Rakamin Academy

Homework
Data Pre-Processing

Final Project - Stage 2

1. Data Cleansing

A. Handle missing values

```
# Mengecek dataset info
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1987 entries, 0 to 1986
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Age                  1987 non-null   int64
 1   Employment Type      1987 non-null   object
 2   GraduateOrNot        1987 non-null   object
 3   AnnualIncome         1987 non-null   int64
 4   FamilyMembers        1987 non-null   int64
 5   ChronicDiseases      1987 non-null   int64
 6   FrequentFlyer        1987 non-null   object
 7   EverTravelledAbroad  1987 non-null   object
 8   TravelInsurance      1987 non-null   int64
```

```
# Cek data kosong
df.isnull().sum()
```

```
Age                    0
Employment Type        0
GraduateOrNot          0
AnnualIncome           0
FamilyMembers          0
ChronicDiseases        0
FrequentFlyer          0
EverTravelledAbroad    0
TravelInsurance        0
```

Pada dataset Travel Insurance dijalankan fungsi info() dan isnull() seperti pada gambar di atas. Diperoleh informasi bahwa data terdiri dari 1987 baris. Pada setiap kolom nya juga terdiri dari 1987 baris data atau tidak ditemukan missing value.

1. Data Cleansing

B. Handle duplicated data

```
# Cek data duplikat
df.duplicated(subset = ['Age','Employment Type','GraduateOrNot','AnnualIncome','FamilyMembers','ChronicDiseases','FrequentFlyer','EverTravelledAbroad',"TravelInsurance"]).sum()
```

```
738
```

```
# Remove baris data duplikat
df = df.drop_duplicates(subset = ['Age','Employment Type','GraduateOrNot','AnnualIncome','FamilyMembers','ChronicDiseases','FrequentFlyer','EverTravelledAbroad',"TravelInsurance"])
df.info()
```

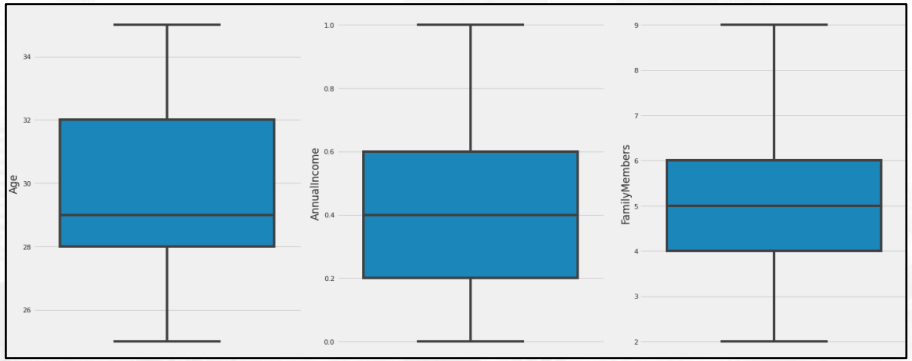
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1249 entries, 0 to 1985
Data columns (total 10 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Unnamed: 0           1249 non-null   int64
 1   Age                  1249 non-null   int64
 2   Employment Type      1249 non-null   object
 3   GraduateOrNot        1249 non-null   object
 4   AnnualIncome         1249 non-null   int64
 5   FamilyMembers        1249 non-null   int64
 6   ChronicDiseases      1249 non-null   int64
 7   FrequentFlyer        1249 non-null   object
 8   EverTravelledAbroad  1249 non-null   object
 9   TravelInsurance      1249 non-null   int64
```

Pada dataset Travel Insurance dijalankan fungsi duplicated() seperti gambar pertama untuk mengetahui baris duplikat dan diperoleh informasi bahwa terdapat 738 baris duplikat.

Selanjutnya, dilakukan drop duplikat untuk menghapus baris duplikat dan tersisa 1249 baris data saja.

1. Data Cleansing

C. Handle outliers (1/2)



Jika dilihat dari boxplot pada gambar di atas, dataset Travel Insurance tidak memiliki outliers untuk kolom dengan data bertipe numerik. Sehingga, tidak diperlukan handle outliers.

1. Data Cleansing

C. Handle outliers (2/2)

```
# Melihat isi Unique pada Coloumn
for column in df.columns:
    print(f"{column} :")
    print(df[column].unique())
    print("")
```

```
Age :
[31 34 28 25 33 26 32 29 35 30 27]

Employment Type :
['Government Sector' 'Private Sector/Self Employed']

GraduateOrNot :
['Yes' 'No']

AnnualIncome :
[ 400000 1250000  500000  700000 1150000 1300000 1350000 1450000  800000
 1400000  850000 1500000 1050000  350000 1100000  600000  900000  550000
  300000  750000 1200000 1000000  950000 1700000 1750000  650000  450000
 1650000 1800000 1550000]

