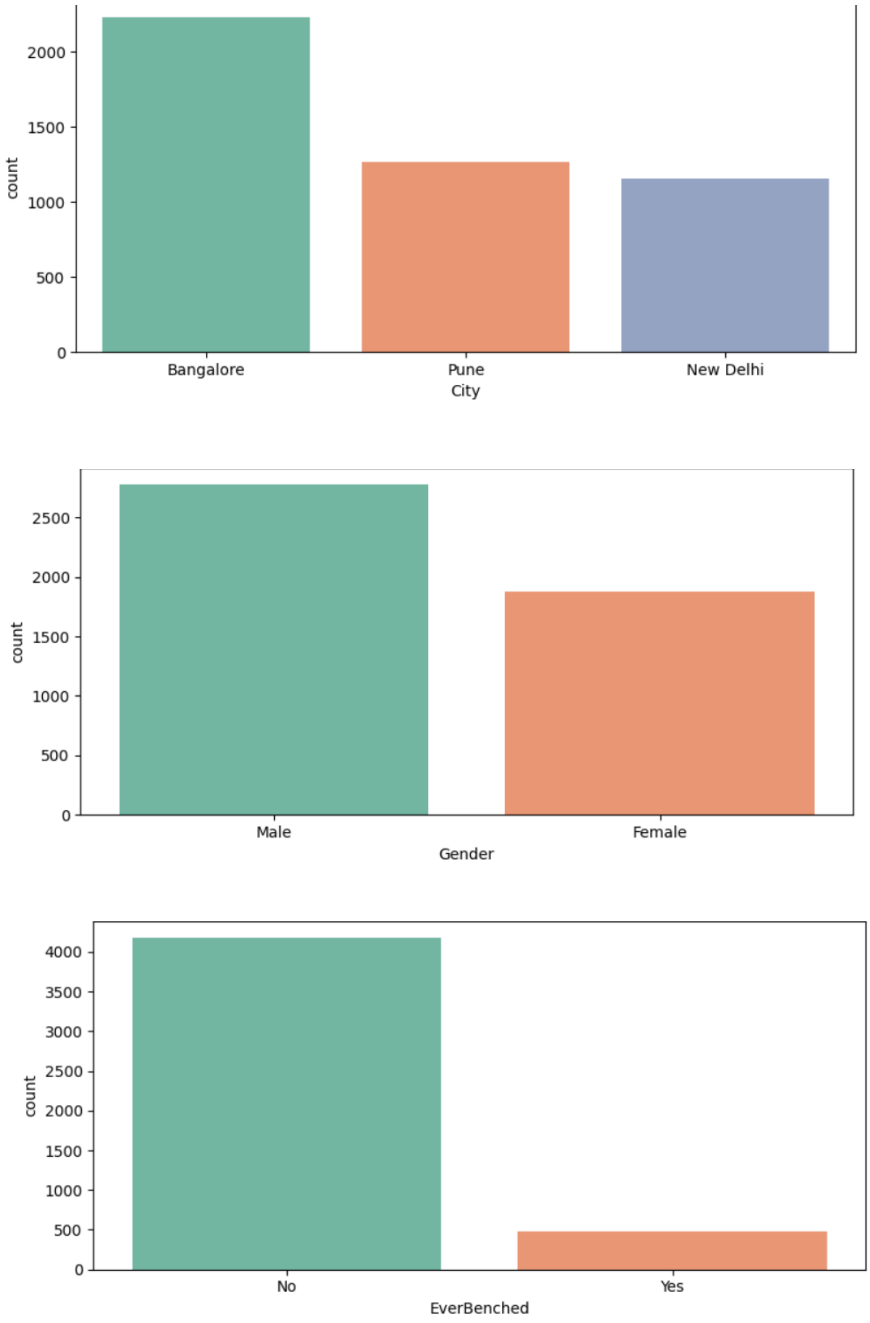




0
1
2
3
4
5
6
7
ExperienceInCurrentDomain


22.5
25.0
27.5
30.0
32.5
35.0
37.5
40.0
Age


3500
3000
2500
2000
1500
1000
500
0
count
Bachelors
Masters
PHD
Education




**Gambar 4.1. 7 Outlier**

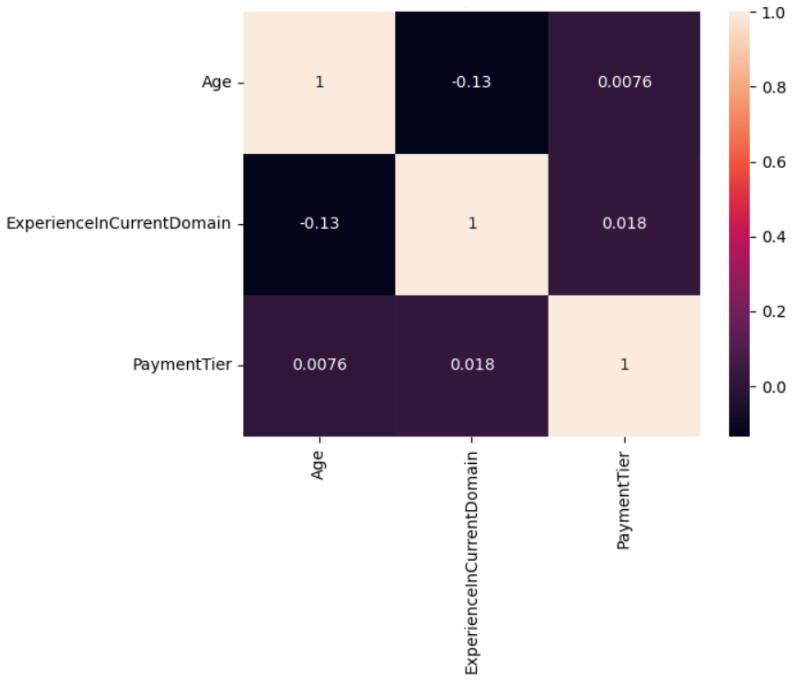
Setelah dilakukan outlier visualisasi dengan boxplot dan countplot, maka dilakukan deteksi outlier untuk menghitung ambang batas atas dan ambang batas bawah.

```
Kolom: Education (Categorical)
  Kategori Langka: []
  Jumlah Kategori Langka: 0
--------------------------------------------------
Kolom: JoiningYear (Numerical)
  Ambang Batas Atas: 2023.0
  Jumlah Outlier di Atas: 0
  Ambang Batas Bawah: 2007.0
  Jumlah Outlier di Bawah: 0
--------------------------------------------------
Kolom: City (Categorical)
  Kategori Langka: []
  Jumlah Kategori Langka: 0
--------------------------------------------------
Kolom: PaymentTier (Numerical)
  Ambang Batas Atas: 3.0
  Jumlah Outlier di Atas: 0
  Ambang Batas Bawah: 3.0
  Jumlah Outlier di Bawah: 1161
--------------------------------------------------
Kolom: Age (Numerical)
  Ambang Batas Atas: 41.0
  Jumlah Outlier di Atas: 0
