SIP 2021 x MTL INTERNSHIP

Data Science for Propensity to Buy

Team Member

Pasakorn Limchuchua

IT for Business Undergraduate

Thiranai Keatpimol

Insurance Undergraduate

Agenda

Introduction

- Project & Scope
- Data Background
- Data Characteristics

Data Preprocessing

1. Handle Categorical Variables
2. Handle Numerical Variables
3. Handle High Correlation Variables
4. Handle Outliers

Model Development

1. Variable Selection
2. Model Selection
3. Model Validation
4. Performance Testing

Model Implementation

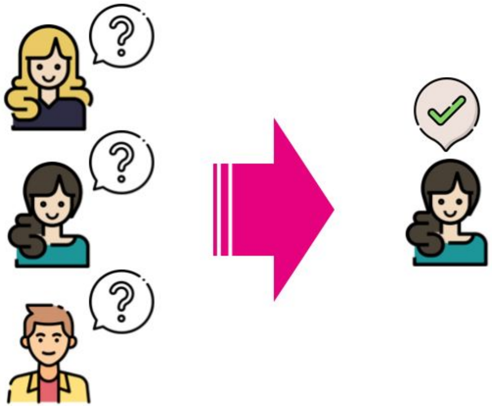
1. Interpretation
2. Implementation

Further Work

Introduction

Insurance Customer Acquisition Using Purchase Propensity Data

The use of purchase propensity data can help you target individuals who are highly likely to purchase an insurance product in the near future.



This increases response rates, lowers the total cost per lead, and improves conversion by optimizing marketing funnel driving efficiency.

Introduction

What do predictive propensity models look like?

Demographic information that tells us “who” a person is based on gender, age etc.

Transactional information that tells us “what” a person has purchased in the past as well as their estimated purchase capacity

Personality information that also tells us something about “why” a person purchases things in terms of their personality biases. This data is usually collected via surveys and can be difficult to acquire

Introduction

Given three product types from raw data

Whole Life insurance

Health insurance

Unit Linked insurance

Introduction

Whole life insurance provides permanent death benefit coverage for the life of the insured

Whole life is paid out to a beneficiary or beneficiaries upon the policyholder's death, provided that the premium payments were maintained

Whole life pays a death benefit, but also has a savings component in which cash can build up

The savings component can be invested; additionally, the policyholder can access the cash while alive, by either withdrawing or borrowing against it, when needed

Project Assignment :

To build model to predict customer propensity to buy product and find customer characteristics

Scope of our team :

We used the data as of '2019' to predict customers who are likely to **repurchase Whole Life** product in '2020' from an **Agent** channel.

Data Background

Our dataset divided into 6 sections with examples.

Target (9 variables)

Policy (52 variables)

Customer Profile (25 variables)

Sum Assured (26 variables)

Risk Indicators (5 variables)

Payment (124 variables)

Data Characteristics

Our dataset divided into 3 data characteristics (for developing model).

- Dataframe
  241 columns
  - Categorical Variables
  - Numerical Variables
    - STD Variables
    - MINMAX Variables
  - Target Variables (Buy or Not Buy)
    - Both Target
    - **Agent Target** **
    - KBank Target

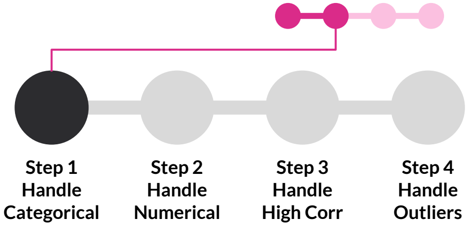
Data Preprocessing

Introduction — Data Preprocessing — Model Development — Model Implementation

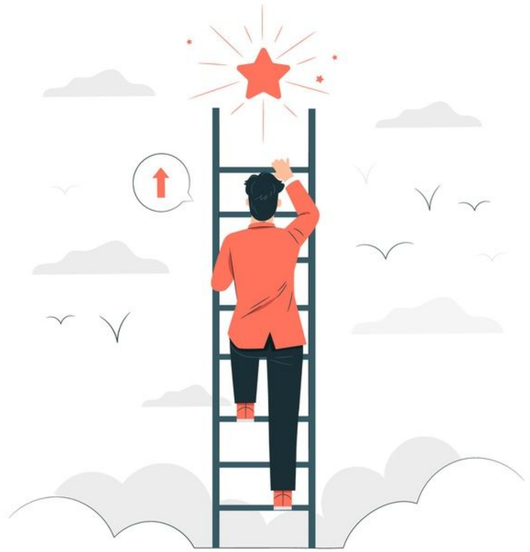
Objective : to get rid useless variable / to decrease bias / to prepare data in proper form to be used in model

- Step 1 : Handle Categorical Variables
- Step 2 : Handle Numerical Variables
- Step 3 : Handle High Correlation Variables
- Step 4 : Handle Outliers

Step 1 : Handle Categorical Variables



1. **Dropping unrelated records**
2. **Dropping variables**
3. **Filling missing records**
4. **One-Hot Encoding**
5. **Ordinal Encoding**



