BUMK 746 Final Report

Starbucks Consumer Data Analysis

By Zhanyi Zhang (Jacky)

Executive Summary

This report investigates how customer demographics, offer design, and delivery channels influence promotional effectiveness at Starbucks. Leveraging a publicly available dataset from Kaggle simulating 30 days of customer behavior, we construct predictive models to (1) determine which customers are most likely to complete a given marketing offer and (2) identify key drivers of customer transaction spending.

We began by cleaning and integrating three datasets—portfolio, profile, and transcript—to construct two modeling-ready datasets: one at the offer-event level for classification, and another at the customer level for regression. Our classification models, implemented in R, compared different ways of representing offers (using offer aliases vs. decomposed attributes) across K-Nearest Neighbors (KNN), Decision Trees, and Random Forests. We then used stepwise linear regression to model spending behavior and identify the most influential customer and offer characteristics.

Random Forests outperformed other classification models in predicting offer completion, though the difference between offer alias and offer attribute representations was minimal. Regression results revealed that customer income, offer duration, and impressions via social and mobile channels were the strongest predictors of total spend. Our findings highlight that promotional success depends more on targeting the right audience and optimizing delivery mechanics than on offer type alone.

We conclude with actionable targeting strategies based on decision tree segmentation and regression coefficients, providing Starbucks with data-driven recommendations to personalize campaigns, improve ROI, and boost customer engagement.

Introduction Background

As digital engagement becomes increasingly central to consumer-brand relationships, data-driven personalization has become more and more important in business decision-making. Starbucks, a global leader in premium coffee, has invested heavily in digital marketing campaigns to drive purchases and brand loyalty. Our analysis is based on the public Starbucks Customer Data dataset provided by Kaggle. This dataset was provided by Starbucks to simulate their customers and transactions to see if there are better approaches to sending customers specific promotional deals. The goal of our project is twofold. First, we aim to predict whether a customer will complete a given offer, using both demographic traits and offer characteristics. Second, we model customer-level spending behavior to identify the key drivers of transaction volume. By utilizing both classification and regression techniques, our work provides actionable insights for Starbucks to optimize its promotional strategies across customer segments and offer combinations. These insights would help Starbucks allocate its marketing spend more properly and maximize the ROI.

Data and Methodology

Data wrangling:

Our project aims to uncover the optimal model to predict the Starbucks customers' offer completion and identify the factors affecting their transaction spending. The analysis began with three separate datasets **(see Appendix 1)**: portfolio.csv (containing 10 unique offers), profile.csv (with 17,000 customer records), and transcript.csv (comprising nearly 300,000 user interactions).

The portfolio.csv dataset reveals details of each marketing offer, such as its type, duration, difficulty, and delivery channels. The profile dataset contains demographic attributes for each customer, including age, gender, income, and the date they joined the program . The transcript dataset tracks all user-level interactions over time, such as when offers are received, viewed, and completed, as well as general transaction activity. The definitions of variables for three dataset are shown in **Appendix 1**.

To prepare the data for modeling, we systematically cleaned and merged these datasets. All the data wrangling work was conducted in Python.

We began by removing the missing values in the gender and income columns from the profile dataset, which is the only dataset that contains NA values. Also, to ensure the became_member_on column can be effectively used for analysis, we converted it to datetime format and extracted the membership year for subsequent analysis. To simplify modeling, we assigned each unique offer ID in the portfolio dataset a corresponding offer_alias label (A–J), creating a more interpretable categorical variable.

The transcript.csv contains data which initially included nested dictionaries in the value column, was unpacked to isolate key variables (offer id, amount) and their values. To allow more of a comprehensive prediction on offer completion, we extracted the offer-level event records from transcript, and then merged it with both portfolio and profile using the person identifier. We then created a new dataset called df_offer_class **(see Appendix 2)**, where original offer events were replaced with binary indicators for whether an offer was viewed or completed.

Separately, during the regression analysis, we have constructed a customer-level dataset called df_transaction_reg **(see Appendix 2)** by summarizing each user's total transaction amount (total_amount) and combining it with offer exposure statistics. To make the regression task reliable, we included the number of offers received by type and computed the average difficulty and duration. In addition, the original channel combination variables were decomposed into individual channel usage counts.

Data analysis:

Following the data preparation, we conducted two types of predictive modeling analyses: classification and regression. All modeling was implemented in R.

For the classification task, the dataset df_offer_class was utilized to predict whether a customer would complete a given offer. The binary variable is_completed was treated as the target variable.

We began with the K-Nearest Neighbors (K-NN) method **(see Appendix 3)**, evaluating from k = 1 to k = 10. To explore how different offers affect model performance and potential optimal combination of offer attributes, we created two different sets of predictors. The first utilized a single categorical variable offer_alias, while the second decomposed offers into separate attributes—offer_type, difficulty, duration, and channels. Profile features were included in each set of predictors. Model performance was assessed using precision, recall, and F1 score on a held-out test set.

To optimize model interpretability and performance, we next moved to decision tree models **(see Appendix 4)** using the rpart package. Both unpruned and pruned trees were trained based on the two sets of predictors mentioned above. The pruned tree was selected based on 1-SE rule.

A Random Forest classifier **(see Appendix 5)** was then employed using the randomForest package, specifying 50 trees. Feature importance was computed to understand which variables contributed most to prediction accuracy. Performance was again evaluated using precision, recall, and F1 score.

For the purpose of predicting the transaction amount (total_amount) and optimizing offer attributes combination, we also conducted a regression analysis **(see Appendix 6)** based on the dataset df_transaction_reg. We utilized the forward stepwise regression to determine the optimal set of predictors. To compare the model performance between using the optimal set of predictors and using all relevant user and offer-level features, we also fit a full linear model. Model fit was assessed based on adjusted R-squared and variable significance.

