Promoting Financial Products to Bank Customers From a Prescriptive Analytics Perspective

Luca-Andrei Manea (lucman@mit.edu)
Zeki Yan (zikaiyan@mit.edu)

December 2023

Contents

1 Introduction

In the competitive sector of financial services, banks must optimize marketing strategies to efficiently allocate resources and maximize profits. Rather than merely selling all products or merely predicting customer purchases, the goal is to craft a model that assigns the most lucrative marketing approach to potential customers based on their buying tendencies.

Our project delves into a retail bank's 2016 transaction data for three products: savings, mortgage, and pension accounts. Following a 2017 merger, the bank aims to deploy tailored marketing campaigns to new customers to maximize profitability.

2 Problem Statement

The problem consists of two stages. We need to make predictions about customer interest regarding the offered products and then make marketing decision based on them. The detailed demonstration are as follows:

1. **Customer interest prediction**: In this stage, models needs to be built to predict whether (or how much) each customer will be interested in each product the bank provides. The products are type of bank accounts that customers can open, namely *Savings accounts*, *Mortgage accounts*, and *Pension accounts*.
2. **Marketing decision making**: The second stage consists of taking marketing decisions based on the prediction result, that is, deciding which product the bank shoud promote (or offer) which customers through which marketing channel. The channels are typical ways to bring advertisement to customers. These include *gifts* as in promotional offerings, *newsletters* via email, and *seminars* such as financial literacy seminars to engage and educate customers.

The main decisions to be made are as follows:

1. **Which customers should we promote our products to?**
2. **Which product should be proposed to each customer?**
3. **Through which channel should we promote the product to the customer?**

To solve the problem, we used the historical data of the customers prior to the merger, where customer behavior and additional demographic data is recorded. This was used to build models and making decisions. Finally, the models' performances were evaluated, selecting the highest expected profit achieving model to make final decisions for the future marketing decisions.

3 Dataset Overview

There are two datasets used for the purpose of this project. The known customer dataset contains both basic customer information and customer behavior, which is their account opening history for different type of bank products, while the unknown data only has basic customer information of the newly acquired customers. The source of these two datasets are the following:

1. **Old Customers Dataset**: `https://raw.githubusercontent.com/vberaudi/utwt/master/known_behaviors.csv` This dataset includes all the customers in 2016, with their financial data and whether they bought savings, mortgage, and pension accounts.
2. **Newly Acquired Customers Dataset**: `https://raw.githubusercontent.com/vberaudi/utwt/master/unknown_behaviors.csv` This dataset includes the new customers in 2017 after the merger with the same variables except for which accounts they bought.

The structure and explanation of the features of the datasets are shown in Table 1. As for the size of the two datasets, there are **11,023** customer records and **23** columns in the old customers dataset and **2,756** records and **20** columns in the newly acquired customers dataset. Furthermore, we are given the following additional information: The unit profit per product is $200 for savings accounts, $300 for mortgage accounts, and $400 for pensions accounts. The bank has a budget of $25,000 for a marketing campaign targeting the new customers. The unit costs and success probability in terms of product purchase are $20 and 0.2 for gifts, $15 and 0.05 for newsletters, and $23 and 0.3 for seminars, respectively.

4 Methodology

Our project used both predictive and prescriptive approaches that given the observational data (customer personal and financial data), output the product to be sold to a customer (interest) that results in the highest profit based on the marketing campaign (decision). The following four methods are implemented independently to make decisions:

Table 1: Dataset Structure and Explanation of Features

Column Name	Explanation
customer_id	id of the customer
age	age of customer
age_youngest_child	age of the customer's youngest child
debt_equity	debt to equity ratio of customer ($\times$ 100)
gender	binary variable, male-0 ,female-1
bad_payment	binary variable, whether customer has bad payment history
gold_card	binary variable, whether customer is a gold card user
pension_plan	binary variable, whether customer has a pension plan
household_debt_to_equity_ratio	household debt to equity ratio ($\times$ 100)
income	annual income of the customer ($)
members_in_household	number of members in the customer's household
months_current_account	number of months passed since the customer opened their account
months_customer	number of months passed since customer registered with the bank
call_center_contacts	amount of time spent in contact with call center
loan_accounts	number of loan accounts the customer has
number_products	number of products (except for those to be promoted) bought
number_transactions	number of transactions the customer has made
non_worker_percentage	percentage of non-workers in the customer's household
white_collar_percentage	percentage of white-collar workers in the customer's household
rfm_score	Recency, Frequency, and Monetary (RFM) score of the customer
****Mortgage***	whether the customer has opened Mortgage account (only in old customer dataset)
****Pension***	whether the customer has opened Pension account (only in old customer dataset)
****Savings***	whether the customer has opened Savings account (only in old customer dataset)

