Forecasting Automobile Sales with Linear Regression

Elias Castro Hernandez - Fall 2017

Forecasting Hyundai Elantra Sales (Adapted from Bertsimas 22.1)

Nearly all companies seek to accurately predict future sales of their product(s). If the company can accurately predict sales before producing the product, then they can better match production with customer demand, thus reducing unnecessary inventory costs while being able to satisfy demand for their product.

In this exercise, we seek to predict the monthly sales in the United States of the Hyundai Elantra. The Hyundai Motor Company is a major automobile manufacturer based in South Korea. The Elantra is a car model that has been produced by Hyundai since 1990 and is sold all over the world, including in the United States. We will use linear regression to predict monthly sales of the Elantra using economic indicators of the United States as well as (normalized) Google search query volumes. The data for this problem is contained in the file **Elantra142-Fall2017.csv.**

Each observation in the file is for a single month, from January 2010 through July 2017. The variables are described in **Table 1**.

Table 1: Variables in the dataset `Elantra142-Fall2017.csv`.

Variable	Description
`MonthNumeric`	The observation month given as a numerical value (1 = January, 2 = February, 3 = March, etc.).
`MonthFactor`	The observation month given as the name of the month (which will be a factor variable in R).
`Year`	The observation year.
`ElantraSales`	The number of units of the Hyundai Elantra sold in the United States in the given month and year.
`Unemployment`	The estimated unemployment rate (given as a percentage) in the United States in the given month and year.
`ElantraQueries`	A (normalized) approximation of the number of Google searches for "hyundai elantra" in the United States in the given month and year.
`CPI.All`	The consumer price index (CPI) for all products for the given month and year. This is a measure of the magnitude of the prices paid by consumer households for goods and services.
`CPI.Energy`	The monthly consumer price index (CPI) for the energy sector of the US economy for the given month and year.

Procedure

We start by splitting the data into a training set and testing set. The training set will contain all observations for 2010, 2011, 2012, 2013, and 2014. The testing set will have all observations for 2015, 2016, and 2017.

To begin, we will consider just the four independent variables *Unemployment, ElantraQueries, CPI.Energy*, and *CPI.All*. Using regression in an iterative way, we will then choose a subset of these four variables and construct a regression model to predict monthly Elantra sales (*ElantraSales*). Based on plots, and Variable Inflation Factor (VIF) analysis, we will choose which of the four variables to use in what will be our initial (naive) linear regression model.

Evaluation and results analysis of naive model will be carried out. The results of our naive model analysis will be used in the development of an improved model. This process will be carried out until finally, we will develop a combined model that considers user-selected variables and the efficacy of any prior model in its structure.

Naive Model

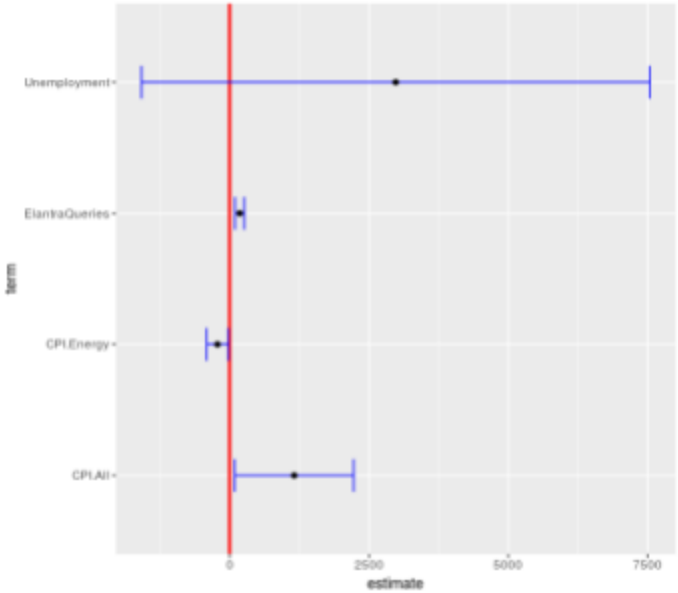
I. The regression equation produced by the naive model follows:

$$Predicted\ Elantra\ Sales\ =\ \beta_0\ +\ \beta_1 * ElantraQueries\ +\beta_2 * CPI.All$$
$$Where\ \beta_i\ are\ our\ estimated\ coefficients\ for\ i = 1, 2\ and\ \beta_0\ is\ the\ intercept$$

Note: When interpreting the coefficients, since the relationship is linear, one can hold all other coefficients as constant, and observe the effects of a unit change on the predictor (say *CPI.All*), and the subsequent effect on the response (*Predicted Elantra Sales*).