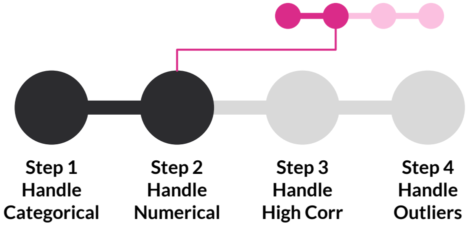
Step 1 : Handle Categorical Variables

Summary of step 1

Steps	Number of Variable Handling	Number of Record Handling
1. Dropping unrelated records	-	*0.289%*
2. Dropping variables	-2	-
3. Filling missing records	-	*87.721%*
4. One-Hot Encoding	+19	-
5. Ordinal Encoding	-	*100%*

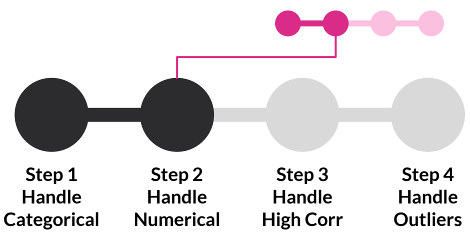
Step	Description	Variables	Var_dif	Record	Rec_dif	Target	Tgt_dif
0		241		100%		100%	
1	Handle Categorical Variables	258	+17	99.711%	-0.289%	100%	0

Step 2 : Handle Numerical Variables



1. **Filling missing records**
2. **Dropping missing records**
3. **Dropping variables**
4. **Dropping variables with condition**

Step 2 : Handle Numerical Variables

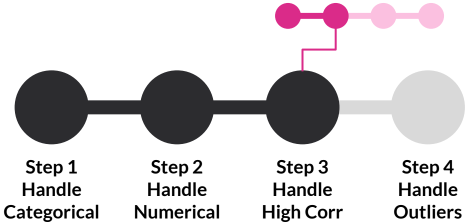


Summary of step 2

Steps	Number of Variable Handling	Number of Record Handling
1. Filling missing records	-	80.411%
2. Dropping missing records	-	0.798%
3. Dropping variables	5	-
4. Dropping variables with conditions	3	-

Step	Description	Variables	Var_dif	Record	Rec_dif	Target	Tgt_dif
0		241		100%		100%	
1	Handle Categorical Variables	258	+17	99.711%	-0.289%	100%	0
2	Handle Numerical Variables	242	-16	98.857%	-0.854%	99.858%	-0.142%

Step 3 : Handle High Correlation Variables



1. Dropping High Correlation of Numerical Variables with conditions

We used **Pearson Correlation** with threshold >= 0.8 to dropped one of pair variables (Corr Value >= 0.8).

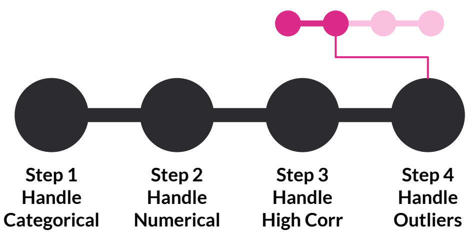
There were 70 variables which we decided to drop them from dataset.

2. Dropping High Correlation of Categorical Variables with conditions

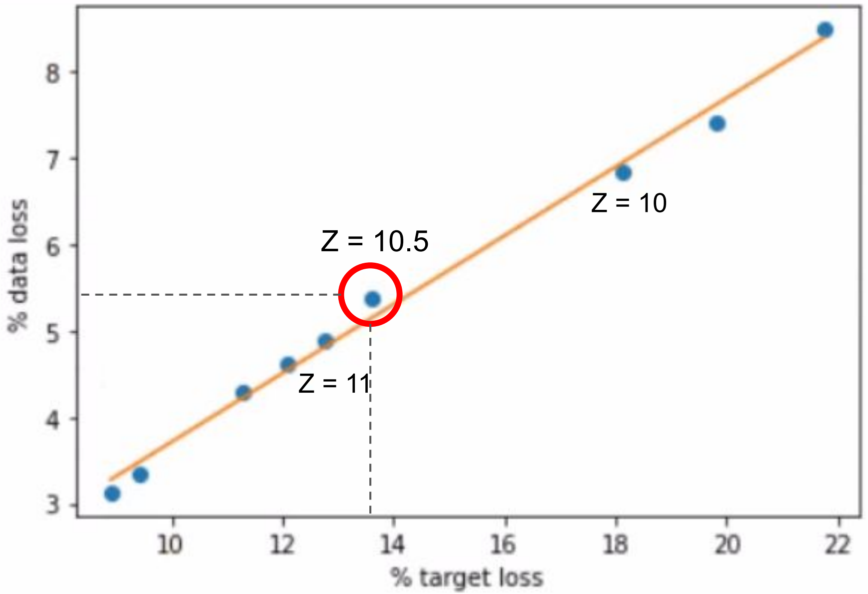
We used **Cramer's V Coefficient** with threshold >= 0.8 to dropped one of pair variables (Corr Value >= 0.8).

There were no variable dropping.

Step 4 : Handle Outliers

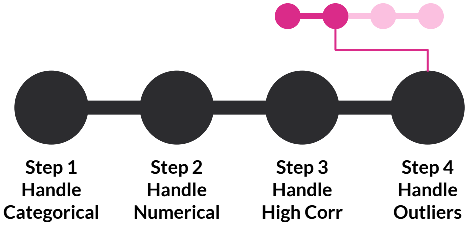


We removed data with STD greater than 10.5 standard deviation.