1. **Greedy Algorithm**: Served as baseline model. The algorithm goes through all the customers, products, and channels, making assignments to those demands (in terms of product interest) that are not satisfied yet and ends once no more offers can be made.
2. **Sample Average Approximation**: Utilizes only training set data to make decisions. This methods accounts for uncertainty but not for auxiliary data.
3. **Point Prediction**: Making predictions of customer's interests utilizing the training set data, then making decisions for customers in the test set using the information we have about them. This method accounts for auxiliary data but not for uncertainty.
4. **Predictive Prescription**: Making predictions and decisions all at once. This method accounts for both auxiliary data and uncertainty.

For each method, we also chose corresponding machine learning models to make predictions. The models that we have used include the following:

1. **Random Forest**: Combines the output of multiple decision trees to reach a single result. Can be used for the Greedy Algorithm, Point Prediction and Predictive Prescription method.
2. **Optimal Classification Tree**[1]: A classification tree that can conduct and can be solved under optimization methods to gain the best structure. Can be used for the Greedy Algorithm and Point Prediction method.
3. **Logistics Regression**: Performs classification. Can be used for the Greedy Algorithm and Point Prediction method.
4. **K-Nearest Neighbor**: Find K nearest neighbours in the known dataset of the new incoming data points. Can be used for the Greedy Algorithm, Point Prediction and Predictive Prescription method.
5. **Kernel Methods**[2]: Local Kernel Methods, similar to k-NN; another way to assign weights which also accounts uncertainty. Can be used for the Greedy Algorithm, Point Prediction and Predictive Prescription method.

We test the performance of all the methods and models based on the test set data that we split from the **Newly Acquired Customers Dataset**, and calculate metrics based on the old customers' behaviors to assess model performance. These metrics include *revenue*, *budget used*, *profit*, *number of customers served*, *how many times each product is offered*, and *how many times each marketing channel is used.*

5 Notation

5.1 General Notation

- i: the i-th customer
- j: the j-th product
- k: the k-th marketing channel
- f: the f-th feature

5.2 Input Data

- I: total number of customers. I_0 for training set and I_1 for test set
- J: total number of products
- K: total number of marketing channels
- F: total number of input features
- X_{if}: the f-th feature of the i-th customer
- Y_{ij}: whether customer j is interested in product j
- p_j: the unit profit of selling product j
- s_k: the success probability of marketing channel k
- c_k: the unit cost of marketing channel k
- B: the marketing budget available

5.3 Interest

- Y_{ij}: whether customer i is interested in product j

$$Y_{ij} = \begin{cases} 1, & \text{if customer } i \text{ is interested in product } j \\ 0, & \text{otherwise} \end{cases}$$

5.4 Decision

- z_{ijk}: whether we would promote product j to customer i through channel k

$$Z_{ijk} = \begin{cases} 1, & \text{if product } j \text{ is to be promoted to customer } i \text{ through marketing channel } k \\ 0, & \text{otherwise} \end{cases}$$

6 Modeling

6.1 Prediction Models

Since all the machine models models employed are widely known and used, we only elaborate on Kernel Methods[2], which we referenced from the paper ***From Predictive to Prescriptive Analytics***.

