Exploring a Fictional Marketing dataset that Causes Quasi-Seperation

Owen Waters

1. Abstract

When evaluating a marketing campaign, several factors contribute to what constitutes a successful campaign, depending on the business context, products sold, inventory availability, demographics, and effective reach of those demographics. This study tries to answer some of these questions by looking at online marketing conversion rates. Specifically, what factors contribute to the success of marketing campaigns with a conversion rate of 10% or greater (1 vs 0 below 10% conversion). Throughout this study, I used many techniques, but the main method was a General Linear Model (GLM) as the outcome is binary 0/1. Trimming down the model parameters I got down to the Customer segment, Language of the advert, and level of engagement being the key factors from this fictional dataset. So, should you find yourself giving a website personal information or accepting cookies with these constraints, you will have accurate targeted ads.

2. Introduction

Digital advertising has become an essential driver of business success, with online sales accounting for 19% of all sales in 2020- a number that continues to rise (Forbes, 2022). Along with targeted advertising, revolutionizing the online market with personalized ads having a 91% more likely to have a positive response with the consumers (Forbes, 2022). This study investigates how campaign characteristics of type, target audience, customer segment, ad source, and ad location predict the likelihood of achieving a conversion rate of 10% or greater. If your campaign is converting 10% of viewership into clicks or a purchase just for having your ad on a specific site or location for your key demographic, it would be an easy decision to send more advertising money to that segment (Barry, 2023). It is clear why it would be necessary to do an EDA of the factors into a successful online campaign to keep your business afloat in an online space where sales are happening more often.

Digital Marketing campaigns can swiftly fall apart with a lack of direction and throwing money at multiple sources to see what sticks. Commonly, the failure points of marketing campaigns are in the budget, unclear objectives or calls to action, lack of clarity for the product, barriers for the customer to successfully convert or the campaign just does not stand out enough compared to its competitors (Forbes, 2021). Such mistakes cost the company thousands in investment for the wasted spend to return on investment or conversions. This could also lead to poor customer engagement, or even worse, poor customer experience, which hinders your company’s growth. Having the ability to measure successes resourcefully and accurately in your digital campaigns is just as important as your ability to put together an unconditionally successful targeted ad. The study operationalizes campaign success as a binary variable, where campaigns with conversion rates of 10% or greater are labeled as ‘successful’ (1), while those below 10% are ‘unsuccessful’ (0). This straightforward binary classification helps to facilitate a quick view into the high-impact advertising strategies with an easy-to-interpret and parsable model.

The objective of this study is to identify the key characteristics of a successful digital campaign. The study will assess how campaign type, target audience, customer segment, ad source, and ad location influence the likelihood of achieving a conversion rate of 10% or higher through EDA and binary classification models. Using a binary classification model while doing my Exploratory Data Analysis helps solidify the easy insights and creates clear paths for data-driven decision-making when looking

at a successful campaign. These methods will lead to a robust statistical plan that helps to optimize an advertising strategy and maximize the return on investment of a digital campaign.

As e-commerce grows to eventually dominate the market, understanding the drivers of a successful digital campaign has never been more important. By identifying these common factors that distinguish a successful campaign from an unsuccessful one, this research will provide businesses with actionable strategies to optimize their own digital campaigns to reduce spend, increase customer engagement, and drive higher conversion rates per dollar used in advertising. Particularly in the digital world, where actionable, data-driven insights can push you ahead of fierce competition and reach a broader customer base for sustained growth in e-commerce.

3. Methods

To begin my investigation into this data set, I first changed all the integers that could be treated as categorical to as.factor. Then ordered Engagement_Score and duration as they would be ordinal variables with meaningful increments. I then summarized the value sets to see if there was anything out of alignment with the dataset. It seemed like there was decent variance between the quantile values, means, and standard deviations. But once I started looking over the graphing solutions in both strip charts for the factors and boxplot/scatterplot combos for the numeric values, I saw something interesting in the data. As this was a fictional data set created to resemble real-life data, there was an exact split of the treatments to my value of success in the Binary outcome >=10% conversion as success and <10% as a failure. This occurred right at 39.5% of the time, but the success values had an equal split in each treatment. Leaving little room for predictability in the dataset as the values already matched the outcome. Causing a semi-quasi separation in the data.

To further investigate this split in the data from the Binary outcome, I tabulated the data in contingency tables and checked the correlation matrix of the numeric plots. Finding no significant correlation between the numeric values and almost equal outcomes per treatment in the factors of the categorical explanatory variables. With 200,000 datapoints the variance between the values I created an all-factor model prediction first to find any available significant predictors of a successful digital marketing campaign.

The First complete categorical factor model showed significance for one value below an alpha <0.05 threshold at Customer_Segment Tech Enthusiasts, and two other borderline significant factors with Engagement at a score 5 and language in Spanish- each with p-values = 0.0428, 0.0781, 0.0905, respectively. I ran a stepwise model build to see if I could reorder the model features and squeeze more explained variance or borderline significant values. However, even when choosing the model feature order, the AIC of the first = 268297.2. The best AIC = 268248.2 was for a model only showing outcome ~ language. Finally, the intercept model AIC = 268248 was the lowest. This model reordering did not tell me much other than that the intercept model was a better predictor than the full factor explanatory model.

To better understand the possibility of multicollinearity, I used the gvif method from the vif function in the car package. Only to find that the explanatory variables all had a VIF value of ~1 or were not likely to be multicollinear or dependent on each other. So, all data points are valid, but there is a split in the data, causing a separation without multicollinearity going on. I then parsed the numeric values into their own data frame and tried to create a glm with a numeric to binary prediction.

Once I separated the numeric values into their own data frame and created a glm model to predict the outcomes of success or >=10% conversion. I found none of the explanatory variables in the numeric or categorical/ factor type had a strong significance with the success outcome. After the numeric model had failed to show significance in predictability, I ran an anova of the numeric vs the

category/factor model to find a deviance of 30.685 and a degree of freedom 37. Using a chi^2 distribution with 37 degrees of freedom, we get a p-val = 0.7856 or >0.05 cutoff, where there is no evidence to support rejecting the null that the category model is outperforming the numeric model in predictive power. Finally, I created a saturated model with the full numeric/category mix of explanatory variables. The saturated model had comparable results to the category mode,l where language, engagement score were both borderline significant, and customer segment was significant. Using this knowledge, I began looking for interaction terms to include the borderline significant categories and the numeric model.