At 10.5 standard deviation,
Data loss : 5.4%
Target loss : 13.6%

Step 4 : Handle Outliers



10.5 standard deviation is the greatest number of data loss while the least number of target class.

Summary of Data Preprocessing

Step	Description	Variables	Var_dif	Record	Rec_dif	Target	Tgt_dif
0	-	241		100%		100%	
1	Handle Categorical Variables	258	+17	99.711%	-0.289%	100%	0
2	Handle Numerical Variables	242	-16	98.857%	-0.854%	99.858%	-0.142%
3	Handle High Correlation Variables	172	-70	98.857%	0	99.858%	0
4	Handle Outliers	172	0	93.533%	-5.324%	86.443%	-13.415%
	Total		-28%		-6.467%		-13.557%

Description	Variables	Record	Target
Before Data Preprocessing	241	100%	100%
After Data Preprocessing	172	93.533%	86.443%
% Data loss after data preprocessing process	28%	6.467%	13.557%

Model Development

Introduction

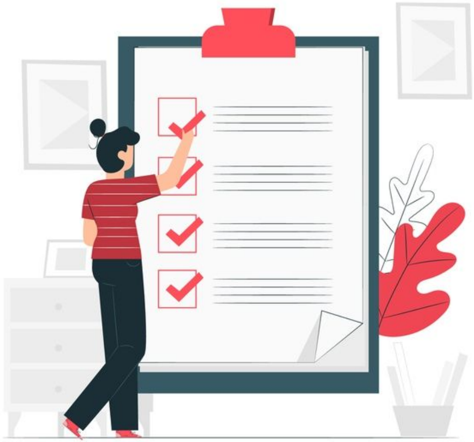
Data Preprocessing

Model Development

Model Implementation

Objective : to predict customer who have propensity to buy

- Step 1 : Variable Selection
- Step 2 : Model Selection
- Step 3 : Model Validation
- Step 4 : Performance Testing