We can estimate $m(x) = \mathbb{E}[Y|X = x]$ by the following equation:

$$\hat{m}_{I_0}(x) = \frac{\sum_{i'=1}^{I_0} y^{i'} K((x^{i'} - x_i)/h_{I_0})}{\sum_{i'=1}^{I_0} K((x^{i'} - x_i)/h_{I_0})} \tag{1}$$

where $K : \mathbb{R}^d \to \mathbb{R}$ is the kernel, $h_{I_0} > 0$ is the bandwidth, i' represents a customer from the training dataset, and $_i$ represents a customer from the test dataset. Here we use the Naïve Kernel, which is defined as followed:

$$K(x) = \mathbb{I}[\|x\| \leq 1] \tag{2}$$

6.2 Optimization Model for Decision-Making Methods

6.2.1 Optimization Objective

Generally, we want to solve the following problem:

$$\nu^*(x) = \min_{z \in Z} \mathbb{E}[c(z; Y)|X = x], \quad z^*(x) \in Z^*(x) = \arg\min_{z \in Z} \mathbb{E}[c(z; Y)|X = x] \tag{3}$$

To be more specific, the objective function of the optimization problem we want to solve consists of the following cost function and revenue function. Combining these two we get the loss function that we want to minimize:

- Cost function

$$C = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} c_k \cdot y_{ij} \cdot z_{ijk} \tag{4}$$

- Revenue function

$$R = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} p_j \cdot s_k \cdot y_{ij} \cdot z_{ijk} \tag{5}$$

- General Objective function / Loss function (the minus of profit function)

$$c(z;Y) = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} (c_k - p_j \cdot s_k) \cdot y_{ij} \cdot z_{ijk} \tag{6}$$

For the methods (except for the Greedy method) that we choose, we outline the specific solutions regarding the decision variables:

- Sample Average Approximation

$$\hat{z}^{SAA} \in \arg\min_{z \in Z} \frac{1}{I_0} \sum_{i=1}^{I_0} \sum_{j=1}^{J} \sum_{k=1}^{K} (c_k - p_j \cdot s_k) \cdot y_{ij} \cdot z_{jk} \tag{7}$$

  where z_{jk} is used instead of z_{ijk} so that SAA prescribes the same prescription to all the customers

- Point-Prediction

$$\hat{z}^{point-pred} \in \arg\min_{z \in Z} \sum_{i=1}^{I_1} \sum_{j=1}^{J} \sum_{k=1}^{K} (c_k - p_j \cdot s_k) \cdot \hat{y}_{ij} \cdot z_{ijk} \tag{8}$$

  where $\hat{y}_{ij}$ is the prediction of y_{ij} by training a random forest on the data X to predict y_{ij} given X_i

- Predictive Prescription

$$\hat{z}_{I_0}(x) \in \arg\min_{z \in Z} \sum_{i'=1}^{I_0} \sum_{i=1}^{I_1} \sum_{j=1}^{J} \sum_{k=1}^{K} w_{I_0}^{i'}(x_i) \cdot (c_k - p_j \cdot s_k) \cdot y_{i'j} \cdot z_{ijk} \tag{9}$$

  where the i' is the index of the customer in the training set while the i is the index of customers in the test set, and $w_{I_0}^{i'}(x_i)$ is the weight of customer i' in the training set which is used in inferring the interest of customer i in the test set.

6.2.2 Optimization Constraints

- Limited Budget Constraint:

$$\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} c_k \cdot z_{ijk} \leq B \tag{10}$$

- Constraint to promote through each marketing channel to at least 10 percent of the customers:

$$\sum_{i=1}^{I} \sum_{j=1}^{J} z_{ijk} \geq 0.1 \cdot I, \quad \forall k \tag{11}$$

- Constraint to promote each product through only one channel:

$$\sum_{k=1}^{K} z_{ijk} \leq 1, \quad \forall i, j \tag{12}$$

- Constraint to promote at most 1 product to each customer:

$$\sum_{j=1}^{J} z_{ijk} \leq 1, \quad \forall i, k \tag{13}$$

7 Implementation

7.1 Data Cleaning and Preprocessing

We performed data cleaning and data preprocessing using the following pipeline:

1. **Drop records with missing values**: There are some records with missing values and since the amount of these records are small, we simply chose to drop them.
2. **Create train and test set**: We created train and test sets using stratified splitting based on the columns *Mortgage*, *Savings* and *Pension* to make sure that customers have similar bank account preference distributions in both datasets.
3. **Standardization**: The features were standardized to all have a mean of 0 and a variance of 1.

7.2 Feature Engineering

As the first step of feature engineering, we calculated the correlation of all the columns to get a better understanding of the data, which is shown in Figure 1.

