DA 6813 Case Study 2

AUTHOR
Will Hytlin, Holly Millazo and Tim Harrison

PUBLISHED
October 6, 2024

1 Executive Summary

This study aimed to develop a predictive model for the Bookbinders Book Club (BBBC) to determine which customers were likely to purchase The Art History of Florence following a direct mail campaign. The analysis was conducted on a dataset containing customer demographics, purchasing behavior, and preferences for different book genres. The key variables considered included gender, amount spent on BBBC books, frequency of purchases, and the number of specific genres purchased (e.g., children's books, cookbooks, art books).

Four different modeling techniques were evaluated: Linear Regression, LDA, Logit, and support vector machines (SVM). Given that the dependent variable was binary (purchase or no purchase), linear regression was found to be unsuitable as it attempts to predict a continuous outcome rather than a classification, leading to misleading results. Logistic regression and SVM were thus compared for their predictive performance on the unbalanced and balanced datasets.

Initial logistic regression results on the unbalanced data showed moderate accuracy (65.61%) and balanced accuracy (71.84%) but low sensitivity (17.78%). However, balancing the dataset improved sensitivity and overall predictive performance. Similarly, the SVM model initially underperformed due to the unbalanced nature of the data but saw significant improvement after balancing, with a final accuracy of 73.77%, sensitivity of 65.69%, and specificity of 81.86%.

The logit model, while having a lower overall accuracy (65.61%) compared to the LDA model (88.91%), shows a much higher sensitivity in detecting the minority class (79.41% vs. 37.75% for LDA). However, the LDA model excels in specificity (93.89% vs. 64.27% for logit) and achieves a higher Kappa value, indicating better agreement overall. Despite this, the logit model has a higher balanced accuracy (71.84% vs. 65.82%), suggesting a better trade-off between detecting both classes. Ultimately, the logit model is more effective at identifying the minority class, while the LDA model performs better for the majority class and overall accuracy.

A key insight was that improving sensitivity, even at the cost of specificity, was critical for maximizing revenue. By lowering the decision threshold in the logistic regression model, the sensitivity increased, resulting in a higher proportion of correctly identified purchasers. Despite the reduction in specificity, the model captured a larger number of likely buyers, ultimately leading to greater profit potential compared to a naïve approach.Capturing more buyers is crucial because the low cost of sending mailers is far outweighed by the potential revenue from correctly identifying additional purchasers. Increasing sensitivity ensures that more potential buyers receive offers, boosting the chances of converting them into sales. Even with some false positives, the higher number of actual buyers leads to greater overall profit compared to a more conservative approach that risks missing out on revenue opportunities.

The profitability analysis showed that while the logistic regression and SVM models performed better than random guessing, there is still room for improvement. Adjusting model parameters such as the decision threshold and considering alternative models, like LDA, could further enhance profitability by improving the balance between targeting the right customers and controlling campaign costs.

The logistic regression model with an optimized threshold provided a solid proof of concept by delivering the best balance of revenue and costs, demonstrating its potential for profitability.

2 Problem Statement

The task of this case study is to develop a predictive model to classify whether customers of the Bookbinders Book Club (BBBC) will purchase The Art History of Florence following a direct mail marketing campaign. The campaign involved sending a specially produced brochure to selected customers in Pennsylvania, New York, and Ohio, aiming to assess the likelihood of each customer making a purchase ('yes') or not ('no').

The goal is to accurately (or rather accurate enough to maximize profit) predict their likelihood of purchasing 'The Art History of Florence' based on various input variables, including demographic factors (such as gender) and past purchasing behaviors, including the total amount spent on BBBC books, the frequency of past purchases, and preferences for different book genres (such as children's books, cookbooks, do-it-yourself, and art books). These factors are believed to significantly influence the decision to purchase the featured book.

The primary objective is to build a classification model that can accurately predict customer purchases, enabling BBBC to target its marketing efforts more effectively. By identifying the most likely purchasers, BBBC can optimize resource allocation, reduce unnecessary mailer costs, and improve the overall conversion rate of its marketing campaigns.

3 Additional Sources

Aldelemy, A., & Abd-Alhameed, R. A. (2023). Binary classification of customer's online purchasing behavior using machine learning. Journal of Techniques, 5(2), 163–186. https://doi.org/10.51173/jt.v5i2.1226

This reference highlights the strong performance of logistic regression compared to other models, which supports our conclusion where logistic regression ultimately outperformed other methods

4 Methodology

The analysis began with data preparation, where the dataset of 12 variables, both categorical and numeric, was cleaned and transformed. The categorical variable Gender was converted to a binary factor, and the target variable, Choice, which indicated whether a customer purchased The Art History of Florence, was transformed into a binary indicator (1 for purchase, 0 for no purchase). Variables representing different genres of books purchased, such as P_Child, P_Youth, P_Cook, P_DIY, and P_Art, were retained as numeric variables reflecting the number of books purchased in each category.