  Ambang Batas Bawah: 17.0
  Jumlah Outlier di Bawah: 0
--------------------------------------------------
Kolom: Gender (Categorical)
  Kategori Langka: []
  Jumlah Kategori Langka: 0
--------------------------------------------------
Kolom: EverBenched (Categorical)
  Kategori Langka: []
  Jumlah Kategori Langka: 0
--------------------------------------------------
Kolom: ExperienceInCurrentDomain (Numerical)
  Ambang Batas Atas: 7.0
  Jumlah Outlier di Atas: 0
  Ambang Batas Bawah: -1.0
  Jumlah Outlier di Bawah: 0
--------------------------------------------------
Kolom: LeaveOrNot (Numerical)
  Ambang Batas Atas: 2.5
  Jumlah Outlier di Atas: 0
  Ambang Batas Bawah: -1.5
  Jumlah Outlier di Bawah: 0
--------------------------------------------------
```

**Gambar 4.1. 8 Outlier Ambang Batas**

e) *Heatmap*

pada tahapan ini dilakukan untuk membuat heatmap korelasi antara beberapa kolom dalam dataset, yaitu Age, ExperienceInCurrentDomain, PaymentTier. Dan digunakan untuk menghitung korelasi antar kolom numerik, lalu hasilnya divisualisasikan. Dengan ini, kita dapat melihat hubungan antar-variabel tersebut secara visual, termasuk kekuatan dan arah korelasinya.