FamilyMembers :
[6 7 4 3 8 9 5 2]

ChronicDiseases :
[1 0]

FrequentFlyer :
['No' 'Yes']

EverTravelledAbroad :
['No' 'Yes']

TravelInsurance :
[0 1]
```

```
df[cat].describe()
```

	Employment Type	GraduateOrNot	FrequentFlyer	EverTravelledAbroad
count	1987	1987	1987	1987
unique	2	2	2	2
top	Private Sector/Self Employed	Yes	No	No
freq	1417	1692	1570	1607

Sedangkan untuk data kategorik pada kolom employment type, graduate or not, frequent flyer, dan ever travelled abroad hanya memiliki 2 nilai unique dengan jumlah unique value yang tidak jauh berbeda untuk masing-masing feature. Sehingga, dapat ditarik kesimpulan juga bahwa tidak ada data outliers untuk feature bertipe kategorik.

1. Data Cleansing

D. Feature Transformation

1. Log-transformation
   Pada data set travel insurance, feature yang bertipe kategorik memiliki skewness sebagai berikut.

```
#skewness value
for i in range(0, len(num)):
    print(f"Skewness {df[num].columns[i]} : {df[num[i]].skew()}")

Skewness Age : 0.1766593709859495
Skewness AnnualIncome : 0.1455499784860766
Skewness FamilyMembers : 0.44079226295946317
```

   Berdasarkan data di atas, tidak ada data yang positively skewed, sehingga tidak perlu dilakukan log-transformation.

1. Normalization
   Dataset tidak perlu dilakukan normalisasi karena terlihat masih cukup balance. Selain itu, data tidak dilakukan normalisasi untuk melihat performa model dengan algoritma yang tidak memperhitungkan distance di stage selanjutnya.

1. Standardization
   Feature transformation ini tidak dilakukan karena data bertipe numerik yang ada sudah mendekati normal.

1. Data Cleansing

E. Feature Encoding (1/3)

```
# Melihat isi Unique pada Coloumn
for column in df.columns:
    print(f"{column} :")
    print(df[column].unique())
    print("")
```

```
Age :
[31 34 28 25 33 26 32 29 35 30 27]

Employment Type :
['Government Sector' 'Private Sector/Self Employed']

GraduateOrNot :
['Yes' 'No']

AnnualIncome :
[ 400000 1250000  500000  700000 1150000 1300000 1350000 1450000  800000
 1400000  850000 1500000 1050000  350000 1100000  600000  900000  550000
  300000  750000 1200000 1000000  950000 1700000 1750000  650000  450000
 1650000 1800000 1550000]

FamilyMembers :
[6 7 4 3 8 9 5 2]

ChronicDiseases :
[1 0]

FrequentFlyer :
['No' 'Yes']

EverTravelledAbroad :
['No' 'Yes']

TravelInsurance :
[0 1]
```

Pada dataset travel insurance terdapat data kategorik pada kolom:

- Employment type (government sector & private sector / self employed)
- Graduate or not (yes & no)
- Frequent flyer (yes & no)
- Ever travelled abroad (yes & no)

1. Data Cleansing

E. Feature Encoding (2/3)

```
# Melakukan Categorial Encoding agar data kategorik bisa ditampilkan di heatmap
df['Employment Type']=df['Employment Type'].map({'Private Sector/Self Employed':1,'Government Sector':0})
df['GraduateOrNot']=df['GraduateOrNot'].map({'Yes':1,'No':0})
df['FrequentFlyer']=df['FrequentFlyer'].map({'No':0,'Yes':1})
df['EverTravelledAbroad']=df['EverTravelledAbroad'].map({'No':0,'Yes':1})
```

Dilakukan feature encoding pada ke empat feature tersebut dengan ketentuan sebagai berikut

- Employment type
  - 0 : government sector
  - 1 : private sector / self employed
- Graduate or not
  - 0 : not graduated
  - 1 : graduated
- Frequent flyer
  - 0 : no
  - 1 : yes
- Ever travelled abroad
  - 0 : no
  - 1 : yes

1. Data Cleansing

E. Feature Encoding (3/3)

Berikut adalah data yang sudah dilakukan feature encoding untuk kolom employment type, graduate or not, frequent flyer, dan ever travel abroad.

	index	Age	Employment Type	GraduateOrNot	AnnualIncome	FamilyMembers	ChronicDiseases	FrequentFlyer	EverTravelledAbroad	TravelInsurance
0	0	31	0	1	400000	6	1	0	0	0
1	1	31	1	1	1250000	7	0	0	0	0
2	2	34	1	1	500000	4	1	0	0	1
3	3	28	1	1	700000	3	1	0	0	0
4	4	28	1	1	700000	8	1	1	0	0
...	...	...	...	...	...	...	...	...	...	...
1976	1976	32	0	1	900000	6	0	0	0	0
1981	1981	27	0	1	850000	3	0	0	0	1
1982	1982	33	1	1	1500000	4	0	1	1	1
1983	1983	28	1	1	1750000	5	1	0	1	0
1985	1985	34	1	1	1000000	6	0	1	1	1

1249 rows × 10 columns

1. Data Cleansing

F. Handle class imbalance

Pada bagian ini, dilakukan analisa terlebih dahulu pada data target yaitu Travel Insurance.

