Rakamin Academy

Kelompok 8A : Astasena

Dokumen Laporan Final Project

(dipresentasikan setiap sesi mentoring)

Stage 2 - Data Pre-Processing

Data Cleansing

A. Handle Missing Values

```
data.isna().sum()
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

Dengan menggunakan syntax seperti pada gambar kita tidak menemukan adanya missing values sehingga kita tidak perlu untuk menghandle missing values, adapun hasil dari syntax seperti pada gambar menunjukan bahwa pada dataset tersebut tidak ada data kosong atau missing values.

Data Cleansing

B. Handle Duplicated Data

```
data.duplicated().sum()
```

```
0
```

Sama seperti sebelumnya pada case ini tidak ada data duplikat yang ditemukan, seperti pada gambar disamping dengan menggunakan syntax tersebut hasil yang ditampilkan adalah 0

Data Cleansing

C. Handle Outliers

Outlier tidak dihandle karena khawatir akan kehilangan informasi yang berharga dan menurut kami, outlier masih dalam batas wajar dan kami akan menggunakan model yang robust terhadap outlier.

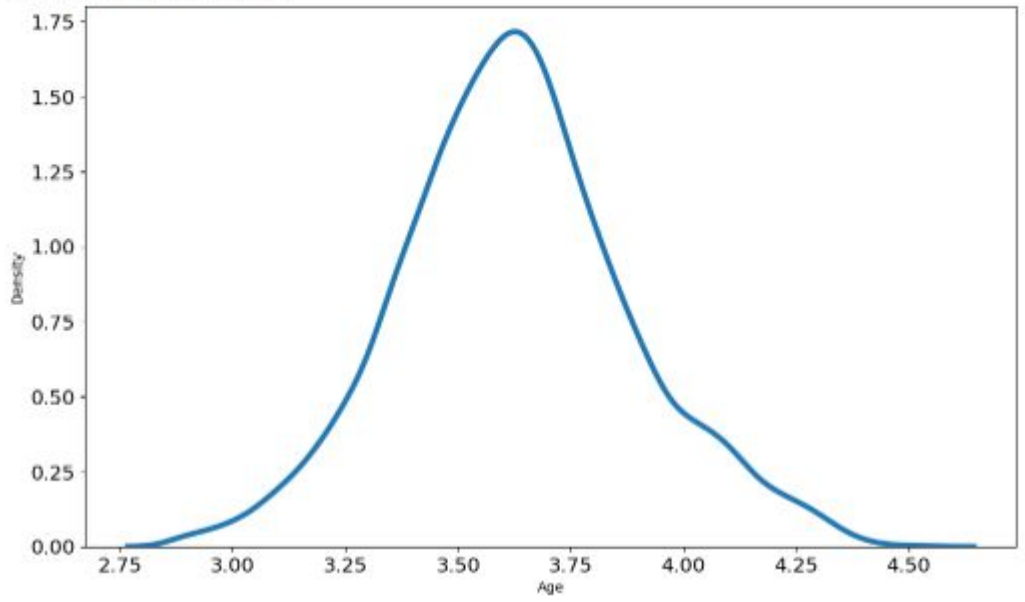
Data Cleansing

D. Feature transformation

Adapun pada dataset ini kita melakukan Feature transformation dengan menggunakan log transformation, yang dimana kita mengubah distribusi data yang berosilasi atau memiliki ekstrem menjadi distribusi yang lebih mendekati distribusi normal. Ini berguna dalam kasus seperti data yang tidak terdistribusi normal, seperti pengukuran berbasis logaritma atau persentase. adapun kolom yang kita transformasi diantaranya adalah :

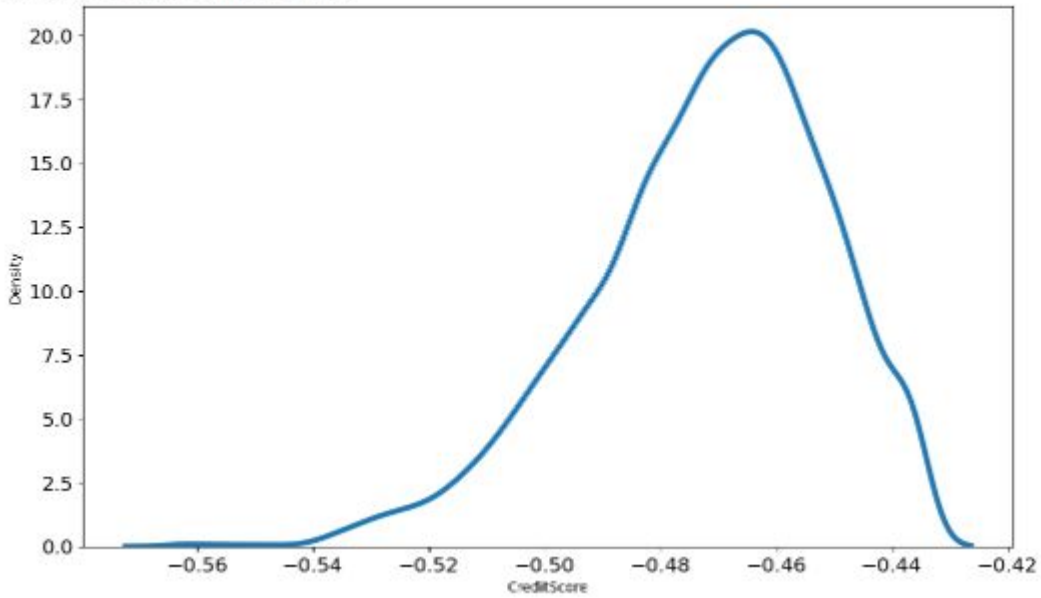
```
# distribusi Age setelah log transformation
sns.kdeplot(np.log(data['Age']))
```