Exploratory data analysis (EDA) was conducted to examine the distribution and relationships within the data. Histograms and box plots were generated to visualize the distribution of numeric variables such as Amount_purchased, Frequency, and Last_purchase based on the outcome variable Choice. Bar plots were used to explore the frequency of categorical variables like Gender. A correlation matrix was constructed to identify relationships among numeric variables and to detect potential multicollinearity issues.

The dataset initially provided was two pre-split sets: one for training and one for testing. However, these datasets were later combined for exploratory data analysis (EDA), correlation analysis, and visualization. The training set contained 80% of the data, and the test set comprised 20%. The combination allowed for comprehensive analysis while ensuring that model evaluation was still conducted on unseen data.

Several modeling techniques were explored, including logistic regression, linear discriminant analysis (LDA), and support vector machines (SVM). Logistic regression was selected as the primary technique due to its suitability for binary classification and its flexibility in optimizing sensitivity and specificity. A stepwise backward selection method, based on the Akaike Information Criterion (AIC), was used to remove insignificant variables and select the most relevant predictors. To address potential multicollinearity, variables with high variance inflation factor (VIF) values were removed.

Model performance was evaluated using accuracy, sensitivity, and specificity, with results summarized in a confusion matrix. To further address class imbalance in the test set, the decision threshold for classifying customers was adjusted to optimize model performance. By iterating over different threshold values, an optimal cutoff was determined that balanced sensitivity and specificity, enhancing the model's ability to predict both purchasers and non-purchasers effectively.

Finally, a profitability analysis was conducted to evaluate the financial impact of the model. The cost of sending mailers and the revenue from book purchases were calculated to determine the overall profit for each modeling approach.

5 Data

The dataset used for this analysis contains a total of 12 variables across both training and testing sets. These variables represent customer demographics, purchasing behavior, and preferences for various book genres at the Bookbinders Book Club (BBBC). The key target variable is Choice, which indicates whether a customer purchased The Art History of Florence. The data consists of both categorical and numeric variables.

Name	combined
Number of rows	3900
Number of columns	11
_______________________	
Column type frequency:	
factor	2
numeric	9
________________________	
Group variables	None

Data summary

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Choice	0	1	FALSE	2	0: 3296, 1: 604
Gender	0	1	FALSE	2	1: 2633, 0: 1267

