Kelompok : Analytic Adventurers

Laporan Final Project

STAGE 0

- **Problem:** Meskipun Trips & Travel.Com memiliki conversion rate yang tinggi, yaitu mencapai 18% (di atas rata-rata industri travel yang hanya 11,9%), tetapi masalah biaya pemasaran masih menjadi kendala. Hal ini disebabkan oleh pemasaran yang dilakukan secara acak tanpa mempertimbangkan potensi pelanggan yang sudah ada.

- **Peran:** Sebagai Data Scientist, bertanggung jawab menganalisis data pelanggan dan mengembangkan model prediktif untuk menentukan pelanggan yang paling potensial membeli paket Wellness Tourism.

- **Goal:** Mengefisiensikan cost marketing yang akan dikeluarkan oleh tim marketing perusahaan Trips & Travel.Com

- **Objective:**

    - **Efisiensi Anggaran Pemasaran:** Memangkas pengeluaran biaya marketing sebesar 60% dengan cara menawarkan package kepada customer yg memiliki potensi pembelian yang tinggi
    - **Optimasi Konversi:** Meningkatkan tingkat konversi pembelian melalui pitching lebih dari 18%.

- **Business Metrics:**

    - **Efisiensi Marketing Ratio:** Mengukur pengeluaran pemasaran terhadap jumlah konversi.
    - **Conversion Rate:** Mengukur persentase pelanggan yang membeli produk.

STAGE 1

Pre-Processing

1. Data Cleansing

A. Handle Missing Value

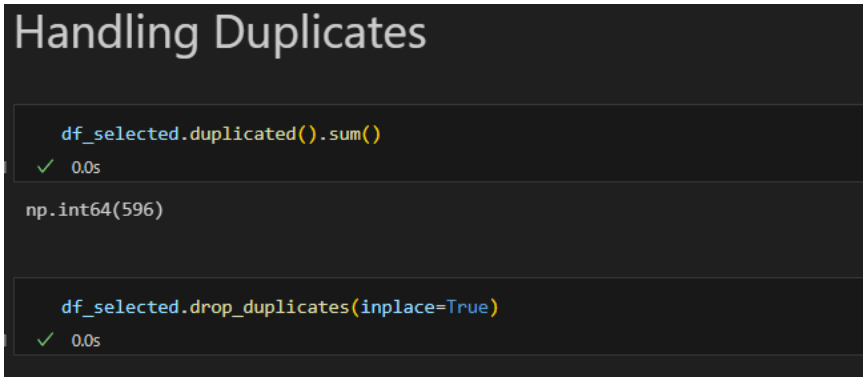
Sebelum melakukan handle missing values, pertama-tama kami memisahkan data berdasarkan kaetgori yaitu data numerikal dan kategorikal. Setelah itu kami baru melakukan pemeriksaan apakah ada missing value dalam data atau tidak. Ditemukan beberapa missing value antara lain :

Kolom	Missing Value
Age	226
DurationOfPitch	251
NumberOfFollowUps	45
PreferredPropertyStar	26
NumberOfChildrenVisiting	66
MonthlyIncome	233

Untuk menangani itu kami melakukan imputasi untuk menggisi missing value tersebut dengan nilai media untuk kolom bertipe data numerik dan mode untuk kolom bertipe data kategori

```
# Mengisi missing values untuk kolom numerik dengan nilai median
num_cols = ['Age', 'MonthlyIncome', 'NumberOfFollowups', 'DurationOfPitch']
for col in num_cols:
    df_selected[col] = df_selected[col].fillna(df_selected[col].median())

# Mengisi missing values untuk kolom kategorikal dengan nilai mode
cat_cols = ['TypeofContact', 'PreferredPropertyStar']
for col in cat_cols:
    df_selected[col] = df_selected[col].fillna(df_selected[col].mode()[0])
```

B. Handle Duplicated Data

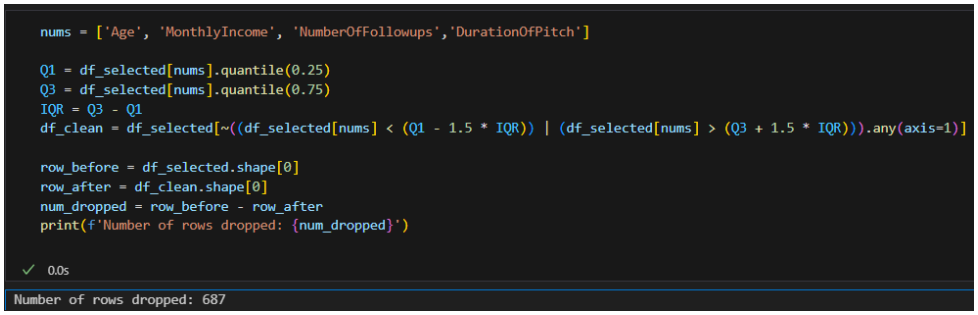


Dapat dilihat bahwa dataset yang digunakan memiliki 596 baris duplikasi data, maka dari itu kami memutuskan untuk menghapus data duplikasi tersebut agar mode tidak melatih baris data yang memiliki value sama lebih dari satu kali.

C. Handle outliers

Pada tahap ini kami melakukan pembersihan data outlier menggunakan metode IQR dikarenakan distribusi data tidak 100% normal, maka dari itu kami anggap metode IQR menjadi metode yg dapat secara optimal untuk menangani masalah tersebut.