Step 1 : Variable Selection

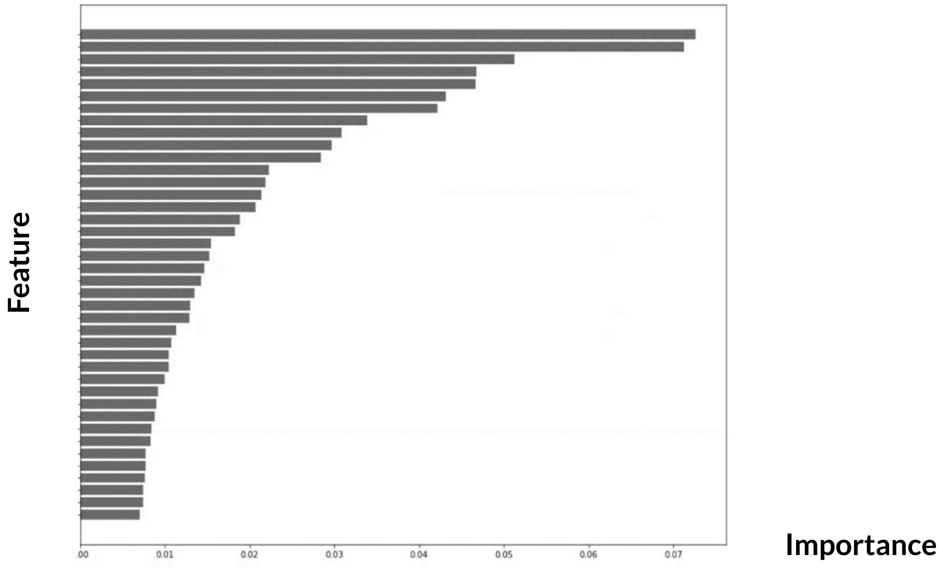
Step 1
Variable
Selection

Step 2
Model
Selection

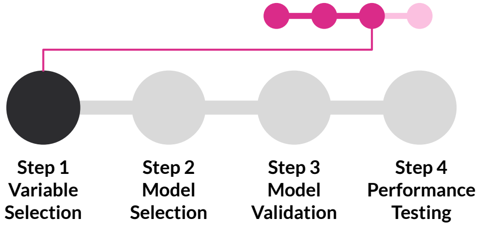
Step 3
Model
Validation

Step 4
Performance
Testing

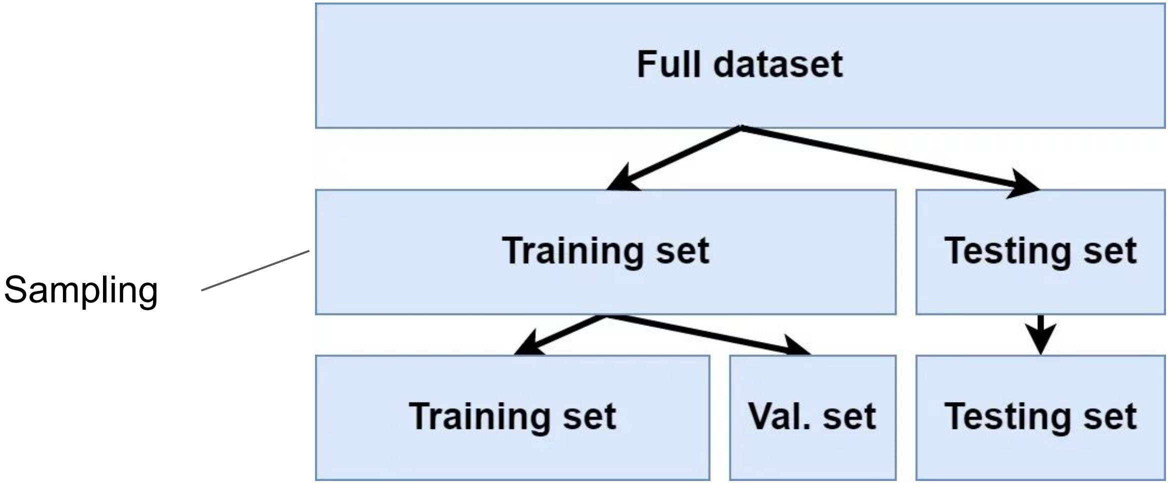
We applied RandomForestClassifier to find variables which we will use in the model.
(We chose the most **40 features** importance)



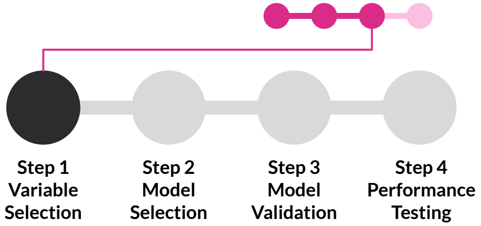
Step 1 : Variable Selection



For splitting data step, we splitted full dataset into training dataset, validation dataset and test dataset in ratio 70%, 10% and 20% respectively.

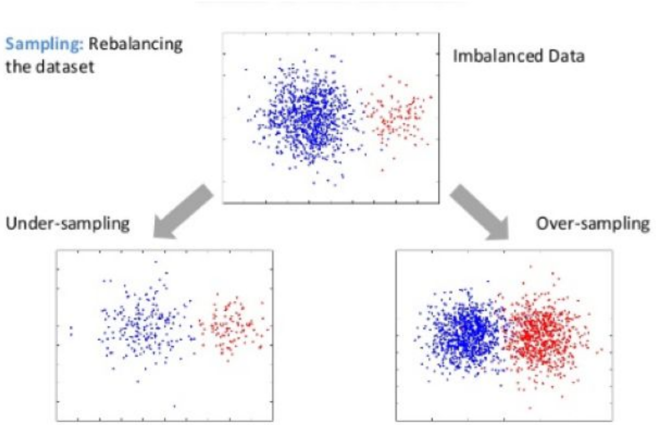


Step 1 : Variable Selection

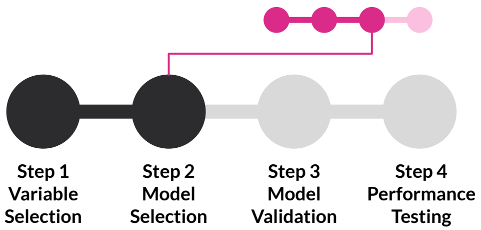


For sampling data step, we decided to use 3 methods,

1. Set parameter **class_weight = 'balanced'** in scikit-learn models.
2. Used imblearn to **oversampling** and **undersampling** data for training set.



Step 2 : Model Selection



Predictive models for binary classification

1. Logistic Regression
2. Random Forest Classifier

Why both?

Our problem is “customer : buy or not buy” which is binary value (0 = not buy and 1 = buy). So, we choose predictive models for binary classification.

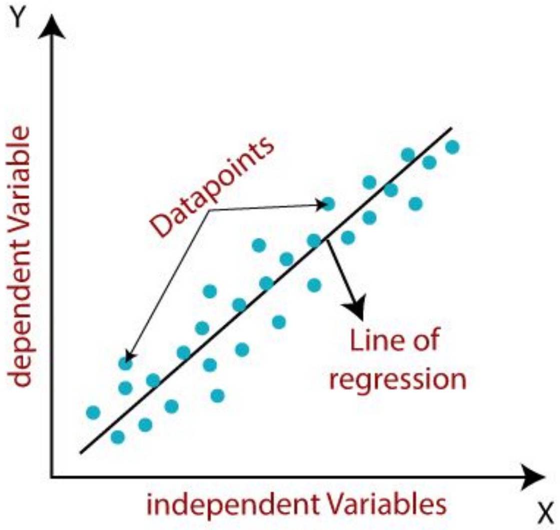
Step 2 : Model Selection

Step 1 Variable Selection | Step 2 Model Selection | Step 3 Model Validation | Step 4 Performance Testing

Linear Regression

Linear Regression is a model that makes a probability of particular prediction by taking the weighted average of the input features of an observation or data point.

