Data Glacier

Your Deep Learning Partner

Retail Case Study EDA Presentation

Virtual Internship

December 16, 2023

Background - Retail Case Study

Problem: *X* is a company that has a beverages business in Australia. They sell their products through various super-markets & also engage in heavy promotions throughout the year. Their demand is influenced by various factors like holidays & seasonality. They need a forecast for each of their products at the item level in weekly buckets.

Business Understanding: The objective is to build a multivariate machine learning model that will be able to forecast sales weekly. The data that is required to build the model may need to be re-coded so that the dates are in weekly buckets.

Data Understanding

1. What type of data have you got for analysis?

The data that I am working with is numeric, with the exception of the features: Product, date, & Price.Discount. The exceptions are strings. Features In.Store.Promo, Catalogue.Promo, Store.End.Promo, Covid_Flag, V_DAY, EASTER, & CHRISTMAS are boolean values: 1 denoting True, 0 denoting False.

2. What are the problems in the data (number of NA values, outliers, skewed, etc)?

The problems I see in the dataset are the features Price.Discount.... & date. The Price.Discount.... feature needs to be renamed. The date feature needs to be re-coded into weekly buckets.

3. What approaches will you apply to your dataset to overcome problems like NA values, outliers, etc. & why?

If there are NA values, I will remove the rows with them. It's better to train a model when we don't have missing data. Although there are many methods to resolve missing data, I feel that doing so will not reflect the true population values of the features. I believe that for this project, the model could be slightly overfit, since only this company will be using this data for its forecasting. Outliers in the sales feature could be the results of promotions, seasonality, holidays, etc. & I want to take that into account when training the model. As such, I will try a model with outliers & without outliers. When building the model without outliers, only extreme outliers will be removed. I believe the systematic way to identify extreme outliers in a statistical research setting is anything over 75th quartile + 3 IQR or anything under 25th quartile - 3 IQR. Anything within will be retained. Extreme outliers will affect the model training process. This model that has removed outliers will be more generalizable. We will determine which is better with the models' predictive accuracy & context.

Selecting the Best Variable

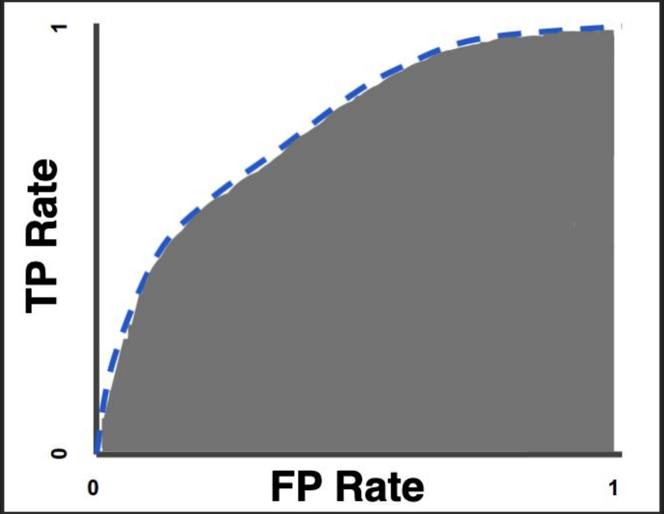
In the machine learning world, there is something called the “curse of dimensionality”. Dimensionality refers to the number of variables (input features, columns) in the data set. When the data set is very large in its number of input features relative to the number observations (rows) in our dataset, some algorithms can have a very hard time producing effective model.

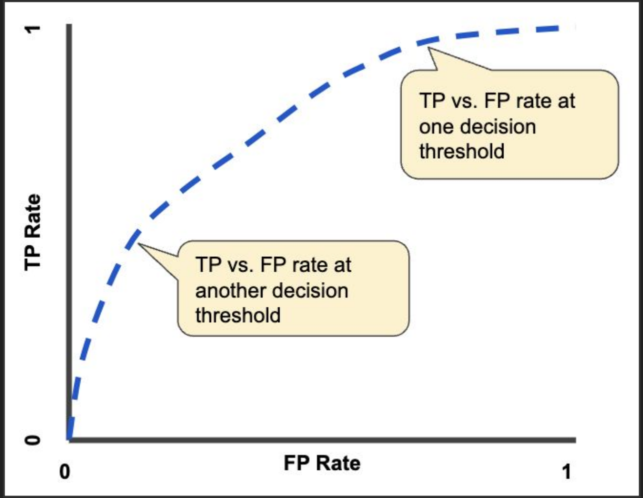
Since there isn’t any missing values in the data, we can move on to dimension reduction. We want to reduce the number of variables to reduce the computational cost of modeling & possible improve the performance of the model. This process is called dimension reduction. The type of dimension reduction we will be performing is feature selection. Feature selection is the process of automatically or manually selecting features that contribute the most to our target variable, i.e. the variable we are trying to predict. I will be performing three types of feature selection & will be using the results of one for my predictive model.