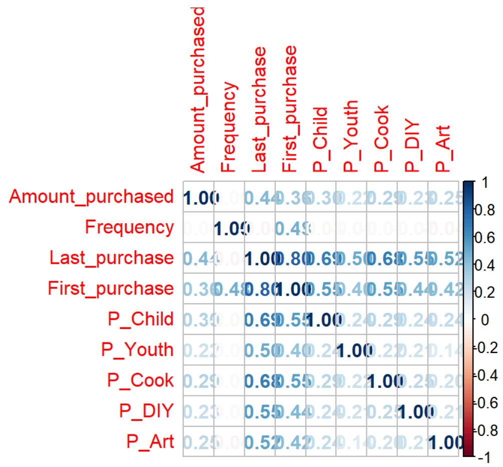
Variable type: numeric

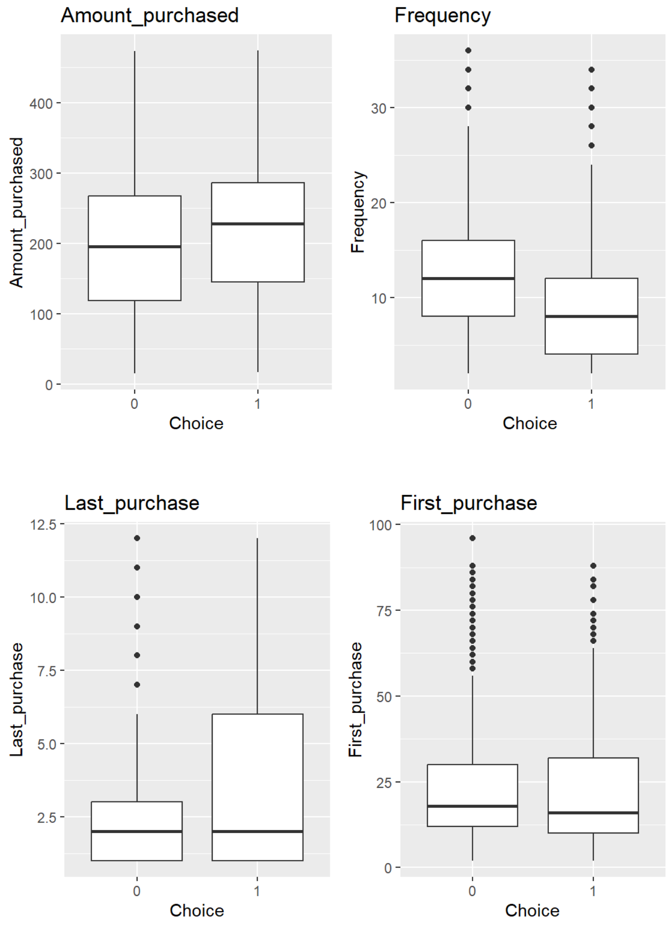
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Amount_purchased	0	1	197.59	95.78	15	122	200	270	474	
Frequency	0	1	12.90	8.09	2	8	12	16	36	
Last_purchase	0	1	3.12	2.94	1	1	2	4	12	
First_purchase	0	1	22.74	15.90	2	12	18	30	96	
P_Child	0	1	0.73	1.03	0	0	0	1	8	
P_Youth	0	1	0.34	0.63	0	0	0	1	5	
P_Cook	0	1	0.78	1.05	0	0	0	1	6	
P_DIY	0	1	0.40	0.70	0	0	0	1	4	
P_Art	0	1	0.37	0.67	0	0	0	1	5	

A check for missing values was performed (anyNA()), and no missing data was detected, so no further imputation or cleaning steps were necessary in that regard.

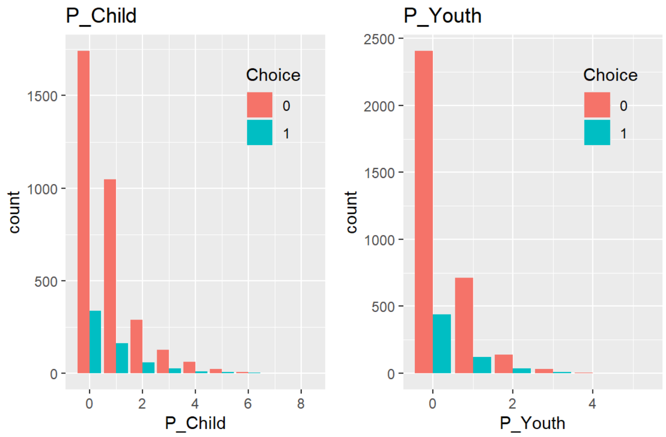
The variable Observation was removed not due to multicollinearity but because it served as a unique identifier for each record and did not provide any predictive value for the analysis.We then converted categorical variables (Choice and Gender) into factors, and combined the training and testing datasets for further analysis or visualization.

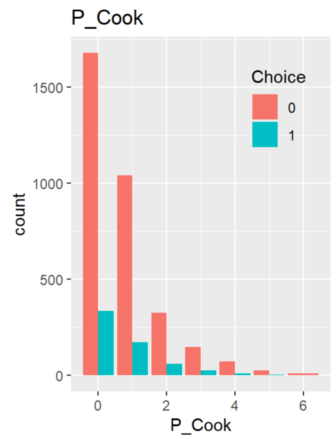
During our exploratory data analysis (EDA), various visualizations were used to examine the distributions and relationships within the dataset. A correlation plot was created to assess the relationships among numeric variables such as Amount_purchased, Frequency, Last_purchase, First_purchase, and the number of different types of books purchased (e.g., P_Child, P_Youth, P_Cook, P_DIY, P_Art). This helped to identify any strong correlations or multicollinearity between the numeric features.

