Store Branches Sales Analysis

Renad Gharz

19/06/2022

1. Overview of Data

The data sample consists of 4 continuous quantitative variables and 896 total observations. The purpose of this analysis is to build a regression model to predict the store's sales in USD using the store area, the items available, and the daily customer count (average) as predictor variables.

The **Store__Area** variable represents the physical area of the store. The values are originally given in square yards (yd^2), however, for better readability, it was converted to squared feet (ft^2) by multiplying the squared yards values by 9.

The **Items__Available** variable represents the number of different items available in the store.

The **Daily__Customer__Count** variable represents the average number of customers who visited the store in a month.

the **Stores__Sales** variable represents the sales made in a store, in USD currency.

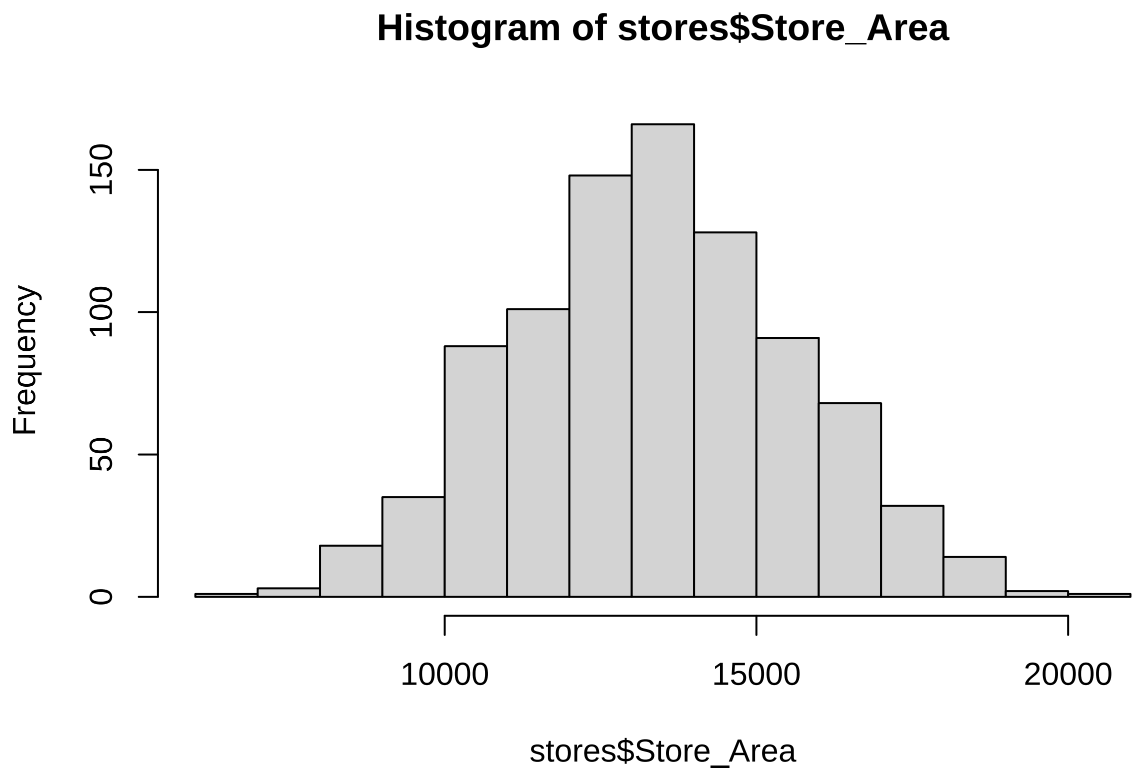
```
##   Store_Area Items_Available Daily_Customer_Count Store_Sales
## 1      14931            1961                  530       66490
## 2      13149            1752                  210       39820
## 3      12060            1609                  720       54010
## 4      13059            1748                  620       53730
## 5      15930            2111                  450       46620
## 6      12978            1733                  760       45260
```