To find interaction effects within the model, I tried interaction plots, but they were too crowded to see any trends. So, I moved on to a Lasso estimation for interactions that would have conservative values for the coefficients that would be required to include in an interaction model summary. Finding the only interactions worth noting were the intercept and success. I wanted to dig deeper and used an elastic net in the glmnet package that would show ‘near zero’ coefficient values and was more liberal in its estimation of effects, as our predictive power had been too weak to capture any associations outside of the segment so far. Using the elastic net, 3 variables not related to the outcome or intercept that had non-zero coefficients: Acquisition_Cost:ROI = -9.085672e-08, ROI: Clicks = -1.360484e-09, and ROI: Impressions = -7.545671e-09. While the coefficients were not zero, they would hardly have enough effect to be considered significant. Trying to use the interactions in a model setup led to a non-convergence as the factors were not strong enough to predict the separation and made too complex to explain the successful outcome. I then created the final model selection for this poorly fitted data.

Finding the blend of the final model and order of features to predict the data as best as possible I used the ROC curve and AUC to find the best predictor from the fully saturated model, the final ordered model, and the y-intercept that had been beating the other model in AIC from a complexity standpoint.

4. Results

After comparing the models, it looked like the intercept model had the correct interpretation of the data. The fully saturated model explained a fraction of a percentage more of the variability but was overly complex with ~ 14 features, from numeric to factor to predict slightly better than the intercept. The intercept model predicted at .5 or 50% of the correct successful outcomes. While the final ordered model was a trimmed version of the fully saturated model. The final order model was also predicted at a .5 with significantly less complexity than the fully saturated model. However, with no difference between the final ordered model and the intercept, the intercept would be a better baseline prediction than our glm’s, no matter how they are set up.

5. Discussion

The results were fascinating in that there were not very many useful features in the data set to pick up the arbitrary successful outcome. Again, this is a fictional data set that is generated by a distribution in an r/python package, as the data landed directly separated in the outcome to the split of the treatment factors. Even so, creating a model using the numeric features that would have a continuous component of variance to the key significant categorical features of language, customer segment, and engagement score allowed the model to predict just as well as the intercept line of best fit to the total model. Had the data not been so evenly split over the factors, I think the significance in these key features would have been more pronounced.

6. Conclusions

Finding significant features to predict whether a digital campaign was successful or not by a conversion rate of >= 10% using a fictional dataset did not prove to be conclusive. Should there be a company that is looking to improve their digital market space cap from the findings in this data it could be said that they would have the most success if they are a tech company, that has good clicks and impressions on their advertisement from an engagement standpoint, perhaps because they cater to a special segment of the market by cornering language. If your product is definitely going to be related to Latino culture, use Spanish in the advert. If you are a tech company wanting to get the word out about your product or hiring through digital marketing campaigns, you would be in the best segment to get a higher conversion rate as people are already using tech and the computer. and if you want to maintain a high conversion rate on your advertisements, getting as much engagement as possible, even to a few clicks, will be a boon in your success as a marketer.

If you are a customer, try to avoid the targeted ads, however. It would look like trying to avoid ads in your native language, being guarded about the info you provide for your hobbies and interests, and avoiding clicking on any ads, as necessary. would afford you the benefit of your targeted ads being less accurate in guessing your likes and dislikes. Now, with TikTok having dwell time, and Instagram showing you a feed for you according to your likes. It would seem that any interaction can create an opportunity to hit one of their funnels for a sale of an ad. Being guarded in your online presence, having a VPN, and using more anonymous data through cookies and settings are about as good as it can get to prevent falling into a digital marketing scheme.

7. References

8. https://www.webfx.com/blog/marketing/what-is-a-good-conversion-rate/
9. https://www.forbes.com/councils/forbestechcouncil/2022/02/24/the-truth-in-user-privacy-and-targeted-ads/
3. https://www.forbes.com/councils/forbesagencycouncil/2021/08/26/8-common-reasons-why-marketing-campaigns-fail/
4. https://stats.oarc.ucla.edu/other/mult-pkg/faq/general/faqwhat-is-complete-or-quasi-complete-separation-in-logistic-regression-and-what-are-some-strategies-to-deal-with-the-issue/
5. https://www.forbes.com/sites/paultalbot/2020/03/14/inside-targets-marketing-strategy/
6. https://www.consumerreports.org/electronics-computers/privacy/how-to-control-annoying-or-upsetting-ads-online-a1056618772/