1) Variable Importance
2) Recursive Feature Elimination
3) Akaike Information Criterion

Variable Importance

An ROC curve is a graph that shows how well a classification model performs. The curve has two axes: one for how often the model correctly identifies positive cases (i.e. true positives) & another for how often it mistakenly identifies negative cases as positive (false positives). It shows the performance of a classification model at all classification thresholds.

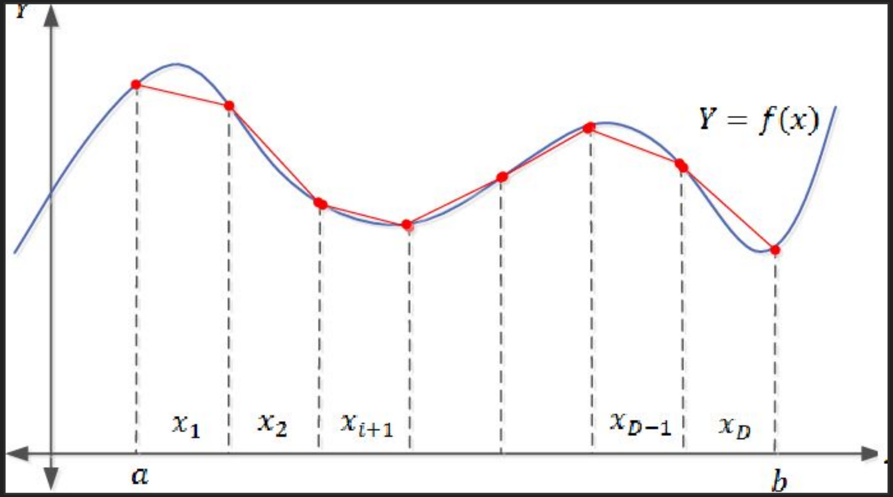


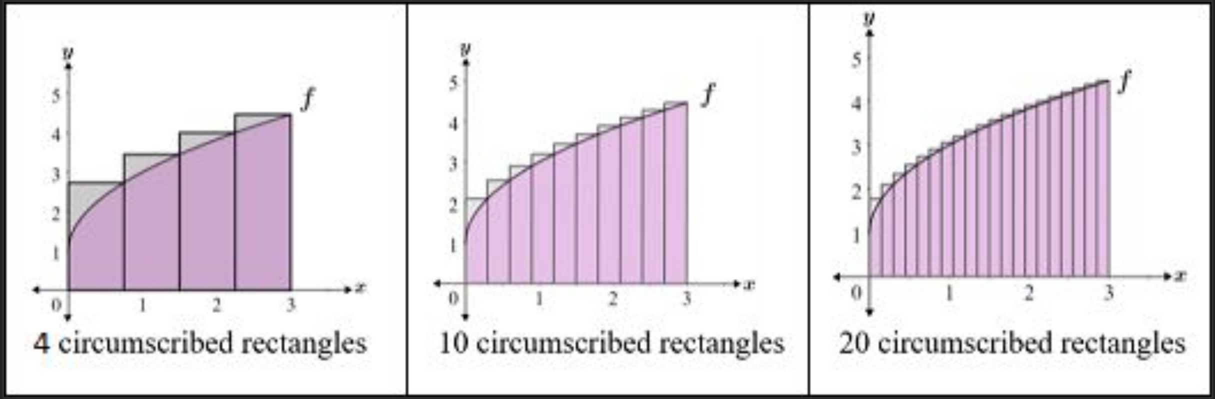


The area under the curve (AUC) is the measure of the ability of a model's performance at distinguishing between positive & negative classes. The higher the AUC, the better the model. If AUC = 1, the classifier can distinguish between all the positive & negative class points. If AUC = 0, the classifier would prejudice negatives as positives & positives as negatives.