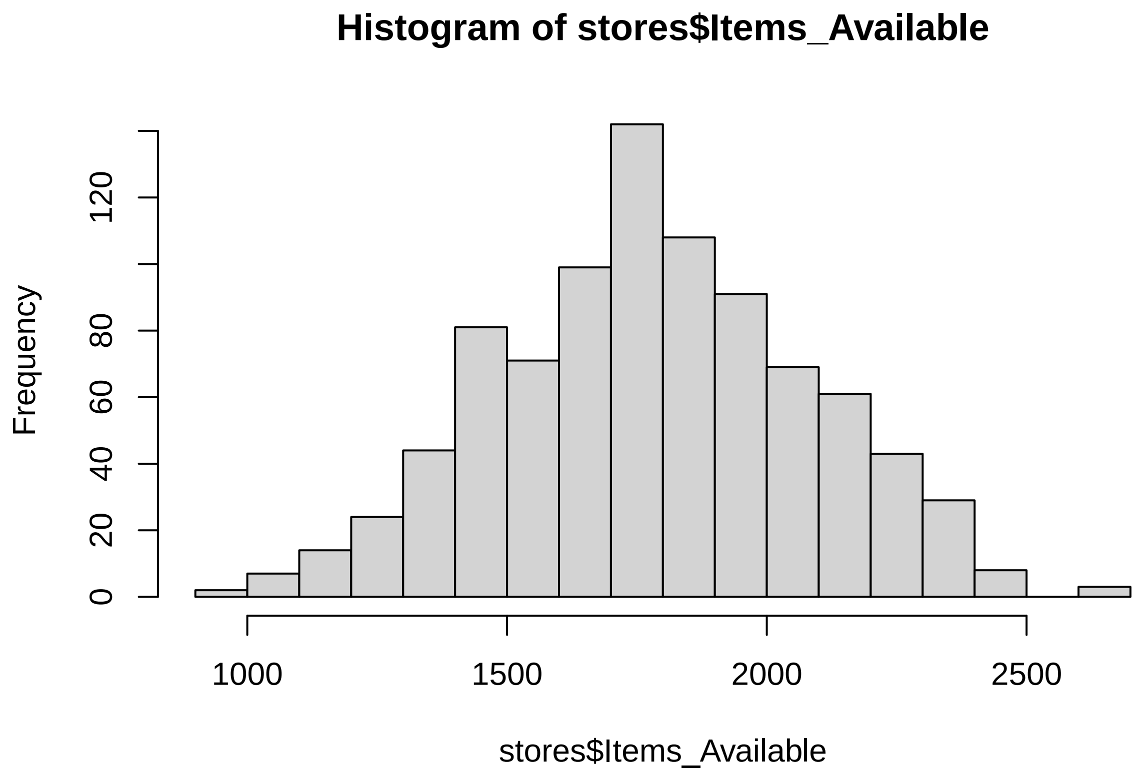
2. Exploratory Analysis

Central tendencies of the sample:

```
##    Store_Area    Items_Available Daily_Customer_Count  Store_Sales
##  Min.   : 6975   Min.   : 932    Min.   :  10.0       Min.   : 14920
##  1st Qu.:11851   1st Qu.:1576    1st Qu.: 600.0       1st Qu.: 46530
##  Median :13293   Median :1774    Median : 780.0       Median : 58605
##  Mean   :13369   Mean   :1782    Mean   : 786.4       Mean   : 59351
##  3rd Qu.:14882   3rd Qu.:1983    3rd Qu.: 970.0       3rd Qu.: 71873
##  Max.   :20061   Max.   :2667    Max.   :1560.0       Max.   :116320
```

From the central tendencies table above, we can see that the means medians of all 4 variables are relatively equal indicating that the sample data is relatively symmetrical. We can further confirm this by looking at the shape of each variable's histogram which support the relative symmetry claim.


Histogram of stores$Store_Area
Frequency
0
50
100
150
10000
15000
20000
stores$Store_Area


Histogram of stores$Items_Available
Frequency
0
20
40
60
80
120
1000
1500
2000
2500
stores$Items_Available


Histogram of stores$Daily_Customer_Count
Frequency
0
20
40
60
80
100
0
500
1000
1500
stores$Daily_Customer_Count