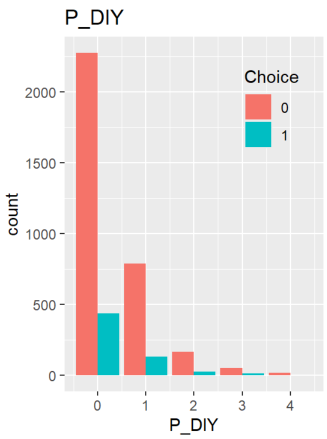


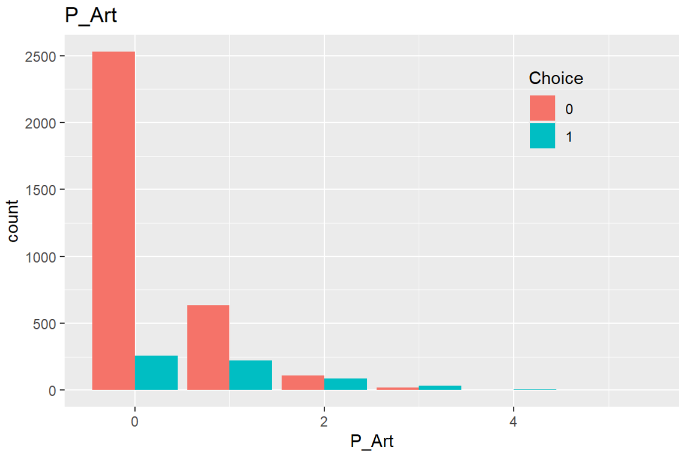


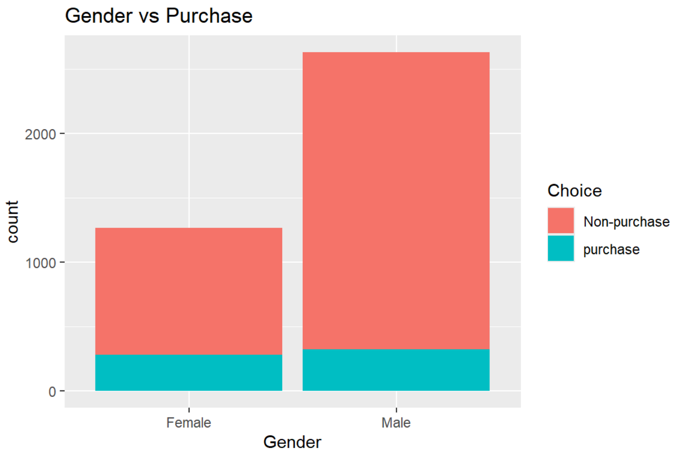
Bar plots were used to explore the distribution of categorical variables such as Gender and Choice (purchase or non-purchase). For example, a bar plot was generated to visualize the relationship between gender and purchase behavior, displaying the frequency of purchases and non-purchases among males and females. These visualizations provided insights into the key factors that might influence the likelihood of a customer purchasing a book.











6 Findings

Upon initial assesment of our SVM on balanced data, the sensitivity of the model was relatively low, but this wasn't a significant issue given our objective. The model predicted that 160 out of 408 observations would likely purchase the book. To ensure the integrity of our model and data, we applied appropriate transformations and balanced the responses in both the training and test sets. However, since we have no knowledge of the distribution of responses in the actual mailing list audience, we cannot assume that it will be balanced. Therefore, it was important to validate our model's performance on an unbalanced dataset to ensure it remained effective in real-world scenarios, where the distribution of purchasers and non-purchasers may differ.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 2060  171
         1   36   33

               Accuracy : 0.91
                 95% CI : (0.8976, 0.9214)
    No Information Rate : 0.9113
    P-Value [Acc > NIR] : 0.6049
```

```
                  Kappa : 0.2062

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.16176
            Specificity : 0.98282
         Pos Pred Value : 0.47826
         Neg Pred Value : 0.92335
             Prevalence : 0.08870
         Detection Rate : 0.01435
   Detection Prevalence : 0.03000
      Balanced Accuracy : 0.57229

       'Positive' Class : 1
```

After applying the SVM model to the original unbalanced test dataset, the sensitivity and specificity metrics remained consistent with those observed in the balanced dataset. This outcome is logical because the distribution between positive (purchasers) and negative (non-purchasers) cases only affects the overall prevalence, not the fundamental calculations of sensitivity and specificity. Each metric remained robust regardless of changes in class distribution because they were calculated independently within each class. As a result, we could apply these performance metrics to a hypothetical, unbalanced dataset of random customers.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1660   70
         1  436  134

               Accuracy : 0.78
                 95% CI : (0.7625, 0.7968)
    No Information Rate : 0.9113
    P-Value [Acc > NIR] : 1

                  Kappa : 0.248

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.65686
            Specificity : 0.79198
         Pos Pred Value : 0.23509
         Neg Pred Value : 0.95954
             Prevalence : 0.08870
         Detection Rate : 0.05826
   Detection Prevalence : 0.24783
      Balanced Accuracy : 0.72442

       'Positive' Class : 1
```