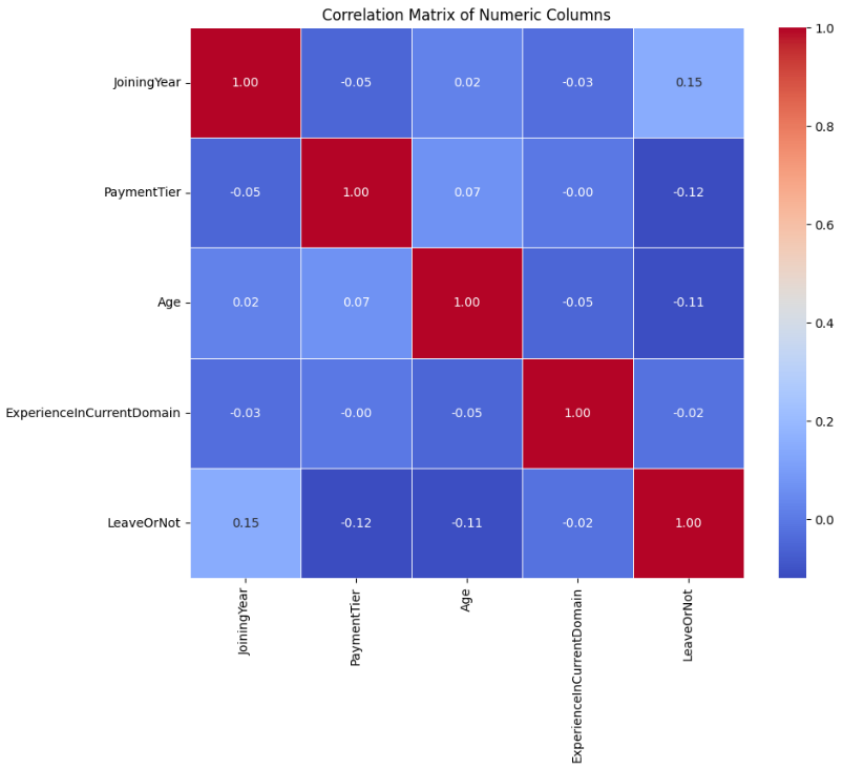

**Gambar 4.1. 9 Heatmap**

Heatmap ini menunjukkan hubungan antarvariabel Age**,** ExperienceInCurrentDomain, dan PaymentTier menggunakan matriks korelasi. Hasilnya, hubungan antarvariabel sangat lemah, dengan nilai korelasi mendekati nol. Terdapat korelasi negatif lemah antara Age dan ExperienceInCurrentDomain (-0.13), menunjukkan bahwa usia tidak selalu

berkaitan dengan pengalaman di domain saat ini. Korelasi antara Age dan PaymentTier (0.0076), serta antara ExperienceInCurrentDomain dan PaymentTier (0.018), sangat kecil, menandakan bahwa tidak ada hubungan anatara variabel-variabel tersebut. Secara keseluruhan, variabel-variabel ini tidak memiliki hubungan signifikan satu sama lain.

f) Matrix Colleration

mengukur dan menganalisis hubungan antar variabel dalam dataset. Matriks ini membantu mengidentifikasi seberapa kuat dan arah hubungan antar variabel (positif, negatif, atau tidak ada hubungan), sehingga dapat digunakan untuk pemilihan fitur, deteksi multikolinearitas, atau memahami pola data sebelum analisis lebih lanjut.




**Gambar 4.1. 10 Matrix Correlation**

### 4.1.3 Data Preprocessing

Berdasarkan Hasil Observasi Exploratory Data Analysis, diperlukannya beberapa preprocessing terlebih dahulu sebelum data dimasukan keakurasi perbandingan algoritma turnover karyawan. Adapun tahapan preprocessing yang dibutuhkan sebagai berikut.

*a. Encoding*

proses transformasi data kategorikal menjadi representasi numerik, seperti 0 dan 1, sehingga data tersebut dapat digunakan pada tahapan selanjutnya. Encodinng memastikan bahwa data non-numerik dapat dipahami oleh algoritma, yang umumnya hanya menerima masukan berupa angka.

| | Education | JoiningYear | City | PaymentTier | Age | Gender | EverBenched | ExperienceInCurrentDomain | LeaveOrNot |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 2017 | 0 | 3 | 34 | 1 | 0 | 0 | 0 |
| **1** | 0 | 2013 | 2 | 1 | 28 | 0 | 0 | 3 | 1 |
| **2** | 0 | 2014 | 1 | 3 | 38 | 0 | 0 | 2 | 0 |
| **3** | 1 | 2016 | 0 | 3 | 27 | 1 | 0 | 5 | 1 |
| **4** | 1 | 2017 | 2 | 3 | 24 | 1 | 1 | 2 | 1 |

**Gambar 4.1. 11 Encoding**

*b. Splitting*

Mengecek pembagian data menjadi data pelatihan (training) dan data pengujian (testing)**,** serta menampilkan jumlah data di masing-masing set.

```
Total train data : 3117
Total test data : 1536
```