II. How variables were selected, and what they tell us

In order to decide which variables to remove, both a confidence interval plot, and Variance Inflation Factors (VIF) were calculated. Variables whose confidence interval overlapped the null hypothesis (that the variable has no effect on the model) where removed, as well as those with a VIF score greater than or equal to 5.




```
> vif(ElantraSales)
Unemployment ElantraQueries      CPI.Energy          CPI.All
   42.988397       2.317007      10.923366       67.437812
```

It makes sense to remove variables due to collinearity despite there being so few. This is because multicollinearity affects the t-statistic, which is in turn used to calculate the p-value, which is itself used to evaluate our hypothesis test - the probability of correctly detecting a non-zero coefficient. Meaning, we may end up accepting variables that we should have discarded.

The coefficients for the intercept, *ElantraQueries*, and *CPI.All* are all significant. We deduced this by backward selection, based on the largest observed p-value (or alternatively, with the lowest number of stars, which are a measure of deviation from the mean - see red box bellow)

```
lm(formula = ElantraSales ~ Unemployment + ElantraQueries + CPI.Energy +
    CPI.All, data = elantra.train)

Residuals:
   Min     1Q Median     3Q    Max
 -6491  -2003   -647   2308   7592

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -232322.14  121172.68  -1.917   0.0604 .
Unemployment     2979.13    2276.31   1.309   0.1961
ElantraQueries    181.34      40.17   4.514  3.4e-05 ***
CPI.Energy       -217.42      96.35  -2.257   0.0280 *
CPI.All          1158.50     532.19   2.177   0.0338 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3312 on 55 degrees of freedom
Multiple R-squared:  0.555,     Adjusted R-squared:  0.5226
F-statistic: 17.15 on 4 and 55 DF,  p-value: 3.477e-09
```

Additionally, we can assume that the signs of our coefficients are correct. This is because one would assume that a consumer looks up a vehicle prior to purchase (+ *ElantraQueries*). Additionally, *CPI.All* is consumer price index, which is a measure of inflation - the purchasing power of a dollar. One would expect, a large CPI to lead to lesser sales (but still positive sales), while a low CPI to lead to higher sales (still positive). Finally, we see that the intercept is negative, since it is the intercept for a linear relationship that could not have a positive intercept (since no consumer in their effort to buy a new Elantra, must first purchase several other prior to doing so)

III. How well does the model predict training set observations?

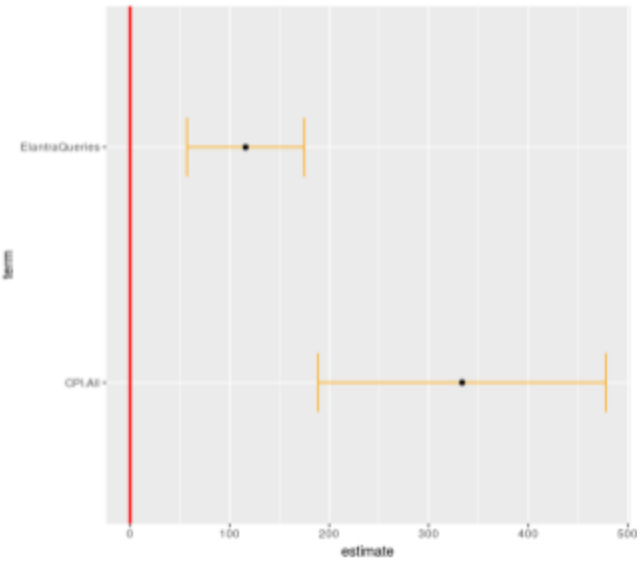
It appears that our model could be better.

```
lm(formula = ElantraSales ~ ElantraQueries + CPI.All, data = elantra.train

Residuals:
   Min     1Q Median     3Q    Max
 -6319  -2625   -380   1800   7867

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -67351.41   15869.82  -4.244 8.17e-05 ***
ElantraQueries   116.09      29.40   3.949 0.000218 ***
CPI.All          333.59      72.19   4.621 2.23e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3418 on 57 degrees of freedom
Multiple R-squared:  0.5088,    Adjusted R-squared:  0.4915
F-statistic: 29.52 on 2 and 57 DF,  p-value: 1.591e-09
```



Although our **F-statistic is 29.52**, which is indicative of a clear relationship between predictors and response, our coefficient of determination causes us to pause. Also note:

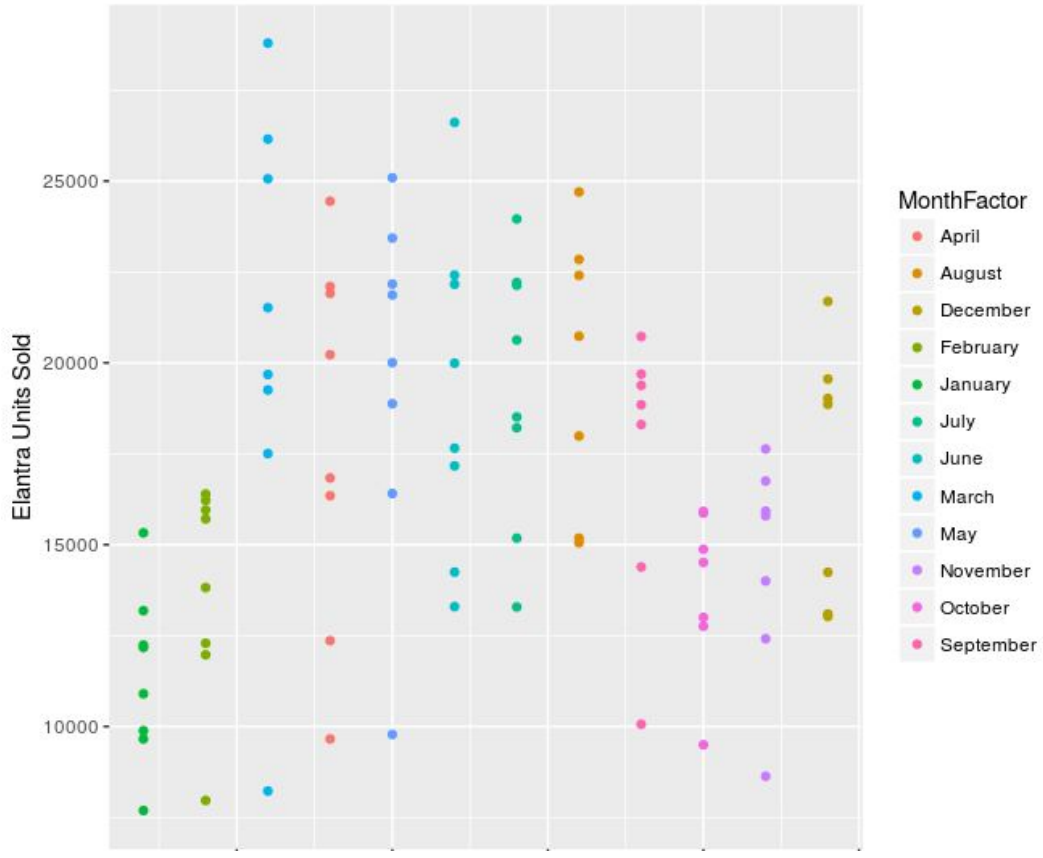
Multiple R-squared: 0.5088, Adjusted R-squared: 0.4915

This measure tells us that only about 50% of the time, our regression line approximates the true data points. Additionally, our Out of Sample Coefficient of Determination (OSR^2) is smaller than the coefficient of determination for the testing data; indicating that we have an overfitting problem.

Seasonal Model

We now try to further improve the linear regression by including seasonality into our model. In predicting demand and sales, seasonality is often very important since demand for most products tends to be periodic in time. For example, demand for heavy jackets and coats tends to be higher in the winter, while demand for sunscreen tends to be higher in the Summer. WIth that consideration in mind:

A plot of sales over years, colorized by month, shows a seasonal trend. Once this was confirmed, the same procedure for feature selection on the naive model was used in the seasonal model. Similar to the above, the coefficients are the scalar by which a unit change in a variable -- holding all others equal -- will affect the response.



However, in this case, the *MonthFactor* variables are subject to the following rule:

$$Predicted\ Elantra\ Sales\ =\ \beta_0\ +\ \sum_{i=1}^{12} \beta_i MonthFactor_i\ +\beta_{13} * ElantraQueries\ +\beta_{14} * CPI.All$$

$$for\ i = \{1 = Jan,\ 2 = Feb, \ldots,\ 12\ =\ Dec\}$$

Where

$\beta_0 =$ *intercept*

$\beta_i =$ *estimated coefficient for month i*

$\beta_{13} =$ *estimated coefficient for ElantraQueries*

$\beta_{14} =$ *estimated coefficient for CPI.All*

That is, our model uses *dummy variables (0,1*) and only considers the coefficient of *MonthFactor*, if it relates to the month in question -- *e.g.* if January, *Monthfactor* coefficient for January multiplied by one else it is multiplied by zero