Key Findings

We began our classification analysis with the K-Nearest Neighbors (K-NN) algorithm, testing values of k from 1 to 10 **(see Appendix 3)**. After tuning, the optimal k was found to be 8 when using the offer_alias setup and 7 for the decomposed offer attributes model. The latter showed slightly better performance, with an F1 score of 0.7368 compared to 0.7094 for the former.

Next, we implemented decision trees **(see Appendix 4)**. For the unpruned models, both versions produced similar results: the offer_alias model achieved an F1 score of 0.7853, while the offer attributes model was nearly identical. To improve generalizability, we applied the 1-SE pruning rule, which slightly increased F1 scores—especially for the offer_alias model, which reached 0.7964, while the offer attributes version followed closely at 0.7934.

Finally, we trained a random forest classifier with 50 trees **(see Appendix 5)**. This ensemble model outperformed both K-NN and single-tree classifiers. Among the two predictor configurations, performance remained very similar, with F1 scores of 0.7895 (offer_alias) and 0.7966 (offer attributes), suggesting both representations are comparably effective in this context.

The results show that there is no notable difference between the performance of offer_alias and offer attributes model when applying tree and random forest approaches. Tree

and random forest has significantly better performance than K-NN in predicting the offer completion.

Variable importance analysis based on random forest shows that in the offer_alias model, both offer_alias and income consistently ranked highest across accuracy and Gini impurity **(see Appendix 5)**. In contrast, in the model using decomposed offer attributes, the importance of membership_year increased notably, while duration and offer_type emerged as the most influential among the offer-related features.

For the regression component of our analysis **(see Appendix 6)**, we applied multiple linear regression with forward stepwise selection, using Mallow's Cp as the stopping rule and benchmarking the selected specification against a full 17-variable model. The algorithm identified a concise 6-variable set—income, average offer duration, membership-year indicators for 2014-2018, social-channel impressions, gender, and mobile-channel impressions—as the point of minimum Cp, driving the statistic down from 2572 to 2.36 and trimming residual sum of squares by roughly 15 percent.

The resulting reduced model delivered an R-squared of 0.1907 and an adjusted R-squared of 0.1901 with a residual standard error of 117.2, confirming that customer affluence, tenure, and digital reach are the most salient drivers of incremental spend.

Income and average offer duration show large, highly significant positive effects; every mobile push adds about $7.5 in expected revenue, and each social-media exposure contributes roughly $8.0. Customers acquired in 2015-2017 spend $18-61 more than the 2013 baseline, whereas the 2018 cohort lags by about $27. Male customers spend nearly $19 less than their female counterparts, while the "other" category shows no meaningful difference.

A full model containing seventeen predictors did not improve explanatory power—overall R-squared remained at 0.1907, but the adjusted metric slipped to 0.1898. Offer-type flags, email exposure, and web impressions were statistically inert, and the email coefficient was dropped for perfect collinearity, underscoring their negligible incremental value.

Conclusions and Recommendations

Based on the pruned decision tree trained on offer attributes **(see Appendix 7)**, we propose a set of targeted marketing strategies. Informational offers were universally ineffective and should be avoided. In contrast, BOGO and discount offers showed promising results, with 63% of respondents being predicted to complete the offer. This proportion is especially higher (70.2%) among customers who joined Starbucks before mid-2017.

For users with membership year < 2017.5 with high income (income ≥ $63,500), campaign completion rates remained high across all offer types, suggesting universal targeting may be appropriate for this segment.

For users with membership year < 2017.5 and income below $63,500, female and non-binary customers showed notably higher probability to complete the offer (70.7%). Therefore, we recommend targeting all of them, regardless of offer type.

However, male customers in this group require more tailored strategies:

For those who joined before 2016.5, a combination of high-channel exposure (e.g., web, email, mobile, and social), bogo and low difficulty levels (<7.5) would be recommended, with 67% of them being predicted to complete the offer. High-channel exposure strategy works better with this segment when combined with discount (81.8%).

For males who joined between 2016.5 and 2017.5, a similar strategy is advised—pair discount offers with broad multi-channel delivery, with 62% of them being predicted to complete the offer.

In addition to audience targeting, our regression findings suggest that marketing outcomes are more responsive to delivery channels and behavioral traits than to the nominal offer category. Therefore, we would recommend Starbucks reallocate campaign budgets to emphasize channels with better performance—specifically mobile push and social media—which have significantly stronger impacts on transaction spending than email or web-based exposures. However, email or web-based exposures still need to be included at a proper proportion to achieve a better overall performance. Longer offer durations are also associated with greater customer spend, implying that extending redemption windows could be an effective lever for boosting revenue.

Overall, this analysis reinforces the importance of personalized, data-driven marketing. By leveraging behavioral and demographic insights alongside offer and channel characteristics, Starbucks can deliver more effective campaigns that align with customer preferences, improve conversion rates, and increase total revenue.

Appendix

Appendix 1: Datasets and definitions of variables

Portfolio.csv

	reward	channels	difficulty	duration	offer_type	id
0	10	['email', 'mobile', 'social']	10	7	bogo	ae264e3637204a6fb9bb56bc8210ddfd
1	10	['web', 'email', 'mobile', 'social']	10	5	bogo	4d5c57ea9a6940dd891ad53e9dbe8da0
2	0	['web', 'email', 'mobile']	0	4	informational	3f207df678b143eea3cee63160fa8bed
3	5	['web', 'email', 'mobile']	5	7	bogo	9b98b8c7a33c4b65b9aebfe6a799e6d9
4	5	['web', 'email']	20	10	discount	0b1e1539f2cc45b7b9fa7c272da2e1d7
5	3	['web', 'email', 'mobile', 'social']	7	7	discount	2298d6c36e964ae4a3e7e9706d1fb8c2
6	2	['web', 'email', 'mobile', 'social']	10	10	discount	fafdcd668e3743c1bb461111dcafc2a4
7	0	['email', 'mobile', 'social']	0	3	informational	5a8bc65990b245e5a138643cd4eb9837
8	5	['web', 'email', 'mobile', 'social']	5	5	bogo	f19421c1d4aa40978ebb69ca19b0e20d
9	2	['web', 'email', 'mobile']	10	7	discount	2906b810c7d4411798c6938adc9daaa5