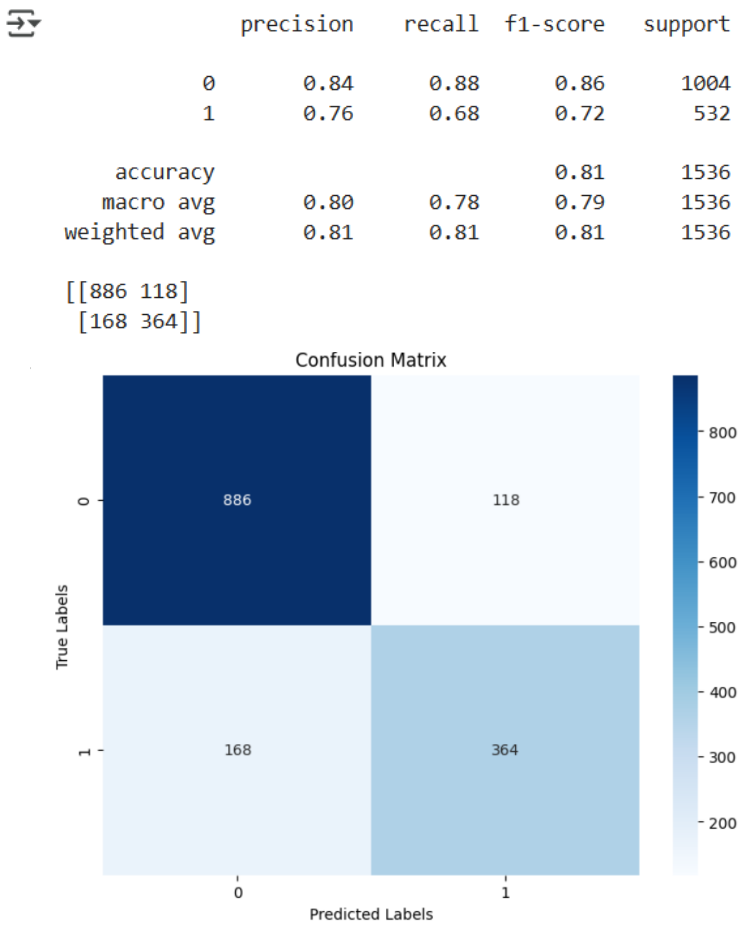
**Gambar 4.1. 12 Splitting**

### 4.1.4 Perbandingan Akurasi Algoritma

perbandingan algoritma untuk turnover karyawan dalam kode digunakan untuk membandingkan kinerja berbagai algoritma (seperti Decision Tree, Random Forest,

KNN) dalam memprediksi keputusan karyawan untuk tetap atau meninggalkan perusahaan berdasarkan data yang ada. kemudian membandingkan kinerja masing-masing algoritma berdasarkan metrik evaluasi seperti akurasi, precision, recall, dan F1-score.

*a) Decision Tree*

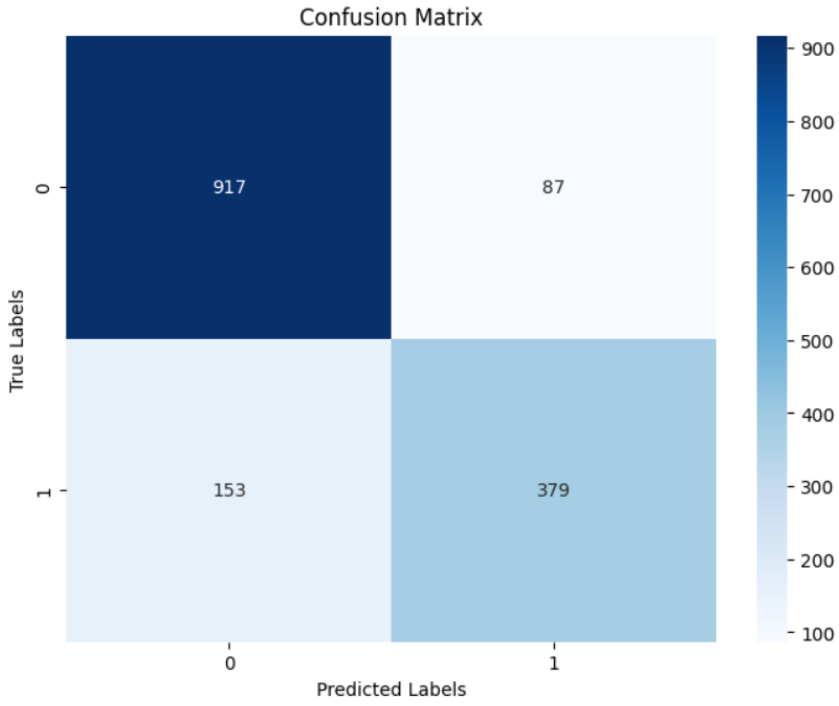



**Gambar 4.1. 13 Decision Tree**

*b) Random Forest*

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.86 | 0.91 | 0.88 | 1004 |
| 1 | 0.81 | 0.71 | 0.76 | 532 |
| accuracy | | | 0.84 | 1536 |
| macro avg | 0.84 | 0.81 | 0.82 | 1536 |
| weighted avg | 0.84 | 0.84 | 0.84 | 1536 |

```
[[917  87]
 [153 379]]
```