```
# Travel Insurance
df[cat].stb.freq(['TravelInsurance'], cum_cols=False)
```

	TravelInsurance	count	percent
0	0	1277	64.26774
1	1	710	35.73226

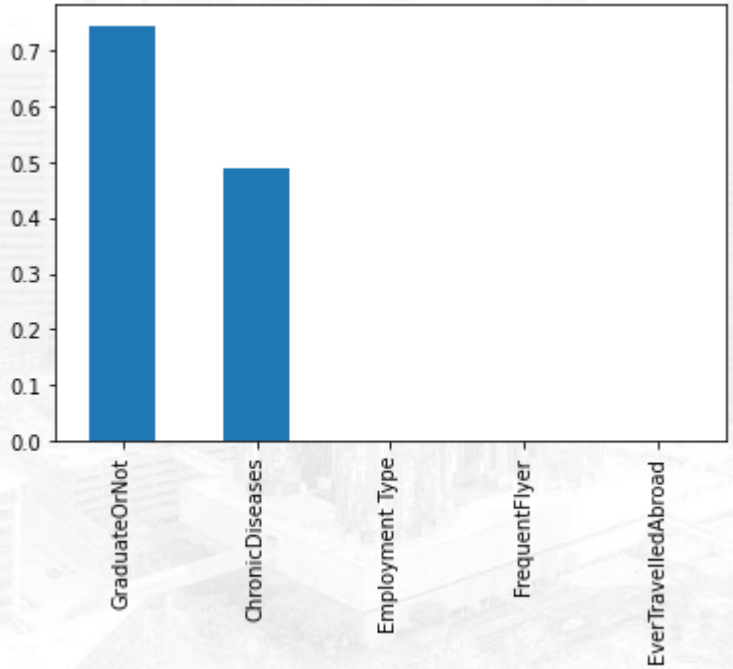
Pada kolom travel insurance, data terbagi menjadi 64.27% tidak membeli travel insurance dan 35.73% membeli travel insurance. Berdasarkan hasil tersebut, ditarik kesimpulan bahwa tidak perlu dilakukan oversampling atau undersampling pada dataset dikarenakan ketimpangan data masih berada pada **range mild** yaitu di antara **20-40%** lebih tepatnya ada di angka 35,73%.

Sumber: https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data ..

2. Feature Engineering

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

A. Chi Square (Fitur Kategorik -> Target Kategorik)



Chi Square digunakan untuk fitur kategorik terhadap target kategorik. Fitur kategorik yang akan dites adalah GraduateOrNot, ChronicDiseasses, Employment Type, FrequentFlyer, EverTravelledAbroad.

Berdasarkan grafik diatas fitur yang digunakan :

- Employment Type
- FrequentFlyer
- EverTravelledAbroad

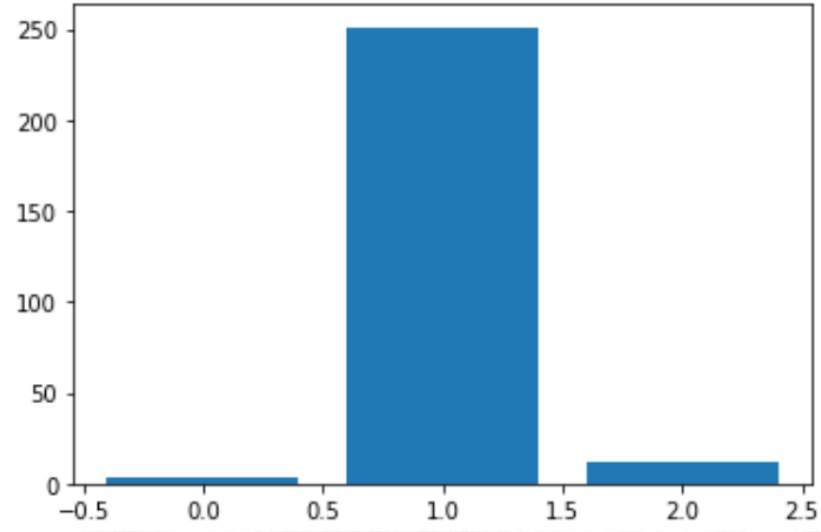
Berdasarkan hasil uji chi-square fitur yang tidak digunakan adalah GraduateOrNot dan ChronicDiseasses.

2. Feature Engineering

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

B. Anova (Fitur Numerik -> Target Kategorik)