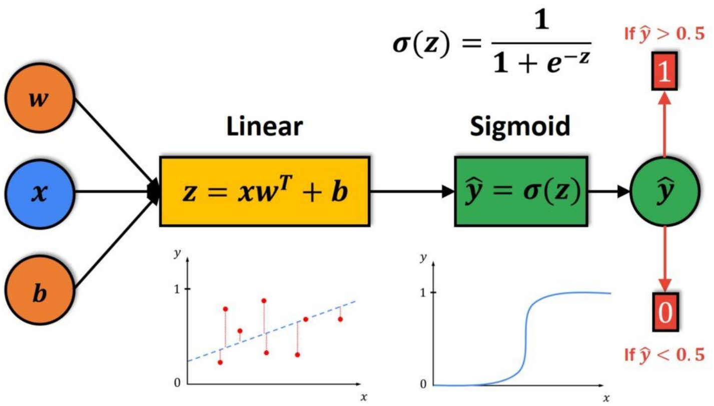
It has a certain fixed number of parameters that depend on the number of input features, and they **output a numeric** prediction.



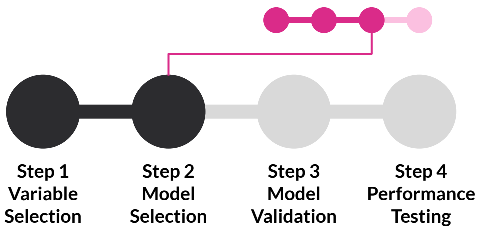
1. Logistic Regression

Logistic Regression is very similar to linear regression.

It has a certain fixed number of parameters that depend on the number of input features, and they **output categorical prediction**.



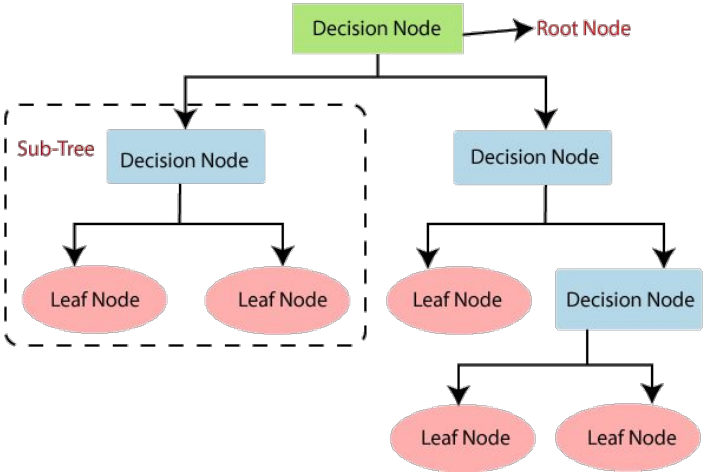
Step 2 : Model Selection



Decision Tree

Decision Tree is an algorithmic approach that identifies ways to split a data set based on different conditions.

It predict the class or value of the target variable by learning simple decision rules inferred from prior data (training data)



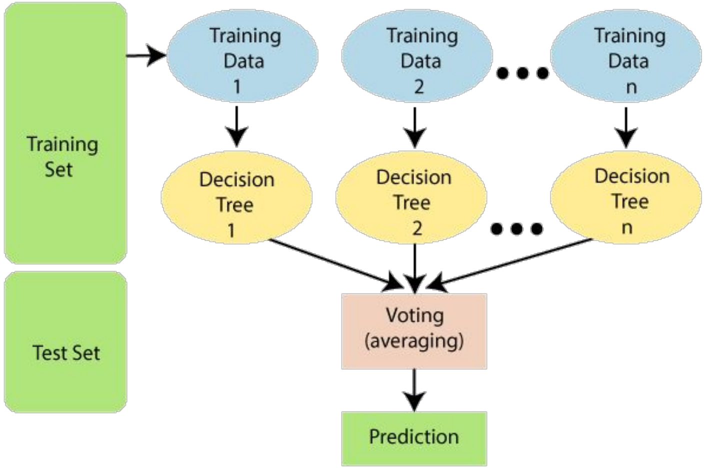
Step 2 : Model Selection

Step 1 Variable Selection | Step 2 Model Selection | Step 3 Model Validation | Step 4 Performance Testing

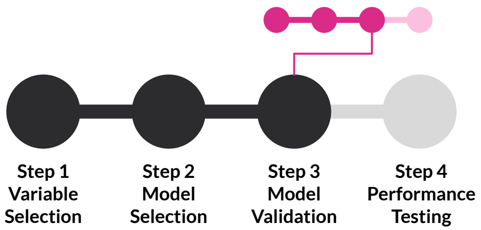
2. Random Forest Classifier

Random Forest Classifier use multiple individual decision trees to obtain better performance.

Each individual tree spits out a class prediction and the class with **majority vote** becomes the model's prediction.