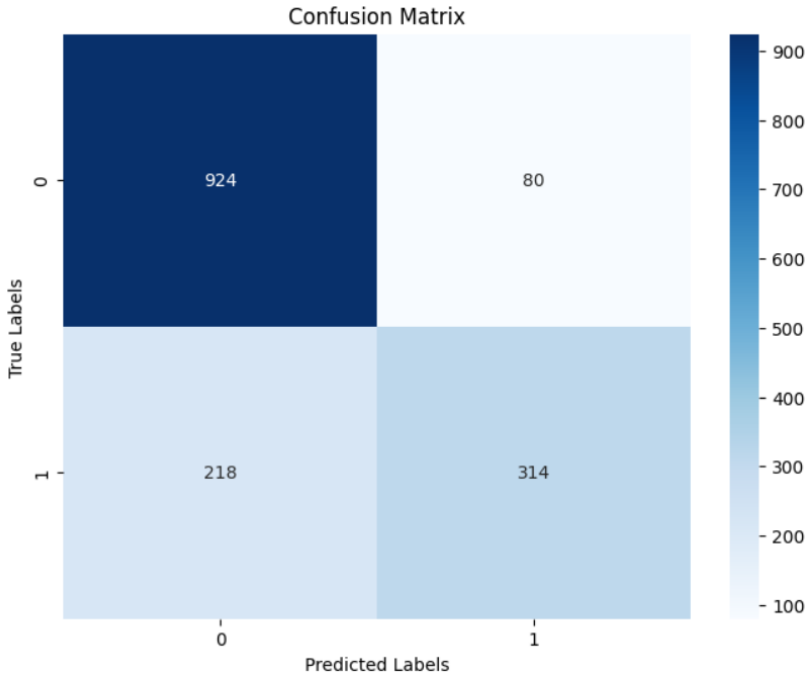



**Gambar 4.1. 14 Random Forest**

*c) K-Nearest Neighbors (KNN)*

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.81 | 0.92 | 0.86 | 1004 |
| 1 | 0.80 | 0.59 | 0.68 | 532 |
| accuracy | | | 0.81 | 1536 |
| macro avg | 0.80 | 0.76 | 0.77 | 1536 |
| weighted avg | 0.80 | 0.81 | 0.80 | 1536 |

```
[[924  80]
 [218 314]]
```




**Gambar 4.1. 15 K-Nearest Neighbors**

*d) Evaluasi Model*

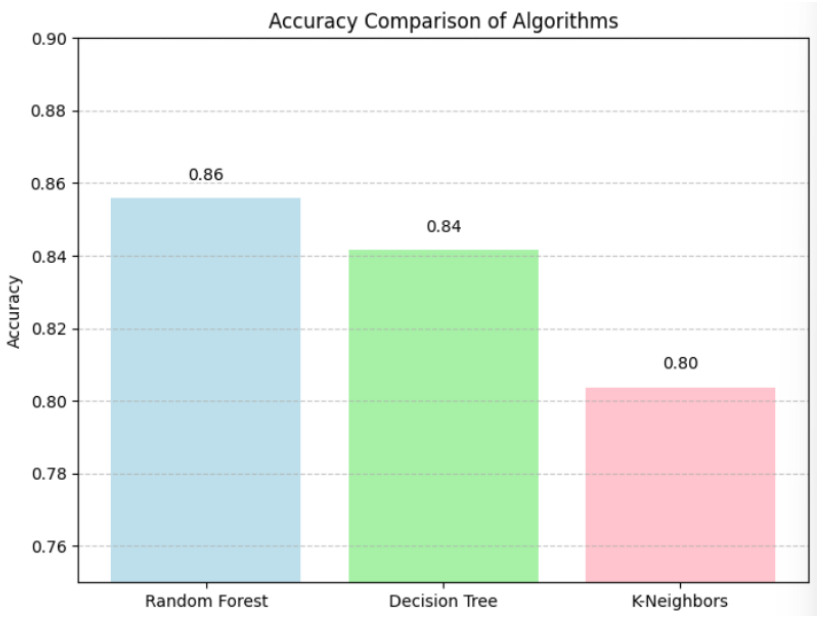
Setelah dilakukan evaluasi terhadap beberapa algoritma, diperoleh hasil akurasi terbaik untuk masing-masing model sebagai berikut.

| | model | best_score |
|---|---|---|
| 0 | RandomForest | 0.855791 |
| 1 | DecisionTree | 0.841607 |
| 2 | KNN | 0.803776 |

**Gambar 4.1. 16 Evaluasi Model**

*e) Visualisasi Perbandingan Algoritma*

Gambar tersebut membandingkan akurasi tiga algoritma machine learning: Random Forest, Decision Tree, dan K-Nearest Neighbors (KNN). Random Forest memiliki akurasi tertinggi sebesar 86%, diikuti oleh Decision Tree dengan 84%, dan KNN dengan 80%. menunjukkan bahwa Random Forest adalah algoritma terbaik yang paling optimal untuk memprediksi turonver karyawan.