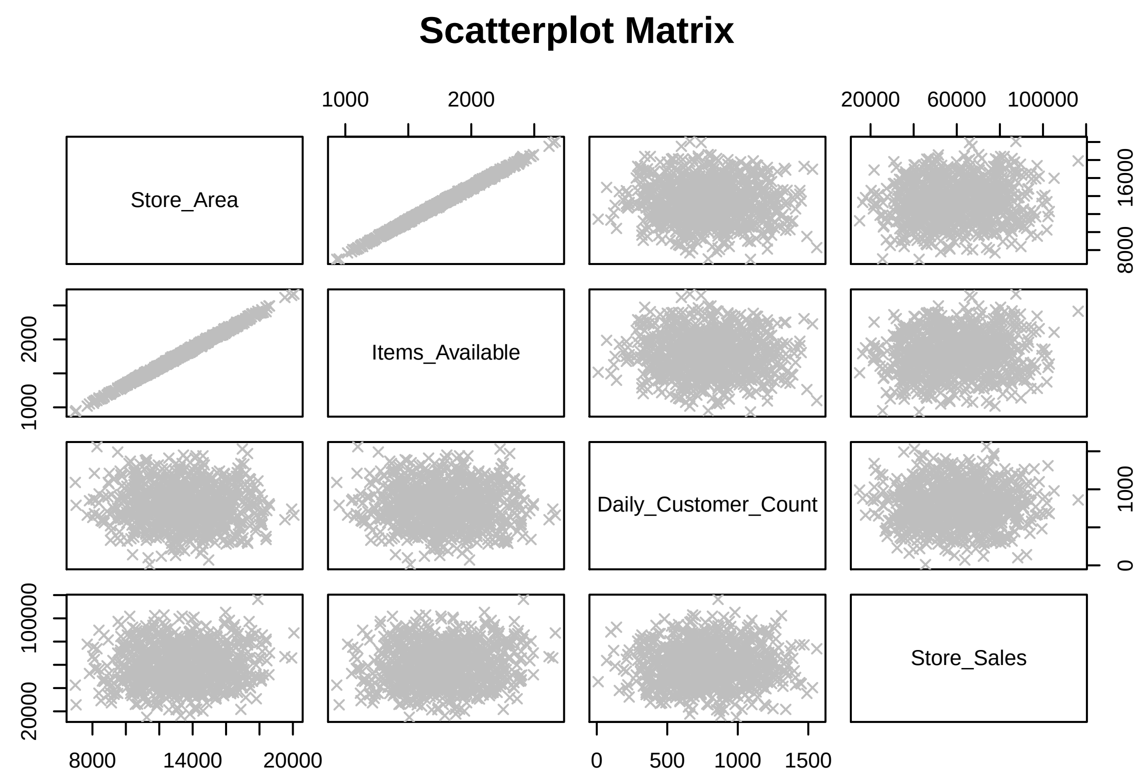
As shown in the histograms, although there a few outlier points (small bars on the edges of the histograms), the majority of the data is symmetrically distributed. Thus, we can assume that the sample meets the relative normality condition of a linear regression model.

Descriptive statistics of the sample:

```
## Warning in describeBy(stores): no grouping variable requested
```

```
##                       vars   n     mean       sd  median  trimmed      mad   min
## Store_Area               1 896 13368.69  2252.13 13293.0 13356.36  2241.69  6975
## Items_Available          2 896  1782.04   299.87  1773.5  1780.30   300.23   932
## Daily_Customer_Count     3 896   786.35   265.39   780.0   784.25   266.87    10
## Store_Sales              4 896 59351.31 17190.74 58605.0 59056.66 18636.28 14920
##                          max  range skew kurtosis     se
## Store_Area             20061  13086 0.03    -0.29  75.24
## Items_Available         2667   1735 0.03    -0.29  10.02
## Daily_Customer_Count    1560   1550 0.07    -0.27   8.87
## Store_Sales           116320 101400 0.15    -0.47 574.30
```

The descriptive statistics also support this as the skew factors are fairly low for 3 of the 4 variables, with the store sales variable being the exception having a 0.15 skew, which is still an acceptable level.




```
##                       Store_Area Items_Available Daily_Customer_Count
## Store_Area             1.0000000      0.99889075         -0.041423095
## Items_Available        0.9988908      1.00000000         -0.040978117
## Daily_Customer_Count -0.0414231     -0.04097812          1.000000000
## Store_Sales            0.0974738      0.09884943          0.008628708
##                       Store_Sales
## Store_Area            0.097473795
## Items_Available       0.098849435
## Daily_Customer_Count  0.008628708
## Store_Sales           1.000000000
```

The scatterplot matrix shows that apart from one relation (Store_Area-Items_Available), the remaining predictor variables have little to no linear relationship with the response variable, which could result in a less accurate regression model (due to collinearity). This is further backed up by the correlation table which indicates low correlation between most of the variables.

3. Model 1 - Raw Data

In order to make sure that we end up with the most accurate model with the highest prediction power, a few different models will be built with various adjustments. Model 1 is the simplest model and will use the raw data without any transformations (except for the square yards to square feet conversions for Store_Area).

```
regression_model_1 <-
  lm(data=stores,
     formula=
       Store_Sales~Store_Area+Items_Available+Daily_Customer_Count)
```

```r
model_1_summary <- summary(regression_model_1)
model_1_summary
```

```
##
## Call:
## lm(formula = Store_Sales ~ Store_Area + Items_Available + Daily_Customer_Count,
##     data = stores)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -43689 -12834   -567  12882  52405
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          48567.7788  3908.8280  12.425   <2e-16 ***
## Store_Area              -4.3373     5.3989  -0.803    0.422
## Items_Available         38.2342    40.5469   0.943    0.346
## Daily_Customer_Count     0.8046     2.1592   0.373    0.710
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17130 on 892 degrees of freedom
## Multiple R-squared:  0.01065,    Adjusted R-squared:  0.007321
## F-statistic:   3.2 on 3 and 892 DF,  p-value: 0.02276
```

```r
model_1_summary$coefficients
```

```
##                          Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)          48567.7788448 3908.828046 12.4251510 8.314683e-33
## Store_Area              -4.3372855    5.398934 -0.8033596 4.219809e-01
## Items_Available         38.2341546   40.546944  0.9429602 3.459566e-01
## Daily_Customer_Count     0.8046133    2.159199  0.3726443 7.095017e-01
```

```r
model_1_summary$adj.r.squared
```

```
## [1] 0.007320654
```

Since this is a multiple regression (more than 1 predictor variable), it is better to use the adjusted R-squared than the multiple R-squared. As we add more predictor variables to models, the R-squared will always go up, thus we need to compensate for this to make sure our model is not inaccurate, thus we would use the adjusted R-squared which accounts for every additional predictor variable added, giving us a more accurate estimation of the model's prediction accuracy.

