Model to Predict Customer Spending

Shankar Haridas

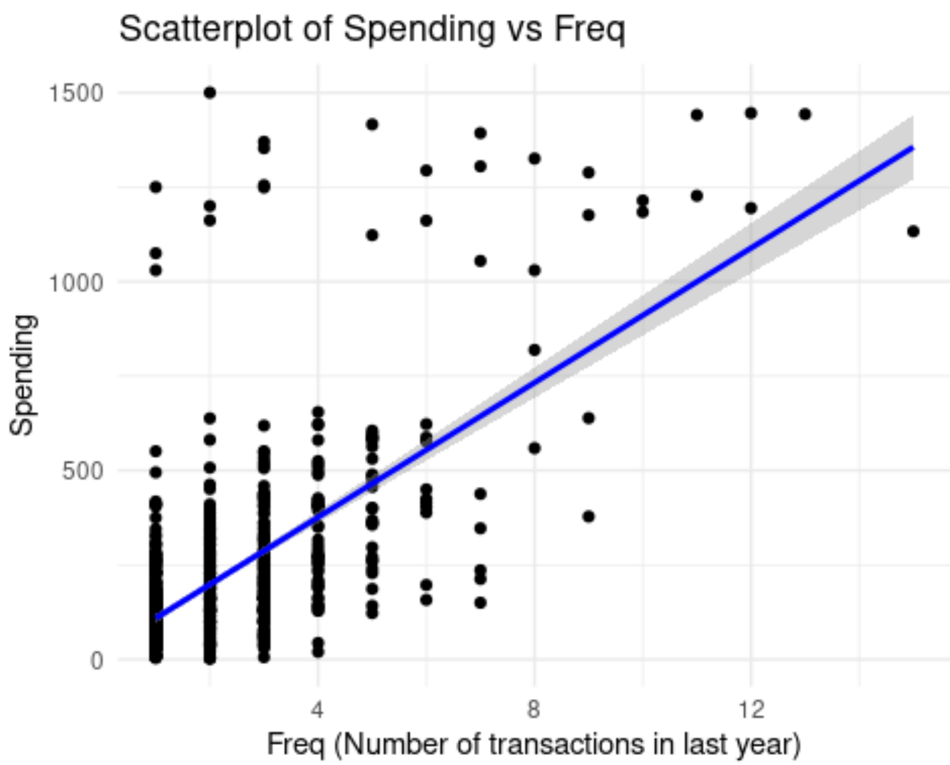
```
# Shankar Haridas

# Installing and Loading the required packages
install.packages("readxl")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)

library(readxl)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(Metrics)


# Load the dataset from Sheet 2 (the actual data)
data <- read_excel("Softwarecompany.xlsx", sheet = 2)

# Making sure the data loaded correctly
head(data)

## # A tibble: 6 × 25
##   sequence_number    US source_a source_c source_b source_d source_e sourc
e_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <d
bl>
## 1               1     1        0        0        1        0        0
0
## 2               3     1        0        0        0        0        0
0
## 3              10     1        1        0        0        0        0
0
## 4              15     0        0        0        0        0        0
0
## 5              21     1        0        0        0        0        0
0
## 6              24     1        0        0        0        0        0
```

```
0
## # ℹ 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>
,
## #   `1st_update_days_ago` <dbl>, `Web order` <dbl>, `Gender=male` <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>

# a) Code for part a

categorical_vars <- c("US", "Web order", "Gender=male", "Address_is_res")

# Use lapply as well as the pipe operator to calculate
# mean and standard deviation for each categorical variable
# The >%> operator allows me to pass the output directly
# into the next function as an argument without the need
# for a temporary variable.

spending_summary <- lapply(categorical_vars, function(var) {
  data %>%
    group_by_at(var) %>%
    summarise(
      mean_spending = mean(Spending, na.rm = TRUE),
      sd_spending = sd(Spending, na.rm = TRUE)
    )
})

# Print the summary for each categorical variable
spending_summary