Variable	Data Type	Description
id	String	Unique identifier for each offer
offer_type	Categorical	Type of offer: `bogo`, `discount`, or `informational`
difficulty	Integer	Minimum amount (in dollars) required to complete the offer
duration	Integer	Validity period of the offer in days
reward	Integer	Reward amount for successful completion (0 for informational offers)
channels	List of Strings	Channels used to deliver the offer (e.g., `['email', 'mobile', 'web']`)

Profile.csv

	gender	age	id	became_member_on	income
0		118	68be06ca386d4c31939f3a4f0e3dd783	20170212	
1	F	55	0610b486422d4921ae7d2bf64640c50b	20170715	112000.0
2		118	38fe809add3b4fcf9315a9694bb96ff5	20180712	
3	F	75	78afa995795e4d85b5d9ceeca43f5fef	20170509	100000.0
4		118	a03223e636434f42ac4c3df47e8bac43	20170804	
5	M	68	e2127556f4f64592b11af22de27a7932	20180426	70000.0
6		118	8ec6ce2a7e7949b1bf142def7d0e0586	20170925	
7		118	68617ca6246f4fbc85e91a2a49552598	20171002	
8	M	65	389bc3fa690240e798340f5a15918d5c	20180209	53000.0
9		118	8974fc5686fe429db53ddde067b88302	20161122	
10		118	c4863c7985cf408faee930f111475da3	20170824	

Variable	Data Type	Description
id	String	Unique customer identifier
gender	Categorical	Gender of the customer: `M`, `F`, or `O` (other)
age	Integer	Age of the customer
income	Integer	Annual income of the customer in dollars
became_member_on	Integer	Date the customer joined Starbucks Rewards (in `YYYYMMDD` format)

Transcript.csv

	person	event	value	time
0	78afa995795e4d85b5d9ceeca43f5fef	offer received	{'offer id': '9b98b8c7a33c4b65b9aebfe6a799e6d9'}	0
1	a03223e636434f42ac4c3df47e8bac43	offer received	{'offer id': '0b1e1539f2cc45b7b9fa7c272da2e1d7'}	0
2	e2127556f4f64592b11af22de27a7932	offer received	{'offer id': '2906b810c7d4411798c6938adc9daaa5'}	0
3	8ec6ce2a7e7949b1bf142def7d0e0586	offer received	{'offer id': 'fafdcd668e3743c1bb461111dcafc2a4'}	0
4	68617ca6246f4fbc85e91a2a49552598	offer received	{'offer id': '4d5c57ea9a6940dd891ad53e9dbe8da0'}	0
5	389bc3fa690240e798340f5a15918d5c	offer received	{'offer id': 'f19421c1d4aa40978ebb69ca19b0e20d'}	0
6	c4863c7985cf408faee930f111475da3	offer received	{'offer id': '2298d6c36e964ae4a3e7e9706d1fb8c2'}	0
125	c0231649f05d40889e3a6e1172303b37	offer complete	{'offer_id': 'f19421c1d4aa40978ebb69ca19b0e20d', '	336
125	8f5592cf479e4fd3b37fdfffa65977ff	transaction	{'amount': 19.02}	336
125	1ac0142abefa459aa275aeb9410b6109	offer viewed	{'offer id': 'ae264e3637204a6fb9bb56bc8210ddfd'}	336
125	1ac0142abefa459aa275aeb9410b6109	transaction	{'amount': 12.11}	336
125	1ac0142abefa459aa275aeb9410b6109	offer complete	{'offer_id': 'ae264e3637204a6fb9bb56bc8210ddfd', '	336
125	313f11a99c9e45aaab99f3af379349ad	transaction	{'amount': 6.43}	336
125	ce2a843de1684511a4276942c4ccead8	offer viewed	{'offer id': 'f19421c1d4aa40978ebb69ca19b0e20d'}	336

Variable	Data Type	Description
person	String	Unique customer identifier (matches `id` in profile.csv)
event	Categorical	Type of event: `offer received`, `offer viewed`, `offer completed`, or `transaction`
value	Dictionary	Nested data containing keys such as `offer id`, `amount`, or `reward`
time	Integer	Time of the event in hours since the start of observation period (0–714)

Appendix 2: Merged datasets

df_offer_class.csv

person	offer_alias	is_viewed	is_completed	gender	age	income	became_member_on	offer_type	difficulty	duration	channels	membership_year
0009655768c64bdeb2e877511632db8f	B	1	1	M	33	72000.0	2017-04-21	bogo	5	5	['web', 'email', 'mobile', 'social']	2017
0009655768c64bdeb2e877511632db8f	F	1	1	M	33	72000.0	2017-04-21	discount	10	10	['web', 'email', 'mobile', 'social']	2017
0009655768c64bdeb2e877511632db8f	G	0	1	M	33	72000.0	2017-04-21	discount	10	7	['web', 'email', 'mobile']	2017
0009655768c64bdeb2e877511632db8f	I	1	0	M	33	72000.0	2017-04-21	informational	0	4	['web', 'email', 'mobile']	2017
0009655768c64bdeb2e877511632db8f	J	1	0	M	33	72000.0	2017-04-21	informational	0	3	['email', 'mobile', 'social']	2017
0011e0d4e6b944f998e987f904e8c1e5	A	1	1	O	40	57000.0	2018-01-09	bogo	5	7	['web', 'email', 'mobile']	2018
0011e0d4e6b944f998e987f904e8c1e5	E	1	1	O	40	57000.0	2018-01-09	discount	7	7	['web', 'email', 'mobile', 'social']	2018
0011e0d4e6b944f998e987f904e8c1e5	H	1	1	O	40	57000.0	2018-01-09	discount	20	10	['web', 'email']	2018
0011e0d4e6b944f998e987f904e8c1e5	I	1	0	O	40	57000.0	2018-01-09	informational	0	4	['web', 'email', 'mobile']	2018
0011e0d4e6b944f998e987f904e8c1e5	J	1	0	O	40	57000.0	2018-01-09	informational	0	3	['email', 'mobile', 'social']	2018
0020c2b971eb4e9188eac86d93036a77	C	0	0	F	59	90000.0	2016-03-04	bogo	10	7	['email', 'mobile', 'social']	2016
0020c2b971eb4e9188eac86d93036a77	D	1	1	F	59	90000.0	2016-03-04	bogo	10	5	['web', 'email', 'mobile', 'social']	2016
0020c2b971eb4e9188eac86d93036a77	F	1	1	F	59	90000.0	2016-03-04	discount	10	10	['web', 'email', 'mobile', 'social']	2016