7. Appendices (including your code if not using something like R Markdown)

```
library(tidyverse)

library(skimr)
library(ggplot2)
library(readr)
digi_data <- read.csv("market_data.csv")
str(digi_data)

digi_data$success <- ifelse(digi_data$Conversion_Rate >= 0.1, 1, 0)
digi_data <- digi_data %>% select(-c(Conversion_Rate, Date, Campaign_ID))
digi_data$Campaign_Type <- as.factor(digi_data$Campaign_Type)
digi_data$Target_Audience <- as.factor(digi_data$Target_Audience)
digi_data$Duration <- as.factor(digi_data$Duration)
digi_data$Channel_Used <- as.factor(digi_data$Channel_Used)
```

```r
digi_data$Engagement_Score <- as.factor(digi_data$Engagement_Score)
digi_data$Customer_Segment <- as.factor(digi_data$Customer_Segment)
digi_data$Acquisition_Cost <- parse_number(digi_data$Acquisition_Cost)
digi_data$Location <- as.factor(digi_data$Location)
digi_data$Language <- as.factor(digi_data$Language)
digi_data$Company <- as.factor(digi_data$Company)
str(digi_data)
```

```
## 'data.frame':    200000 obs. of  14 variables:
##  $ Company         : Factor w/ 5 levels "Alpha Innovations",..: 3 4 1 2 4 2 4 2 1 5 ..
##  $ Campaign_Type   : Factor w/ 5 levels "Display","Email",..: 2 2 3 1 2 1 2 4 5 2 ...
##  $ Target_Audience : Factor w/ 5 levels "All Ages","Men 18-24",..: 2 5 3 1 3 1 5 2 5 5
##  $ Duration        : Factor w/ 4 levels "15 days","30 days",..: 2 4 2 4 1 1 4 3 1 1 ..
##  $ Channel_Used    : Factor w/ 6 levels "Email","Facebook",..: 3 3 6 6 6 4 5 3 2 4 ...
##  $ Acquisition_Cost: num  16174 11566 10200 12724 16452 ...
##  $ ROI             : num  6.29 5.61 7.18 5.55 6.5 4.36 2.86 5.55 6.73 3.78 ...
##  $ Location        : Factor w/ 5 levels "Chicago","Houston",..: 1 5 3 4 3 5 3 3 1 3 ..
##  $ Language        : Factor w/ 5 levels "English","French",..: 5 3 2 4 4 3 5 4 3 1 ...
##  $ Clicks          : int  506 116 584 217 379 100 817 624 861 642 ...
##  $ Impressions     : int  1922 7523 7698 1820 4201 1643 8749 7854 1754 3856 ...
##  $ Engagement_Score: Factor w/ 10 levels "1","2","3","4",..: 6 7 1 7 3 1 10 7 6 3 ...
##  $ Customer_Segment: Factor w/ 5 levels "Fashionistas",..: 3 1 4 3 3 2 5 4 5 5 ...
##  $ success         : num  0 1 0 1 0 0 1 0 0 0 ...
```

```r
levels(digi_data$Duration)
```

```r
levels(digi_data$Engagement_Score)
```

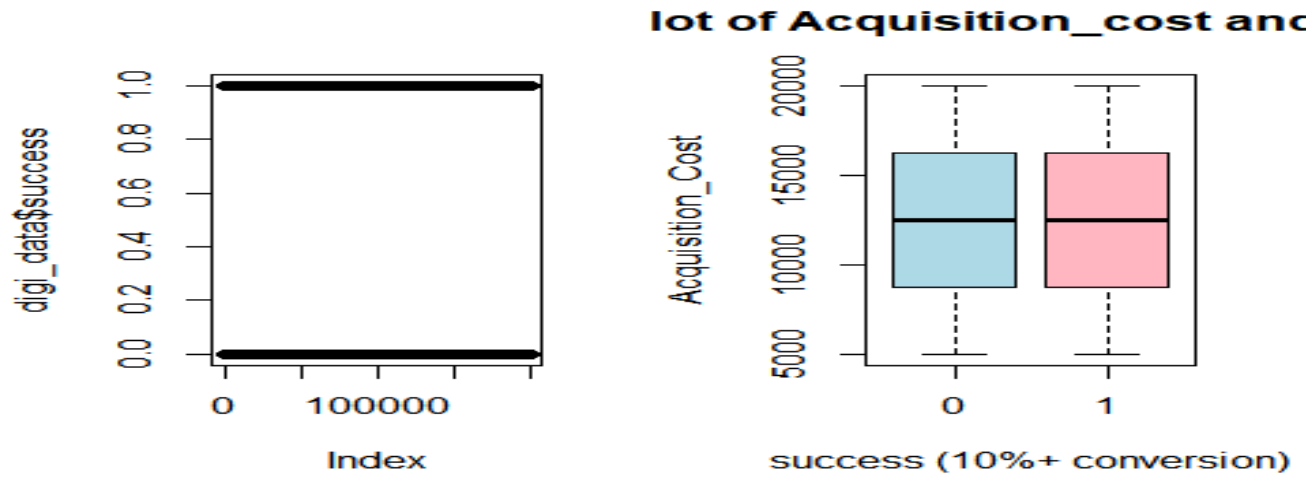
```r
digi_data$Duration <- factor(digi_data$Duration,
                             levels = c("15 days", "30 days", "45 days", "60 days"),
                             ordered = TRUE)
digi_data$Engagement_Score <- factor(digi_data$Engagement_Score,
                            levels = c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10"),
                                     ordered = TRUE)
skim(digi_data)
```

```r
summary(digi_data)
```

```
##                 Company           Campaign_Type      Target_Audience
##  Alpha Innovations  :40051   Display     :39987   All Ages   :40019
##  DataTech Solutions :40012   Email       :39870   Men 18-24  :40258
##  Innovate Industries:39709   Influencer  :40169   Men 25-34  :40023
##  NexGen Systems     :39991   Search      :40157   Women 25-34:40013
##  TechCorp           :40237   Social Media:39817   Women 35-44:39687
##
##     Duration         Channel_Used   Acquisition_Cost      ROI
##  15 days:49779   Email     :33599   Min.   : 5000    Min.   :2.000
##  30 days:50255   Facebook  :32819   1st Qu.: 8740    1st Qu.:3.500
##  45 days:50100   Google Ads:33438   Median :12496    Median :5.010
##  60 days:49866   Instagram :33392   Mean   :12504    Mean   :5.002
##                  Website   :33360   3rd Qu.:16264    3rd Qu.:6.510
##                  YouTube   :33392   Max.   :20000    Max.   :8.000
##         Location         Language         Clicks        Impressions
##  Chicago    :40010   English :39896   Min.   : 100.0   Min.   : 1000
##  Houston    :39750   French  :39764   1st Qu.: 325.0   1st Qu.: 3266
##  Los Angeles:39947   German  :39983   Median : 550.0   Median : 5518
##  Miami      :40269   Mandarin:40255   Mean   : 549.8   Mean   : 5507
```