$$for\ i\ in\ MonthFactor = \{1 = Jan,\ 2 = Feb, \ldots,\ 12\ =\ Dec\}$$

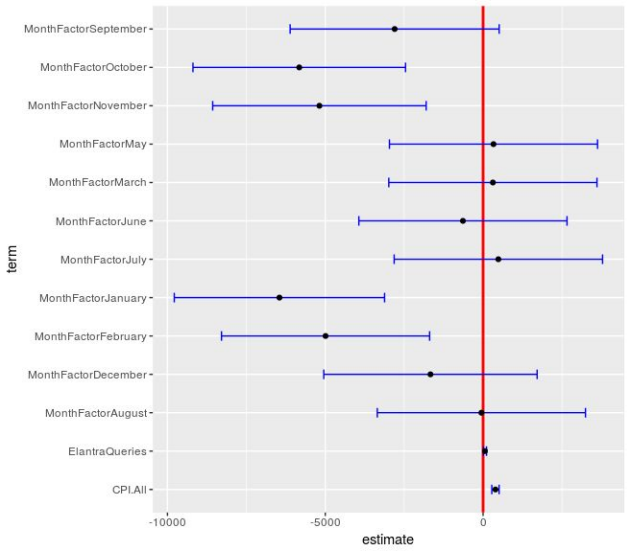
$$If\ MonthFactor\ = i:$$

$$\beta_i *\ 1\ else\ \beta_i * 0$$

I. Evaluating the Seasonal Model:

```
Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         -74707.81   12431.95  -6.009 2.80e-07 ***
MonthFactorAugust       -53.75    1638.56  -0.033 0.973973
MonthFactorDecember   -1667.46    1679.18  -0.993 0.325895
MonthFactorFebruary   -4990.98    1637.96  -3.047 0.003820 **
MonthFactorJanuary    -6452.65    1655.34  -3.898 0.000313 ***
MonthFactorJuly         484.35    1640.15   0.295 0.769088
MonthFactorJune        -638.14    1638.54  -0.389 0.698738
MonthFactorMarch        310.97    1638.49   0.190 0.850306
MonthFactorMay          329.74    1637.85   0.201 0.841334
MonthFactorNovember   -5185.64    1680.11  -3.086 0.003425 **
MonthFactorOctober    -5824.86    1672.96  -3.482 0.001103 **
MonthFactorSeptember  -2802.01    1645.10  -1.703 0.095271 .
ElantraQueries           58.88      24.91   2.364 0.022366 *
CPI.All                 391.97      57.27   6.844 1.56e-08 ***

Residual standard error: 2589 on 46 degrees of freedom
Multiple R-squared:  0.7725,    Adjusted R-squared:  0.7082
F-statistic: 12.02 on 13 and 46 DF,  p-value: 9.218e-11
```



Note that adding the *MonthFactor* variable indeed improved our model. One can look at Multiple R-Squared on both models and see an improvement (R-squared is closer to 1) in the seasonal model. Additionally, the (OSR^2) also improved. However, it should be observed that there is a glaring fault with the seasonality application, and it is that it considers months as seasons. Car sales tend to experience seasonalities around quarter and/or sales events. That is to say, events such as the holiday season, and after tax season (labor day sales), tend to more accurately show increased sales, while summer break tends to show a decrease. A seasonality model, specific to the car market would more aptly suit our efforts.