Kolom yang akan dihilangkan outliernya diantaranya kolom 'Age', 'MonthlyIncome', 'NumberOfFollowups','DurationOfPitch'. Mengapa hanya kolom kolom tersebut, karena kolom kolom tersebutlah yg kami anggak memiliki nilai yang tidak wajar untuk terjadi.



```
nums = ['Age', 'MonthlyIncome', 'NumberOfFollowups','DurationOfPitch']

Q1 = df_selected[nums].quantile(0.25)
Q3 = df_selected[nums].quantile(0.75)
IQR = Q3 - Q1
df_clean = df_selected[~((df_selected[nums] < (Q1 - 1.5 * IQR)) | (df_selected[nums] > (Q3 + 1.5 * IQR))).any(axis=1)]

row_before = df_selected.shape[0]
row_after = df_clean.shape[0]
num_dropped = row_before - row_after
print(f'Number of rows dropped: {num_dropped}')
```

✓ 0.0s

```
Number of rows dropped: 687
```

D. Feature Encoding

```
df_selected['Designation'] = df_selected['Designation'].map({'Executive': 1, 'Manager': 2, 'Senior Manager': 3, 'AVP': 4, 'VP': 5})

df_selected = pd.concat([df_selected, pd.get_dummies(df_selected['ProductPitched'], prefix='ProductPitched', drop_first=True)], axis=1)
df_selected = pd.concat([df_selected, pd.get_dummies(df_selected['Occupation'], prefix='Occupation', drop_first=True)], axis=1)
df_selected = pd.concat([df_selected, pd.get_dummies(df_selected['MaritalStatus'], prefix='MaritalStatus', drop_first=True)], axis=1)
df_selected = pd.concat([df_selected, pd.get_dummies(df_selected['TypeofContact'], prefix='TypeofContact', drop_first=True)], axis=1)

df_selected.drop(['MaritalStatus','Occupation', 'ProductPitched', 'TypeofContact'], axis=1, inplace=True)
```

✓ 0.0s

Disini kami mengubah valu dari kolom Designation menggunakan label encoding dikarenakan value dari kolom tersebut dapat dibilang ordinal (memiliki urutan). Sedangkan untuk kolom 'MaritalStatus','Occupation', 'ProductPitched', 'TypeofContact' kami mengubah nya menggunakan one hot encoding dikarenakan value dari kolom kolom tersebut bersifat kategori tanpa memiliki urutan.

E. Feature Transformation

Feature transformation (transformasi fitur) adalah teknik dalam machine learning dan statistik yang digunakan untuk mengubah fitur data agar lebih mudah dianalisis atau dimanfaatkan oleh model prediktif. metode yang digunakan adalah Min-Max Scaling, yaitu mengubah data sehingga berada dalam rentang yang ditentukan, biasanya antara 0 dan 1, tanpa mengubah distribusi asli data tersebut.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
df_clean_scl = df_clean.copy()
df_clean_scl[num_cols] = scaler.fit_transform(df_clean[num_cols])
```

F. Handle Class Imbalance

Meskipun dataset yang kami miliki bersifat tidak seimbang (imbalanced), kami memutuskan untuk tidak melakukan balancing data. Hal ini disebabkan oleh fakta bahwa saat balancing diterapkan, model cenderung mengalami overfitting. Sebaliknya, ketika balancing tidak diterapkan, beberapa algoritma justru tidak mengalami overfitting.

2. Feature Engineering

A. Feature selection

Disini kami menggunakan Chi2 untuk menentukan feature apasaja yang akan digunakan untuk memasuki tahap pemodelan, dan untuk menghindari data yang redundan, pada saat melakukan one-hot encoding gunakan parameter drop_first = True

B. Feature Extraction

C. Tuliskan minimal 4 feature tambahan (selain yang sudah tersedia di dataset)

- Menambahkan fitur HealthRate (mengukur tingkat kesehatan customer)
- Membuat klasifikasi kepuasan pelanggan
- Membuat klasifikasi berdasarkan pendapatan customer
- Membuat fitur easy to persuade (membandingkan antara numberOfPitch dengan ProdTaken, semakin kecil numberOfPitch dari pelanggan yang

mengambil produk (ProdTaken = 1), maka semakin mudah dibujuk untuk membeli produk

STAGE 2
EXPLORATORY DATA ANALYSIS

1. Descriptive Statistics

Feature Numeric

	count	mean	std	min	25%	50%	75%	max
CustomerID	4888.0	202443.500000	1411.188388	200000.0	201221.75	202443.5	203665.25	204887.0
ProdTaken	4888.0	0.188216	0.390925	0.0	0.00	0.0	0.00	1.0
Age	4662.0	37.622265	9.316387	18.0	31.00	36.0	44.00	61.0
CityTier	4888.0	1.654255	0.916583	1.0	1.00	1.0	3.00	3.0
DurationOfPitch	4637.0	15.490835	8.519643	5.0	9.00	13.0	20.00	127.0
NumberOfPersonVisiting	4888.0	2.905074	0.724891	1.0	2.00	3.0	3.00	5.0
NumberOfFollowups	4843.0	3.708445	1.002509	1.0	3.00	4.0	4.00	6.0
PreferredPropertyStar	4862.0	3.581037	0.798009	3.0	3.00	3.0	4.00	5.0
NumberOfTrips	4748.0	3.236521	1.849019	1.0	2.00	3.0	4.00	22.0
Passport	4888.0	0.290917	0.454232	0.0	0.00	0.0	1.00	1.0
PitchSatisfactionScore	4888.0	3.078151	1.365792	1.0	2.00	3.0	4.00	5.0
OwnCar	4888.0	0.620295	0.485363	0.0	0.00	1.0	1.00	1.0
NumberOfChildrenVisiting	4822.0	1.187267	0.857861	0.0	1.00	1.0	2.00	3.0
MonthlyIncome	4655.0	23619.853491	5380.698361	1000.0	20346.00	22347.0	25571.00	98678.0

Feature Categorical

	TypeofContact	Occupation	Gender	ProductPitched	MaritalStatus	Designation
count	4863	4888	4888	4888	4888	4888
unique	2	4	3	5	4	5
top	Self Enquiry	Salaried	Male	Basic	Married	Executive
freq	3444	2368	2916	1842	2340	1842

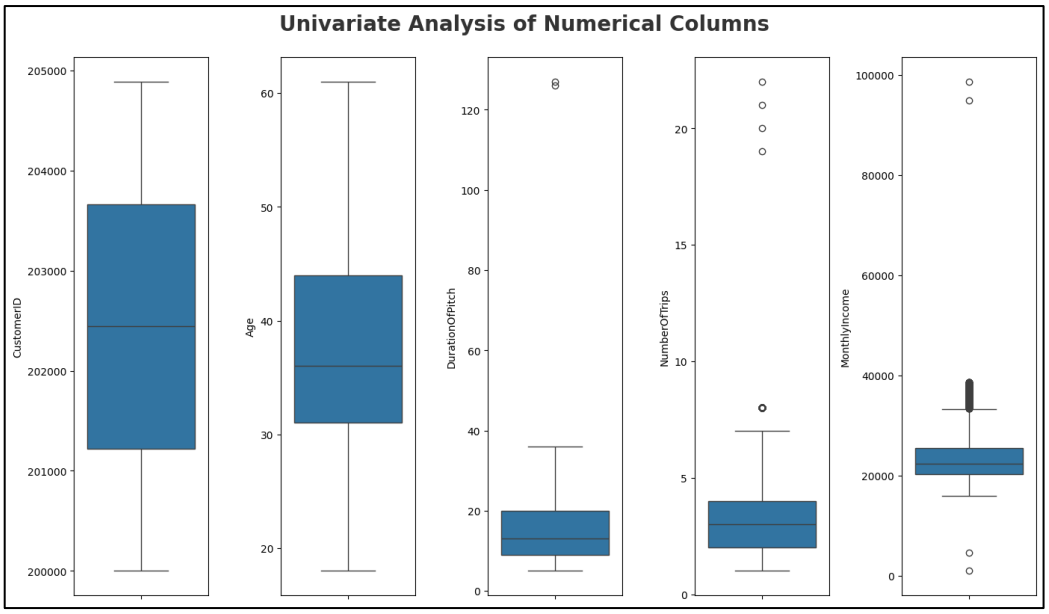
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4888 entries, 0 to 4887
Data columns (total 20 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   CustomerID                4888 non-null   int64
 1   ProdTaken                 4888 non-null   int64
 2   Age                       4662 non-null   float64
 3   TypeofContact             4863 non-null   object
 4   CityTier                  4888 non-null   int64
 5   DurationOfPitch           4637 non-null   float64
 6   Occupation                4888 non-null   object
 7   Gender                    4888 non-null   object
 8   NumberOfPersonVisiting    4888 non-null   int64
 9   NumberOfFollowups         4843 non-null   float64
 10  ProductPitched            4888 non-null   object
 11  PreferredPropertyStar     4862 non-null   float64
 12  MaritalStatus             4888 non-null   object
 13  NumberOfTrips             4748 non-null   float64
 14  Passport                  4888 non-null   int64
 15  PitchSatisfactionScore    4888 non-null   int64
 16  OwnCar                    4888 non-null   int64
 17  NumberOfChildrenVisiting  4822 non-null   float64
 18  Designation               4888 non-null   object
 19  MonthlyIncome             4655 non-null   float64
dtypes: float64(7), int64(7), object(6)
memory usage: 763.9+ KB
```

