**Nicholas Lynch**

**C2T3 - Multiple Regressions in R**

**Algorithm summary**

The linear model, random forest(rf), gradient boosting machines (gbm), and support vector machine(svm) data model were used to predict new product volumes. These model were trained and tested on the existing product dataset. Before training these models, the “BestSellersRank” and “x5StarReviews” were removed to prevent errors due to NA values and overfitting, respectively. In addition, the models were trained on a 10 fold repeated cross validation sampling model. The rf model was chosen for new product volumes prediction.

The linear model was not used as our data model has over 20 attributes, but linear models are best suited for datasets with a few parameters and normal distribution. In addition, non-parametric methods (rf, gbm, and svm) are better for datasets that have numerous attributes. The rf model was selected because the svm and gbm predicted negative volume on the existing product dataset. Negative volumes are not possible for that dataset. See R Model data models at the end of the report for [prediction model summaries](#P_Summaries). As shown below, the optimal rf model was mtry = 26 based on the “Rsquared” value, which represents accuracy. The model has a mean absolute error (MAE) of 386.19 and a root-mean-square error (RMSE) of 788.03. MAE represents the margin of error and RMSE represents the squared deviation. The confidence interval cannot be calculate as this is a regression model, not a classification model.

|  |  |  |  |
| --- | --- | --- | --- |
| mtry | RMSE | Rsquared | MAE |
| 2 | 868.09 | 0.82 | 495.26 |
| 14 | 839.74 | 0.86 | 412.12 |
| 26 | 788.03 | 0.88 | 386.19 |

Note that as the number of mtry increased, the model shows signs of overfitting. Although overfitting occurred in this model, it is better than the other models discussed. When this model was applied to the test dataset, the results were:

|  |  |  |  |
| --- | --- | --- | --- |
| > postResample(testPred\_rf1, testing$Volume) | | | |
| **RMSE** | **Rsquared** | **MAE** |
| 262.30 | 0.83 | 110.72 |

The MAE and the RMSE were much lower for once applied to the testing dataset. The RF predictive model was then applied to the new products dataset.

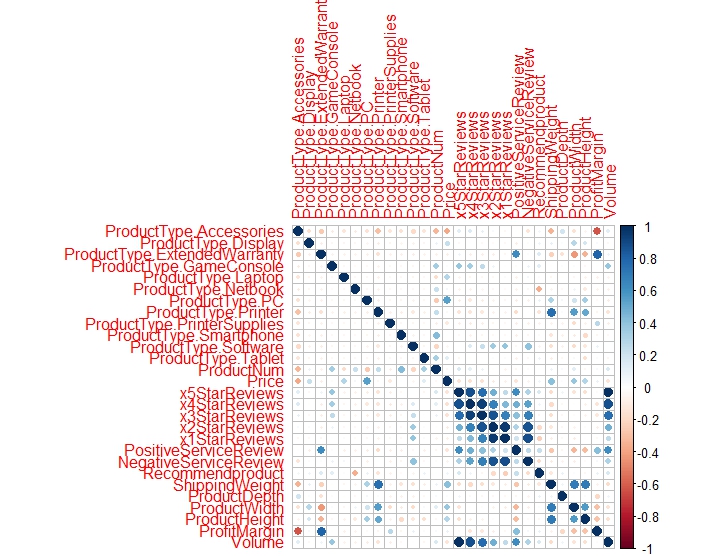
**New product sales prediction** **for Laptops, PCs, Netbooks, and Smartphones**

|  |  |  |
| --- | --- | --- |
| **Product Category** | **Predicted Sales Volume** | **Predicted Profit** |
| PC | 621 | $108,118.25 |
| Netbook | 1,880 | $58,373.48 |
| Laptop | 206 | $29,916.89 |
| Smartphone | 1,522 | $22,091.10 |
| **Grand Total** | **4,229** | **$218,499.72** |

PCs are most profitable and Smartphone are the least profitable. For predictions at individual level, see below

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Rank** | **Product Type** | **Product #** | **Price** | **Profit Margin** | **Predictions Volume** | **Total  Profit** |
| 1 | PC | 171 | $699.00 | 25% | 475 | $83,006.25 |
| 2 | Netbook | 180 | $329.00 | 9% | 1660 | $49,152.60 |
| 3 | PC | 172 | $860.00 | 20% | 146 | $25,112.00 |
| 4 | Laptop | 173 | $1,199.00 | 10% | 175 | $20,982.50 |
| 5 | Smartphone | 193 | $199.00 | 11% | 417 | $9,128.13 |
| 6 | Netbook | 181 | $439.00 | 11% | 136 | $6,567.44 |
| 7 | Smartphone | 196 | $300.00 | 11% | 188 | $6,204.00 |
| 8 | Laptop | 176 | $1,999.00 | 23% | 12 | $5,517.24 |
| 9 | Smartphone | 194 | $49.00 | 12% | 834 | $4,903.92 |
| 10 | Laptop | 175 | $1,199.00 | 15% | 19 | $3,417.15 |
| 11 | Netbook | 178 | $399.99 | 8% | 69 | $2,207.94 |
| 12 | Smartphone | 195 | $149.00 | 15% | 83 | $1,855.05 |
| 13 | Netbook | 183 | $330.00 | 9% | 15 | $445.50 |
|  |  |  |  |  | Grand Total | $218,499.72 |