The R-squared measures how much variance the model accounts for, thus the higher the value, the better the model will be at predicting the response variable. The adjusted R-squared is 0.073% which is extremely low, indicating that the model has almost no prediction power to accurately regress the response variable.

As suspected previously, collinearity seems to be a factor here as the **Store__Area** and the **Items__Available** predictors are almost perfectly correlated. This pairwise collinearity is affecting how we interpret the coefficients, which is why the Store_Area coefficient is negative which does not make sense from a domain perspective.

We can verify this property by looking at the Variance Inflation Factor of model's predictors.

```r
vif(regression_model_1)
```

```
##           Store_Area      Items_Available Daily_Customer_Count
##           451.054471           451.037905             1.001791
```

As shown, the 2 correlated predictors demonstrate a very high VIF (greater than 10), which indicates multicollinearity in the model. In order to make the model more parsimonuous, we can run new models by dropping one of those 2 collinear variables to improve the model.

4. Model 2 - Adjusting for Collinearity

In Model 2, we will compare 2 submodels in which one of the 2 collinear variables will be dropped in each model to see which one demonstrates better prediction power.

4.1 Dropping Store_Area

We will first drop the **Store_Area** variable.

```
stores2 <- stores %>% select(-Store_Area)

regression_model_2 <- lm(data=stores2,
                         formula=
                           Store_Sales~Items_Available+Daily_Customer_Count)

model_2_summary <- summary(regression_model_2)
model_2_summary
```

```
##
## Call:
## lm(formula = Store_Sales ~ Items_Available + Daily_Customer_Count,
##     data = stores2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -43104 -12913   -660  12686  53308
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.855e+04  3.908e+03  12.424  < 2e-16 ***
## Items_Available      5.697e+00  1.910e+00   2.982  0.00294 **
## Daily_Customer_Count 8.227e-01  2.159e+00   0.381  0.70321
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17120 on 893 degrees of freedom
## Multiple R-squared:  0.009932,	Adjusted R-squared:  0.007715
## F-statistic: 4.479 on 2 and 893 DF,  p-value: 0.0116
```

```
model_2_summary$coefficients
```

```
##                          Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)          48552.888556 3908.007928 12.4239483 8.370588e-33
## Items_Available          5.696570    1.910426  2.9818316 2.943120e-03
## Daily_Customer_Count     0.822695    2.158653  0.3811149 7.032086e-01
```

```
model_2_summary$adj.r.squared
```

```
## [1] 0.007714851
```

In this new model we can see that the adjusted R-squared has minutely improved over the model that used the raw data to 0.077%, however this model is still not good enough.

4.2 Dropping Items_Available

We can now move on to dropping the **Items_Available** variable.

```
stores3 <- stores %>% select(-Items_Available)

regression_model_3 <- lm(data=stores3,
                         formula=
                           Store_Sales~Store_Area+Daily_Customer_Count)

model_3_summary <- summary(regression_model_3)
model_3_summary
```

```
##
## Call:
## lm(formula = Store_Sales ~ Store_Area + Daily_Customer_Count,
##     data = stores3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -43014 -12916   -683  12654  53518
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.870e+04  3.906e+03  12.470  < 2e-16 ***
## Store_Area           7.480e-01  2.544e-01   2.940  0.00336 **
## Daily_Customer_Count 8.219e-01  2.159e+00   0.381  0.70353
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17130 on 893 degrees of freedom
## Multiple R-squared:  0.009662,   Adjusted R-squared:  0.007444
## F-statistic: 4.356 on 2 and 893 DF,  p-value: 0.0131
```

```
model_3_summary$coefficients
```

```
##                          Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)          4.870473e+04 3905.8866344 12.4695703 5.143580e-33
## Store_Area           7.480384e-01    0.2544131  2.9402508 3.364059e-03
## Daily_Customer_Count 8.218818e-01    2.1589878  0.3806792 7.035318e-01
```

```
model_3_summary$adj.r.squared
```

```
## [1] 0.007443851
```

5. Model 3 - Log-Transformed Data

6. Model 4 - Data Without Outliers

7. Conclusion