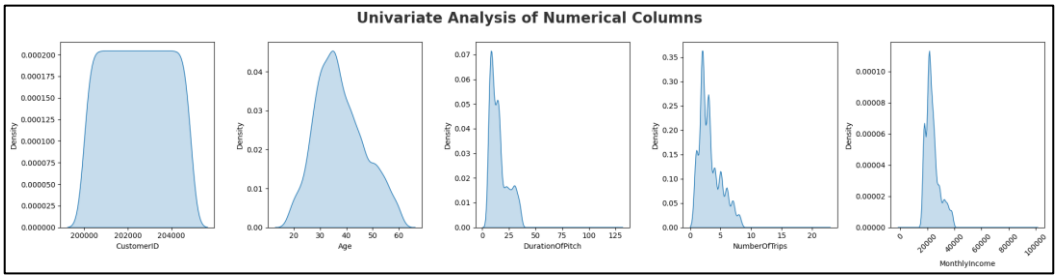
Terdapat 4888 observasi (baris) dan 20 variabel (kolom) dalam dataframe. Dari 20 variabel tersebut: 15 diantaranya adalah kategorikal. 5 adalah numerik. Tidak ada kolom yang tergolong sebagai kategorikal yang seharusnya numerik (cat_but_car). Ada 9 kolom numerik yang memiliki jumlah nilai unik kurang dari batas ambang 10 dan seharusnya kategorikal (num_but_cat).

Terdapat pula 7 kolom/fitur yang memiliki nilai kosong, yaitu kolom Age (226), TyopeofContact (25), DurationOfPitch (251), NumberOfFollowups (45), PreferredPropertyStar (26), NumberOfTrips (140), NumberOfChildrenVisiting (66), dan MonthlyIncome (133).