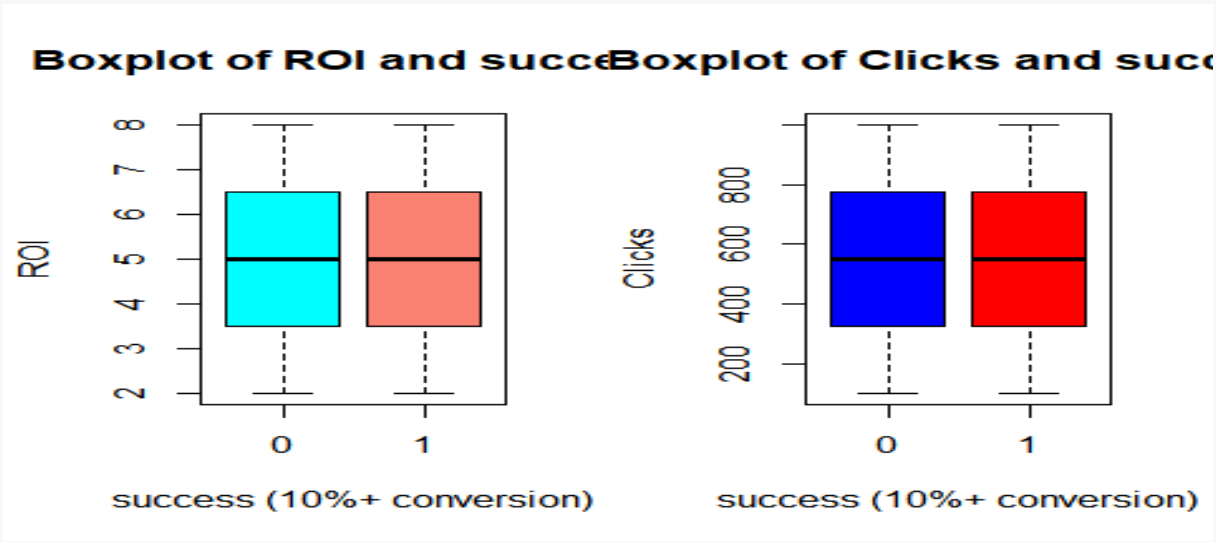
```
##  New York   :40024   Spanish :40102   3rd Qu.: 775.0   3rd Qu.: 7753
##                                       Max.   :1000.0   Max.   :10000
##  Engagement_Score           Customer_Segment    success
##  4      :20141    Fashionistas       :39742   Min.   :0.0000
##  2      :20113    Foodies            :40208   1st Qu.:0.0000
##  9      :20106    Health & Wellness  :39888   Median :0.0000
##  1      :20027    Outdoor Adventurers:40011   Mean   :0.3943
##  5      :20023    Tech Enthusiasts   :40151   3rd Qu.:1.0000
##  3      :19947                                Max.   :1.0000
##  (Other):79643
```

```r
par(mfrow = c(1, 2))
plot(digi_data$success)
boxplot(digi_data$Acquisition_Cost ~ digi_data$success,
        main = "Boxplot of Acquisition_cost and success",
        xlab = "success (10%+ conversion)", ylab = "Acquisition_Cost",
        col = c("lightblue", "lightpink"))
```




```r
boxplot(digi_data$ROI ~ digi_data$success,
        main = "Boxplot of ROI and success",
        xlab = "success (10%+ conversion)", ylab = "ROI",
        col = c("cyan", "salmon"))
boxplot(digi_data$Clicks ~ digi_data$success,
        main = "Boxplot of Clicks and success",
        xlab = "success (10%+ conversion)", ylab = "Clicks",
```