Variable Importance (cont'd)

Ok, you're telling me about ROC & AUC, but what has that got to do with variable importance? Well, that's how the importance of each variable is approximated. For each predictor variable, the AUC is approximated using the trapezoidal rule. The trapezoid rule is like using Riemann sums to approximate the AUC, except it differs in that it uses trapezoids instead of rectangles. From the figures, you can deduce that using the trapezoidal rule is better for getting an accurate AUC. By finding the importance of each variable, we are able to see which variables contribute the most in predicting our outcome variable, Sales.



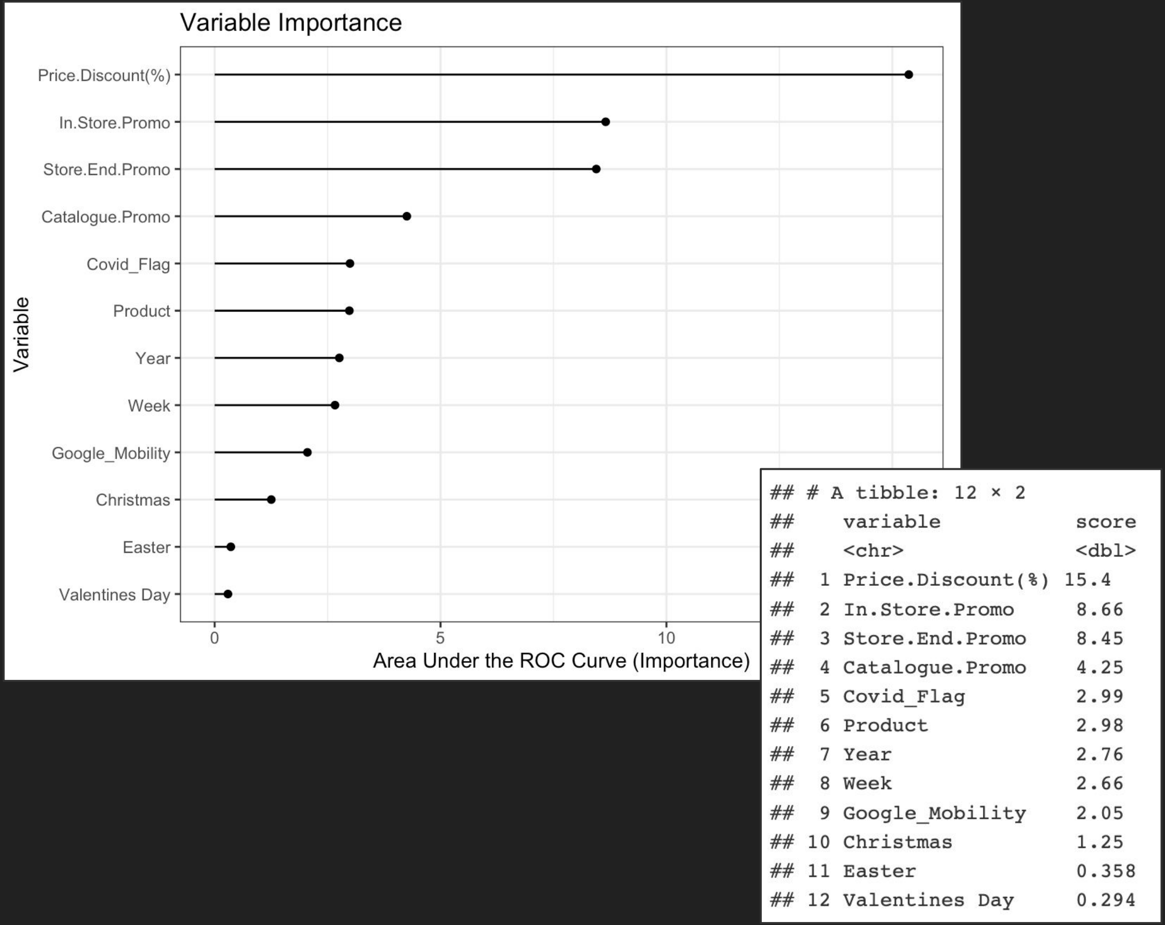


4 circumscribed rectangles

10 circumscribed rectangles

20 circumscribed rectangles

Variable Importance (cont'd)