2. Univariate Analysis

- Numerical Columns



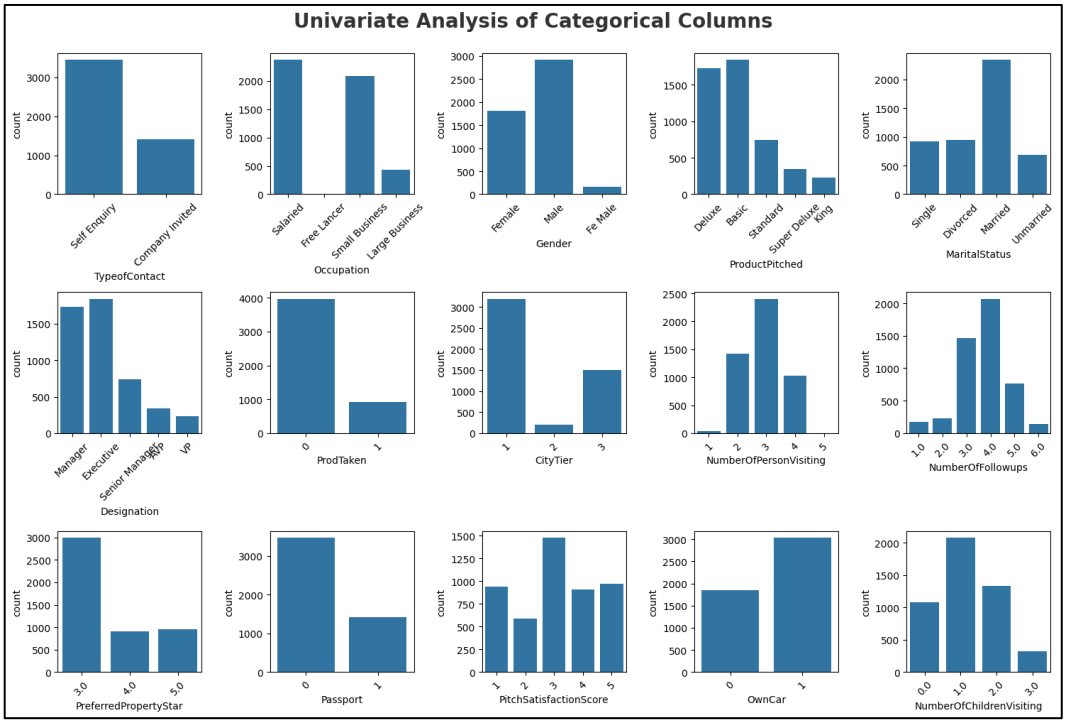


Kesimpulan:

- Kolom CustomerID memiliki sebaran data yang terlalu banyak sehingga kolom tersebut bisa dihapus nantinya.

- Kolom Age memiliki distribusi yang hampir normal.
- Kolom DurationOfPitch, NumberOfTrips, dan MonthlyIncome sepertinya memiliki distribusi data positive skewed yang mengindikasikan terdapat outlier.
- kolom lain yang sisanya termasuk jenis data diskrit atau ordinal.