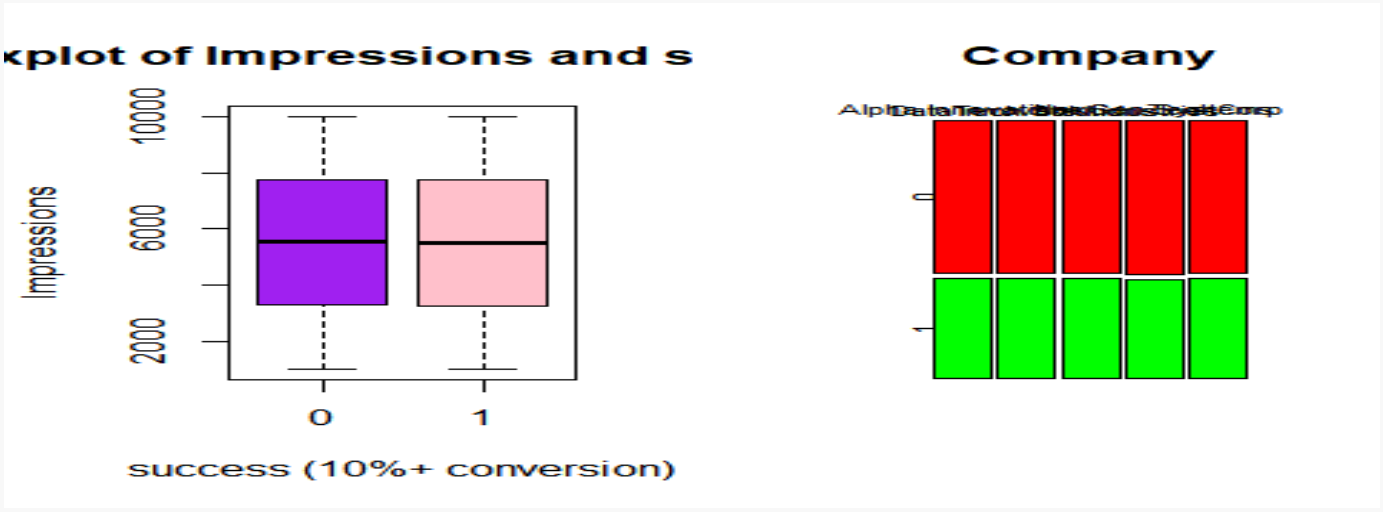
```
        col = c("blue", "red"))
```

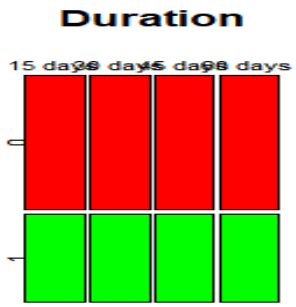


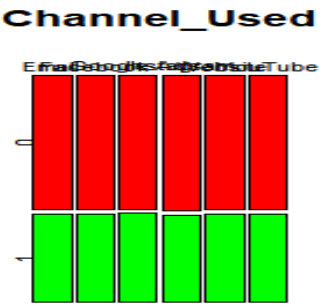

```
boxplot(digi_data$Impressions ~ digi_data$success,
        main = "Boxplot of Impressions and success",
        xlab = "success (10%+ conversion)", ylab = "Impressions",
        col = c("purple", "pink"))
for(var_name in names(digi_data)) {
  if(is.factor(digi_data[[var_name]]) & var_name != "success") {
    mosaicplot(table(digi_data[[var_name]], digi_data$success),
               main = var_name,
               col = c("red", "green"),
               xlab = "",
               ylab = "")}}
```

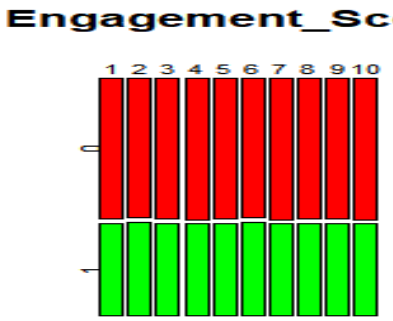

Campaign_Type
Social Media
0
1
Target_Audience
35–44
0
1

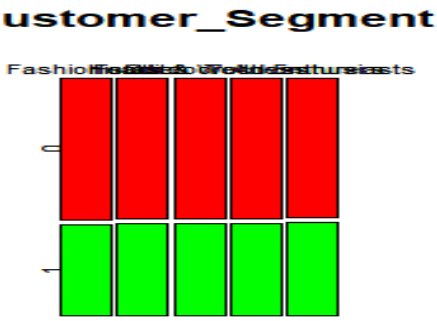

Location
York
0
1
Language
Spanish
0
1


xplot of Impressions and s
Impressions
10000
6000
2000
0
1
success (10%+ conversion)
Company
0
1