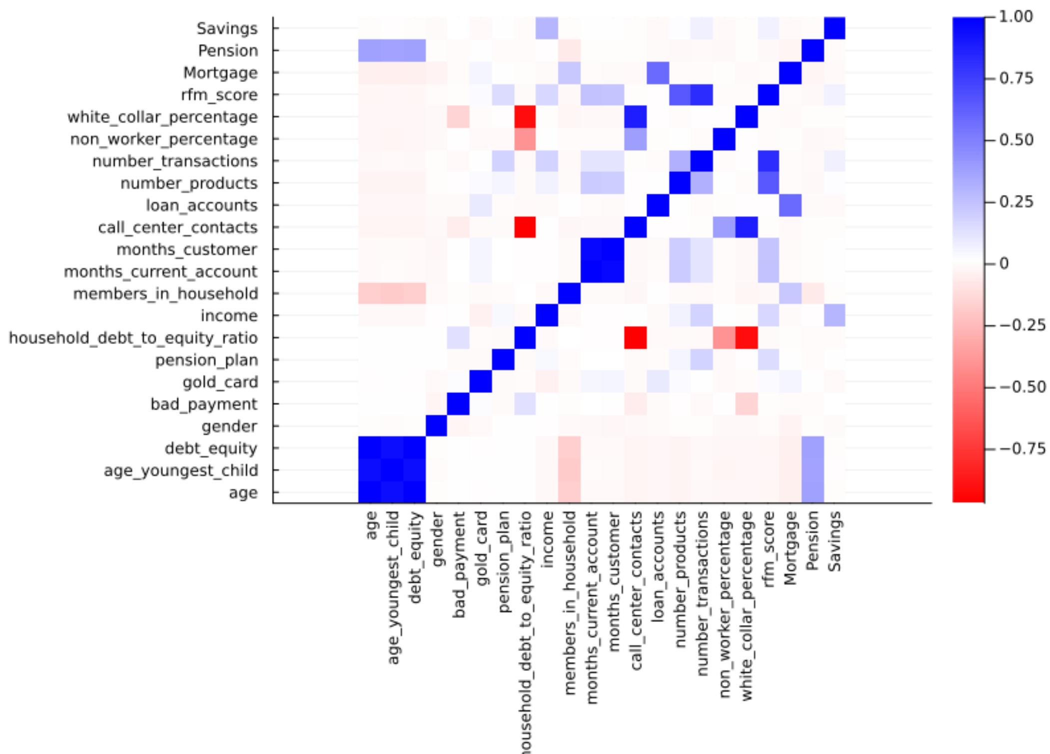

Figure 1: Heatmap for feature correlation

For savings accounts, the highest correlated feature is ***income***, with a correlation of around 0.2987. This makes sense as one would generally assume that people with higher income have more money to save and are therefore more likely to have saving accounts. For pension accounts, the highest correlated features are ***age***, ***age_youngest_child***, and ***debt_equity***, at correlation of approximately 0.3814. This is because the three features are have perfect positive correlation with each other. This is suprising since the dataset contains 8,818 different customers. Out of these three features, age seems the most relevant one because it is more general and easier to source. For mortgage accounts, the highest correlated feature is ***loan_accounts***, with a correlation of about 0.5916. ***Members_in_household*** also seems to describe a part of the Mortgage target variable. Even though the correlation is only around 0.2278, it might help to include it.

Overall, these four features seem the most important in explaining the three target varibles, whereas the other features do not seem to add much information. We therefore restrict the dataset to only contain the following features: ***age***, ***income***, ***members_in_household***, and ***loan_accounts***.

Figure 2 visualizes the relationship for each product of their most correlated features. From the scatter plots, it becomes visible that the features that have been chosen are indeed important in understanding whether a not a product was bought