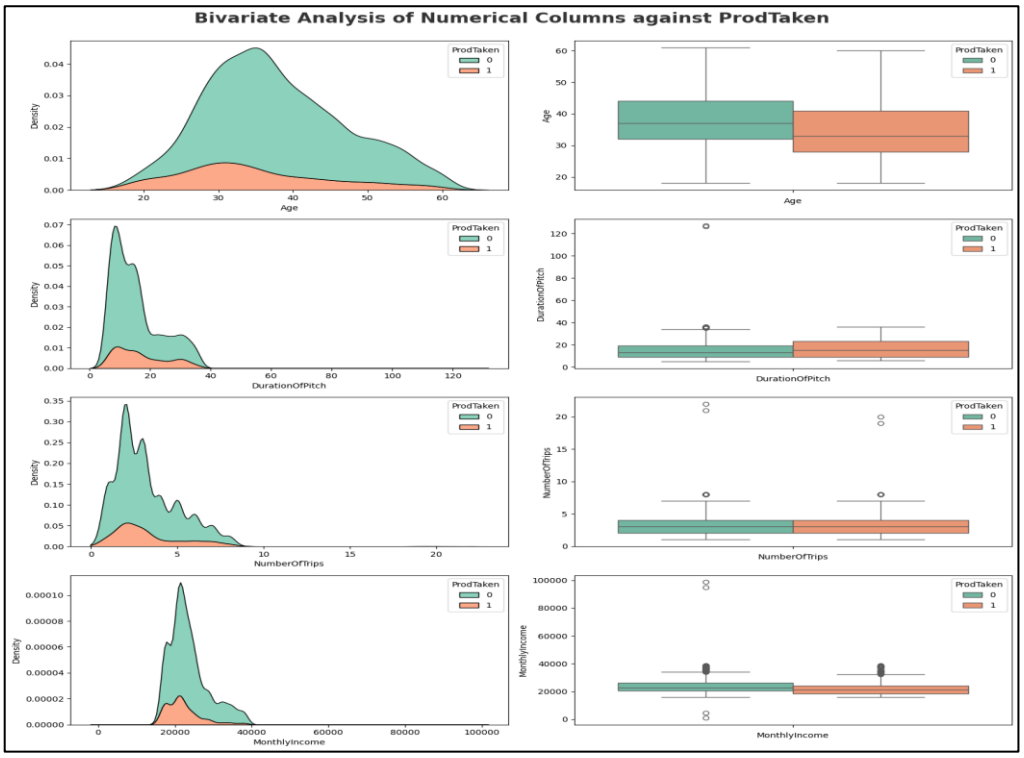
- Categorical Columns

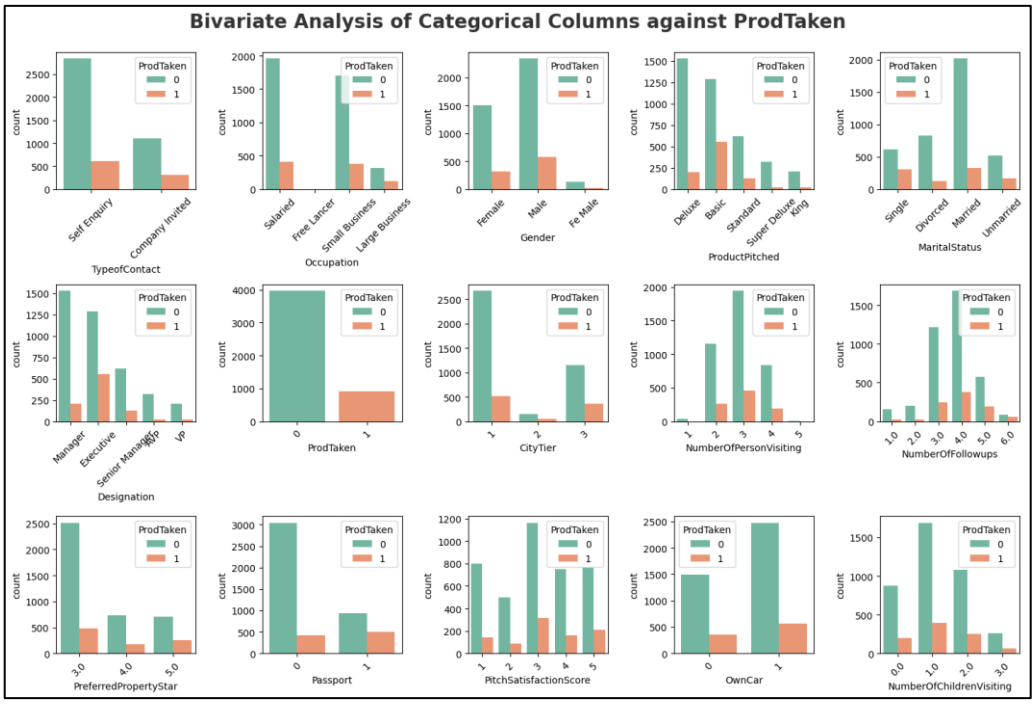


Kesimpulan:

- Pada tiap kolom memiliki perbedaan yang signifikan tiap kategori seperti pada kolom Productpitched,Designition dan occupation didominasi 2 kategori.
- NumberofFollowups didominasi diterima pada 4 kali promosi
- 50% TypeofContact oleh Self Enquiry
- Kepimilikan pasport masih sedikit

3. Multivariate Analysis

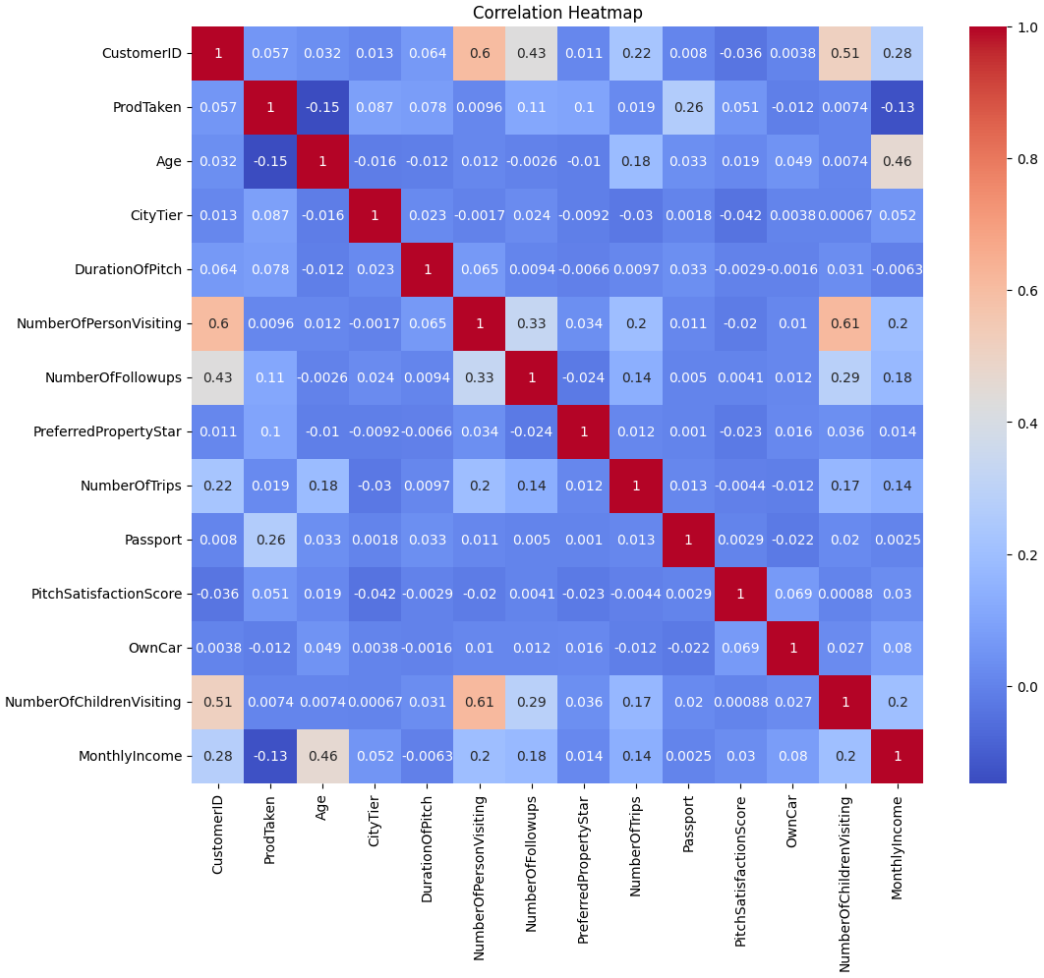




Berdasarkan Visualisasi diatas dapat disimpulkan bahwa:

- Customer dengan tipe kontrak Self Enquiry membeli paket lebih banyak daripada customer dengan tipe kontrak Company Invited
- Customer yang berada di city tier 3 memiliki persentase pembelian paket lebih tinggi setelah ditawarkan oleh sales