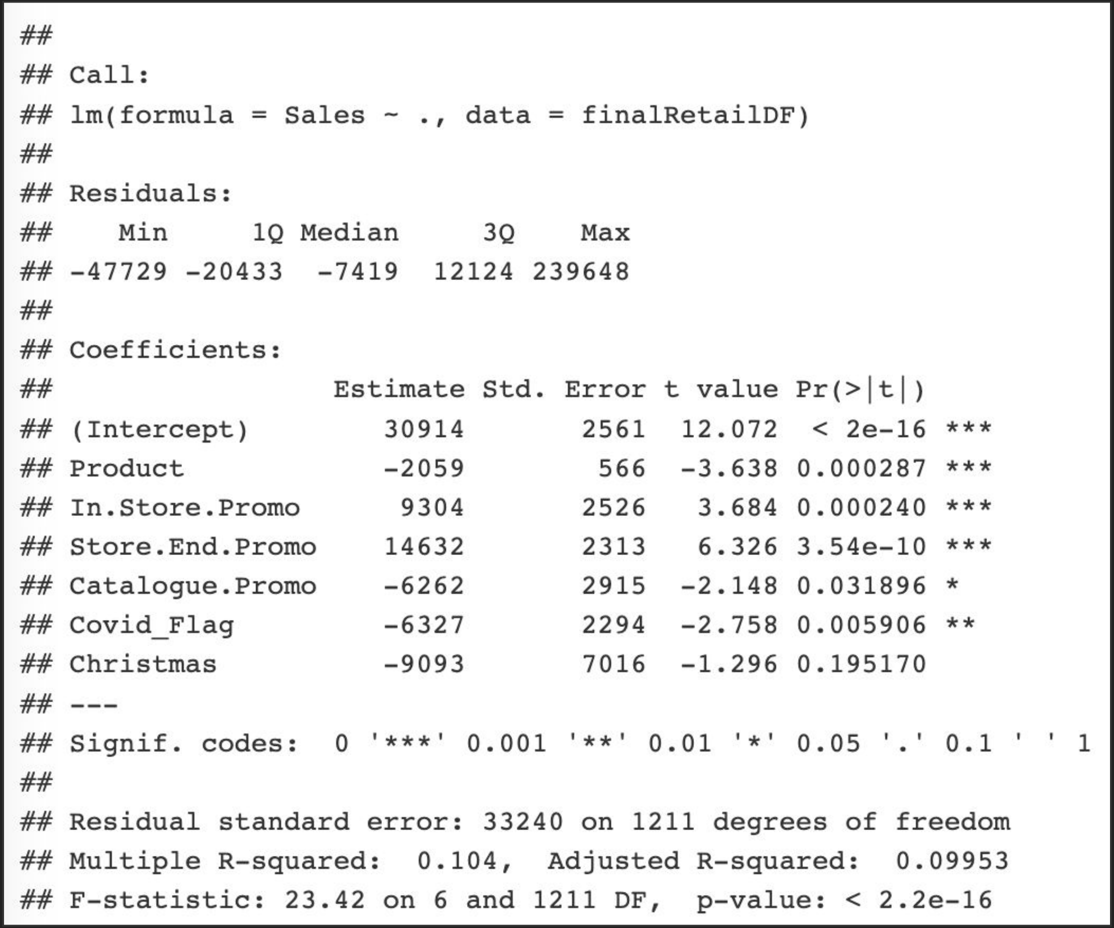


```
## # A tibble: 12 × 2
##    variable            score
##    <chr>               <dbl>
##  1 Price.Discount(%)  15.4
##  2 In.Store.Promo      8.66
##  3 Store.End.Promo     8.45
##  4 Catalogue.Promo     4.25
##  5 Covid_Flag          2.99
##  6 Product             2.98
##  7 Year                2.76
##  8 Week                2.66
##  9 Google_Mobility     2.05
## 10 Christmas           1.25
## 11 Easter              0.358
## 12 Valentines Day      0.294
```

Based on the visualisation, it seems that discounts & promotions are the best predictors of sales. This seems very intuitive, because people are quick to capitalise on good deals. I'm surprised to see Christmas not having higher importance because people do a lot of shopping on the holidays. Valentine's having low sales is also pretty hilarious. This is interesting because the company's statement says that their demand is influenced by various factors like holidays & seasonality, but it doesn't seem to be the case. Perhaps, their promotions & discounts come during certains times of the year, for example, the holidays, which is why we see such a high uptick in importance for those variables, but not the holidays.