Variable	Data Type	Description
is_viewed	Binary (0/1)	Indicates whether the offer was viewed by the customer
is_completed	Binary (0/1)	Indicates whether the offer was completed by the customer (target variable)
offer_alias	Categorical	Simplified ID representing one of the 10 offers (`A` to `J`)
offer_type	Categorical	Type of offer (`bogo`, `discount`, `informational`)
difficulty	Integer	Dollar threshold required to redeem the offer
duration	Integer	Validity of the offer in days
channels	List	Delivery channels for the offer (email, mobile, social, web)
gender	Categorical	Customer's gender
age	Integer	Customer's age
income	Integer	Customer's income
membership_year	Integer	Year the customer joined Starbucks Rewards

df_transaction_reg.csv

person	total_amount	count_type_bogo	count_type_discount	count_type_informational	avg_duration	avg_difficulty	gender	age	income	became_member_on	count_channel_email	count_channel_mobile	count_channel_social	count_channel_web	membership_year
0009655768c64bdeb2e877511632db8f	127.6	1.0	2.0	2.0	5.8	5.0	M	33.0	72000.0	2017-04-21	5.0	5.0	3.0	4.0	2017
0011e0d4e6b944f998e987f904e8c1e5	79.460000000000000	1.0	2.0	2.0	6.2	6.4	O	40.0	57000.0	2018-01-09	5.0	4.0	2.0	4.0	2018
0020c2b971eb4e9188eac86d93036a77	196.860000000000000	2.0	2.0	1.0	7.0	8.0	F	59.0	90000.0	2016-03-04	5.0	5.0	5.0	3.0	2016
0020ccbbb6d84e358d3414a3ff76cffd	154.05	2.0	1.0	1.0	5.5	4.25	F	24.0	60000.0	2016-11-11	4.0	4.0	3.0	3.0	2016
003d66b6608740288d6cc97a6903f4f0	48.34	0.0	3.0	2.0	7.4	8.0	F	26.0	73000.0	2017-06-21	5.0	4.0	3.0	4.0	2017
00426fe3ffde4c6b9cb9ad6d077a13ea	68.510000000000000	0.0	4.0	1.0	7.4	10.0	F	19.0	65000.0	2016-08-09	5.0	4.0	2.0	4.0	2016
004b041fbfe44859945daa2c7f79ee64	138.36	1.0	1.0	1.0	6.333333333333330	5.0	F	55.0	74000.0	2018-05-08	3.0	3.0	2.0	3.0	2018
004c5799adbf42868b9cff0396190900	347.38	3.0	2.0	0.0	7.4	8.0	M	54.0	99000.0	2016-03-31	5.0	5.0	5.0	4.0	2016
005500a7188546ff8a767329a2f7c76a	20.36	4.0	1.0	0.0	7.0	9.0	M	56.0	47000.0	2017-12-09	5.0	5.0	3.0	2.0	2017
0056df74b63b4296809f0b375a304cf4	144.14	1.0	2.0	1.0	7.0	8.0	M	54.0	91000.0	2016-08-21	4.0	3.0	1.0	4.0	2016
00715b6e55c3431cb56ff7307eb19675	375.12	2.0	3.0	1.0	7.166666666666670	11.666666666666700	F	58.0	119000.0	2017-12-07	6.0	4.0	2.0	5.0	2017
0082fd87c18f45f2be70dbcbb0fb8aad	121.85	2.0	0.0	3.0	5.0	2.0	F	28.0	68000.0	2017-09-08	5.0	5.0	1.0	4.0	2017
00840a2ca5d2408e982d56544dc14ffd	62.93	1.0	4.0	1.0	7.5	9.166666666666670	M	26.0	61000.0	2014-12-21	6.0	5.0	1.0	6.0	2014
00857b24b13f4fe0ad17b605f00357f5	6.26	4.0	1.0	0.0	6.8	10.0	M	71.0	41000.0	2017-10-23	5.0	4.0	3.0	4.0	2017
008d7088107b468893889da0ede0df5c	16.18	3.0	1.0	2.0	5.0	5.833333333333330	M	24.0	42000.0	2017-09-10	6.0	6.0	5.0	3.0	2017
0091d2b6a5ea4defaa8393e4e816db60	279.16	4.0	1.0	0.0	6.0	10.0	F	62.0	81000.0	2016-06-17	5.0	4.0	4.0	5.0	2016