**Gambar 4.1. 17 Visualisasi Perbandingan Algoritma**

### 4.1.5 Proses Algoritma Decision Tree

Langkah-langkah dalam pembentukan pohon keputusan pada prediksi turnover karyawan salah satu menggunakan metode decision tree dilakukan sesuai pada tabel 4.5.1, berikut perhitungan decision tree:

**NILAI ENTROPY DAN GAIN UNTUK MENENTUKAN SIMPUL AKAR**

| Atribut | Nilai | Jumlah kasus | LeaveOrNot=0 | LeaveOrNot=1 | Entropy | Gain | Akurasi |
|---|---|---|---|---|---|---|---|
| Total | | 4653 | 3053 | 1600 | 0,9284685 | | 0,656135826 |
| Leave | | 1600 | | | | | |
| Not | | 3053 | | | | | |
| Education | | | | | | 0,3903654557 | |
| | Bachelors | 3601 | 2472 | 1129 | 0,8971995 | | 0,686475978 |
| | Masters | 873 | 447 | 426 | 0,9995825 | | 0,512027491 |
| | PHD | 179 | 134 | 45 | 0,8134842 | | 0,748603352 |
| Joining Year | | | | | | 0,9228294393 | |
| | 2012 | 504 | 395 | 109 | 0,7532976 | | 0,783730158 |
| | 2013 | 669 | 445 | 224 | 0,9197833 | | 0,665171898 |
| | 2014 | 699 | 526 | 173 | 0,8072858 | | 0,752503576 |
| | 2015 | 781 | 463 | 318 | 0,9749907 | | 0,592829705 |
| | 2016 | 525 | 408 | 318 | 0,7207918 | | 0,777142857 |
| | 2017 | 1108 | 811 | 297 | 0,8386533 | | 0,731949458 |
| | 2018 | 367 | 5 | 362 | 0,1039581 | | 0,013623978 |
| City | | | | | | 0,4789659861 | |
| | Bangalore | 2228 | 1633 | 595 | 0,8372078 | | 0,732944344 |
| | New Delh | 1157 | 791 | 366 | 0,9003516 | | 0,68366465 |
| | Pune | 1268 | 629 | 639 | 0,9999551 | | 0,496056782 |
| PaymentTier | | | | | | 0,4336019544 | |
| | 1 | 243 | 154 | 89 | 0,9477532 | | 0,633744856 |
| | 2 | 918 | 368 | 550 | 0,9714580 | | 0,400871459 |
| | 3 | 3492 | 2531 | 961 | 0,8488283 | | 0,724799541 |
| Age | | | | | | 1,033501223 | |
| | 22 | 49 | 30 | 19 | 0,9633355 | | 0,612244898 |
| | 23 | 48 | 32 | 16 | 0,9182958 | | 0,666666666 |
| | 24 | 385 | 232 | 153 | 0,9694109 | | 0,602597402 |
| | 25 | 418 | 242 | 176 | 0,9819407 | | 0,578947368 |
| | 26 | 645 | 422 | 223 | 0,9302025 | | 0,654263565 |
| | 27 | 625 | 399 | 226 | 0,9440032 | | 0,6384 |
| | 28 | 630 | 444 | 186 | 0,8753918 | | 0,704761904 |
| | 29 | 230 | 155 | 75 | 0,9108783 | | 0,673913043 |
| | 30 | 220 | 143 | 77 | 0,9340680 | | 0,65 |
| | 31 | 125 | 88 | 37 | 0,8763462 | | 0,704 |
| | 32 | 132 | 78 | 54 | 0,9760206 | | 0,590909090 |
| | 33 | 124 | 84 | 40 | 0,9071657 | | 0,677419354 |
| | 34 | 136 | 94 | 42 | 0,8918107 | | 0,691176470 |
| | 35 | 123 | 78 | 45 | 0,9474351 | | 0,634146341 |
| | 36 | 139 | 94 | 45 | 0,9084033 | | 0,676258992 |
| | 37 | 141 | 98 | 43 | 0,8872751 | | 0,695035461 |
| | 38 | 136 | 96 | 40 | 0,8739810 | | 0,705882352 |
| | 39 | 131 | 92 | 39 | 0,8784734 | | 0,702290076 |
| | 40 | 134 | 93 | 41 | 0,8884667 | | 0,694029850 |
| | 42 | 82 | 59 | 23 | 0,8561146 | | 0,719512195 |
| Gender | | | | | | 1,017997561 | |
| | Male | 2778 | 2062 | 716 | 0,8233158 | | 0,742260619 |
| | Female | 1875 | 991 | 884 | 0.9976495 | | 0.528533333 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Everbench | | | | | | 0,2084845036 | |
| | Yes | 478 | 261 | 217 | 0,9938791 | | 0,546025104 |
| | No | 4175 | 2792 | 1383 | 0,9162059 | | 0,668742515 |
| Experience | | | | | | 0,9519932168 | |
| | 0 | 355 | 231 | 124 | 0,9334375 | | 0,650704225 |
| | 1 | 558 | 370 | 188 | 0,9218384 | | 0,663082437 |
| | 2 | 1087 | 688 | 399 | 0,9483920 | | 0,632934682 |
| | 3 | 786 | 487 | 299 | 0,9583290 | | 0,619592875 |
| | 4 | 931 | 634 | 297 | 0,9033036 | | 0,680988184 |
| | 5 | 919 | 631 | 288 | 0,8970412 | | 0,686615886 |
| | 6 | 8 | 6 | 2 | 0,8112781 | | 0,75 |
| | 7 | 9 | 6 | 3 | 0,9182958 | | 0,666666666 |