Variable Importance (cont’d)

Having created a simple model with some of our most important variables, it does not seem that our selection of variables will help us create a model that best explains most of our variability in Sales. While individually, the important variables are assigned a score that indicates that they contribute a ton to predicting our outcome variable, they do not combine together well to help us build a model that will explain the majority of the variability in sales. Let’s try another feature selection method.



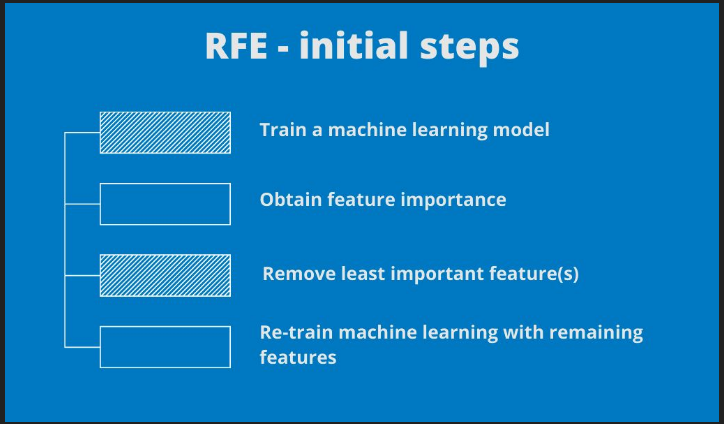
```
##
## Call:
## lm(formula = Sales ~ ., data = finalRetailDF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -47729 -20433  -7419  12124 239648
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)         30914       2561  12.072  < 2e-16 ***
## Product             -2059        566  -3.638 0.000287 ***
## In.Store.Promo       9304       2526   3.684 0.000240 ***
## Store.End.Promo     14632       2313   6.326 3.54e-10 ***
## Catalogue.Promo     -6262       2915  -2.148 0.031896 *
## Covid_Flag          -6327       2294  -2.758 0.005906 **
## Christmas           -9093       7016  -1.296 0.195170
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33240 on 1211 degrees of freedom
## Multiple R-squared:  0.104,  Adjusted R-squared:  0.09953
## F-statistic: 23.42 on 6 and 1211 DF,  p-value: < 2.2e-16
```

Recursive Feature Elimination

Recursive Feature Elimination (RFE) is a feature selection method used to identify a data set's key features (predictor variables). The process involves developing a model with the remaining features after repeatedly removing the least significant parts until the desired number of features is obtained. Here's how the RFE algorithm works:

1) Rank the importance of all features using the RFE machine learning algorithm.
2) Eliminate the least important feature.
3) Build a model using the remaining features.
4) Repeat steps 1-3 until the desired number of features is reached.

How will the algorithm know what is the desired number of features? It won't. This is why we cross-validate our model, which means we resample different portions of the data to test & train the model across different iterations.