However, our overall analysis revealed that the logistic regression outperformed the other models in terms of overall prediction accuracy, particularly after the decision threshold was optimized. By adjusting the threshold, the model’s sensitivity significantly improved, allowing it to correctly identify a larger number of customers who were likely to purchase the featured book. While the support vector machine (SVM) model initially demonstrated poor sensitivity due to the unbalanced nature of the dataset, its performance improved once the data was balanced. The SVM model showed strong specificity, meaning it effectively reduced false positives, but this came at the cost of lower sensitivity. Linear discriminant analysis (LDA) performed similarly to logistic regression but did not achieve the same level of sensitivity as the threshold-optimized logistic model.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1347   42
         1  749  162

               Accuracy : 0.6561
                 95% CI : (0.6363, 0.6755)
    No Information Rate : 0.9113
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1703

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.79412
            Specificity : 0.64265
         Pos Pred Value : 0.17783
         Neg Pred Value : 0.96976
             Prevalence : 0.08870
         Detection Rate : 0.07043
   Detection Prevalence : 0.39609
      Balanced Accuracy : 0.71839

       'Positive' Class : 1
```

How we gathered these findings were by first using data from the training and test sets, the proportion of people who are expected to purchase the book out of the 50,000 people in the mailing audience is calculated then storing this estimate in a column called 'newcnt'. We then used the model to estimate how many of the individuals in both the purchasing and non-purchasing groups would be predicted to buy the book called 'est_targets'.

The cost of sending mailers was calculated by multiplying the number of predicted buyers ("est_targets") by $0.65 (the cost of each mailer) and called 'mailercst'.

For those who are predicted to buy the book, the total cost of the books and overhead (calculated as $15 x 1.45) was estimated. These costs are only applied to those predicted to purchase the book ("newcnt" where Choice == 1), "purchcst" variable. The total revenue from book sales is calculated by multiplying the number of predicted buyers by $31.95 (the price of the book). 'Revenue' is only generated when Choice == 1, and 'Profit' is calculated by subtracting both the mailer and book purchase costs from the total revenue.

We see by summing up the values in the "profit" we get a total expected profit from the mailer campaign. We see the comparison results here:

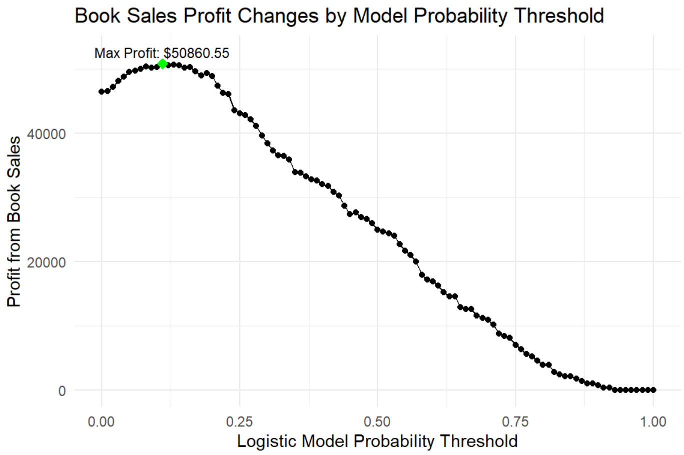
Model <chr>	**Profit** <dbl>
Naive	46488.80
LDA	44004.35
Logit	48917.50
SVM	42867.35

4 rows

One of the primary challenges in the dataset was the inherent class imbalance, with significantly more non-purchasers than purchasers. To address this, the dataset was balanced by oversampling the minority class (purchasers), which improved the performance of both logistic regression and SVM models, particularly in terms of sensitivity. Even when tested on the original unbalanced dataset, the sensitivity and specificity metrics for both models remained consistent, indicating that the models were robust against changes in class distribution.

The analysis also highlighted a trade-off between sensitivity and specificity. Improving sensitivity was crucial for identifying a greater proportion of potential purchasers, which is the primary goal of the direct mail campaign. However, this improvement came at the cost of specificity, meaning that some mailers would be sent to non-purchasers, resulting in false positives. Despite this, the increased sensitivity was considered an acceptable trade-off, as the cost of sending mailers to non-purchasers is relatively low compared to the revenue generated from correctly identified purchasers.

Finally, the profitability analysis showed that the logistic regression model with an optimized threshold provided the best balance between sensitivity and specificity, leading to the highest potential profit for the campaign. The SVM model, while strong in terms of specificity, identified fewer purchasers overall, limiting its potential revenue generation. This analysis demonstrated that balancing the dataset and fine-tuning the decision threshold were critical steps in maximizing the effectiveness and profitability of the direct mail campaign.



	threshold <dbl>	**profit** <dbl>
12	0.11	50860.55

1 row

In a similar study on "Binary Classification of Customer's Online Purchasing Behavior Using Machine Learning", the strength of logistic regression compared to other models was also found: "A comparative study of ten classifiers is presented in [18]. Their accuracy indicator, i.e., the area under the curve (AUC), highlighted logistic regression as the best classifier. Naive Bayes, neural network, and support vector machine classifiers followed as runners-up, while decision tree-based classifiers tended to underperform."

7 Conclusion

In conclusion, the logistic regression model ultimately performed the best in predicting customer purchases for the mailer campaign.

By iterating through different decision thresholds, we identified the optimal threshold that maximized profit, adjusting the threshold down to 0.2 for the logistic model. While the naive method yielded higher revenue by reaching all potential buyers, using a predictive model like logistic regression or LDA with an optimized threshold helped balance mailer costs and capture more actual purchasers. This approach resulted in higher overall profitability by efficiently targeting customers most likely to buy the book, demonstrating that a carefully tuned predictive model provides a more cost-effective solution than the naive approach.

8 Appendix