**Customer and service reviews effect on sales volume**

****

|  |  |
| --- | --- |
| **Review Attribute** | **Correlation to Volume** |
| x5StarReviews | 1.00 |
| x4StarReviews | 0.88 |
| x3StarReviews | 0.76 |
| x2StarReviews | 0.49 |
| x1StarReviews | 0.26 |
| PositiveServiceReview | 0.62 |
| NegativeServiceReview | 0.31 |

I created a correlation matrix based on the existing products dataset. There is an increasing correlation for product volume from negative to positive reviews. The star reviews have a stronger correlation than the service reviews. The 5 star review attribute had a perfect correlation, so I removed it from my data model to prevent overfitting.

**R data models output where seed = 123**

**1. Linear Model <-** lm(formula = Volume ~ ., data = ready\_Data)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Residuals:** |  |  |  |  |
| **Min** | **1Q** | **Median** | **3Q** | **Max** |
| -1939.29 | -189.45 | 45.52 | 219.5 | 1153.12 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Coefficients: (1 not defined because of singularities)** | | | |  |  |
| **Estimate** | **Std.** | **Error** | **t value** | **Pr(>|t|)** | **Sig Codes** |
| (Intercept) | 427.03 | 972.87 | 0.439 | 0.66 |  |
| ProductType.Accessories | -3.08 | 460.91 | -0.007 | 0.99 |  |
| ProductType.Display | 416.43 | 444.51 | 0.937 | 0.35 |  |
| ProductType.ExtendedWarranty | 79.20 | 810.23 | 0.098 | 0.92 |  |
| ProductType.GameConsole | 1,607.09 | 546.62 | 2.94 | 0.00 | \*\* |
| ProductType.Laptop | -11.55 | 542.74 | -0.021 | 0.98 |  |
| ProductType.Netbook | 860.62 | 511.52 | 1.682 | 0.10 | . |
| ProductType.PC | 201.59 | 567.33 | 0.355 | 0.72 |  |
| ProductType.Printer | 1,156.79 | 569.27 | 2.032 | 0.05 | \* |
| ProductType.PrinterSupplies | -136.61 | 694.84 | -0.197 | 0.84 |  |
| ProductType.Smartphone | 467.99 | 390.15 | 1.2 | 0.24 |  |
| ProductType.Software | -344.50 | 529.66 | -0.65 | 0.52 |  |
| ProductType.Tablet | NA | NA | NA | NA |  |
| Price | 0.53 | 0.36 | 1.471 | 0.15 |  |
| x4StarReviews | 27.71 | 5.00 | 5.539 | 0.00 | \*\*\* |
| x3StarReviews | -39.80 | 13.63 | -2.92 | 0.01 | \*\* |
| x2StarReviews | 34.63 | 17.72 | 1.955 | 0.06 | . |
| x1StarReviews | -3.09 | 3.52 | -0.879 | 0.38 |  |
| PositiveServiceReview | 1.90 | 1.22 | 1.553 | 0.13 |  |
| NegativeServiceReview | -44.50 | 12.24 | -3.637 | 0.00 | \*\*\* |
| Recommendproduct | 454.52 | 381.77 | 1.191 | 0.24 |  |
| ShippingWeight | 1.13 | 10.87 | 0.104 | 0.92 |  |
| ProductDepth | 3.32 | 1.77 | 1.874 | 0.07 | . |
| ProductWidth | -69.74 | 16.66 | -4.187 | 0.00 | \*\*\* |
| ProductHeight | -11.46 | 14.14 | -0.81 | 0.42 |  |
| ProfitMargin | 414.82 | 2,138.01 | 0.194 | 0.85 |  |
| ProductNum | -4.74 | 4.96 | -0.957 | 0.34 |  |

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Residual standard error**: 490.7 on 54 degrees of freedom | | | |
| |  |  | | --- | --- | | **Multiple R-squared:** | 0.9285 | | **Adjusted R-squared:** | 0.8953 | |  |  |  |
| **F-statistic**: 28.03 on 25 and 54 DF | **p-value:** | < 2.2e-16 | |

**2. Random Forest** <- train(Volume ~ ., data = training, method = "rf", metric = "Rsquared", trControl = fitcontrol1)

61 samples

26 predictors

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 10 times)

Summary of sample sizes: 55, 56, 55, 55, 55, 54, ...