RFE - initial steps

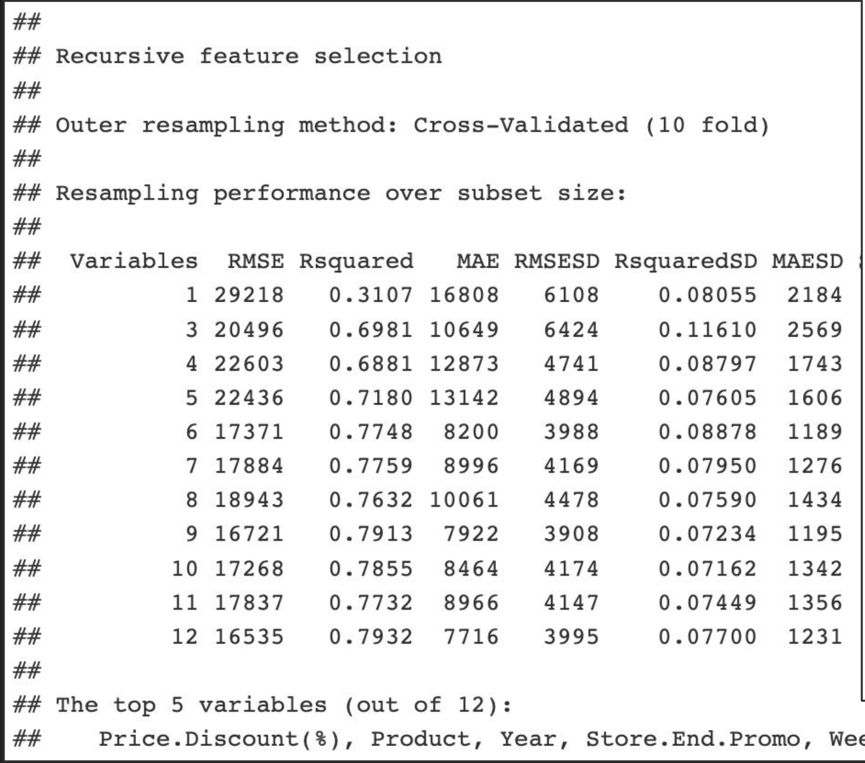
Train a machine learning model

Obtain feature importance

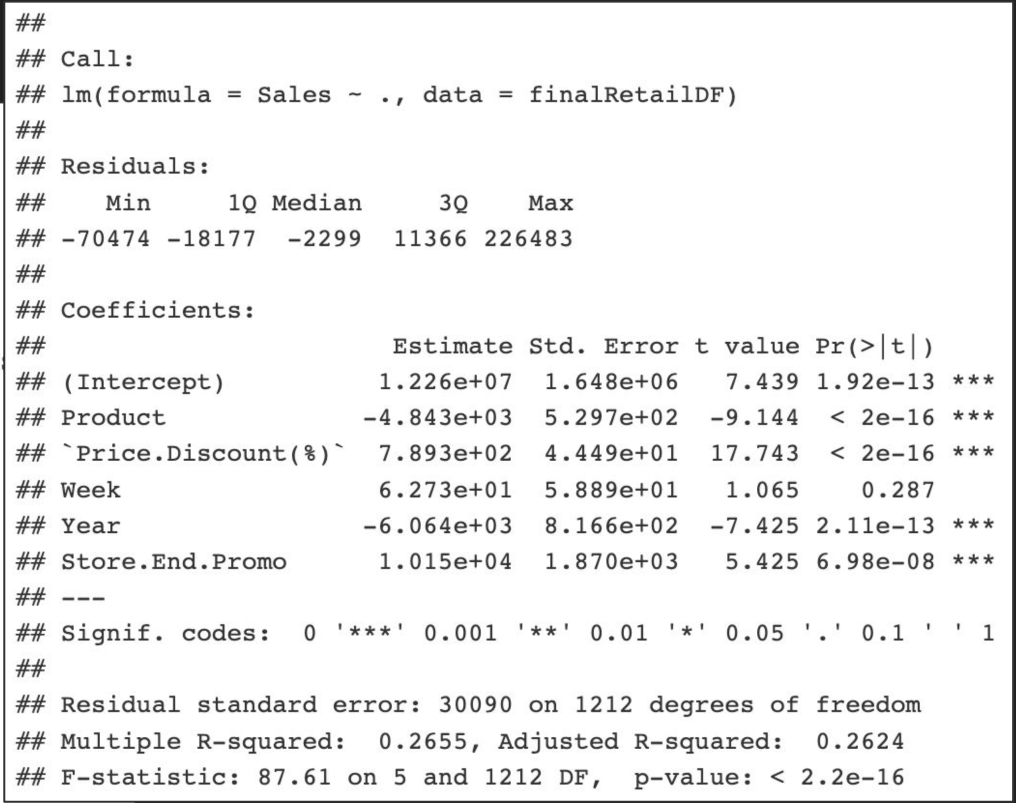
Remove least important feature(s)

Re-train machine learning with remaining features

Recursive Feature Elimination (cont’d)



```
##
## Recursive feature selection
##
## Outer resampling method: Cross-Validated (10 fold)
##
## Resampling performance over subset size:
##
##  Variables  RMSE Rsquared   MAE RMSESD RsquaredSD MAESD
##          1 29218   0.3107 16808   6108    0.08055  2184
##          3 20496   0.6981 10649   6424    0.11610  2569
##          4 22603   0.6881 12873   4741    0.08797  1743
##          5 22436   0.7180 13142   4894    0.07605  1606
##          6 17371   0.7748  8200   3988    0.08878  1189
##          7 17884   0.7759  8996   4169    0.07950  1276
##          8 18943   0.7632 10061   4478    0.07590  1434
##          9 16721   0.7913  7922   3908    0.07234  1195
##         10 17268   0.7855  8464   4174    0.07162  1342
##         11 17837   0.7732  8966   4147    0.07449  1356
##         12 16535   0.7932  7716   3995    0.07700  1231
##
## The top 5 variables (out of 12):
##    Price.Discount(%), Product, Year, Store.End.Promo, Week
```



