# Agenda

- Data Preparation
- Conversion trend
- Model Prediction to enhance conversion rate
- Customer Segmentation by clustering

# Data Preparation

Initial Data Cleaning and Feature Readiness Process

## Profiling

shape: (215993, 26)

| | Column | Type | %Null | #Null | #Unique | MinMeanMax_or_Unique |
|---|---|---|---|---|---|---|
| 0 | campaign_month | object | 0.0 | 0 | 12 | [Mar, Jul, Jan, Nov, Oct, Sep, Apr, Jun, Dec, ... |
| 1 | marital_sta | object | 2.1 | 4635 | 8 | [โสด, สมรส, สมรสจด ทะเบียน, หย่าร้าง, ม่าย, อื... |
| 2 | main_occupation | object | 0.7 | 1500 | 10 | [Salary man, Self-employed, Freelance, Housewi... |
| 3 | customer_segment | object | 0.7 | 1500 | 3 | [Lower Mass, Mass, Upper Mass, nan] |
| 4 | gender | object | 0.7 | 1500 | 2 | [Female, Male, nan] |
| 5 | have_acc_planet | object | 1.1 | 2401 | 2 | [N, Y, nan] |
| 6 | have_cc | object | 0.7 | 1500 | 2 | [N, Y, nan] |
| 7 | scb_payroll | object | 0.7 | 1500 | 2 | [Y, N, nan] |
| 8 | num_children | float64 | 0.7 | 1500 | 9 | [0.0, 0.04038826441888547, 21.0] |
| 9 | age | float64 | 0.7 | 1500 | 39 | [23.0, 37.78451511238129, 61.0] |
| 10 | income | float64 | 1.1 | 2401 | 129477 | [0.0, 16867.35048068841, 199625.67] |
| 11 | maxosdc_last_30d | float64 | 1.1 | 2401 | 76455 | [0.0, 11354.821829843762, 4374289.15] |
| 12 | dcspend_last_30d | float64 | 1.1 | 2401 | 4335 | [-87030.88, 165.36634630510457, 1852987.91] |
| 13 | easypymt_last_30d | float64 | 1.1 | 2401 | 10545 | [0.0, 1196.3680287651223, 2000000.0] |
| 14 | savacc_bal | float64 | 1.1 | 2401 | 130874 | [0.0, 20856.298326998018, 4793828.33] |
| 15 | currentacc_bal | float64 | 1.1 | 2401 | 2 | [0.0, 0.0012172740552080603, 1.0] |
| 16 | avg_savaccbal_30d | float64 | 1.1 | 2401 | 150382 | [0.0, 15175.681839535351, 2471112.58] |
| 17 | avg_currentaccbal_30d | float64 | 1.1 | 2401 | 174 | [0.0, 30.42249236862803, 1232520.06] |
| 18 | mob | float64 | 1.1 | 2401 | 436 | [0.0, 54.56240870444586, 887.0] |
| 19 | inflow30d | float64 | 1.1 | 2401 | 74018 | [0.0, 25289.75274734142, 21471692.25] |
| 20 | outflow30d | float64 | 1.1 | 2401 | 80606 | [0.0, 25545.924888338872, 21471691.0] |
| 21 | inflow1_15 | float64 | 1.1 | 2401 | 74018 | [0.0, 25289.75274734142, 21471692.25] |
| 22 | outflow1_15 | float64 | 1.1 | 2401 | 80606 | [0.0, 25545.924888338872, 21471691.0] |
| 23 | net_flow_30d | float64 | 1.1 | 2401 | 72928 | [-1813780.11, -256.1721409977902, 2295697.16] |
| 24 | net_flow_15d | float64 | 1.1 | 2401 | 72928 | [-1813780.11, -256.1721409977902, 2295697.16] |
| 25 | label | int64 | 0.0 | 0 | 3 | [0, 0.01671350460431588, 2] |

## Check Null

[3]:
```
df[df['have_acc_planet'].isnull()]
```

[3]:

| | campaign_month | marital_sta | main_occupation | customer_segment | gender | have_acc_planet | have_cc | scb_payroll | num_children | age | income | maxosdc_last_30d | dcspend_last_30d | easypymt_last_3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 147 | Dec | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 187 | Feb | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 230 | Dec | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 250 | Feb | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 495 | Dec | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| 215733 | Mar | โสด | Self-employed | Mass | Male | NaN | N | N | 0.0 | 45.0 | NaN | NaN | NaN | Na |
| 215814 | Feb | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 215830 | Feb | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |
| 215945 | Mar | สมรส | Self-employed | Mass | Male | NaN | N | N | 0.0 | 47.0 | NaN | NaN | NaN | Na |
| 215954 | Feb | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | Na |