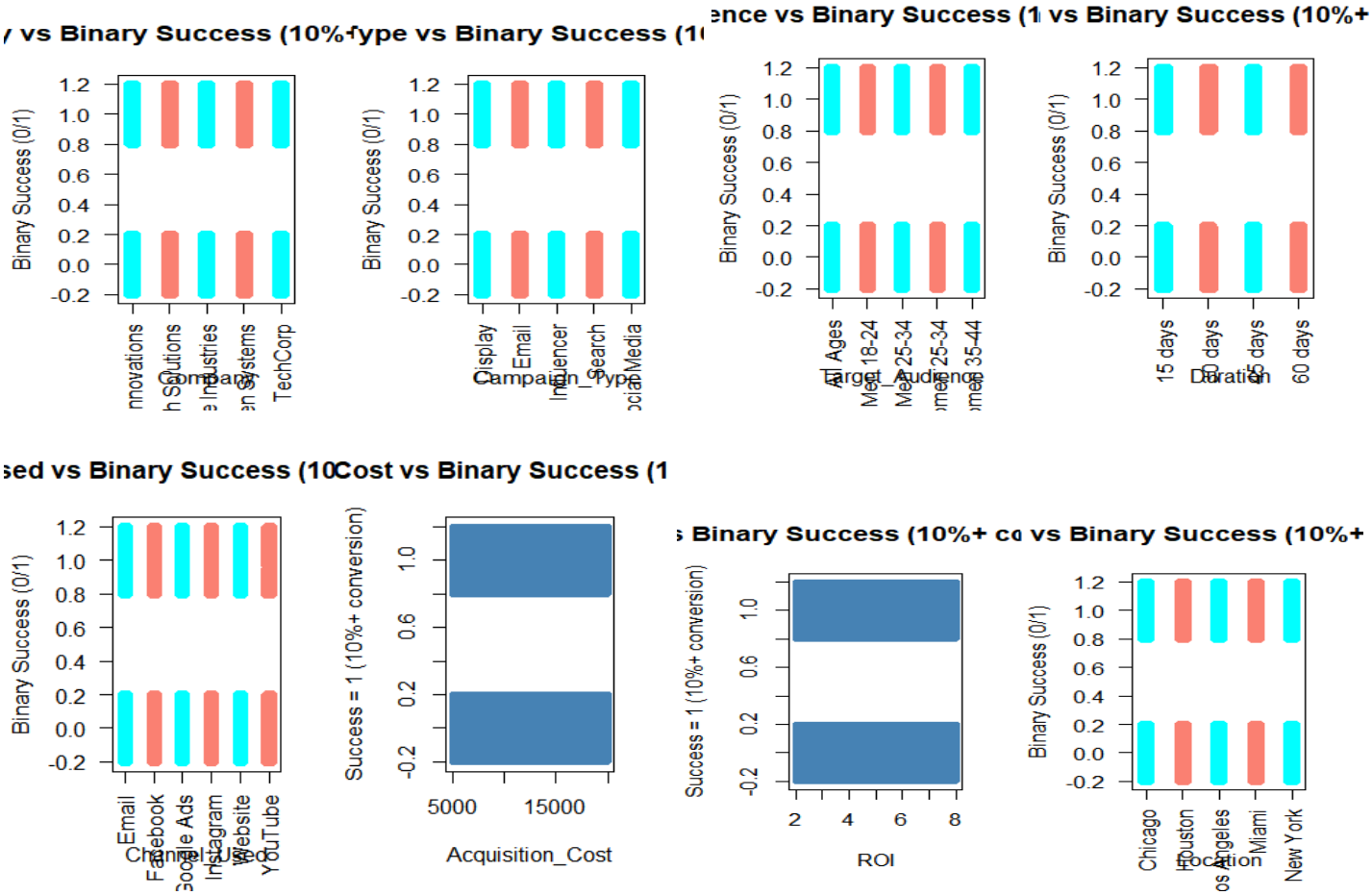


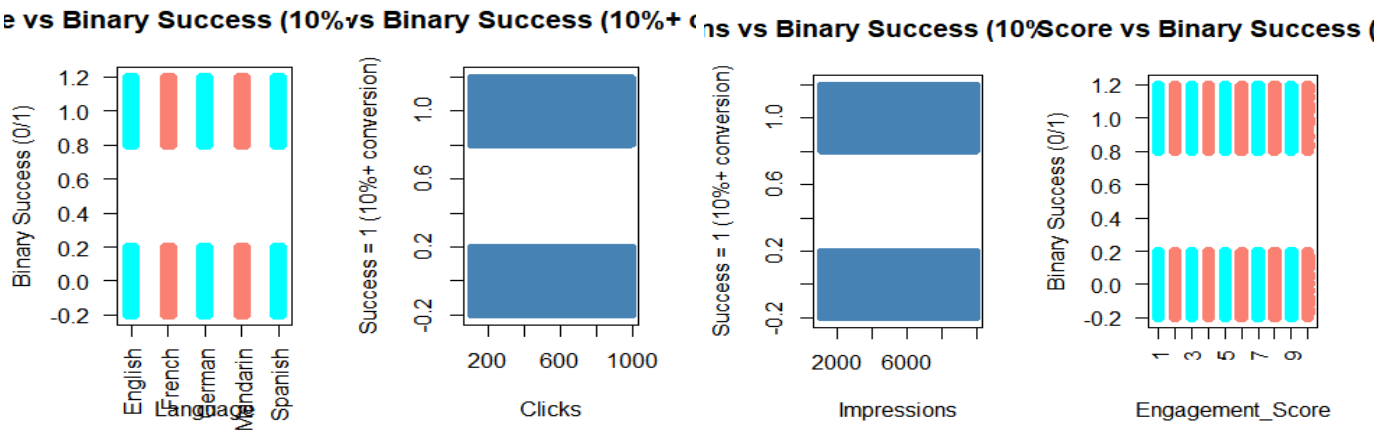


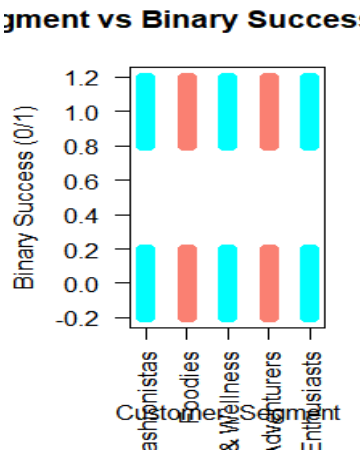





```r
for (var_name in names(digi_data)) {
  if (var_name != "success") {
    if (is.numeric(digi_data[[var_name]])) {
      plot(
        digi_data[[var_name]],
        jitter(digi_data$success),
        main = paste(var_name, "vs Binary Success (10%+ conversion)"),
        col = "steelblue",
        xlab = var_name,
        ylab = "Success = 1 (10%+ conversion)",
        pch = 19,
        cex = 0.7
      )}
    else if (is.factor(digi_data[[var_name]])) {
      stripchart(
        jitter(digi_data$success) ~ digi_data[[var_name]],
        main = paste(var_name, "vs Binary Success (10%+ conversion)"),
        xlab = var_name,
        ylab = "Binary Success (0/1)",
        vertical = TRUE,
        method = "jitter",
        col = c("cyan", "salmon"),
        pch = 19,
        las = 2
      )
    }}}
```