Step 3 : Model Validation



F Score is calculated from Precision and Recall.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

We focus on "From the customers that we list, how many customers who buy the product"

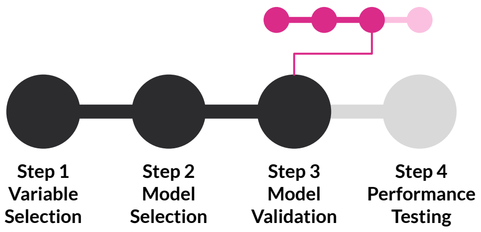
$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

We focus on " % of predicted customers who tend to buy were correctly captured by model comparing to actual buyer"

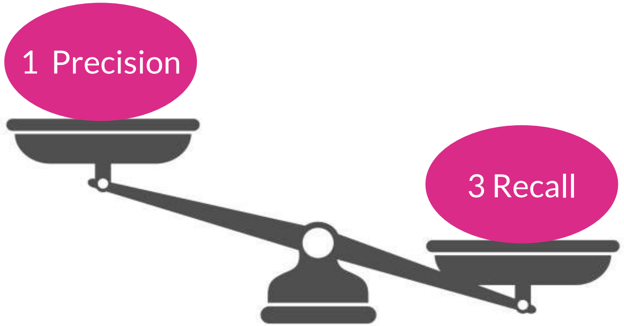
	Actual	Predict
True Positive	✓	✓
True Negative	✗	✗
False Positive	✗	✓
False Negative	✓	✗

Positive = Buy
Negative = Not Buy

Step 3 : Model Validation



Metrics to be considered : **F3 Score**



F3 Score is calculated from Precision and Recall by putting more weight on Recall.

Why F3 Score?

We want the model to find the group of customers who buy the product from all customers that we have. However, the agent must contact them carefully because we don't want to annoy customers.

Step 4 : Performance Testing

Step 1 Variable Selection
Step 2 Model Selection
Step 3 Model Validation
Step 4 Performance Testing

We created a **baseline model** to compare the performance of the generated model and used **most-frequent strategy** (always predicts the most frequent label in the training set (ignores the input data))

Baseline Model (sklearn dummy classifier)

Strategy	Accuracy	Balanced Accuracy	Recall	Precision	F3
most_frequent	0.9911	0.5	0	0	0

Step 4 : Performance Testing

Step 1 Variable Selection | Step 2 Model Selection | Step 3 Model Validation | Step 4 Performance Testing

Model

Logistic Regression

Random Forest

Sampling Data

Under Sampling

Over Sampling

Class Weight

Parameter

Default

Tuning

12 Models

Step 4 : Performance Testing

Step 1 Variable Selection
Step 2 Model Selection
Step 3 Model Validation
Step 4 Performance Testing

Summary of model performance

Model	Accuracy	Balanced Accuracy	Recall	Precision	F3
Baseline Model (Most Frequently)	0.9911	0.5	0	0	0
RandomForest (Default)	0.9911	0.5003	0.0006	0.1429	0.0007
Logistic (Default)	0.9911	0.5012	0.0032	0.2632	0.0036
:					
Logistic (Undersampling, Default)	0.7644	0.7266	0.6883	0.0254	0.1907
Logistic (Undersampling, Tuning)	0.7625	0.7283	0.6934	0.0254	0.1909
RandomForest (Class_weight = 'balanced', Tuning)	0.8272	0.7345	0.6402	0.0322	0.2218

Based on the performance, We decided to use **Random Forest Classifier (Tuning)** with **Class_weight balanced** as our **final model**.

Model Implementation

Objective : to interpret our final model and give recommendations

- Step 1 : Interpretation
- Step 2 : Implementation

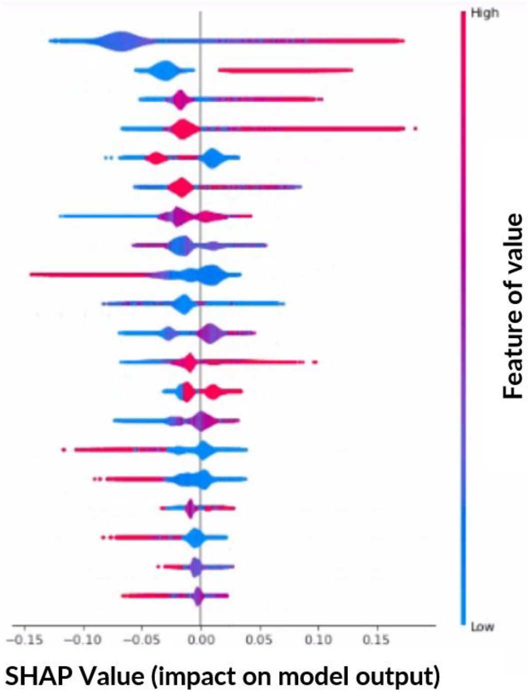
Step 1 : Interpretation

Step 1
Interpretation

Step 2
Implementation

SHAP value

SHAP value interprets the impact of having a certain value for a given feature in comparison to the prediction.