## [[1]]
## # A tibble: 2 × 3
##      US mean_spending sd_spending
##   <dbl>         <dbl>       <dbl>
## 1     0          213.        201.
## 2     1          204.        225.
##
## [[2]]
## # A tibble: 2 × 3
##   `Web order` mean_spending sd_spending
##         <dbl>         <dbl>       <dbl>
## 1           0          209.        223.
## 2           1          202.        219.
##
## [[3]]
## # A tibble: 2 × 3
##   `Gender=male` mean_spending sd_spending
##           <dbl>         <dbl>       <dbl>
## 1             0          210.        223.
## 2             1          201.        219.
```

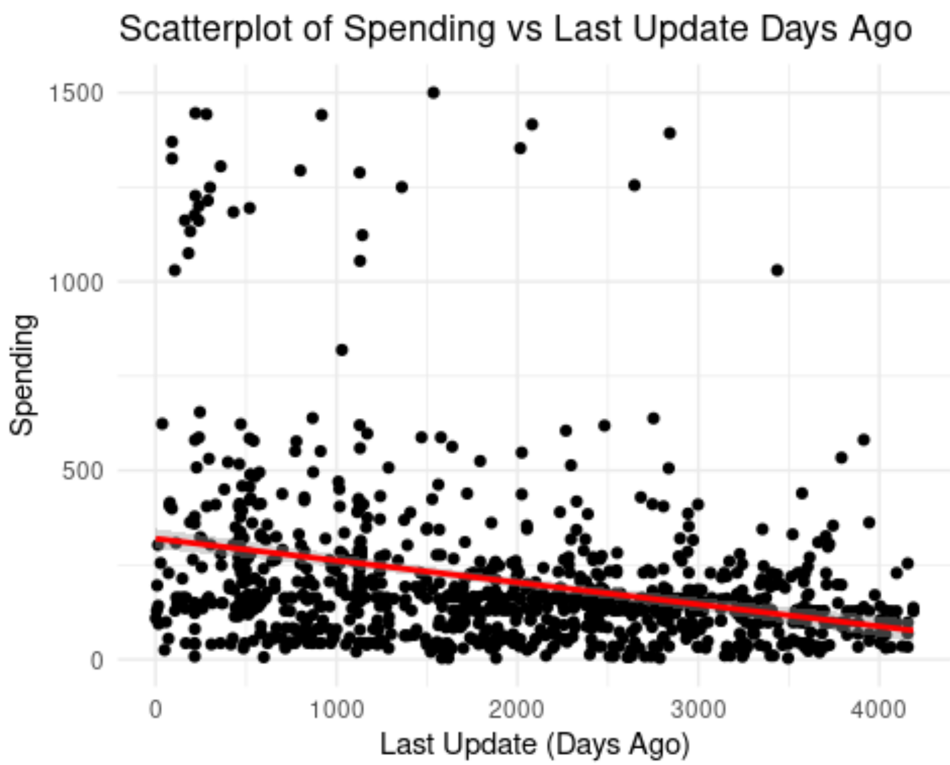
```
##
## [[4]]
## # A tibble: 2 × 3
##   Address_is_res mean_spending sd_spending
##            <dbl>         <dbl>       <dbl>
## 1              0          211.        240.
## 2              1          185.        133.
```

```
Takeaways from summary
1. Non-US customers, on average, spend a little more than
US customers, but the spending variability (standard deviation)
is higher among US customers.

2. Customers who ordered via the web spent slightly less on average
than those who didn’t, though the difference is minimal. The standard
deviation is almost the same, indicating a similar spread in spending
behavior for both groups

3. Female customers, on average, spent more than male customers, though
the difference is quite small. The standard deviation is also slightly
higher for females, indicating slightly more variability in spending
among female customers.

4. Customers with non-residential addresses spent more on average than
those with residential addresses. The spending variability is also much
higher among non-residential customers, indicating more diverse spending
patterns, while spending among residential customers is more consistent

# b) Code for part b

# Creating a scatterplot for Spending v Freq
ggplot(data, aes(x = Freq, y = Spending)) +
    geom_point() + # adding points to the scatterplot
    geom_smooth(method = "lm", col = "blue") + # adding linear regression lin
e
    labs(title = "Scatterplot of Spending vs Freq", x = "Freq (Number of tran
sactions in last year)", y = "Spending") +
    theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
Interpretation:

I do not think that the scatterplot shows a strong linear
relationship between Spending and Freq. There is high
variability in spending at the lower frequencies and the
data is very sparse at higher frequencies. Looking at just
the scatterplot, I do not see a reason to assume that there is
a linear relationship between spending and frequency.