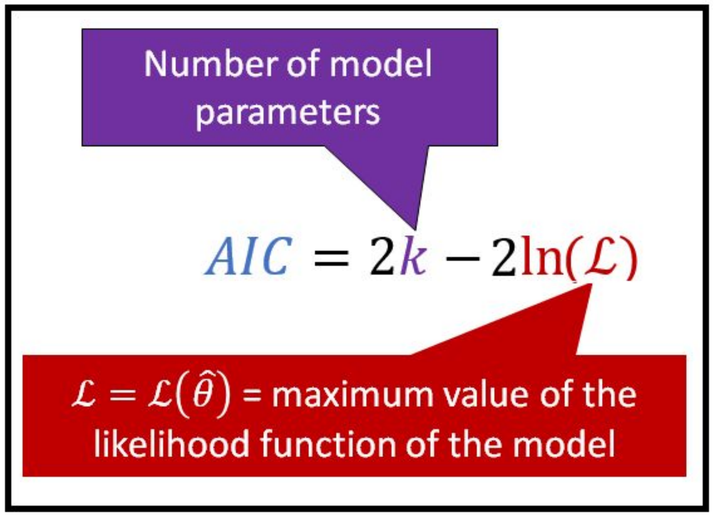
```
##
## Call:
## lm(formula = Sales ~ ., data = finalRetailDF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -70474 -18177  -2299  11366 226483
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.226e+07  1.648e+06   7.439 1.92e-13 ***
## Product             -4.843e+03  5.297e+02  -9.144  < 2e-16 ***
## `Price.Discount(%)`  7.893e+02  4.449e+01  17.743  < 2e-16 ***
## Week                 6.273e+01  5.889e+01   1.065    0.287
## Year                -6.064e+03  8.166e+02  -7.425 2.11e-13 ***
## Store.End.Promo      1.015e+04  1.870e+03   5.425 6.98e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30090 on 1212 degrees of freedom
## Multiple R-squared:  0.2655, Adjusted R-squared:  0.2624
## F-statistic: 87.61 on 5 and 1212 DF,  p-value: < 2.2e-16
```

Recursive Feature Elimination (cont’d)

The result of our cross-validated RFE algorithm states that the best combination of features is Price Discount, Product, year, Promo End, & Week. We popped those variables into a simple model to see if they do well in predicting sales numbers. From the results of the model, it seems that our RFE selected variables created a better model in explaining the variability in Sales than with just variable importance selected variables. Although RFE is better, that does not mean that it is useful. Right, just because a feature is important – just because a model relies on a feature to make predictions, it may not contribute at all to the overall accuracy of the model. Since our model fit isn’t very great, we’ll try another method once again.

Akaike Information Criterion

Much like RFE, AIC iteratively removes predictor variables least significantly related to the outcome variable until all of them are significantly associated to the outcome variable. AIC's goal is to create the best model with the least number of coefficients as possible measuring based on AIC score. This score is a compromise between the quality of the model fit (R^2) & its complexity (number of predictors).