Variable	Data Type	Description
total_amount	Numeric	Total dollar amount spent by the customer during the observation period
count_type_bogo	Integer	Number of BOGO offers received by the customer
count_type_discount	Integer	Number of discount offers received
count_type_informational	Integer	Number of informational offers received
avg_duration	Numeric	Average duration (in days) of offers received
avg_difficulty	Numeric	Average difficulty (minimum spend threshold) of offers received
count_channel_email	Integer	Number of offers received via email channel
count_channel_mobile	Integer	Number of offers received via mobile app
count_channel_social	Integer	Number of offers received via social media
count_channel_web	Integer	Number of offers received via web
gender	Categorical	Customer's gender
age	Integer	Customer's age
income	Integer	Customer's income
membership_year	Integer	Year the customer joined Starbucks Rewards

Appendix 3: K-NN

1. offer_alias model

Predictors: offer_alias, gender, age, income, membership_year

```
> print(cbind(k = 1:10, training_error = err_by_k))
      k training_error
 [1,] 1            874
 [2,] 2           6412
 [3,] 3           6929
 [4,] 4           8151
 [5,] 5           8436
 [6,] 6           8970
 [7,] 7           9268
 [8,] 8           9702
 [9,] 9           9810
[10,] 10         10091
```

```
> print(cbind(k = 1:10, testing_error = err_by_k))
      k testing_error
 [1,] 1          3550
 [2,] 2          3542
 [3,] 3          3412
 [4,] 4          3419
 [5,] 5          3348
 [6,] 6          3371
 [7,] 7          3393
 [8,] 8          3339
 [9,] 9          3355
[10,] 10         3346
```

```
pred_test    0    1
        0 3574 1557
        1 1807 4107
```

```
Precision = 0.6944538    Recall = 0.7251059    F1 = 0.709449
```

2. offer attributes model

Predictors: gender, age, income, membership_year, offer_type, difficulty, duration, channels

```
> print(cbind(k = 1:10, training_error = err_by_k))
      k training_error
 [1,] 1            874
 [2,] 2           6896
 [3,] 3           7100
 [4,] 4           8294
 [5,] 5           8382
 [6,] 6           8896
 [7,] 7           9013
 [8,] 8           9256
 [9,] 9           9486
[10,] 10          9739
```

```
> print(cbind(k = 1:10, testing_error = err_by_k))
      k testing_error
 [1,] 1          3459
 [2,] 2          3440
 [3,] 3          3252
 [4,] 4          3255
 [5,] 5          3137
 [6,] 6          3185
 [7,] 7          3127
 [8,] 8          3143
 [9,] 9          3166
[10,] 10         3152
```

```
pred_test    0    1
        0 3478 1250
        1 1903 4414
```

```
Precision = 0.6987494    Recall = 0.7793079    F1 = 0.7368333
```

Appendix 4: Tree

1. offer_alias model

Predictors: offer_alias, gender, age, income, membership_year

	CP	nsplit	rel error	xerror	xstd
1	0.40298644	0	1.00000	1.00000	0.0047987
2	0.06849628	1	0.59701	0.59701	0.0043811
3	0.01685008	2	0.52852	0.52852	0.0042203
4	0.01085285	3	0.51167	0.50952	0.0041702
5	0.00347048	6	0.47911	0.48226	0.0040936
6	0.00340198	7	0.47564	0.48153	0.0040914
7	0.00260286	9	0.46883	0.47998	0.0040869
8	0.00120249	11	0.46363	0.46879	0.0040537
9	0.00090187	14	0.46002	0.46701	0.0040483
10	0.00077629	18	0.45641	0.46235	0.0040341
11	0.00047947	20	0.45486	0.46089	0.0040296
12	0.00041098	26	0.45171	0.45970	0.0040260
13	0.00039576	28	0.45089	0.46121	0.0040306
14	0.00038815	32	0.44924	0.46112	0.0040303
15	0.00034248	34	0.44847	0.46116	0.0040305
16	0.00031965	36	0.44778	0.46180	0.0040324
17	0.00029682	39	0.44682	0.46052	0.0040285
18	0.00029225	42	0.44568	0.46116	0.0040305
19	0.00028921	47	0.44422	0.46107	0.0040302
20	0.00027399	54	0.44217	0.46066	0.0040289
21	0.00025115	60	0.44052	0.45993	0.0040267
22	0.00024354	68	0.43806	0.46011	0.0040272
23	0.00022832	71	0.43733	0.46025	0.0040276
24	0.00021919	78	0.43564	0.46080	0.0040293
25	0.00021690	84	0.43431	0.46025	0.0040276
26	0.00020549	88	0.43344	0.45993	0.0040267
27	0.00018266	96	0.43180	0.46034	0.0040279
28	0.00015982	119	0.42714	0.46153	0.0040316
29	0.00015412	143	0.42312	0.46235	0.0040341
30	0.00015221	154	0.42134	0.46235	0.0040341

The model with 26 splits has the lowest xerror (0.45970). According to 1-SE rule, The model with 18 splits would be selected (cp = 0.00077629, xerror = 0.46235).