# Creating a scatterplot for Spending v Last_Update
ggplot(data, aes(x = last_update_days_ago, y = Spending)) +
    geom_point() +
    geom_smooth(method = "lm", col = "red") +
    labs(title = "Scatterplot of Spending vs Last Update Days Ago", x = "Last
Update (Days Ago)", y = "Spending") +
    theme_minimal()

## `geom_smooth()` using formula = 'y ~ x'
```



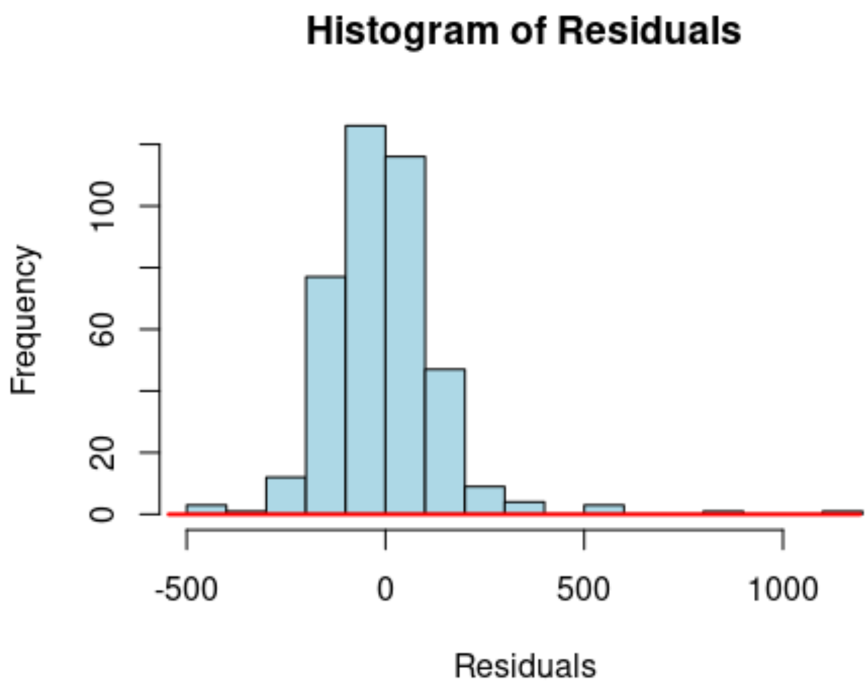

```r
Interpretation:

I also think that this graph does not provide convincing
evidence of a linear relationship between spending and the
days since last update

# c) Code for part c

# c1
set.seed(12345)
training_index <- sample(1:nrow(data), 0.6 * nrow(data))

# Split into training and validation sets
training_set <- data[training_index, ]
validation_set <- data[-training_index, ]

# c2

# Fit the multiple linear regression model
model <- lm(Spending ~ Freq + last_update_days_ago + `Web order` + `Gender=ma
le` + Address_is_res + US, data = training_set)

# Display the summary of the model
summary(model)
```

```
##
## Call:
## lm(formula = Spending ~ Freq + last_update_days_ago + `Web order` +
##     `Gender=male` + Address_is_res + US, data = training_set)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -418.31  -93.03  -21.29   44.66 1272.23
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          103.175086  26.639343   3.873 0.000119 ***
## Freq                  89.180162   4.690189  19.014  < 2e-16 ***
## last_update_days_ago  -0.022742   0.006977  -3.260 0.001179 **
## `Web order`            6.145211  14.471627   0.425 0.671254
## `Gender=male`          0.976507  14.485405   0.067 0.946276
## Address_is_res       -99.913716  17.593036  -5.679 2.12e-08 ***
## US                   -19.754342  19.236431  -1.027 0.304875
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 176.5 on 593 degrees of freedom
## Multiple R-squared:  0.4376, Adjusted R-squared:  0.4319
## F-statistic: 76.89 on 6 and 593 DF,  p-value: < 2.2e-16

Estimated Regression Equation:

Spending = 103.18 + 89.18(Freq)  - 0.02(last_update_days_ago)\n +
6.15(Web order) + 0.98(Gender=male) - 99.91(Address_is_res) - 19.75(US)

# c3

The type of purchaser most likely to spend a large amount of money is

(1) A frequent purchaser
(2) A customer with recent updates
(3) A non-residential purchaser

The other variables (web order, male, US, do not have a statistically
significant impact on spending. This is known because they have large
p-value, as can be seen in the summary statistics

# c4

The first predictor to be dropped from the model using backward
elimination will be Gender=male, since it has the highest p-value
and provides no statistically significant contribution to predicting spending

# c5
```

```r
# Storing the coefficients into variables
intercept <- 103.18
freq_coef <- 89.18
last_update_coef <- -0.02
web_order_coef <- 6.15
gender_male_coef <- 0.98
address_is_res_coef <- -99.91
us_coef <- -19.75

# Extract the first observation from the validation set
first_observation <- validation_set[1, ]

# Compute the predicted spending manually using the rounded coefficients
predicted_spending <- intercept +
  freq_coef * first_observation$Freq +