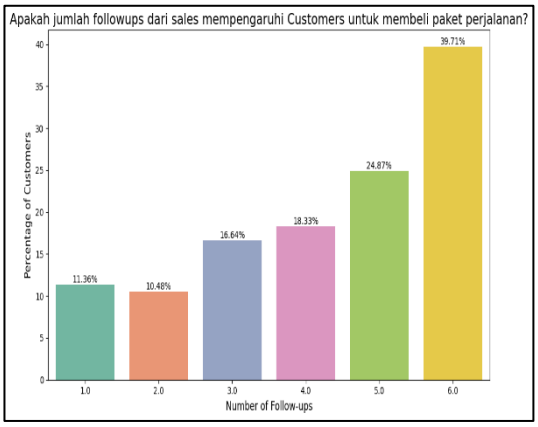
- Customer dengan Occupation Salaried dan Small Business memiliki ketertarikan untuk membeli paket yang ditawarkan
- Customer dengan gender Male lebih banyak mengambil paket yang ditawarkan daripada female atau fe male
- Distribusi jumlah orang yang ikut dalam perjalanan dengan customer yang mengambil penawaran paket travel paling banyak adalah 3 orang
- Customner yang di-follow up antara 3-5 kali lebih banyak yang mengambil penawaran travel dibandingkan dengan yang ditawarkan kurang dari 3 kali atau lebih dari 5 kali
- Product basic yang ditawarkan oleh sales lebih banyak diambil daripada produk lainnya
- Customer yang menerima penawaran paket travel lebih banyak memilih property bintang tiga dibanding bintang empat dan lima
- Customer dengan status single atau unmarried lebih banyak menerima penawaran paket travel
- Customer yang memiliki passport memiliki persentase menerima penawaran paket travel lebih tinggi daripada yang tidak memiliki passport
- Customer yang memberikan score kepuasan >=3 lebih banyak membeli paket perjalanan.
- Customer yang memiliki mobil lebih banyak menerima penawaran paket travel
- Customer dengan jumlah anak 1 lebih banyak menerima penawaran paket travel
- Customer dengan jabatan Executive lebih banyak menerima penawaran paket travel

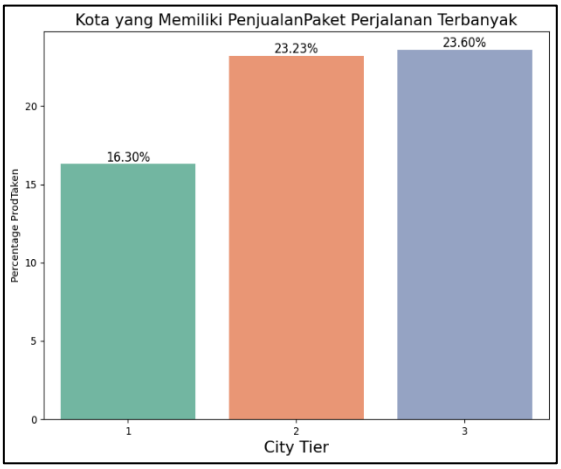


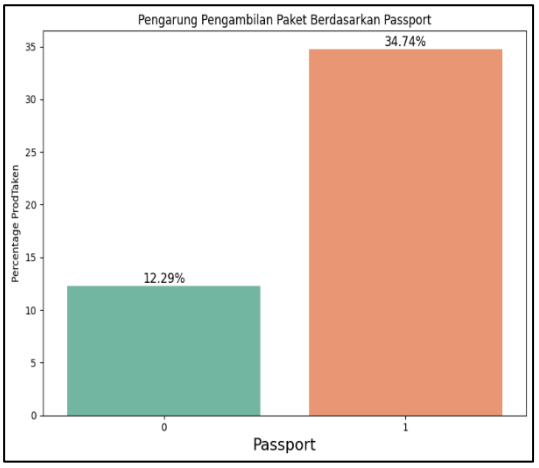
Kesimpulan :

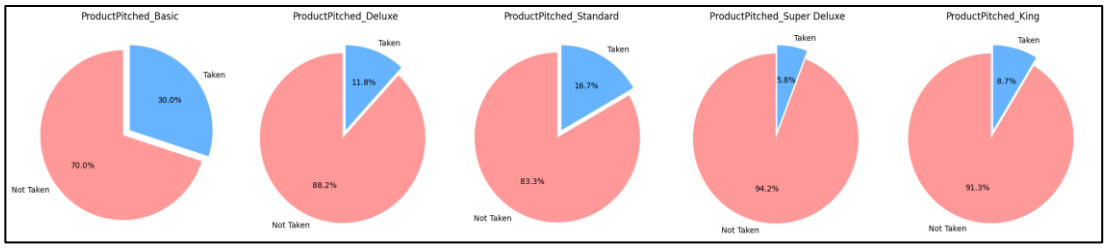
Pada feature kategori yang telah dilakukan one-hot encoding, banyak menghasilkan kolom redundan dan hal ini disebut Dummy Variable Trap, dikarenakan memasukan semua hasil one-hot encoding, untuk menghindarinya dapat di drop salah satu kolom hasil one-hot encoding

4. Business Insight









- Product mana yang diminati oleh pelanggan?

  Dari diagram batang dapat dilihat bahwa 60% paket yang diambil tawarannya adalah pake Basic, kemudian Deluxe, Standard, serta Super Deluxe dan King. Jika ditelaah lebih dalam pada setiap paketnya, tetap lebih banyak paket Basic yang diambil setelah ditawarkan, kemudian paket standar, paket deluxe, paket king, dan terakhir adalah paket super deluxe.Dalam presentasi ini, harga paket bukanlah masalah utama seseorang dalam mengambil paket setelah ditawarkan asalkan sales dapat dengan tepat memilih segmen pelanggan yang disesuaikan untuk di-pitch.

- Apakah Customer dari tiap City Tier yang berbeda memiliki ketertarikan dalam membeli paket perjalanan?

  Jumlah follow-up yang lebih tinggi oleh sales berkorelasi positif dengan peningkatan pembelian paket perjalanan. Persentase pembelian naik dari 11.36% pada satu follow-up menjadi 39.71% pada enam follow-up, menunjukkan bahwa

semakin banyak tindak lanjut, semakin tinggi kemungkinan pelanggan membeli. Strategi follow-up yang intens terbukti efektif meningkatkan konversi pelanggan.

- Apakah Customers yang memiliki passport lebih tertarik mengambil paket perjalanan?

  Tingkat konversi pelanggan lebih tinggi di CityTier 3 (23.60%) dan CityTier 2 (23.23%) dibandingkan dengan CityTier 1 (16.30%), meskipun jumlah pelanggan di CityTier 1 lebih besar. Ini menunjukkan peluang menarik untuk strategi pemasaran di CityTier 2 dan 3. Selain itu, pelanggan dengan paspor memiliki tingkat konversi lebih tinggi (34.74%) dibandingkan yang tidak memiliki (12.29%), menunjukkan adanya potensi pengaruh kepemilikan paspor terhadap keputusan pembelian, meskipun analisis lebih lanjut diperlukan.