2401 rows × 26 columns

row ที่ null ส่วนใหญ่จะ null เกือบทุก columns และ null เพียง 1% -> drop row

[4]:
```
df['label'].value_counts(normalize=True)*100
```

[4]:
```
0    98.786072
1     0.756506
2     0.457422
Name: label, dtype: float64
```

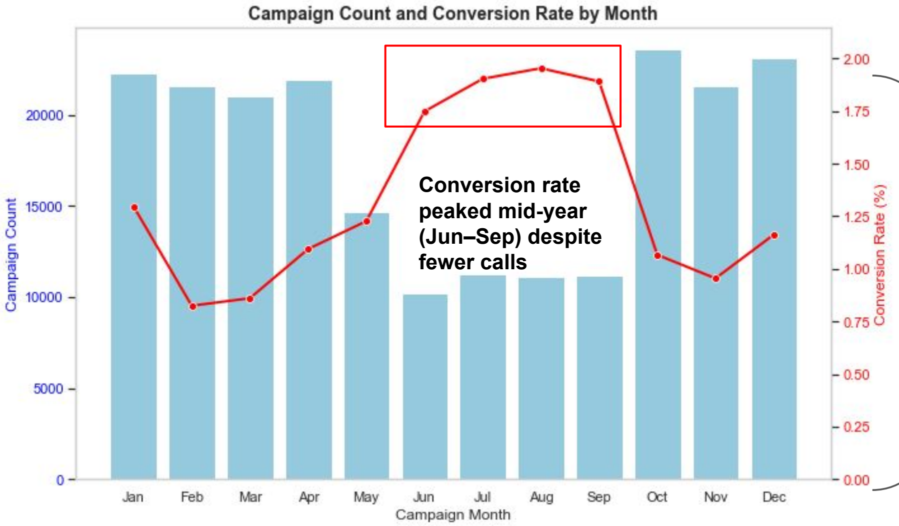
[5]:
```
df[df['have_acc_planet'].notna()]['label'].value_counts(normalize=True)*100
```

## Impute Unknown

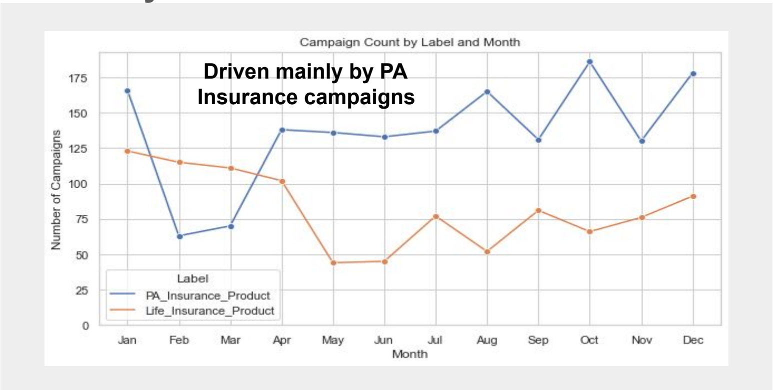
```
df_prep['marital_sta'] = df_prep['marital_sta'].fillna("unknown")
```

# Conversion trend

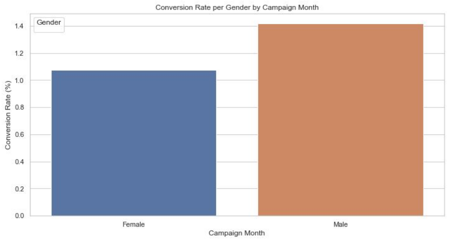
Monthly and Product-level insights on campaign performance

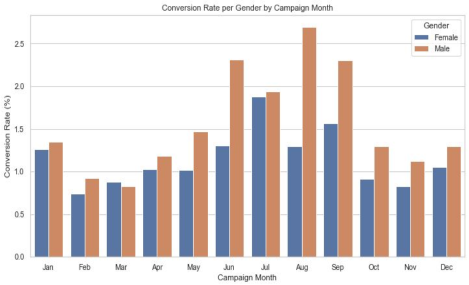



## Trend by Product




- Conversion rate สูงในช่วง Jun-Sep แม้มีการโทรที่ต่ำ ที่ 1.88 %
- Product ที่ขายดีกว่าจะเป็น PA Insurance ที่ 1.3 %
- Trend การซื้อของ Product Life Insurance ลดลงตั้งแต่ต้นปี 0.55% เป็น 0.33%



# Predictive Model

Objective: Used to identify customers with high potential to become leads

Campaign Response
0 = Reject Offer
1 = Accept Offer

Feature Engineering

- Number of Products Held: The total number of products a customer has with us, including travel cards, credit cards, and SCB Payroll.
- Spend-to-Income Ratio: The ratio of a customer's spending in the last 30 days (dcspend_last_30d) to their income.