```
pacman::p_load(MASS, tidyverse, e1071, here, readxl, skimr, corrplot, patchwork)
```

```
raw_train <- read_xlsx(here('Case Study 2', 'BBBC-Train.xlsx'))
```

```
raw_test <- read_xlsx(here('Case Study 2', 'BBBC-Test.xlsx'))
```

```
str(raw_train)
```

```
anyNA(raw_train)
```

```
skim(raw_train)
```

```
train1 <- raw_train %>%
  select(-Observation) %>%
  mutate(
    Choice = as.factor(Choice),
    Gender = as.factor(Gender)
    )
test1 <- raw_test %>%
  select(-Observation) %>%
  mutate(
    Choice = as.factor(Choice),
    Gender = as.factor(Gender)
    )

combined <- rbind(train1, test1)
```

```
combined %>% select_if(is.numeric) %>% cor() %>% corrplot(method = 'number')
```

```
combined %>% mutate(
  Gender = ifelse(Gender == 0, 'Female', 'Male'),
  Choice = ifelse(Choice == 0, 'Non-purchase', 'purchase')
  ) %>%
  ggplot(aes(x = Gender, fill = Choice)) +
  geom_bar() + ggtitle('Gender vs Purchase')
```

```
combox <- lapply(colnames(select_if(combined, is.numeric)),
       function(col) {
        ggplot(combined,
                aes(y = .data[[col]], x = .data$Choice)) + geom_boxplot() + ggtitle(col)
       }
)

combox[[1]] + combox[[2]]
combox[[3]] + combox[[4]]
```

Boxplots are a bad visual for the variables starting with "P_". Contingency table below helps, but maybe grouped bar charts for those too? I have those below, let me know what yall think.

```
combbar <- lapply(colnames(select_if(combined, startsWith(names(combined), 'P_'))),
       function(col) {
        ggplot(combined,
                aes(x = .data[[col]], fill = .data$Choice)) + geom_bar(position = 'dodge') +
           ggtitle(col) +
           theme(legend.position = c(0.8,0.8), legend.background = element_blank())
       }
)

combbar[[1]] + combbar[[2]]
combbar[[3]] + combbar[[4]]
combbar[[5]]
```

```
combined %>% group_by(Choice, P_Art) %>%
  summarize(cnt = n()) %>% pivot_wider(id_cols = P_Art, names_from = Choice, values_from = cnt)
```

### Linear Regression

```
#Creating data frames that don't make Choice a factor only to run Linear Regression. This is done to show it isn't the appropriate model.
linregtrain1 <- raw_train %>%
  select(-Observation) %>%
  mutate(
    Gender = as.factor(Gender)
    )
linregtest1 <- raw_test %>%
  select(-Observation) %>%
  mutate(
    Gender = as.factor(Gender)
    )

#combined <- rbind(linregtrain1, linregtest1)
resultsLinReg <- lm(Choice ~ ., data = linregtrain1)
summary(resultsLinReg)



predict(resultsLinReg, linregtest1, type = 'response') %>% summary()
predict(resultsLinReg, linregtest1, type = 'response') %>% head()