```
Feature Age: 3.610883
Feature AnnualIncome: 251.094542
Feature FamilyMembers: 12.293209
```



Anova digunakan untuk fitur numerik terhadap target kategorik.

Berdasarkan grafik diatas fitur yang dapat digunakan :

- Age
- AnnualIncome
- FamilyMembers

Berdasarkan hasil Anova semua fitur bisa digunakan untuk pembuatan model. Walaupun begitu fitur FamilyMembers tidak akan digunakan dalam data preprocessing alasannya karena berdasarkan pandangan bisnis terkait produk yang ingin ditawarkan kepada customer yaitu produk untuk satu individu saja dan hanya mengcover individu yang membeli tersebut. Oleh sebab itu, fitur yang dipilih adalah **Age** dan **AnnualIncome** saja.

Selain itu, berdasarkan hasil analisa melalui heatmap di stage sebelumnya feature family member tidak berkorelasi kuat dengan target.

2. Feature Engineering

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

Berdasarkan business point of view dan didukung dengan hasil dari uji anova dan chi-square yang sudah dilakukan, maka disimpulkan bahwa feature-feature yang dipilih adalah sebagai berikut

	index	Age	Employment Type	AnnualIncome	FrequentFlyer	EverTravelledAbroad	TravelInsurance
0	0	31	Government Sector	400000	No	No	0
1	1	31	Private Sector/Self Employed	1250000	No	No	0
2	2	34	Private Sector/Self Employed	500000	No	No	1
3	3	28	Private Sector/Self Employed	700000	No	No	0
4	4	28	Private Sector/Self Employed	700000	Yes	No	0
...	...	...	...	...	...	...	...
1976	1976	32	Government Sector	900000	No	No	0
1981	1981	27	Government Sector	850000	No	No	1
1982	1982	33	Private Sector/Self Employed	1500000	Yes	Yes	1
1983	1983	28	Private Sector/Self Employed	1750000	No	Yes	0
1985	1985	34	Private Sector/Self Employed	1000000	Yes	Yes	1

2. Feature Engineering

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

Feature Extraction:

1. Income Bracket

Pengkategorian annual income menjadi 2 kategori yaitu high dan low yang mana annual income high adalah income yang berada di atas 1300000.

2. Traveller

Fitur yang berasal dari hasil penggabungan 2 fitur yaitu fitur frequent flyer dan ever travelled abroad yang mana apabila customer masuk kategori orang yang sering melakukan penerbangan dan pernah melakukan perjalanan ke luar negeri akan dikategorikan sebagai seorang traveller dan selain itu tidak.

3. Age bracket

Fitur didapatkan dari pembagian 2 kategori umur yaitu 30 tahun ke bawah dan di atas 30 tahun.

income_bracket	Traveller	age_bracket
0	0	0
0	0	0
0	0	0
0	0	1
0	0	1

2. Feature Engineering

C. Tuliskan minimal 4 feature tambahan

1. Foreign or Native
   Asumsi yang dipakai bahwa bepergian ke luar negeri memiliki risiko tertular berbagai macam penyakit yang bisa ditemukan di negara tujuan. Sehingga, dapat diasumsikan bahwa foreign cenderung aware untuk membeli asuransi sebagai antisipasi tertular penyakit.

1. Male or Female
   Asumsi yang dipakai adalah untuk mengetahui seberapa besar pengaruh gender dalam membeli travel insurance.

1. Married or Not Married
   Asumsi yang dipakai bahwa married person akan selalu bersama dengan pasangannya dan sangat mungkin untuk tidak mematuhi peraturan menjaga jarak 1 - 2 meter dan mengakibatkan risiko tertular penyakit lebih tinggi.

1. Disable or Not
   Asumsi yang dipakai adalah untuk mengetahui seberapa besar pengaruh gender dalam membeli travel insurance.

1. Smoker or Not
   Asumsi yang dipakai adalah smoker memiliki risiko tinggi terhadap covid19. Sehingga, kemungkinan besar smoker cenderung aware untuk membeli asuransi.

1. Destination (Dalam Negeri atau Luar Negeri)
   Asumsi yang dipakai adalah bepergian ke luar negeri berisiko tertular berbagai macam penyakit di negara tujuan. Sehingga, diasumsikan bahwa orang yang bepergian ke luar negeri akan aware untuk membeli asuransi.

fitur-fitur ini sangat berkaitan dengan personal para customer dan mungkin bisa menjadi bahan pertimbangan para customer untuk membeli atau menjadi sebab para customer untuk cenderung membeli travel insurance.

3. Git

Rakamin Academy

Link Repository GIT Group 1 UNO

Rakamin Academy

Link Google Drive Group 1 UNO