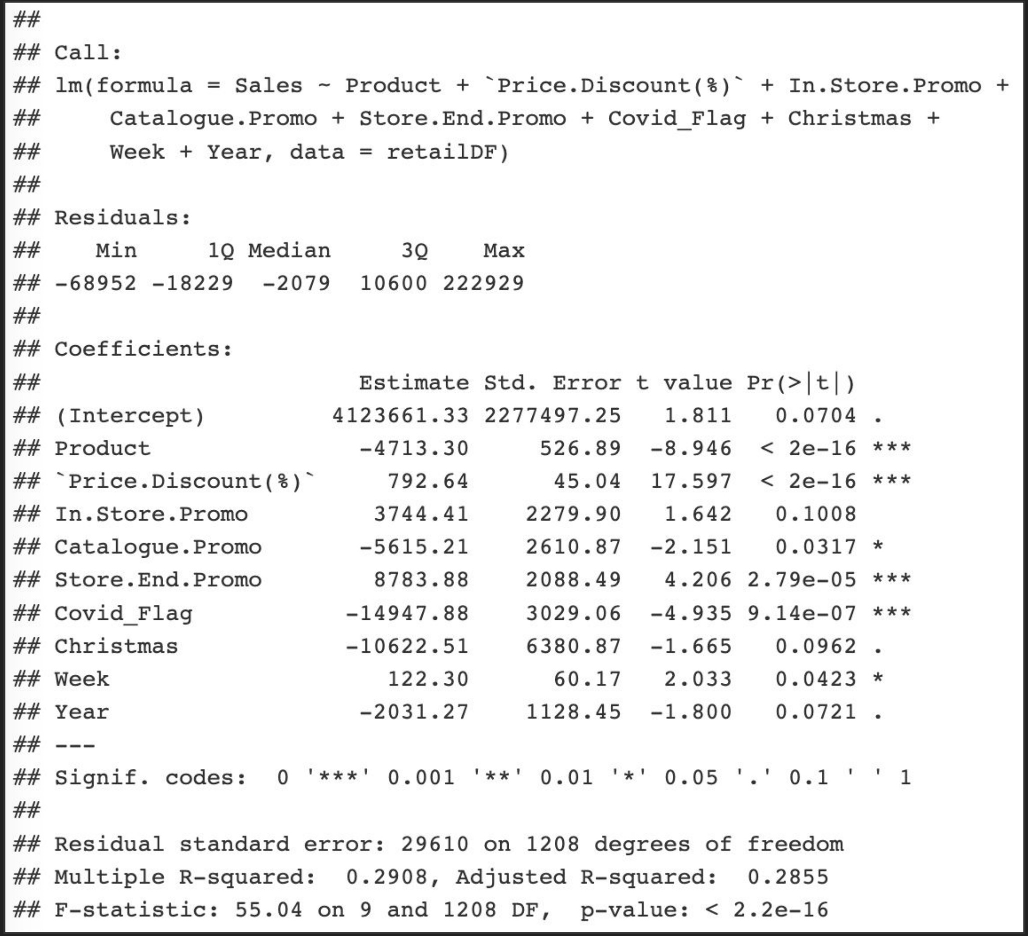


Number of model parameters

$$AIC = 2k - 2\ln(\mathcal{L})$$

$\mathcal{L} = \mathcal{L}(\hat{\theta})$ = maximum value of the likelihood function of the model

Akaike Information Criterion (cont’d)



```
##
## Call:
## lm(formula = Sales ~ Product + `Price.Discount(%)` + In.Store.Promo +
##     Catalogue.Promo + Store.End.Promo + Covid_Flag + Christmas +
##     Week + Year, data = retailDF)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -68952 -18229  -2079  10600 222929
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         4123661.33 2277497.25   1.811   0.0704 .
## Product               -4713.30     526.89  -8.946  < 2e-16 ***
## `Price.Discount(%)`     792.64      45.04  17.597  < 2e-16 ***
## In.Store.Promo         3744.41    2279.90   1.642   0.1008
## Catalogue.Promo       -5615.21    2610.87  -2.151   0.0317 *
## Store.End.Promo        8783.88    2088.49   4.206 2.79e-05 ***
## Covid_Flag           -14947.88    3029.06  -4.935 9.14e-07 ***
## Christmas            -10622.51    6380.87  -1.665   0.0962 .
## Week                    122.30      60.17   2.033   0.0423 *
## Year                  -2031.27    1128.45  -1.800   0.0721 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29610 on 1208 degrees of freedom
## Multiple R-squared:  0.2908, Adjusted R-squared:  0.2855
## F-statistic: 55.04 on 9 and 1208 DF,  p-value: < 2.2e-16
```

The results of our AIC selected model builds on our variance importance selected model & adds Year & Week. The week & year features may aid in explaining the change in sales numbers throughout the holidays & seasons, since they would be able to capture the time of them. Our model fit is also the best so far.

Model Selection

Since most of our variables are binary/categorical, & our outcome variable is continuous, we would want a regression model. There are various regression algorithms; here are just some of them:

1) Linear regression
2) Ridge regression
3) LASSO regression
4) Elastic Net regression
5) k-Nearest neighbours
6) Decision trees with Random Forest Boosting
7) XGBoost

My recommendation is to use decision trees with boosting because the categorical variables make the decision tree easier to understand & to reduce overfitting.

THANK YOU