  last_update_coef * first_observation$last_update_days_ago +
  web_order_coef * first_observation$`Web order` +
  gender_male_coef * first_observation$`Gender=male` +
  address_is_res_coef * first_observation$Address_is_res +
  us_coef * first_observation$US

# Display the predicted spending
cat("Predicted Spending: ", predicted_spending)

## Predicted Spending:  184.13

actual_spending <- first_observation$Spending
cat("Actual Spending:", actual_spending)

## Actual Spending: 127.48

# Calculate the prediction error
prediction_error <- actual_spending - predicted_spending
cat("Prediction Error:", prediction_error)

## Prediction Error: -56.65

# c6

train_predictions <- predict(model, newdata = training_set)
validation_predictions <- predict(model, newdata = validation_set)

# RMSE and MAPE for the training set
rmse_train <- rmse(training_set$Spending, train_predictions)
mape_train <- mape(training_set$Spending, train_predictions) * 100  # Convert to percentage

# RMSE and MAPE for the validation set
rmse_validation <- rmse(validation_set$Spending, validation_predictions)
mape_validation <- mape(validation_set$Spending, validation_predictions) * 100  # Convert to percentage
```

```r
# Display the results
cat("RMSE for Training Set:", round(rmse_train, 2))

## RMSE for Training Set: 175.47

cat("MAPE for Training Set:", round(mape_train, 2), "%")

## MAPE for Training Set: 130 %

cat("RMSE for Validation Set:", round(rmse_validation, 2))

## RMSE for Validation Set: 145.19

cat("MAPE for Validation Set:", round(mape_validation, 2), "%")

## MAPE for Validation Set: 124.95 %

# c7

1. RMSE for the Validation Set:

An RMSE of 145.19 means that, on average, the models predictions differ
actual spending by $145.19. This is a relatively high error.

2. MAPE for the Validation Set:

The MAPE of 124.95% is very high. This means that, on average, the model's
predictions are off by 124.95%. This tells us that the predictive accuracy of
the model is low.

3. Comparison between Training and Validation Set Performance:

The RMSE on the validation set is lower than that on the training
set, which is a good sign, suggesting that the model is not
overfitting to the training data. However, both values are still
relatively high, indicating poor predictive performance overall.
The MAPE for both sets is extremely high, indicating that the model
struggles to predict spending accurately in both the training and validation
sets

# c8

# Compute residuals for the validation set
residuals <- validation_set$Spending - validation_predictions

# Create a histogram of the residuals
hist(residuals,
     main = "Histogram of Residuals",
     xlab = "Residuals",
     col = "lightblue",
     breaks = 20)
```

```r
# Add a density line to the histogram to check for normality
lines(density(residuals), col = "red", lwd = 2)
```




```
The histogram of the residuals is not normally distributed. The
residuals are right-skewed with a tail of large positive values.
This suggests that the model under-predicts spending, especially those with
higher spending amounts Additionally, since the histogram does not follow
a normal distribution
```