Resampling results across tuning parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **mtry** | **RMSE** | **Rsquared** | **MAE** |
| 2 | 868.09 | 0.82 | 495.26 |
| 14 | 839.74 | 0.86 | 412.12 |
| 26 | 788.03 | 0.88 | 386.19 |

Rsquared was used to select the optimal model using the largest value.

The final value used for the model was mtry = 26.

**3. Support Vector Machine with Linear Kernel** <- train(Volume ~ ., data = training, method = "svmLinear2", metric = "Rsquared", trControl = fitcontrol1)

61 samples

26 predictors

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 10 times)

Summary of sample sizes: 54, 55, 57, 54, 56, 54, ...

Resampling results across tuning parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **cost** | **RMSE** | **Rsquared** | **MAE** |
| 0.25 | 1,247.03 | 0.84 | 648.31 |
| 0.50 | 1,273.68 | 0.82 | 682.62 |
| 1.00 | 1,312.09 | 0.81 | 712.41 |

Rsquared was used to select the optimal model using the largest value.

The final value used for the model was cost = 0.25.

There were 36 warnings (use warnings() to see them) > warnings()

Warning messages: 1: In svm.default(x = as.matrix(x), y = y, kernel = "linear", ... : Variable(s) ProductType.Smartphone’ constant. Cannot scale data.

**4. Stochastic Gradient Boosting Machines** <- train(Volume ~ ., data = training, method = "gbm", metric = "Rsquared", trControl = fitcontrol1,verbose = FALSE)

61 samples

26 predictors

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 10 times)

Summary of sample sizes: 54, 53, 54, 56, 55, 57, ...

Resampling results across tuning parameters:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **interaction.depth** | **n.trees** | **RMSE** | **Rsquared** | **MAE** |
| 1 | 50 | 966.50 | 0.78 | 592.47 |
| 1 | 100 | 1,013.07 | 0.76 | 651.58 |
| 1 | 150 | 1,043.32 | 0.74 | 679.96 |
| 2 | 50 | 955.87 | 0.78 | 586.55 |
| 2 | 100 | 1,011.98 | 0.74 | 656.12 |
| 2 | 150 | 1,054.73 | 0.72 | 697.31 |
| 3 | 50 | 973.94 | 0.77 | 596.69 |
| 3 | 100 | 1,021.42 | 0.75 | 658.49 |
| 3 | 150 | 1,059.24 | 0.72 | 699.09 |

Tuning parameter 'shrinkage' was held constant at a value of 0.1

Tuning parameter 'n.minobsinnode' was held constant at a value of 10

Rsquared was used to select the optimal model using the largest value.

The final values used for the model were n.trees = 50, interaction.depth = 1, shrinkage = 0.1 and n.minobsinnode = 10.

There were 50 or more warnings (use warnings() to see the first 50)

1: In (function (x, y, offset = NULL, misc = NULL, distribution = "bernoulli", ... : variable 4: ProductType.GameConsole has no variation.

5. **Support Vector Machine with no scaling** <- train(Volume ~ ., data = training, method = "svmLinear2", metric = "Rsquared", trControl = fitcontrol1, scale = FALSE)

61 samples

26 predictors

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 10 times)

Summary of sample sizes: 54, 55, 55, 53, 56, 57, ...

Resampling results across tuning parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **cost** | **RMSE** | **Rsquared** | **MAE** |
| 0.25 | 815.84 | 0.85 | 421.76 |
| 0.50 | 919.04 | 0.81 | 468.93 |
| 1.00 | 1,038.49 | 0.84 | 517.05 |

Rsquared was used to select the optimal model using the largest value.The final value used for the model was cost = 0.25.

**Prediction Model Summaries**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| > summary(testPred\_rf1) | | |  |  |  |
| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
| 14.11 | 48.26 | 330.08 | 547.79 | 1218.32 | 1894.93 |
| > summary(testPred\_svm1) | | |  |  |  |
| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
| -191.1 | 119.2 | 337.5 | 520 | 535.2 | 2060.2 |
| > summary(testPred\_gbm) | | |  |  |  |
| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
| -399 | 467.5 | 793.1 | 941.8 | 1622.9 | 2372.7 |
| > summary(testPred\_svm\_noscale) | | | |  |  |
| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
| -12.28 | 65.38 | 316.88 | 507.37 | 633.04 | 2226.62 |