```
<Axes: xlabel='Age', ylabel='Density'>
```




```
# distribusi CreditScore setelah log transformation
sns.kdeplot(np.log(data['CreditScore']))
```

```
<Axes: xlabel='CreditScore', ylabel='Density'>
```



Data Cleansing

E. Feature Encoding

```
import pandas as pd
data = pd.read_csv('Churn_Modelling_modified.csv')

# Melakukan one hot encoding pada kolom 'Geography'
one_hot_encoding= pd.get_dummies (data['Geography'],prefix = 'country')

 # Menggabungkan data asli dengan hasil one hot encoding
data_encoded = pd.concat([data, one_hot_encoding], axis=1)
print(data_encoded.head())
```

```
   RowNumber  CustomerId   Surname  CreditScore Geography  Gender  Age  \
0          1    15634602  Hargrave          619    France       1   42
1          2    15647311      Hill          608     Spain       1   41
2          3    15619304      Onio          502    France       1   42
3          4    15701354      Boni          699    France       1   39
4          5    15737888  Mitchell          850     Spain       1   43

   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       2       0.00              1          1               1
1       1   83807.86              1          0               1
2       8  159660.80              3          1               0
3       1       0.00              2          0               0
4       2  125510.82              1          1               1

   EstimatedSalary  Exited  country_France  country_Germany  country_Spain
0        101348.88       1               1                0              0
1        112542.58       0               0                0              1
2        113931.57       1               1                0              0
3         93826.63       0               1                0              0
4         79084.10       0               0                0              1
```

Pada tahap ini kami menggunakan metode label encoding dan one hot encoding untuk mengubah feature kategorikal menjadi feature numeric. Label encoding digunakan untul kolom gender karena kolom tersebut memiliki urutan atau tingkatan yang dapat diirutkan. Misalnya kita dapat menganggap bahwa nilai 'male' memiliki nilai lebih rendah dan 'female' memiliki nilai lebih tinggi. Pada label encoding nilai' male di ganti dengan 0 dan nilai 'female' diganti dengan 1.

One hot encoding digunakan untuk kolom geography karena kolom tersebut tidak memiliki urutan atau tidak ada hubungan ordinal. Misalnya, pada kolom geography dengan nilai spain, germany, dan france tidak ada urutan yang jelas antar ketiga negara tersebut. Oleh karena ini, one hot encoding akan membuat kolom terpisah untuk setiap nilai unik dalam kolom geography dan mengindetitikasi kehadiran nilainya dengan menggunakan 0 dan 1.

Data Cleansing

F. Handle class imbalance

```
import pandas as pd
from imblearn.over_sampling import RandomOverSampler
from imblearn.under_sampling import RandomUnderSampler

data = pd.read_csv('Churn_Modelling.csv')

# Memisahkan fitur dan target
X = data.drop('Exited', axis=1)
y = data['Exited']

# Oversampling dengan RandomOverSampler
oversampler = RandomOverSampler()
X_oversampled, y_oversampled = oversampler.fit_resample(X, y)

# Undersampling dengan RandomUnderSampler
undersampler = RandomUnderSampler()
X_undersampled, y_undersampled = undersampler.fit_resample(X, y)

# Menampilkan informasi tentang jumlah sampel di setiap kelas setelah oversampling
print("Jumlah sampel setelah oversampling:")
print(y_oversampled.value_counts())

# Menampilkan informasi tentang jumlah sampel di setiap kelas setelah undersampling
print("Jumlah sampel setelah undersampling:")
print(y_undersampled.value_counts())
```