- Segmentasi Umur dan Income apa yang mendominasi di setiap paket perjalanan?

  Dari diagram batang, terlihat bahwa setiap paket memiliki segmen pasar berdasarkan income. Paket Basic didominasi oleh pelanggan berpenghasilan rendah, Deluxe oleh pelanggan berpenghasilan menengah, dan paket Standard, Super Deluxe, serta King oleh pelanggan berpenghasilan tinggi. Paket Wellness Tourism yang lebih mahal dari King dapat dipasarkan dengan segmentasi pelanggan berpenghasilan tinggi. Dari segi umur, paket Basic didominasi oleh anak muda dan orang tua, Deluxe dan Super Deluxe oleh pelanggan usia menengah, serta Standard dan King oleh orang tua. Oleh karena itu, paket Wellness Tourism cocok dipasarkan ke orang tua berusia 50 tahun ke atas dengan high income.

STAGE 3
MACHINE LEARNING MODELING & EVALUATION

1. Split Data Train & Test

Pada tahap ini, data dibagi menjadi empat bagian, yaitu X_train, X_test, y_train, dan y_test. Pembagian data dilakukan dengan proporsi 80% untuk data train dan 20% untuk data test dan menghasilkan 2884 baris untuk data train dan 721 untuk data test

2. Modeling

Pada tahap ini, data yang telah dibagi (X_train, X_test, y_train, dan y_test) digunakan untuk melatih beberapa model. Setiap model dilatih menggunakan data X_train dan y_train, kemudian dievaluasi pada X_test dengan metrik seperti Precision untuk memprediksi pelanggan yang tidak akan melakukan pembelian, Recall untuk memprediksi pelanggan yang akan melakukan pembelian, serta F1-score untuk mempertimbangkan keduanya, tetapi disini lebih memfokuskan pada matriks Recall, dikarenakan kita memiliki tujuan agar model dapat memprediksi customer yang benar benar berpotensi atau benar benar akan membeli package yang ditawarkan

Disini kami mecoba beberapa algoritma klasifikasi yang ada diataranya Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, AdaBoost, dan XGBoost. Dari percobaan pada algoritma tersebut menghasilkan:

	Model	Accuracy (train)	Precision (train)	Recall (train)	F1 Score (train)	ROC AUC (train)	Accuracy (Test)	Precision (Test)	Recall (Test)	F1 Score (Test)	ROC AUC (Test)
0	Logistic Regression	0.844313	0.686957	0.295327	0.413072	0.632338	0.829404	0.671642	0.308219	0.422535	0.634979
1	Decision Tree	1.000000	1.000000	1.000000	1.000000	1.000000	0.879334	0.718519	0.664384	0.690391	0.799148
2	Random Forest	1.000000	1.000000	1.000000	1.000000	1.000000	0.890430	0.860215	0.547945	0.669456	0.762668
3	Gradient Boosting	0.881068	0.850365	0.435514	0.576020	0.709030	0.855756	0.762500	0.417808	0.539823	0.692382
4	AdaBoost	0.841193	0.657143	0.300935	0.412821	0.632587	0.822469	0.650000	0.267123	0.378641	0.615301
5	XGBoost	1.000000	1.000000	1.000000	1.000000	1.000000	0.919556	0.872881	0.705479	0.780303	0.839696

Dapat dilihat dari gambar di atas bahwa banyak model yang mengalami overfitting, sementara beberapa model, seperti Logistic Regression dan AdaBoost, tidak menunjukkan gejala tersebut. Namun, setelah melakukan hyperparameter tuning pada algoritma Logistic Regression, model tersebut malah mengalami overfitting. Sebaliknya, algoritma AdaBoost menghasilkan skor yang sangat baik dalam hal precision.

3. Hyperparameter Tuning

```python
from sklearn.ensemble import AdaBoostClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import RandomizedSearchCV
from sklearn.metrics import precision_score, make_scorer, f1_score
from scipy.stats import uniform, randint

# Base estimator (Decision Tree)
base_estimator = DecisionTreeClassifier(random_state=42)

# Model AdaBoost dengan base estimator
best_model = AdaBoostClassifier(estimator=base_estimator, random_state=42)

# Parameter distributions untuk RandomizedSearchCV
param_dist = {
    'n_estimators': randint(50, 500),
    'learning_rate': uniform(0.01, 0.99),
    'estimator__max_depth': randint(1, 10),
    'estimator__min_samples_split': randint(2, 20),
    'estimator__min_samples_leaf': randint(1, 10),
    'algorithm': ['SAMME', 'SAMME.R']
}

# Inisialisasi RandomizedSearchCV dengan precision score
random_search = RandomizedSearchCV(
    estimator=best_model,
    param_distributions=param_dist,
    n_iter=50,
    scoring=make_scorer(f1_score),
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)

# Melakukan pencarian hyperparameter
random_search.fit(X_train_resampled, y_train_resampled)

# Menyimpan model dengan parameters terbaik
best_model = random_search.best_estimator_

# Evaluasi model pada test set
y_pred = best_model.predict(X_test)
test_precision = precision_score(y_test, y_pred)
print("\nTest Set Precision Score:", test_precision)
```

✓ 7m 7.9s

```
Fitting 5 folds for each of 50 candidates, totalling 250 fits

Test Set Precision Score: 0.8508771929824561
```

Optimisasi hyperparameter dilakukan untuk mencari kombinasi parameter yang memberikan hasil paling optimal terhadap performa model AdaBoost. Dalam kasus ini, proses ini berfokus pada beberapa parameter kunci, yaitu:

- **n_estimators**: Jumlah decision trees yang digunakan dalam AdaBoost (50-500). Lebihbanyak estimators meningkatkan akurasi tapi memperlambat komputasi.
- **learning_rate**: Mengontrol kontribusi setiap tree (0.01-0.99). Learning rate kecil butuh lebih banyak estimators, sedangkan besar mempercepat tapi berisiko overfitting.