/ vs Binary Success (10%+
ype vs Binary Success (1
ence vs Binary Success (1
vs Binary Success (10%+
Binary Success (0/1)
Company
Campaign_Type
Target_Audience
Duration
sed vs Binary Success (10
Cost vs Binary Success (1
Binary Success (10%+ co
vs Binary Success (10%+
Success = 1 (10%+ conversion)
Channel_Used
Acquisition_Cost
ROI
Location
Email
Facebook
Google Ads
Instagram
Website
YOUTube
5000
15000
2
4
6
8
Chicago
Houston
Miami
New York
15 days
60 days






```
table(digi_data$Target_Audience, digi_data$success)

table(digi_data$Duration, digi_data$success)

table(digi_data$Channel_Used, digi_data$success)

table(digi_data$Engagement_Score, digi_data$success)
##          0     1
##   1  12142  7885
##   2  12056  8057
##   3  12066  7881
##   4  12249  7892
##   5  12168  7855
##   6  11938  7944
##   7  12129  7804
##   8  12074  7870
##   9  12227  7879
##   10 12099  7785

table(digi_data$Customer_Segment, digi_data$success)

table(digi_data$Location, digi_data$success)

table(digi_data$Language, digi_data$success)

table(digi_data$Company, digi_data$success)

table(digi_data$Campaign_Type, digi_data$success)
```

```r
numeric_vars <- digi_data %>% select_if(is.numeric)
cor_matrix <- cor(numeric_vars, method = "spearman")
cor_matrix

cate_var <- digi_data %>% select_if(is.factor)
cate_var$success <- digi_data$success
cate_mod <- glm(success ~ ., data = cate_var, family = binomial)
summary(cate_mod)
## glm(formula = success ~ ., family = binomial, data = cate_var)

anova(cate_mod)

step_cate <- step(cate_mod, direction = "both")

## Start:  AIC=268297.2
## success ~ Company + Campaign_Type + Target_Audience + Duration +
##     Channel_Used + Location + Language + Engagement_Score + Customer_Segment
##                     Df Deviance    AIC
## - Engagement_Score  9   268221 268287
## - Channel_Used      5   268216 268290
## - Location          4   268214 268290
## - Target_Audience   4   268215 268291
## - Campaign_Type     4   268215 268291
## - Company           4   268216 268292
## - Duration          3   268215 268293
## - Customer_Segment  4   268219 268295
## - Language          4   268221 268297

## Step:  AIC=268248.2
## success ~ 1
##                     Df Deviance    AIC
## <none>                  268246 268248
## + Language          4   268239 268249
## + Customer_Segment  4   268241 268251
## + Duration          3   268245 268253
## + Company           4   268244 268254
## + Campaign_Type     4   268244 268254
## + Target_Audience   4   268245 268255
## + Location          4   268245 268255
## + Channel_Used      5   268244 268256
## + Engagement_Score  9   268238 268258

summary(step_cate)

library(car)

vif(cate_mod)

##                       GVIF Df GVIF^(1/(2*Df))
## Company           1.000605  4        1.000076
## Campaign_Type     1.000631  4        1.000079
## Target_Audience   1.000816  4        1.000102
## Duration          1.000555  3        1.000092
## Channel_Used      1.000874  5        1.000087
## Location          1.000633  4        1.000079
## Language          1.000798  4        1.000100
```

```
## Engagement_Score 1.001327  9        1.000074
## Customer_Segment 1.000705  4        1.000088
```

```
num_mod <- glm(success ~ ., data = numeric_vars, family = binomial)
summary(num_mod)
```

```
## glm(formula = success ~ ., family = binomial, data = numeric_vars)
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -4.230e-01  2.356e-02 -17.957   <2e-16 ***
## Acquisition_Cost  6.222e-07  1.055e-06   0.590    0.555
## ROI              -1.746e-03  2.638e-03  -0.662    0.508
## Clicks            9.929e-06  1.760e-05   0.564    0.573
## Impressions      -1.982e-06  1.762e-06  -1.125    0.261
##
##     Null deviance: 268246  on 199999  degrees of freedom
## Residual deviance: 268244  on 199995  degrees of freedom
## AIC: 268254
```

```
anova(num_mod)
```

```
anova(num_mod, cate_mod)
```

```
## Analysis of Deviance Table
## Model 1: success ~ Acquisition_Cost + ROI + Clicks + Impressions
## Model 2: success ~ Company + Campaign_Type + Target_Audience + Duration +
##     Channel_Used + Location + Language + Engagement_Score + Customer_Segment
##   Resid. Df Resid. Dev Df Deviance
## 1    199995     268244
## 2    199958     268213 37   30.685
```

```
p_value <- pchisq(30.685, df = 37, lower.tail = FALSE)
p_value
```

```
## [1] 0.7586104
```

```
full_mod <- glm(success ~ ., data = digi_data, family = binomial)
summary(full_mod)
```

```
## glm(formula = success ~ ., family = binomial, data = digi_data)
##     Null deviance: 268246  on 199999  degrees of freedom
## Residual deviance: 268211  on 199954  degrees of freedom
## AIC: 268303
```

```
digi_data$Engagement_Score <- as.numeric(digi_data$Engagement_Score)
engage_num_mod <- glm(success ~ ., data = digi_data, family = binomial)
summary(engage_num_mod)
```

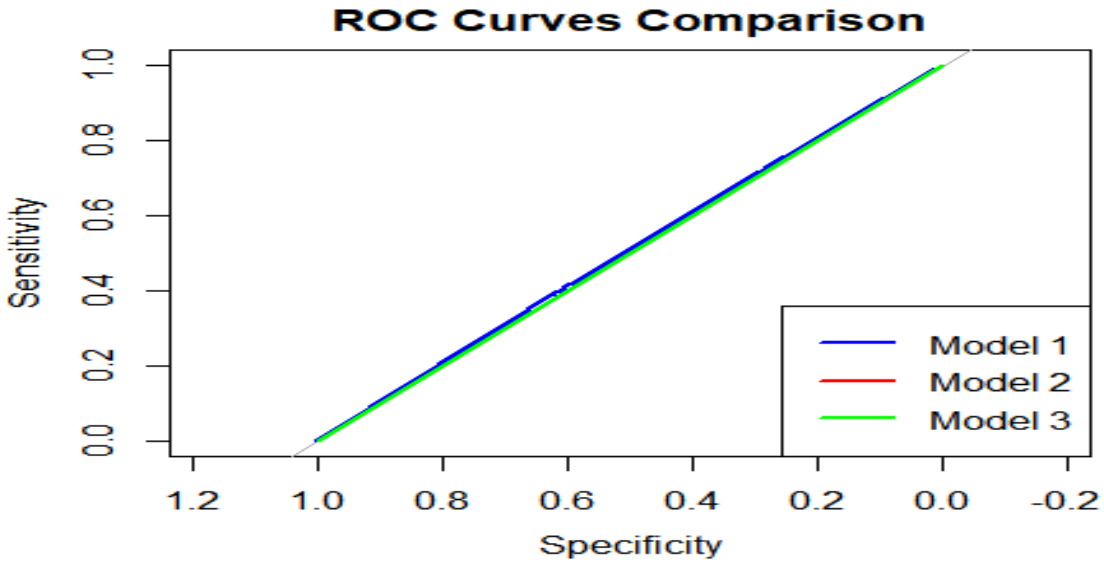
```
## glm(formula = success ~ ., family = binomial, data = digi_data)
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      -4.069e-01  3.521e-02 -11.558   <2e-16 ***
## Engagement_Score                 -1.932e-03  1.593e-03  -1.213   0.2252
## Customer_SegmentFoodies           2.548e-02  1.448e-02   1.760   0.0785 .
## Customer_SegmentTech Enthusiasts  2.931e-02  1.448e-02   2.024   0.0430 *
## AIC: 268294
```

```
digi_data$Engagement_Score <- as.factor(digi_data$Engagement_Score)
```

```
library(glmnet)
```