```
Jumlah sampel setelah oversampling:
1    7963
0    7963
Name: Exited, dtype: int64
Jumlah sampel setelah undersampling:
0    2037
1    2037
Name: Exited, dtype: int64
```

Dataset “Churn_Modelling ” memiliki jumlah data yang tidak seimbang antara kelas churn dan tidak churn. Hal ini dapat dilihat dari distribusi kelas pada kolom target “Exited”, di mana hanya sekitar 20% data yang termasuk ke dalam kelas churn. Ketidakseimbangan ini dapat mempengaruhi performa model machine learning, di mana model cenderung lebih memperhatikan kelas mayoritas dan mengabaikan kelas minoritas.

Untuk mengatasi masalah ini, kita dapat melakukan teknik class balancing seperti oversampling atau undersampling. Oversampling dilakukan dengan menambahkan data pada kelas minoritas, sedangkan undersampling dilakukan dengan menghapus data pada kelas mayoritas. Selain itu, kita juga dapat menggunakan teknik class weighting pada model machine learning, di mana bobot yang lebih besar diberikan pada kelas minoritas.

Feature Engineering

A. Feature selection (membuang feature yang kurang relevan atau redundan)

Di tahap ini kami menghapus feature yg kurang relevan yaitu feature RowNumber, CostumerId, dan Surname, karena RowNumber Ini hanya nomor baris dan tidak memberikan informasi berharga. Customer ID dan Surname ini adalah identifikasi unik atau nama pelanggan, mereka mungkin tidak relevan untuk analisis churn dan bisa dihapus. Lalu kami melakukan uji korelasi feature antar feature kategorikal dengan menggunakan heatmap, dari heatmap ini bisa di simpulkan bahwa ada feature yg memiliki korelasi kuat yaitu feature Numofproduct-Balance, setelah itu kami menggunakan analisis untuk melihat multicolinearity nya dengan menggunakan metode VIF, dan mendapatkan hasil bahwa tingkat multicolinearity untuk Numofproduct 1.7554292400724105 dan Balance 1.7554292400724092, sehingga dapat di katakan tidak ada redundant dalam feature² numeric ini

```
import pandas as pd
data = pd.read_csv('Churn_Modelling.csv')

data = data.drop(columns = ['RowNumber', 'CustomerId','Surname'])
display (data)
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	619	France	Female	42	2	0.00	1	1	1	101348.88	1
1	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
2	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0
...	...	...	...	...	...	...	...	...	...	...	...

Feature Engineering

B. Feature extraction (membuat feature baru dari feature yang sudah ada)

Disini kita menambahkan fitur age_grup yang dimana kita mengkelompokkan usia dari range yang telah kita tentukan agar memudahkan dalam menganalisa pelanggan dari segi umur

```
import pandas as pd

# Membaca data dari file CSV
data = pd.read_csv('Churn_Modelling.csv')

# Membuat grup usia dengan menggunakan fungsi cut
bins = [0, 18, 30, 40, 50, 60, 100]  # Batas-batas grup usia
labels = ['<18', '18-30', '31-40', '41-50', '51-60', '60+']  # Label untuk setiap grup usia
data['age_group'] = pd.cut(data['Age'], bins=bins, labels=labels, right=False)
data
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited	age_group
0	1	15634602	Hargrave	619	France	Female	42	2	0.00	1	1	1	101348.88	1	41-50
1	2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0	41-50
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1	41-50
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0	31-40
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0	41-50
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9995	9996	15606229	Obijiaku	771	France	Male	39	5	0.00	2	1	0	96270.64	0	31-40
9996	9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0	31-40
9997	9998	15584532	Liu	709	France	Female	36	7	0.00	1	0	1	42085.58	1	31-40
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	92888.52	1	41-50
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0	18-30

10000 rows × 15 columns

Feature Engineering

C. 4 feature tambahan

1. Feature Importance :

   Feature Importance adalah teknik untuk mengukur seberapa besar pengaruh suatu fitur terhadap hasil prediksi model machine learning. Fitur yang memiliki nilai feature importance tinggi berarti memiliki kontribusi besar terhadap prediksi churn, sedangkan fitur yang memiliki nilai feature importance rendah berarti kurang relevan atau bahkan dapat mengganggu prediksi churn

2. Riwayat Transaksi (Transaction History) :

   Fitur ini juga dapat kita tambahkan agar kita mengetahui tentang pola transaksi pelanggan, seperti jumlah transaksi bulanan, jumlah transfer, jumlah setoran, dll. Ini dapat memberikan wawasan tentang aktivitas keuangan pelanggan dan apakah mereka cenderung berpindah ke bank lain

3. Status Pekerjaan (Employment Status) :

   Fitur ini dapat menggambarkan apakah pelanggan bekerja penuh waktu, paruh waktu, pengangguran, atau memiliki pekerjaan lainnya. Meskipun sudah ada sallary, akan tetapi status pekerjaan dapat memengaruhi stabilitas keuangan pelanggan.

4. Rating Layanan Pelanggan (Customer Service Rating) :

   Informasi ini perlu kita dapatkan untuk mengetahui bagaimana kepuasaan pelanggan terhadap service kita, sehingga kita dapat selalu mengevaluasi atau mengupdate service kita sesuai dengan kebutuhan customer