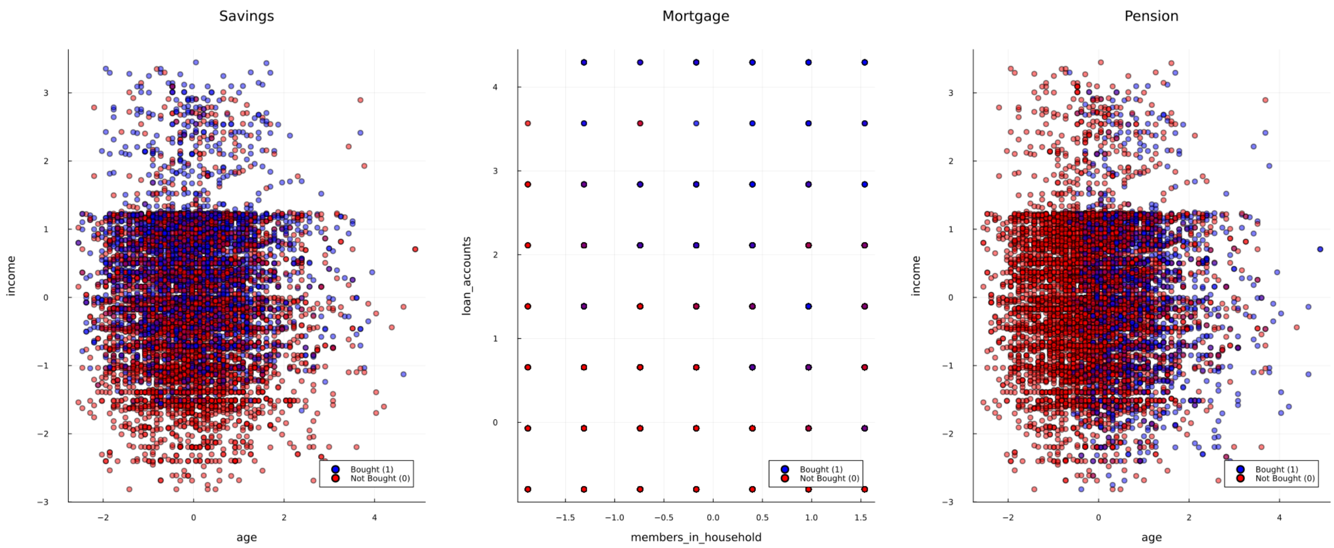

Figure 2: Scatter Plots for each product with most correlated feature

by a customer. Figure 2 incicates that customers with higher incomes tend to have savings accounts more than customers with lower income. For pension accounts, the focus lies on the age column. In the dataset, there are more older customers than younger customers that have pension accounts. Higher numbers of loan accounts and members in the household tend to jointly lead to higher likelihood of owning mortgage accounts.

8 Result and Analysis

The evaluation metrics for different methods and models are shown in Table 2, Table 3, and Table 4.

For the four methods, the ranking (based on profit) of the models' performances is as follows:

- *Prescriptive* > *Point Prediction* > *Greedy* > *Sample Average Approximation*

Note that Sample Average Method generates empty results because it is infeasible in this case since the same decision is prescribed to all customers which is not feasible under the budget constraint.

For the different prediction models, the ranking (based on profit) is the following.

- Kernel Method > k-NN > Logistics Regression > OCT > XGBoost > Random Forest

As far as the ability to utilize the budget as well serving customers is concerned, out of all the methods that we applied, only prescriptive methods utilized all the budget ($25K) to to promote products. Point Prediction used nearly all the budget and Greedy Method used the least amount of the budget. One reason is that compared to the prescriptive method, the other methods do not consider either the information in auxiliary variable or the uncertainty of decision making, which results in imperfect solutions when applying these methods. The second reason is that compared to k-NN and Kernel Method, other prediction models are not good at predicting the interest variables. In most of the cases, they predicted less customer interests than apparent in the test set. Because of this the optimization model is not able to allocate all of the budget as there would be no gain in doing so.

As for the distribution of products that these methods offer, the "smarter" prescriptive methods tend to assign the most profitable products, which is *Pension account*, first, then *Mortgage account*, and finally *Savings account*. The same thing is done for the marketing channels, ensuring that all the channels serve at least 10% customers and then making all of the remaining product promotions through the *Seminar channel*.

Figure 3 shows the profits and number of customers sold to for four selected models. It becomes apparent that the prescriptive approaches are much better at prescribing to maximize profits. Therefore, they also sell to more customers. Point prediction using logisitic regression also almost sells to as many customers as the prescriptive methods, however, with a lower profit. This is because the quality of the product interest predictions is not high enough to ensure that the predictions reflect the actual test set. This means that valuable customers are being misinterpreted.