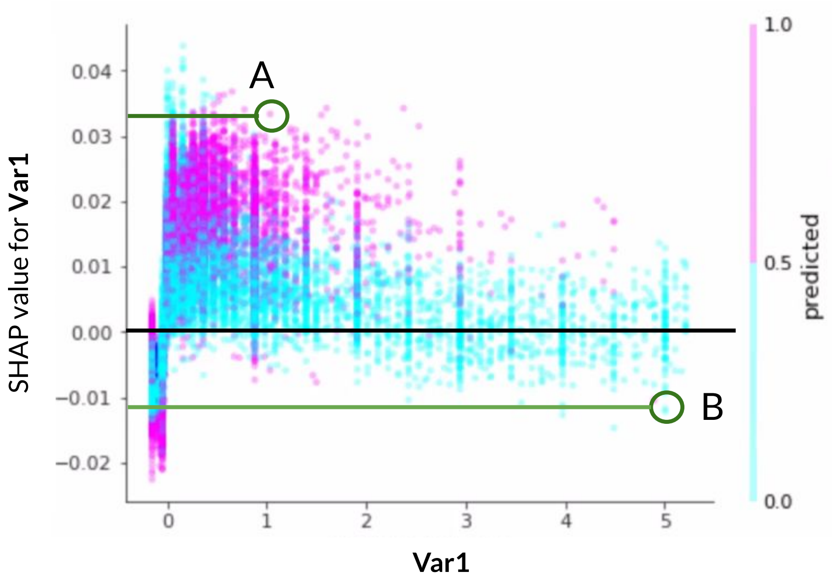


Step 1 : Interpretation

Step 1
Interpretation

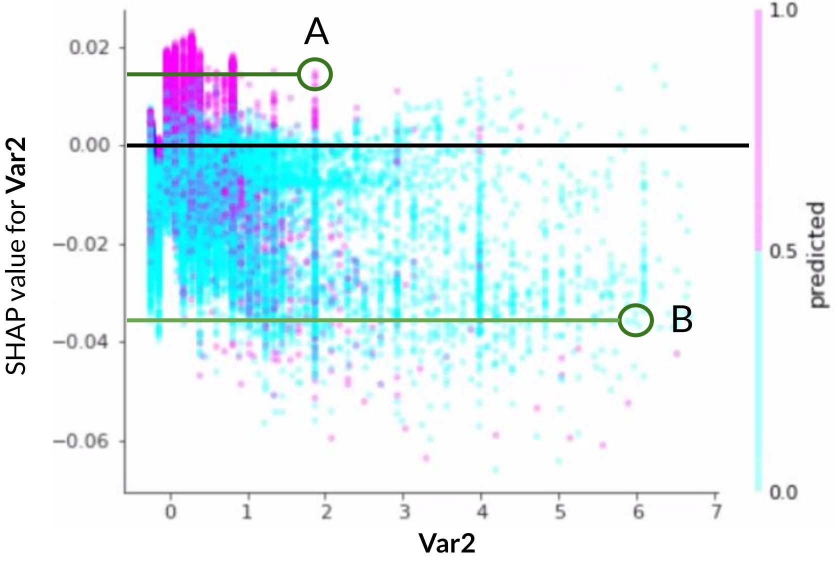
Step 2
Implementation

SHAP value



The less Var1, the more propensity to buy Whole Life Product from an agent channel.

SHAP value



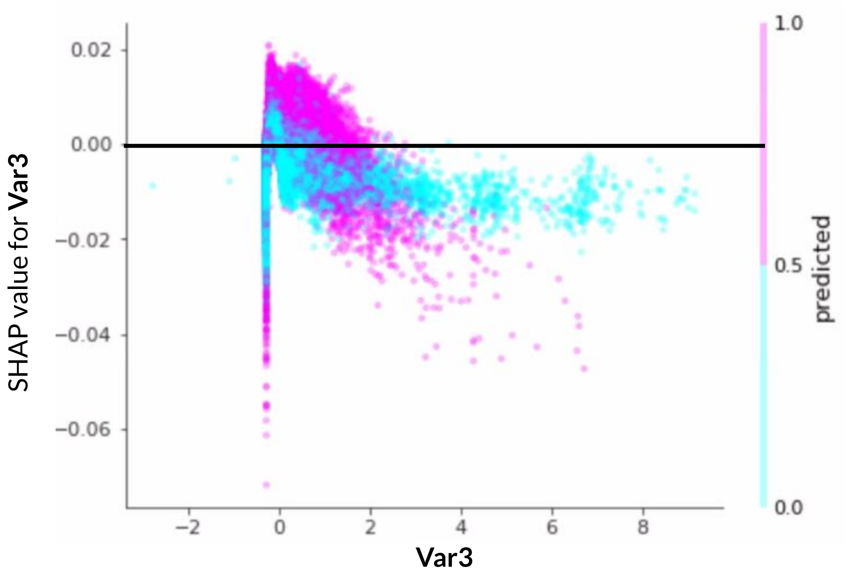
The less Var2, the more propensity to buy Whole Life Product from an agent channel.

Step 1 : Interpretation

Step 1
Interpretation

Step 2
Implementation

SHAP value



The less Var3, the more propensity to buy Whole Life Product from an agent channel.