### Logistic Regression


set.seed(321)
logfit <- step(glm(Choice ~ .,
                   data = train1, family = binomial),
               direction = "backward", trace = 0)

summary(logfit)


car::vif(logfit)



set.seed(321)
logfit2 <- step(glm(Choice ~ . -Last_purchase,
                   data = train1, family = binomial),
               direction = "backward", trace = 0)



car::vif(logfit2)


set.seed(321)
logfit3 <- step(glm(Choice ~ . -Last_purchase -First_purchase,
                   data = train1, family = binomial),
               direction = "backward", trace = 0)



car::vif(logfit3)



predprob_log <- predict(logfit3, newdata = test1, type = "response")
pr_class_log <- ifelse(predprob_log > 0.2, 1, 0)

log_CM_unbal <- caret::confusionMatrix(as.factor(pr_class_log), as.factor(test1$Choice), positive = '1')
log_CM_unbal


### Linear Discriminant Analysis (LDA)

set.seed(321)
ldafit <- lda(Choice ~ ., data = train1)

ldafit



pr_class_lda <- predict(ldafit, test1)

lda_CM_unbal <- caret::confusionMatrix(as.factor(pr_class_lda$class), as.factor(test1$Choice), positive = "1")
lda_CM_unbal


### Support Vector Machines (SVM)



set.seed(321)
form1 <- Choice ~ .

# TAKES A LONG TIME TO RUN!
svmtune <- tune.svm(form1, data = train1, gamma = seq(.01,.1, by = .01), cost = seq(.1, 1, by = .1))
```

```
best_params <- svmtune$best.parameters
print(best_params)
#best parameters: gamma 0.02, cost 0.5


svmtune$performances


svmfit <- svm(formula = form1, data = train1, gamma = best_params$gamma, cost = best_params$cost)
summary(svmfit)


svmpredict <- predict(svmfit, test1, type = 'response')
caret::confusionMatrix(svmpredict, test1$Choice, positive = '1')


### Balancing Dataset


set.seed(321)
trn_art = train1 %>% filter(Choice == '1')
trn_no_art = train1 %>% filter(Choice == '0')

tst_art = test1 %>% filter(Choice == '1')
tst_no_art = test1 %>% filter(Choice == '0')

sample_no_art_trn = sample_n(trn_no_art, nrow(trn_art))
train_bal = rbind(sample_no_art_trn,trn_art)

sample_no_art_tst = sample_n(tst_no_art, nrow(tst_art))
test_bal = rbind(sample_no_art_tst,tst_art)


### Logistic Regression (Balanced)


set.seed(321)
logfit_bal <- step(glm(Choice ~ .,
                   data = train_bal, family = binomial),
               direction = "both", trace = 0)


summary(logfit_bal)


predprob_log_bal <- predict(logfit_bal, newdata = test_bal, type = "response")
pr_class_log_bal <- ifelse(predprob_log_bal > 0.5, 1, 0)

log_CM_unbal_bal <- caret::confusionMatrix(as.factor(pr_class_log_bal), as.factor(test_bal$Choice), positive = '1')
log_CM_unbal_bal


predprob_log_imbal <- predict(logfit_bal, newdata = test1, type = "response")
pr_class_log_imbal <- ifelse(predprob_log_imbal > 0.22, 1, 0)

log_CM_imbal <- caret::confusionMatrix(as.factor(pr_class_log_imbal), as.factor(test1$Choice), positive = '1')
log_CM_imbal


### Linear Discriminant Analysis (LDA) (Balanced)


set.seed(321)
ldafit_bal <- lda(Choice ~ ., data = train_bal)

ldafit_bal


pr_class_lda_bal <- predict(ldafit_bal, test_bal)

lda_CM_unbal_bal <- caret::confusionMatrix(as.factor(pr_class_lda_bal$class), as.factor(test_bal$Choice), positive = "1")
lda_CM_unbal_bal


pr_class_lda_imbal <- predict(ldafit_bal, test1)
```

```
lda_CM_imbal <- caret::confusionMatrix(as.factor(pr_class_lda_imbal$class), as.factor(test1$Choice), positive = "1")
lda_CM_imbal
```

```
### Support Vector Machines (SVM) (Balanced)
```

```
set.seed(321)
svmtune_bal <- tune.svm(form1, data = train_bal, gamma = seq(.005,.1, by = .005), cost = seq(.1, 1.5, by = .05))
```

```
best_params_bal <- svmtune_bal$best.parameters
print(best_params_bal)
#best parameters: gamma 0.01, cost 1
```

```
svmfit_bal <- svm(formula = form1, data = train_bal, gamma = best_params_bal$gamma, cost = best_params_bal$cost)
summary(svmfit_bal)
```

```
svmpredict_bal <- predict(svmfit_bal, test_bal, type = 'response')
caret::confusionMatrix(svmpredict_bal, test_bal$Choice, positive = '1')
```