### 4.1.6 Proses Algoritma Random Forest

Langkah-langkah pada prediksi turnover karyawan salah satu menggunakan metode Random Forest dilakukan sesuai pada tabel 4.5.1, berikut perhitungan Random Forest:

| Education | JoiningYear | City | PaymentTier | Age | Gender | EverBenched | ExperienceInCurrentDomain | Actual | Predicted | True/False | Total Benar | Akurasi |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2016 | 0 | 3 | 24 | 1 | 0 | 2 | 1 | 0 | Salah | 4001 | 0.86006 |
| 0 | 2013 | 0 | 3 | 26 | 1 | 0 | 4 | 0 | 0 | Benar | | |
| 0 | 2017 | 2 | 2 | 25 | 1 | 0 | 3 | 1 | 0 | Salah | | |
| 1 | 2015 | 1 | 2 | 28 | 1 | 0 | 2 | 0 | 0 | Benar | | |
| 0 | 2012 | 0 | 3 | 33 | 0 | 0 | 1 | 0 | 0 | Benar | | |
| 1 | 2012 | 2 | 3 | 24 | 1 | 0 | 2 | 1 | 0 | Salah | | |
| 0 | 2016 | 1 | 3 | 27 | 1 | 0 | 5 | 1 | 1 | Benar | | |
| 0 | 2015 | 1 | 2 | 25 | 1 | 1 | 3 | 1 | 1 | Salah | | |
| 0 | 2015 | 2 | 3 | 28 | 0 | 0 | 1 | 0 | 0 | Benar | | |
| 0 | 2015 | 1 | 2 | 27 | 1 | 0 | 5 | 1 | 1 | Benar | | |
| 0 | 2013 | 0 | 3 | 27 | 0 | 0 | 5 | 0 | 0 | Benar | | |
| 0 | 2017 | 0 | 3 | 24 | 1 | 0 | 2 | 0 | 0 | Benar | | |
| 0 | 2018 | 1 | 2 | 39 | 1 | 0 | 4 | 1 | 1 | Benar | | |
| 0 | 2012 | 0 | 3 | 27 | 0 | 1 | 5 | 0 | 0 | Benar | | |
| 1 | 2017 | 2 | 2 | 34 | 0 | 1 | 2 | 0 | 0 | Benar | | |
| 0 | 2015 | 1 | 2 | 26 | 1 | 0 | 4 | 1 | 1 | Benar | | |
| 0 | 2014 | 1 | 3 | 24 | 0 | 0 | 2 | 0 | 0 | Benar | | |
| 2 | 2016 | 2 | 2 | 28 | 0 | 0 | 3 | 1 | 0 | Salah | | |
| 1 | 2012 | 1 | 3 | 27 | 0 | 0 | 5 | 1 | 1 | Benar | | |
| 0 | 2016 | 0 | 3 | 40 | 0 | 0 | 5 | 0 | 0 | Benar | | |
| 0 | 2012 | 0 | 3 | 28 | 0 | 0 | 0 | 0 | 1 | Salah | | |
| 0 | 2015 | 1 | 2 | 25 | 0 | 0 | 3 | 1 | 1 | Benar | | |
| 0 | 2017 | 1 | 2 | 33 | 1 | 0 | 2 | 1 | 1 | Benar | | |
| 0 | 2015 | 0 | 3 | 27 | 1 | 0 | 5 | 0 | 0 | Benar | | |
| 0 | 2012 | 0 | 1 | 27 | 1 | 0 | 5 | 0 | 0 | Benar | | |
| 0 | 2015 | 0 | 3 | 26 | 0 | 0 | 4 | 0 | 0 | Benar | | |