On the bar chart the two prescriptive models seem similar. Hence, it is interesting to further investigative the decisions made by these models to better understand the actual differences to see why the profit is different. Figure 4 illustrated the

Table 2: Basic evaluation of each model

Method	Model	Revenue	Budget	Profit	Customers
Greedy	Random Forest	$7,525	$7,075	$3,025	398
Greedy	Logistics Regression	$22,660	$12,818	$15,026	663
Greedy	OCT	$21,595	$12,818	$13,952	663
Greedy	XGBoost	$20,820	$12,818	$13,291	663
SAA	Sample Average	$0	$0	$0	0
Point Prediction	Random Forest	$23,670	$16,889	$15,608	840
Point Prediction	Logistics Regression	$52,130	$24,985	$38,752	1,192
Point Prediction	OCT	$45,200	$24,985	$32,836	1,192
Point Prediction	XGBoost	$43,095	$24,295	$31,374	1,162
Prescriptive	KNN	$52,840	$25,000	$40,946	1,193
Prescriptive	Kernel Method	$59,090	$25,000	$46,327	1,193

Table 3: Products offered under each model

Method	Model	Savings	Mortgage	Pension
Greedy	Random Forest	335	10	53
Greedy	Logistics Regression	206	112	345
Greedy	OCT	289	100	274
Greedy	XGBoost	320	6	337
SAA	Sample Average	0	0	0
Point Prediction	Random Forest	436	121	283
Point Prediction	Logistics Regression	308	219	665
Point Prediction	OCT	451	210	531
Point Prediction	XGBoost	501	78	583
Prescriptive	KNN	0	65	1,128
Prescriptive	Kernel Method	0	2	1,191

Table 4: Channels used under each model

Method	Model	Gift	Newsletter	Seminar
Greedy	Random Forest	221	177	0
Greedy	Logistics Regression	221	221	221
Greedy	OCT	221	221	221
Greedy	XGBoost	221	221	221
SAA	Sample Average	0	0	0
Point Prediction	Random Forest	221	221	398
Point Prediction	Logistics Regression	221	221	750
Point Prediction	OCT	221	221	750
Point Prediction	XGBoost	221	221	720
Prescriptive	KNN	221	222	750
Prescriptive	Kernel Method	221	222	750

differences. The plots show how many customers should be contacted about which products and through which marketing channels. It is easy to see that the reason why the prescriptive kernel method approach achieves higher profit is because it can allocate more customers to the most profitable product (Pension account) through the most successful marketing channel (seminar). This comes from the fact that the weighting on the product interesting using the train dataset is better when using the kernel method in comparison with k-NN.

9 Business Impact

Our method improves the expected profit of the bank notably. Previously, the bank used the Greedy method to make marketing decisions. For the subset of customers (20% of total customers) we chose, the Greedy method generates profits of **$11,323** (average of all the corresponding models), while our best model (Prescriptive method powered by Kernel