- **estimator__max_depth**: Kedalaman maksimal decision tree (1-10). Kedalaman lebih besar menangkap lebih banyak kompleksitas tapi bisa overfitting.
- **estimator__min_samples_split**: Jumlah minimum sampel untuk membagi node tree (2-20). Nilai lebih tinggi mengurangi kompleksitas pohon.
- **estimator__min_samples_leaf**: Jumlah minimum sampel di setiap leaf node (1-10). Mencegah overfitting dengan membatasi ukuran leaf node.
- **algorithm**: Jenis boosting yang digunakan:
    - **SAMME**: Diskrit, klasifikasi berbasis voting.
    - **SAMME.R**: Real, menggunakan probabilitas untuk prediksi, biasanya lebih akurat.

Dari hasil hyperparameter tuning menghasilkan score :

AdaBoost	Before Tunning	After Tunning
Precission	0.65	0.85

4. Cross-Validation

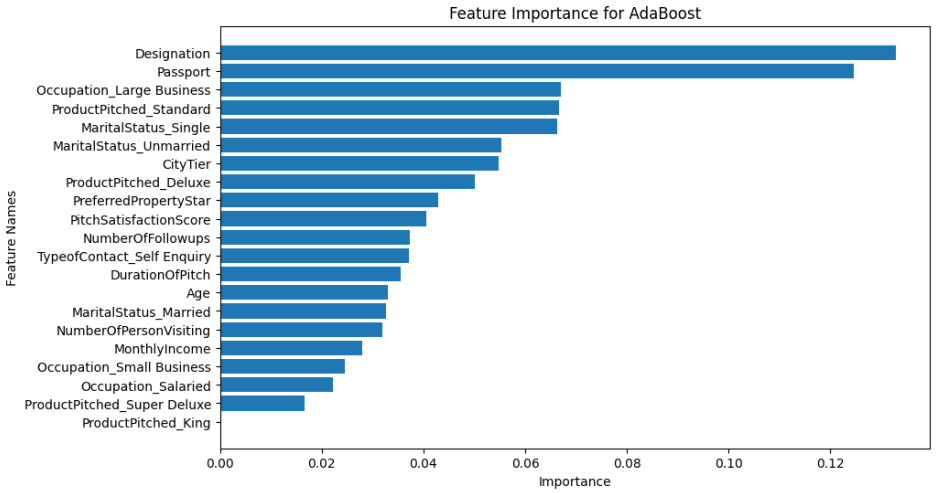
```
from sklearn.metrics import roc_auc_score
from sklearn.model_selection import cross_val_score

# Lakukan cross-validation untuk model terbaik
print(f"Evaluating {best_model_name} with Cross-Validation...")
cv_scores = cross_val_score(best_model, X_train_resampled, y_train_resampled, cv=5, scoring='precision')


print(f"precision: {cv_scores}")
print(f"mean score: {cv_scores.mean()}")
```

✓ 16.3s

```
Evaluating None with Cross-Validation...
precision: [0.89705882 0.81428571 0.88405797 0.76623377 0.7721519 ]
mean score: 0.8267576347595125
```

Berdasarkan hasil cross-validation, model AdaBoost menunjukkan performa yang stabil dengan skor yang berkisar antara 0.72 hingga 0.88. Tidak ada perbedaan signifikan antar fold, yang mengindikasikan bahwa model tidak mengalami overfitting dan mampu generalisasi dengan baik terhadap data baru. Dengan demikian, model memiliki kinerja yang cukup konsisten dan tidak terlalu sensitif terhadap variasi data.

5. Feature Importance



Business Insight :

Pelanggan yang memiliki passport, menjabat sebagai excecutive atau pengusaha skala besar, serta berstatus single/unmarried memiliki potensi tinggi untuk melakukan pembelian terhadap paket yang ditawarkan, yaitu Wellness Tourism Package.

Pelanggan yang tinggal di perkotaaan besar memiliki kemungkinan lebih besar untuk melakukan pembelian dibandingkan dengan pelanggan dari kota-kota lainnya. Faktor-faktor ini menekankan pentingnya memahami demografi dan karakteristik pelanggan untuk merancang strategi pemasaran yang lebih efektif dan menargetkan segmen pasar yang tepat.

Business Recommendation :

Berdasarkan Business Insight yang telah diuraikan sebelumnya, kita dapat merumuskan kesimpulan yang memberikan arahan untuk action plan atau rekomendasi bisnis. Rekomendasi ini bertujuan untuk meningkatkan efektivitas pemasaran dan penjualan paket yang ditawarkan, serta menciptakan pengalaman yang lebih baik bagi pelanggan. Beberapa langkah strategis yang dapat diambil meliputi

- **Fokus Pemasaran**

Fokuskan penawaran kepada customer yang memiliki paspor.

- **Mengadakan Payday Sale**

Diharapkan dapat menarik perhatian customer dengan posisi eksekutif dan meningkatkan penjualan secara signifikan pada periode gajian.

- **Special Offer Price**

Berikan harga khusus untuk pembelian dalam jumlah besar guna menarik customer yang memiliki bisnis berskala besar, sehingga mereka bisa mengajak keluarga atau para staf-nya.

- **Memberikan Perlakuan Khusus**

Memberikan perlakuan khusus kepada pelanggan single/unmarried bisa menjadi strategi efektif untuk menarik minat mereka, misalnya dengan memberikan merchandise pada hari Valentine.

- **Promosi Didaerah Perkotaan**

Tingkatkan promosi di daerah dengan pertumbuhan ekonomi yang lebih tinggi.

STAGE 4
FINAL PREPARATION

Link presentation : https://drive.google.com/file/d/1c4sG7F1VG-hy3Meh16bCdKaa0Y9pgIFJ/view?usp=sharing