Unpruned tree

pred_test	0	1
0	3833	1001
1	1548	4663

Precision = 0.7507648 Recall = 0.8232698 F1 = 0.7853474

Pruned tree

pred_test_pruned	0	1
0	3747	835
1	1634	4829

Precision = 0.7471762 Recall = 0.8525777 F1 = 0.7964047

2. offer attributes model

Predictors: gender, age, income, membership_year, offer_type, difficulty, duration, channels

```
           CP nsplit rel error  xerror      xstd
1  0.40298644      0   1.00000 1.00000 0.0047987
2  0.06849628      1   0.59701 0.59701 0.0043811
3  0.01685008      2   0.52852 0.52852 0.0042203
4  0.00598201      3   0.51167 0.51075 0.0041735
5  0.00534271      6   0.49372 0.49660 0.0041345
6  0.00406411      7   0.48838 0.49039 0.0041170
7  0.00360747      9   0.48025 0.48724 0.0041080
8  0.00296817     10   0.47664 0.48660 0.0041061
9  0.00237454     11   0.47367 0.48468 0.0041006
10 0.00189506     13   0.46893 0.47902 0.0040841
11 0.00121010     15   0.46514 0.47075 0.0040596
12 0.00098178     17   0.46272 0.46906 0.0040545
13 0.00073063     19   0.46075 0.46714 0.0040487
14 0.00070018     20   0.46002 0.46541 0.0040434
15 0.00041098     23   0.45792 0.46249 0.0040345
16 0.00036531     29   0.45545 0.46331 0.0040370
17 0.00034248     31   0.45472 0.46326 0.0040369
18 0.00030443     35   0.45335 0.46367 0.0040382
19 0.00027399     38   0.45244 0.46226 0.0040338
20 0.00025115     45   0.45052 0.46262 0.0040349
21 0.00024659     47   0.45002 0.46413 0.0040395
22 0.00024354     52   0.44879 0.46413 0.0040395
23 0.00022832     55   0.44806 0.46518 0.0040427
24 0.00021690     60   0.44673 0.46463 0.0040411
25 0.00020549     64   0.44587 0.46445 0.0040405
26 0.00020092     76   0.44326 0.46477 0.0040415
27 0.00019407     97   0.43874 0.46468 0.0040412
28 0.00018266    108   0.43573 0.46546 0.0040436
29 0.00016744    138   0.42979 0.46518 0.0040427
30 0.00015982    151   0.42714 0.46504 0.0040423
```

The model with 38 splits has the lowest xerror (0.46226). According to 1-SE rule, The model with 18 splits would be selected (cp = 0.00070018, xerror = 0.46541).

Unpruned tree

```
pred_test    0    1
        0 3860 1019
        1 1521 4645

Precision = 0.7533247   Recall = 0.8200918   F1 = 0.7852916
```

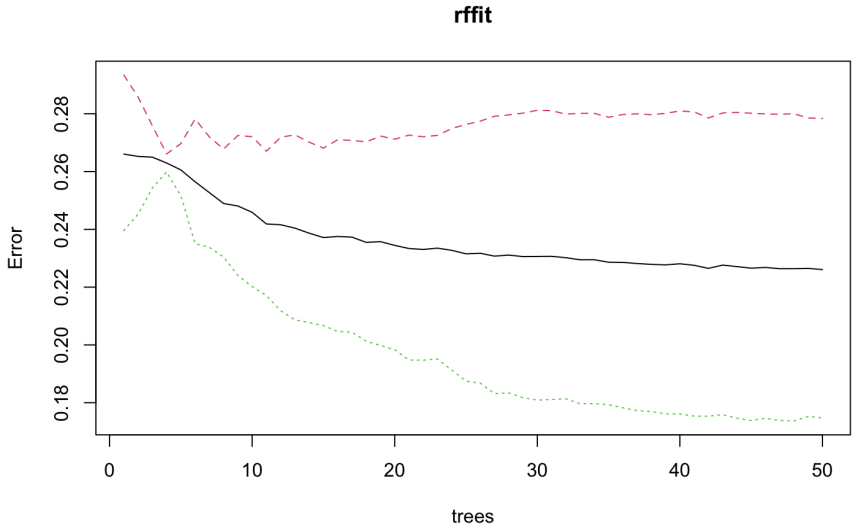
Pruned tree

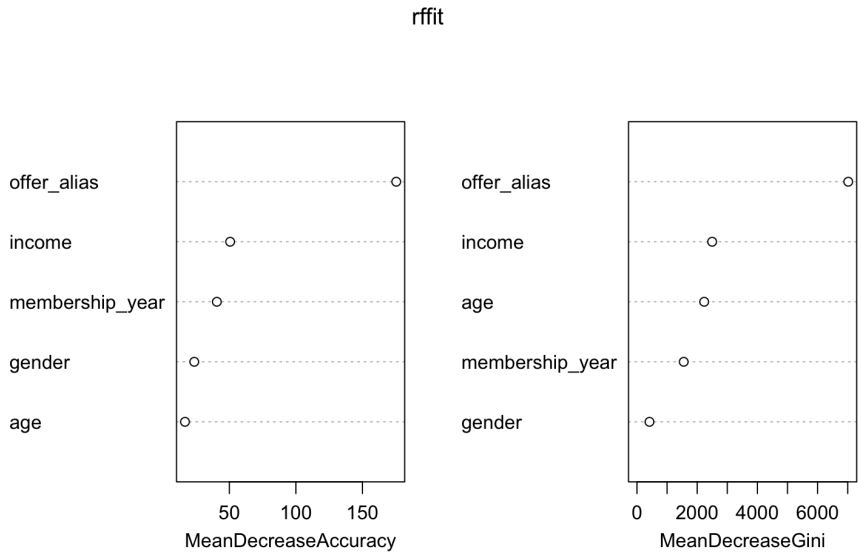
```
pred_test_pruned    0    1
               0 3715  844
               1 1666 4820

Precision = 0.7431391   Recall = 0.8509887   F1 = 0.7934156
```