Step 1 : Interpretation

Step 1
Interpretation

Step 2
Implementation

Summary of SHAP value

Characteristics	Low	High
Var1	Buy	Not Buy
Var2	Buy	Not Buy
Var3	Buy	Not Buy

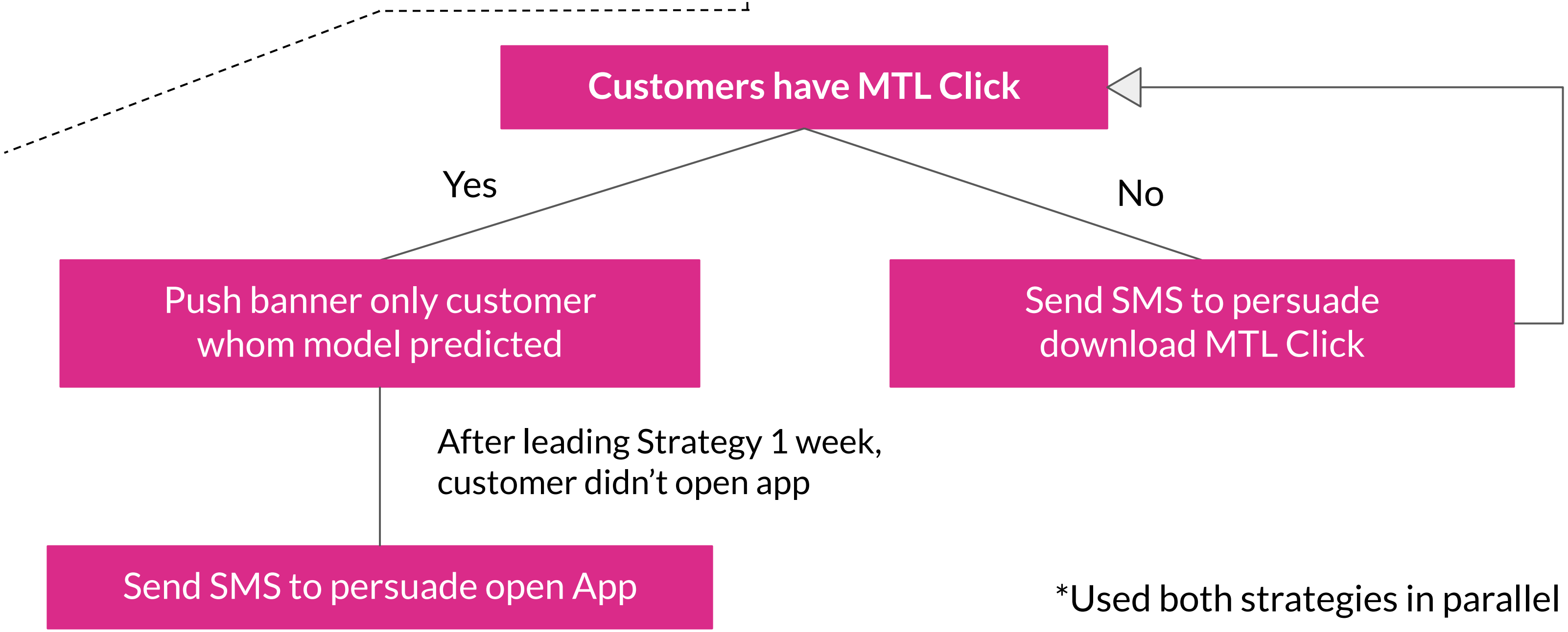
Step 2 : Implementation

Step 1
Interpretation

Step 2
Implementation

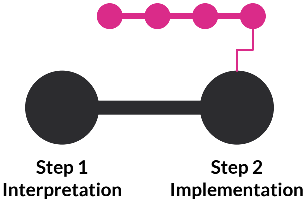
Strategy 1: Send customer contact to Agent

Strategy 2: Push Banner on MTL Click



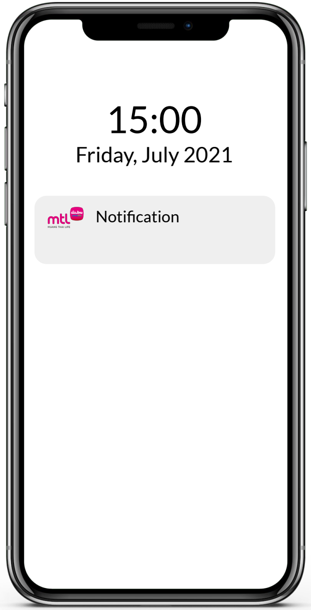
*Used both strategies in parallel

Step 2 : Implementation



Push Banner on MTL Click

Notification Pop-Up for Whole Life Product



Further Work

1. **To build model to approach new customer**
2. **Increasing customer likely to buy**

	Predicted Buy	**Predicted Not Buy**
Actual Buy	0.565% TP	FN 0.317%
Actual Not Buy	16.957% FP	TN 82.161%

The prediction was wrong because **customer's characteristics were different** from what the model sees as a high propensity to buy.

- Need to find more information on why they buy it.
- Maybe try asking the agent who takes care of this customer how they can sell him
- What reason did they buy it?

Summary

Data Exploring Raw Data : 241 columns

Data Preprocessing Filling with mode imputation / Dropping variables / Dropping missing records / Encoding / High correlation handling / Outlier Handling

Model Development Variable Selection (40 features) -> Random Forest Classifier -> F3 Score

Model Interpretation

Characteristics	Low	High
Var1	Buy	Not Buy
Var2	Buy	Not Buy
Var3	Buy	Not Buy

Model Implementation
1. Send customer contact to Agent
2. Push Banner on MTL Click

THANK YOU