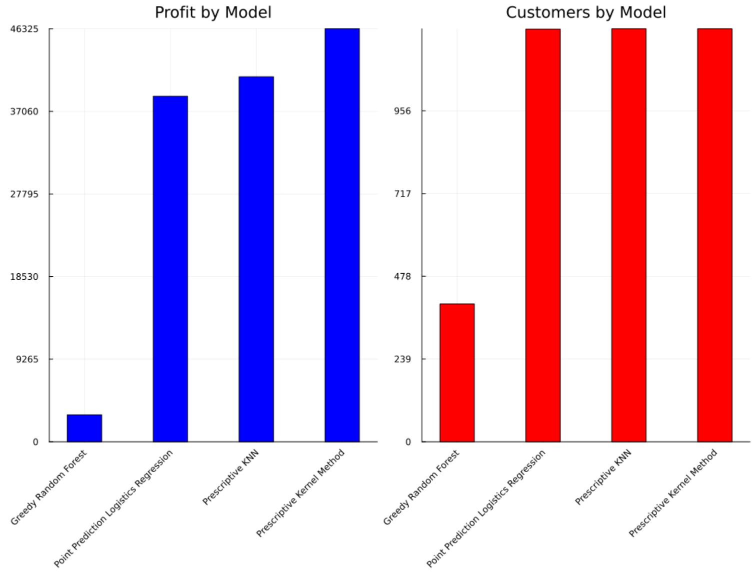

Figure 3: Comparison of different methods' profit and number of customers served

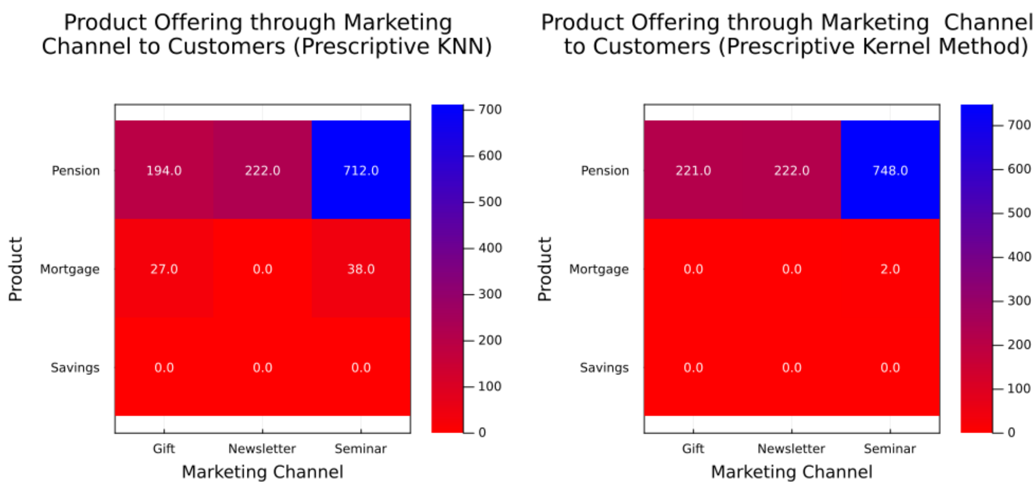

Figure 4: Comparison of different methods' product offering through marketing channel to customers

Method) can generate a profit of **$46,327**, which signifies an increase of **309%**. If our method were to be implemented completely by the bank, it could generate estimated profits of **$175K** for all 11K customers.

10 Potential Future Improvement

The cutting-edge methods we applied generated much better profit than the baseline method that the bank has been using. But some parts can still be improved:

- **Applying more advanced cutting-edged methods**: The current models we used to make predictions (assigning weights) for the Predictive Prescription methods are both local methods (k-NN and local Kernel Method), which have limitations. Global Kernel Methods[3,4] could be implemented as well.
- **Refining feature engineering**: We only conducted feature engineering based on correlations with the interest variables, which increased the model performance but still has improvement potential. More refined ways can be applied to create composite features.
- **Hyperparameter tuning**: We only conducted simple grid searches when fitting our models. These can be expand to potential improve the results.

References

[1] Bertsimas, Dimitris, and Jack Dunn. ”Optimal classification trees.” Machine Learning 106 (2017): 1039-1082.

[2] Bertsimas, Dimitris, and Nathan Kallus. ”From predictive to prescriptive analytics.” Management Science 66.3 (2020): 1025-1044.

[3] Bertsimas, Dimitris, and Nihal Koduri. ”Data-driven optimization: A reproducing kernel Hilbert space approach.” Operations Research 70.1 (2022): 454-471.

[4] Bertsimas, Dimitris, and Kimberly Villalobos Carballo. ”Multistage Stochastic Optimization via Kernels.” arXiv preprint arXiv:2303.06515 (2023).

Appendix

Contribution

For the most part of this project, we met up regularly to work together. These working sessions at the beginning consisted of mainly understanding the datasets, formulating the project, and then formulating models. Once we started coding, we worked in a sort of peer-coding way, where sometimes Zeki would code and something Luca would code and the person currently not coding would take a step back to think about the current approach and provide critical comments. Like this, we achieved everything from data processing to implementing the SAA, Point Prediction, and Prescriptive KNN models.

We then decided to split up the tasks and work individually, coming together later to unify the work. Zeki worked on implementing the Greedy model, while Luca worked on the Prescriptive kernel methods model. Zeki also worked on starting the report and making the slides, whereas Luca conducted the feature engineering and created the necessary graphs for the report. Together, we then reran all the models, finished the report and put the finishing on the slides.