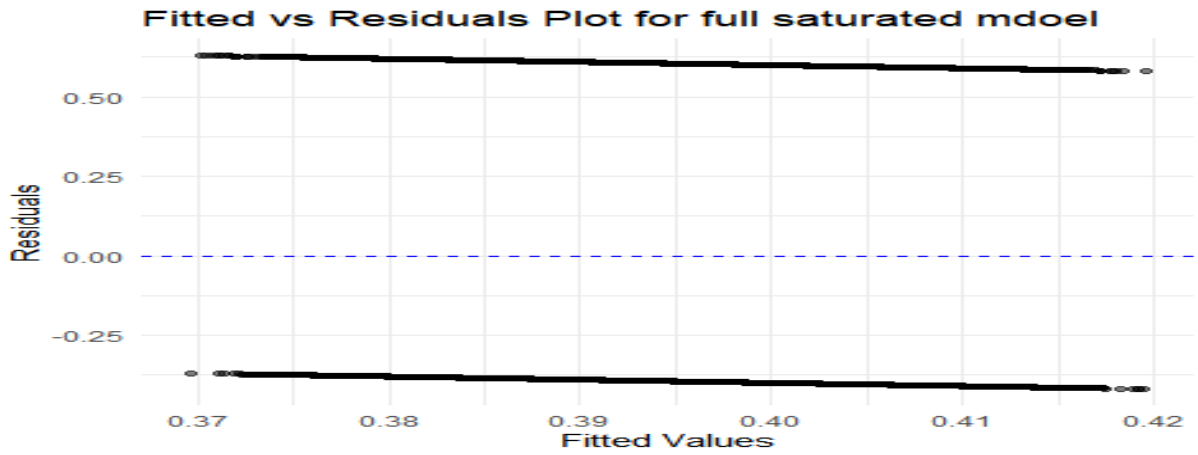
```r
X <- model.matrix(~ (.)^2, data = digi_data)[, -1]

lasso_model <- cv.glmnet(X, digi_data$success, alpha = 1)
coef(lasso_model, s = "lambda.min")

## 997 x 1 sparse Matrix of class "dgCMatrix"

elastic_net_model <- cv.glmnet(X, digi_data$success, alpha = 0.5, standardize = TRUE)

non_zero_coefs <- coef(elastic_net_model, s = "lambda.min")
non_zero_coefs_matrix <- as.matrix(non_zero_coefs)
feature_names <- rownames(non_zero_coefs_matrix)
coefs <- non_zero_coefs_matrix[, 1]
coef_df <- data.frame(feature = feature_names, coefficient = coefs)
non_zero_df <- coef_df[coef_df$coefficient != 0, ]
print(non_zero_df)

select_features <- non_zero_df$feature
select_features <- select_features[select_features != "(Intercept)"]
select_features <- select_features[select_features != "(success)"]
y <- digi_data$success

x_final <- X[, select_features, drop = FALSE]
select_inter_model <- glm(y ~ ., data = as.data.frame(x_final), family = binomial)

summary(select_inter_model)
## glm(formula = y ~ ., family = binomial, data = as.data.frame(x_final))

anova(full_mod, select_inter_model)

final_model <- glm(success ~ Customer_Segment + Engagement_Score + Language + ROI +
Acquisition_Cost + Clicks + Impressions, data = digi_data, family = binomial)
summary(final_model)
## glm(formula = success ~ Customer_Segment + Engagement_Score +
##     Language + ROI + Acquisition_Cost + Clicks + Impressions,
##     family = binomial, data = digi_data)

order_final <- step(final_model)

summary(order_final)
## Call:
## glm(formula = success ~ 1, family = binomial, data = digi_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.429440   0.004576  -93.85   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 268246  on 199999  degrees of freedom
## Residual deviance: 268246  on 199999  degrees of freedom
## AIC: 268248
##
## Number of Fisher Scoring iterations: 4
```

```r
intercept <- glm(success ~ 1, data = digi_data, family = binomial)
summary(intercept)
## Call:
## glm(formula = success ~ 1, family = binomial, data = digi_data)
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.429440   0.004576  -93.85   <2e-16 ***
##     Null deviance: 268246  on 199999  degrees of freedom
## Residual deviance: 268246  on 199999  degrees of freedom
## AIC: 268248

library(pROC)

pred_1 <- predict(full_mod, newdata = digi_data, type = "response")
pred_2 <- predict(order_final, newdata = digi_data, type = "response")
pred_3 <- predict(intercept, newdata = digi_data, type = "response")

roc1 <- roc(digi_data$success, pred_1)
roc2 <- roc(digi_data$success, pred_2)
roc3 <- roc(digi_data$success, pred_3)

plot(roc1, col = "blue", main = "ROC Curves Comparison")
plot(roc2, col = "red", add = TRUE)  # Adding the second ROC curve on the same plot
plot(roc3, col = "green", add = TRUE)
legend("bottomright", legend = c("Model 1", "Model 2", "Model 3"), col = c("blue", "red",
"green"), lwd = 2)
```




```r
auc_1 <- auc(roc1)
print(paste("AUC for Model 1:", auc_1)) ## [1] "AUC for Model 1: 0.507774319014089"

auc_2 <- auc(roc2)
print(paste("AUC for Model 2:", auc_2)) ## [1] "AUC for Model 2: 0.5"

auc_3 <- auc(roc3)
print(paste("AUC for Model 3:", auc_3)) ## [1] "AUC for Model 3: 0.5"

fitted_1 <- predict(full_mod, newdata = digi_data, type = "response")
residuals_1 <- digi_data$success - fitted_1
```

```r
fitted_2 <- predict(order_final, newdata = digi_data, type = "response")
residuals_2 <- digi_data$success - fitted_2
fitted_3 <- predict(intercept, newdata = digi_data, type = "response")
residuals_3 <- digi_data$success - fitted_3

ggplot(data = digi_data, aes(x = fitted_1, y = residuals_1)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "blue") +
  labs(title = "Fitted vs Residuals Plot for full saturated mdoel",
       x = "Fitted Values",
       y = "Residuals") +
  theme_minimal()
```




```r
ggplot(data = digi_data, aes(x = fitted_2, y = residuals_2)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(title = "Fitted vs Residuals Plot for final ordered model",
       x = "Fitted Values",
       y = "Residuals") +
  theme_minimal()
```


```r
ggplot(data = digi_data, aes(x = fitted_3, y = residuals_3)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "green") +
  labs(title = "Fitted vs Residuals Plot for intercept model",
       x = "Fitted Values",
```

```
      y = "Residuals") +
theme_minimal()
```