**Gambar 4.1 19 Proses Algoritma Random Forest**

### 4.1.7 Proses Algoritma K-Nearest Neighbor

Langkah-langkah pada prediksi turnover karyawan salah satu menggunakan metode K-Nearest Neighbor dilakukan sesuai pada tabel 4.5.1, berikut perhitungan K-Nearest Neighbor:

| Education | JoiningYear | City | PaymentTier | Age | Gender | EverBenched | ExperienceInCurrentDomain | Actual | Predicted | Correct Prediction | Akurasi |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.666666667 | 0 | 1 | 0.105263 | 0 | 0 | 0.285714286 | 1 | 0 | 0 | 0.78981 |
| 0 | 0.166666667 | 0 | 1 | 0.210526 | 0 | 0 | 0.571428571 | 0 | 0 | 1 | |
| 0 | 0.833333333 | 1 | 0.5 | 0.157895 | 0 | 0 | 0.428571429 | 1 | 0 | 0 | |
| 1 | 0.5 | 2 | 0.5 | 0.315789 | 0 | 0 | 0.285714286 | 0 | 0 | 1 | |
| 0 | 0 | 0 | 1 | 0.578947 | 1 | 0 | 0.142857143 | 0 | 0 | 1 | |
| 1 | 0 | 1 | 1 | 0.105263 | 0 | 0 | 0.285714286 | 1 | 0 | 0 | |
| 0 | 0.666666667 | 2 | 1 | 0.263158 | 0 | 0 | 0.714285714 | 1 | 0 | 0 | |
| 0 | 0.5 | 2 | 0.5 | 0.157895 | 0 | 1 | 0.428571429 | 1 | 1 | 1 | |
| 0 | 0.5 | 1 | 1 | 0.315789 | 1 | 0 | 0.142857143 | 0 | 0 | 1 | |
| 0 | 0.5 | 2 | 0.5 | 0.263158 | 0 | 0 | 0.714285714 | 1 | 1 | 1 | |
| 0 | 0.166666667 | 0 | 1 | 0.263158 | 1 | 0 | 0.714285714 | 0 | 1 | 0 | |
| 0 | 0.833333333 | 0 | 1 | 0.105263 | 0 | 0 | 0.285714286 | 0 | 0 | 1 | |
| 0 | 1 | 2 | 0.5 | 0.894737 | 0 | 0 | 0.571428571 | 1 | 0 | 0 | |
| 0 | 0 | 0 | 1 | 0.263158 | 1 | 1 | 0.714285714 | 0 | 0 | 1 | |
| 1 | 0.833333333 | 1 | 0.5 | 0.631579 | 1 | 1 | 0.285714286 | 0 | 0 | 1 | |
| 0 | 0.5 | 2 | 0.5 | 0.210526 | 0 | 0 | 0.571428571 | 1 | 0 | 0 | |
| 0 | 0.333333333 | 2 | 1 | 0.105263 | 1 | 0 | 0.285714286 | 0 | 0 | 1 | |
| 2 | 0.666666667 | 1 | 0.5 | 0.315789 | 1 | 0 | 0.428571429 | 1 | 1 | 1 | |
| 1 | 0 | 2 | 1 | 0.263158 | 1 | 0 | 0.714285714 | 1 | 0 | 0 | |
| 0 | 0.666666667 | 0 | 1 | 0.947368 | 1 | 0 | 0.714285714 | 0 | 0 | 1 | |
| 0 | 0 | 0 | 1 | 0.315789 | 1 | 0 | 0 | 0 | 1 | 0 | |
| 0 | 0.5 | 2 | 0.5 | 0.157895 | 1 | 0 | 0.428571429 | 1 | 0 | 0 | |
| 0 | 0.833333333 | 2 | 0.5 | 0.578947 | 0 | 0 | 0.285714286 | 1 | 1 | 1 | |
| 0 | 0.5 | 0 | 1 | 0.263158 | 0 | 0 | 0.714285714 | 0 | 0 | 1 | |
| 0 | 0 | 0 | 0 | 0.263158 | 0 | 0 | 0.714285714 | 0 | 0 | 1 | |
| 0 | 0.5 | 0 | 1 | 0.210526 | 1 | 0 | 0.571428571 | 0 | 0 | 1 | |

**Gambar 4.1. 20 Proses algoritma K-Nearest Neighbor**

### 4.1.8 Pohon Keputusan

Pohon keputusan atau graph view menunjukan hasil dari percabangan yang dapat dihasilkan kesimpulan. Berikut adalah hasil dari *decision tree.*