```
svmpredict_imbal <- predict(svmfit_bal, test1, type = 'response')
SVM_CM_imbal <- caret::confusionMatrix(svmpredict_imbal, test1$Choice, positive = '1')
SVM_CM_imbal
#caret::confusionMatrix(svmpredict_imbal, test1$Choice, positive = '1')
```

```
summary_table_svm <- combined %>% group_by(Choice) %>%
  summarize(percent = n()/nrow(combined),
            newcnt = round(percent * 50000)) %>% as.data.frame() %>%
  mutate(
    est_targets = ifelse(
      Choice == 0, round(newcnt*(1-SVM_CM_imbal$byClass[["Specificity"]])), round(newcnt*(SVM_CM_imbal$byClass[["Sensitivity"]]))
      ),
    mailercst = est_targets * 0.65,
    purchcst = ifelse(Choice == 1, 15 * 1.45 * est_targets, 0),
    revenue = ifelse(Choice == 1, 31.95 * est_targets, 0),
    profit = revenue - purchcst - mailercst
    )
summary_table_svm
```

```
sum(summary_table_svm$profit)
```

```
summ_tab_fun <- function(base_data, hcount, CM) {
  base_data %>% group_by(Choice) %>%
  summarize(percent = n()/nrow(base_data),
            newcnt = round(percent * hcount)) %>% as.data.frame() %>%
  mutate(
    est_targets = ifelse(
      Choice == 0, round(newcnt*(1-CM$byClass[["Specificity"]])), round(newcnt*(CM$byClass[["Sensitivity"]]))
      ),
    mailercst = est_targets * 0.65,
    purchcst = ifelse(Choice == 1, 15 * 1.45 * est_targets, 0),
    revenue = ifelse(Choice == 1, 31.95 * est_targets, 0),
    profit = revenue - purchcst - mailercst
    )
}
```

```
summary_table_log <- summ_tab_fun(combined, 50000, log_CM_unbal)
summary_table_log
```

```
summary_table_lda <- summ_tab_fun(combined, 50000, lda_CM_imbal)
summary_table_lda
```

```
naive_table <- combined %>% group_by(Choice) %>%
  summarize(percent = n()/nrow(combined),
            newcnt = round(percent * 50000)) %>% as.data.frame() %>%
  mutate(
    mailercst = newcnt * 0.65,
    purchcst = ifelse(Choice == 1, 15 * 1.45 * newcnt, 0),
    revenue = ifelse(Choice == 1, 31.95 * newcnt, 0),
    profit = revenue - purchcst - mailercst
    )
naive_table


data.frame(Model = c('Naive', 'LDA', 'Logit', 'SVM'),
  Profit = c(
    sum(naive_table$profit),
    sum(summary_table_lda$profit),
    sum(summary_table_log$profit),
    sum(summary_table_svm$profit)
    )
  )

.


thresh <- data.frame(threshold = -0.01, profit = 0)
for (i in seq(0, 1, by = 0.01)) {
  preds <- ifelse(predprob_log >= i, 1, 0)
  CM_for <- caret::confusionMatrix(as.factor(preds), as.factor(test1$Choice), positive = '1')
  summ_for <- summ_tab_fun(combined, 50000, CM_for)
  thresh = rbind(thresh, data.frame(threshold = i, profit = sum(summ_for$profit)))
}

thresh <- thresh %>% filter(threshold >= 0)
thresh

thresh %>% ggplot(aes(x = threshold, y = profit)) +
  geom_line() +
  geom_point() +
  annotate('text', x = thresh[which(thresh$profit == max(thresh$profit)),]$threshold, y = thresh[which(thresh$profit == max(thresh$profit)),]$profit + 1800, label =
paste0('Max Profit: $', thresh[which(thresh$profit == max(thresh$profit)),]$profit), size = 3) +
  annotate('point', x = thresh[which(thresh$profit == max(thresh$profit)),]$threshold, y = thresh[which(thresh$profit == max(thresh$profit)),]$profit, color = 'green',
shape = 'diamond', size = 3) +
  theme_minimal() +
  ggtitle('Book Sales Profit Changes by Model Probability Threshold') +
  xlab('Logistic Model Probability Threshold') +
  ylab('Profit from Book Sales')


thresh[which(thresh$profit == max(thresh$profit)),]



predslog_best <- ifelse(predprob_log >= thresh[which(thresh$profit == max(thresh$profit)),]$threshold, 1, 0)
  CMlog_best <- caret::confusionMatrix(as.factor(predslog_best), as.factor(test1$Choice), positive = '1')
CMlog_best
```