Appendix 5: Random forest

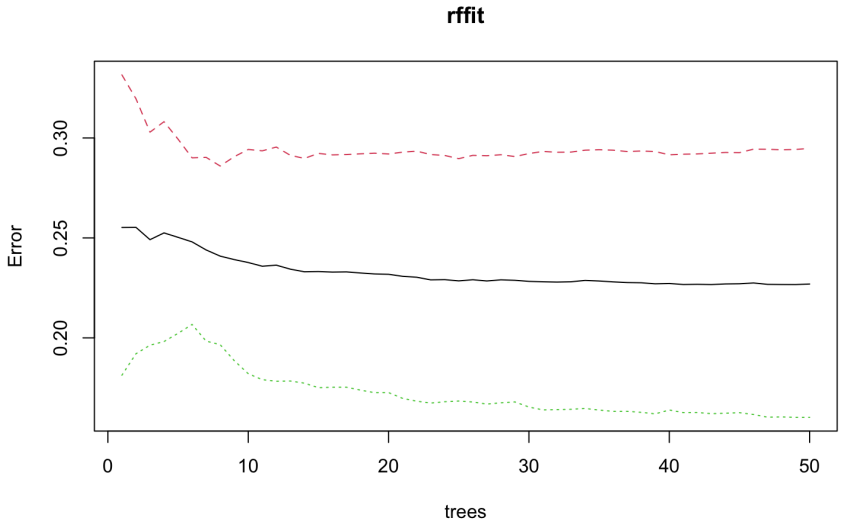
1. offer_alias model

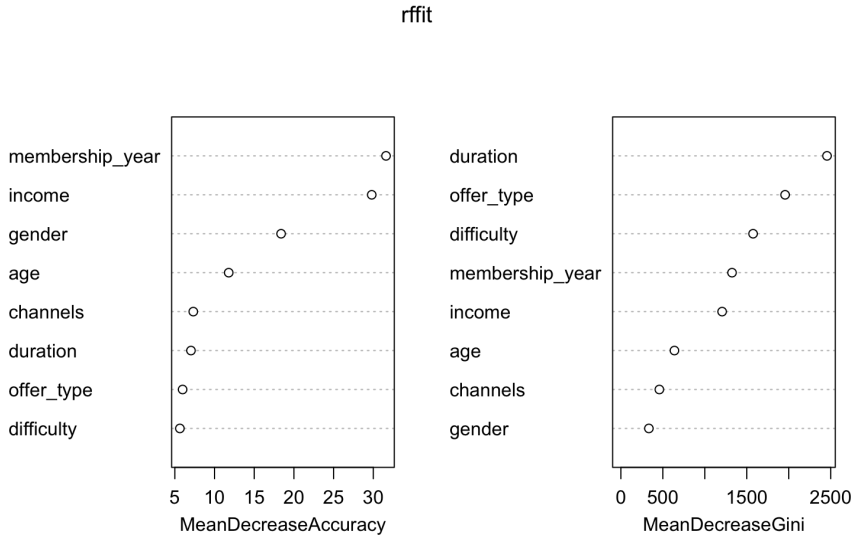





```
> print(model_comparison)
               Precision    Recall  F1 Score
Unpruned Tree 0.7507648 0.8232698 0.7853474
Pruned Tree   0.7471762 0.8525777 0.7964047
Random Forest 0.7573792 0.8245056 0.7895182
```

2. offer attributes model






```
> print(model_comparison)
               Precision    Recall  F1 Score
Unpruned Tree 0.7533247 0.8200918 0.7852916
Pruned Tree   0.7431391 0.8509887 0.7934156
Random Forest 0.7559043 0.8419845 0.7966257
```

Appendix 6: Regression

Forward stepwise regression

```
> res <- lars(x, y, type = "stepwise")
> print(summary(res))
LARS/Forward Stepwise
Call: lars(x = x, y = y, type = "stepwise")
   Df       Rss        Cp
0   1 245696837 2572.0646
1   2 219013615  721.6295
2   3 214111152  383.2848
3   4 211471881  202.0581
4   5 209936712   97.4817
5   6 209111881   42.2193
6   7 208508869    2.3562
7   8 208500811    3.7968
8   9 208495424    5.4228
9  10 208494370    7.3497
10 11 208491953    9.1819
11 12 208490173   11.0583
12 13 208489333   13.0000
```

```
> print(res)

Call:
lars(x = x, y = y, type = "stepwise")
R-squared: 0.151
Sequence of Forward Stepwise moves:

Var  8 9 12 6 4 11 7 13 1  5  2  3
Step 1 2  3 4 5  6 7  8 9 10 11 12
```

Optimized linear model

```
Call:
lm(formula = total_amount ~ income + membership_year + count_channel_social +
    gender + avg_duration + count_channel_mobile, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-207.30  -60.29  -20.56   30.35 1441.80

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -1.057e+02  1.116e+01  -9.473  < 2e-16 ***
income               1.665e-03  4.682e-05  35.558  < 2e-16 ***
membership_year2014 -2.317e+00  8.437e+00  -0.275 0.783610
membership_year2015  4.982e+01  7.693e+00   6.476 9.72e-11 ***
membership_year2016  6.063e+01  7.439e+00   8.150 3.92e-16 ***
membership_year2017  1.791e+01  7.283e+00   2.459 0.013943 *
membership_year2018 -2.752e+01  7.369e+00  -3.735 0.000189 ***
count_channel_social 8.047e+00  1.030e+00   7.811 6.07e-15 ***
genderM             -1.884e+01  2.055e+00  -9.166  < 2e-16 ***
genderO              6.906e+00  8.358e+00   0.826 0.408674
avg_duration         8.825e+00  9.449e-01   9.340  < 2e-16 ***
count_channel_mobile 7.531e+00  1.112e+00   6.773 1.31e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 117.2 on 14475 degrees of freedom
Multiple R-squared:  0.1907,    Adjusted R-squared:  0.1901
F-statistic:   310 on 11 and 14475 DF,  p-value: < 2.2e-16
```