Imbalance Data: We used XGBoost and adjusted scale_pos_weight to handle class imbalance.

```
df_model['label'].value_counts(normalize=True)*100
```

```
0    98.775235
1     1.224765
Name: label, dtype: float64
```

```
df_model['label'].value_counts()
```

```
0    210976
1      2616
Name: label, dtype: int64
```

```
# XGBoost recommends: scale_pos_weight = (num_neg / num_pos)
num_pos = sum(y_train == 1)
num_neg = sum(y_train == 0)
scale_pos_weight = num_neg / num_pos

print("scale_pos_weight:", scale_pos_weight)
```

```
scale_pos_weight: 80.64022933588151
```

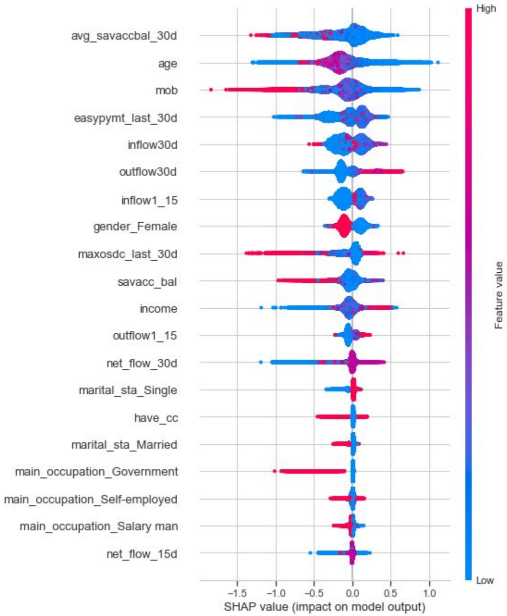
GridSearch Logging: Tracks all experiments with parameters and scores for reproducibility.

```
param_grid = {
    'max_depth': [5,6,7],
    'learning_rate': [0.1,0.3,0.5],
    'n_estimators': [100,300,500],
    'subsample': [0.7,0.8,1.0],
    'colsample_bytree': [0.7,0.8,1.0],
    'scale_pos_weight': [80,90,95]
}
```

# Predictive Model

Model Selection: Pick the model with the highest PR AUC of test set. (PR AUC = 0.0296)

Model Interpretation with SHAP Values: Explain model behavior and feature impact.



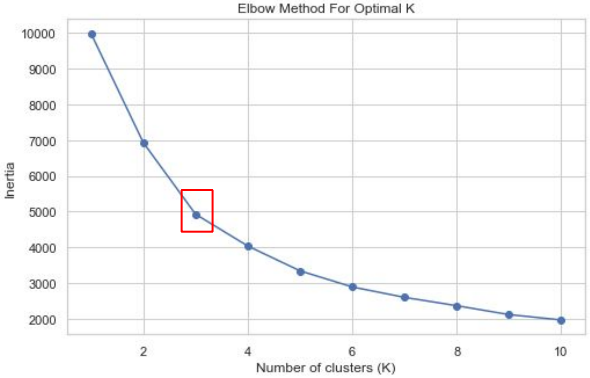

The variable avg_savaccbal_30d, which has the greatest impact on the model's prediction, shows that lower avg_savaccbal_30d values (in blue) contribute positively to the prediction, whereas higher avg_savaccbal_30d values (in red) have the opposite effect (negative impact).

Output: Probability of accepting the offer

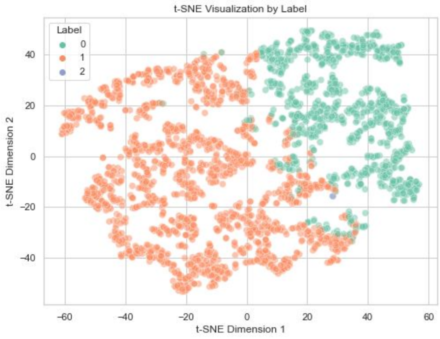
# Clustering

Objective: Customer clustering to better understand customers and offer products that match their needs.

## Elbow method to identify Group of customer




## Scale and clustering model to identify customer segment




## Median of customer groups

| Cluster | 0 | 1 | 2 |
|---|---|---|---|
| **age** | 46.00 | 29.00 | 41.50 |
| **mob** | 50.00 | 41.00 | 33.00 |
| **avg_savaccbal_30d** | 1449.03 | 1057.49 | 13762.95 |
| **easypymt_last_30d** | 561.00 | 414.00 | 1666.50 |
| **inflow30d** | 12418.80 | 8246.02 | 4577131.42 |

### Cluster 0

**Stable Savers**

Mature, mid-value customers with consistent but modest activity. They maintain small balances and modest inflows.

### Cluster 1

**Young Starters**

Young, low-value, possibly **new or less-engaged customers**. Low balance and limited transactions indicate **entry-level accounts**.

### Cluster 2


**High Rollers**

Mid-aged but very high balances and massive inflows. High-value, premium clients with high financial activity.