Full linear model

```
Call:
lm(formula = total_amount ~ count_type_bogo + count_type_discount +
    count_type_informational + avg_duration + avg_difficulty +
    gender + age + income + membership_year + count_channel_email +
    count_channel_mobile + count_channel_social + count_channel_web,
    data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-208.20  -60.25  -20.51   30.09 1439.12

Coefficients: (1 not defined because of singularities)
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              -1.024e+02  1.493e+01  -6.858 7.28e-12 ***
count_type_bogo           1.519e+00  3.423e+00   0.444 0.657227
count_type_discount       1.375e+00  3.711e+00   0.371 0.710997
count_type_informational  5.296e-01  2.497e+00   0.212 0.832008
avg_duration              9.025e+00  1.893e+00   4.768 1.88e-06 ***
avg_difficulty           -6.913e-01  9.920e-01  -0.697 0.485906
genderM                  -1.884e+01  2.063e+00  -9.132  < 2e-16 ***
genderO                   6.879e+00  8.361e+00   0.823 0.410659
age                       1.396e-03  5.896e-02   0.024 0.981114
income                    1.665e-03  4.873e-05  34.166  < 2e-16 ***
membership_year2014      -2.286e+00  8.440e+00  -0.271 0.786497
membership_year2015       4.987e+01  7.695e+00   6.481 9.39e-11 ***
membership_year2016       6.067e+01  7.441e+00   8.153 3.84e-16 ***
membership_year2017       1.793e+01  7.285e+00   2.461 0.013868 *
membership_year2018      -2.751e+01  7.371e+00  -3.732 0.000191 ***
count_channel_email              NA         NA      NA       NA
count_channel_mobile      5.308e+00  2.937e+00   1.807 0.070800 .
count_channel_social      8.446e+00  1.220e+00   6.923 4.60e-12 ***
count_channel_web         7.820e-01  1.544e+00   0.507 0.612474
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 117.2 on 14469 degrees of freedom
Multiple R-squared:  0.1907,    Adjusted R-squared:  0.1898
F-statistic: 200.6 on 17 and 14469 DF,  p-value: < 2.2e-16
```

Appendix 7: Output of pruned tree model (offer attributes setting)

```
> print(tfit_pruned)
n= 44177

node), split, n, loss, yval, (yprob)
      * denotes terminal node

   1) root 44177 21899 1 (0.4957104 0.5042896)
     2) offer_type=informational 8825     0 0 (1.0000000 0.0000000) *
     3) offer_type=bogo,discount 35352 13074 1 (0.3698235 0.6301765)
       6) membership_year>=2017.5 8814  3657 0 (0.5850919 0.4149081)
        12) income< 76500 6853  2492 0 (0.6363636 0.3636364)
          24) gender=M 4729  1484 0 (0.6861916 0.3138084) *
          25) gender=F,O 2124  1008 0 (0.5254237 0.4745763)
            50) difficulty>=8.5 1317   563 0 (0.5725133 0.4274867) *
            51) difficulty< 8.5 807   362 1 (0.4485750 0.5514250) *
        13) income>=76500 1961   796 1 (0.4059153 0.5940847) *
       7) membership_year< 2017.5 26538  7917 1 (0.2983269 0.7016731)
        14) income< 63500 12641  5068 1 (0.4009176 0.5990824)
          28) gender=M 8165  3755 1 (0.4598898 0.5401102)
            56) membership_year>=2016.5 4207  1907 0 (0.5467079 0.4532921)
             112) channels=['email', 'mobile', 'social'],['web', 'email'] 1049   350 0 (0.6663489 0.3336511) *
             113) channels=['web', 'email', 'mobile', 'social'],['web', 'email', 'mobile'] 3158  1557 0 (0.5069664 0.4930336)
               226) offer_type=bogo 1594   686 0 (0.5696361 0.4303639)
                 452) difficulty>=7.5 552   178 0 (0.6775362 0.3224638) *
                 453) difficulty< 7.5 1042   508 0 (0.5124760 0.4875240)
                   906) income< 44500 457   189 0 (0.5864333 0.4135667) *
                   907) income>=44500 585   266 1 (0.4547009 0.5452991) *
               227) offer_type=discount 1564   693 1 (0.4430946 0.5569054)
                 454) channels=['web', 'email', 'mobile'] 515   225 0 (0.5631068 0.4368932)
                   908) income< 48500 289   104 0 (0.6401384 0.3598616) *
                   909) income>=48500 226   105 1 (0.4646018 0.5353982) *
                 455) channels=['web', 'email', 'mobile', 'social'] 1049   403 1 (0.3841754 0.6158246) *
            57) membership_year< 2016.5 3958  1455 1 (0.3676099 0.6323901)
             114) channels=['email', 'mobile', 'social'],['web', 'email'] 999   441 0 (0.5585586 0.4414414)
               228) membership_year< 2015.5 582   193 0 (0.6683849 0.3316151) *
               229) membership_year>=2015.5 417   169 1 (0.4052758 0.5947242) *
             115) channels=['web', 'email', 'mobile', 'social'],['web', 'email', 'mobile'] 2959   897 1 (0.3031430 0.6968570)
               230) offer_type=bogo 1484   628 1 (0.4231806 0.5768194)
                 460) difficulty>=7.5 504   200 0 (0.6031746 0.3968254)
                   920) membership_year< 2014.5 138    22 0 (0.8405797 0.1594203) *
                   921) membership_year>=2014.5 366   178 0 (0.5136612 0.4863388)
                    1842) income< 52500 219    83 0 (0.6210046 0.3789954) *
                    1843) income>=52500 147    52 1 (0.3537415 0.6462585) *
                 461) difficulty< 7.5 980   324 1 (0.3306122 0.6693878) *
               231) offer_type=discount 1475   269 1 (0.1823729 0.8176271) *
          29) gender=F,O 4476  1313 1 (0.2933423 0.7066577)
            58) income< 50500 2197   792 1 (0.3604916 0.6395084)
             116) difficulty>=8.5 1396   594 1 (0.4255014 0.5744986)
               232) membership_year< 2014.5 106    30 0 (0.7169811 0.2830189) *
               233) membership_year>=2014.5 1290   518 1 (0.4015504 0.5984496) *
             117) difficulty< 8.5 801   198 1 (0.2471910 0.7528090) *
            59) income>=50500 2279   521 1 (0.2286090 0.7713910) *
        15) income>=63500 13897  2849 1 (0.2050083 0.7949917) *
```