Combined Naive and Seasonal Model

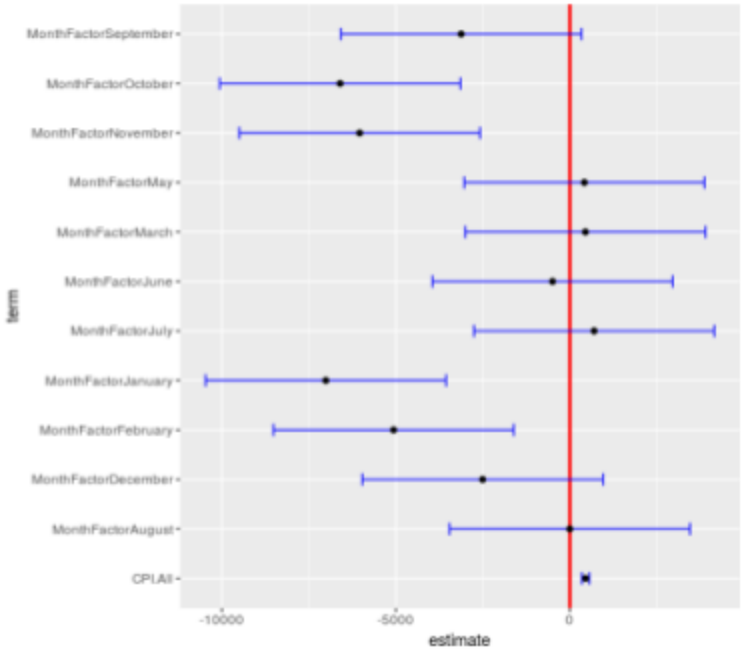
We now build a model using a subset of the independent variables used in the naive and seasonal models previously constructed.

```
lm(formula = ElantraSales ~ MonthFactor + CPI.All, data = elantra.train)

Residuals:
    Min      1Q  Median      3Q     Max
-6156.5 -1219.3   101.8  1486.3  5118.4

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       -84309.214  12309.792  -6.849 1.39e-08 ***
MonthFactorAugust      -2.965   1716.526  -0.002 0.998629
MonthFactorDecember -2505.800   1719.537  -1.457 0.151696
MonthFactorFebruary -5063.524   1715.739  -2.951 0.004924 **
MonthFactorJanuary  -7011.472   1716.473  -4.085 0.000170 ***
MonthFactorJuly       696.156   1715.771   0.406 0.686775
MonthFactorJune      -496.864   1715.510  -0.290 0.773373
MonthFactorMarch      446.327   1715.550   0.260 0.795872
MonthFactorMay        416.636   1715.490   0.243 0.809165
MonthFactorNovember -6039.347   1719.049  -3.513 0.000990 ***
MonthFactorOctober  -6600.706   1718.645  -3.841 0.000366 ***
MonthFactorSeptember -3122.468  1717.664  -1.818 0.075463 .
CPI.All               452.004     53.777   8.405 6.37e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2712 on 47 degrees of freedom
Multiple R-squared:  0.7449,    Adjusted R-squared:  0.6798
F-statistic: 11.44 on 12 and 47 DF,  p-value: 3.043e-10
```



I. Evaluating the Combined Model:

Observing the statistical metrics outputted by our combined model, we see that we did not improve on our prior models. Recall that Total Sum of Squares (TSS) is the amount of variability inherent in the response prior to regression being performed. While Residual Sum of Squares (RSS) is a measure of the variability left unexplained after regression. This implies that $TSS - RSS$ is a measure of the amount of variability in the response that is explained (or removed) by regression. Since R^2 is a proportion of variances, independent of the scale of the response, then we clearly get the following:

$$R^2 = \frac{TSS-RSS}{TSS} = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}$$

By observing OSR^2 through the given relationship, we see that all of our models are flawed. $RSS > TSS$ shows us that our regression does not explain much of the variability in the response. But perhaps most importantly, since OSR^2 has been negative for all of our models, and backward feature selection has only served to make that negative number smaller, we can assume one or both of the following:

1) Our linear model is wrong
2) There is an inherent error in our data, meaning variance will always be high

Toward and Improved Model

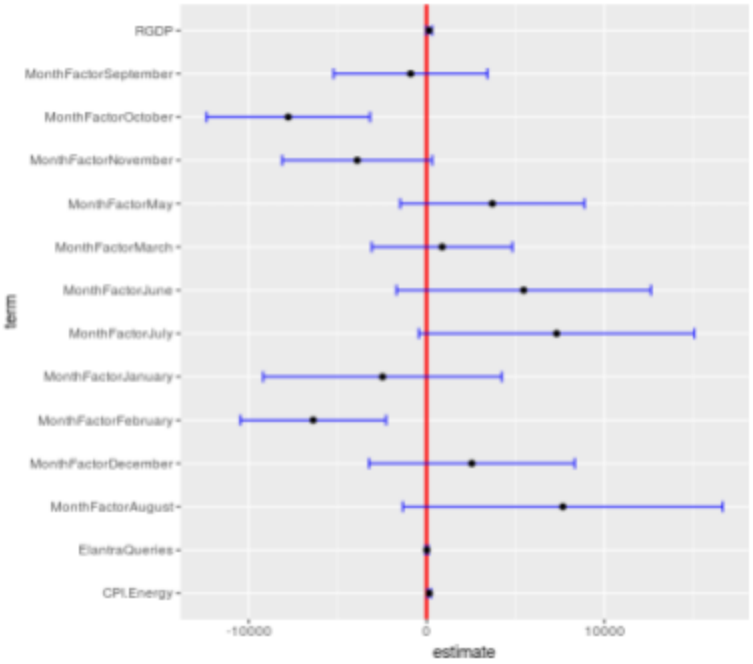
We now build a final model by incorporating an additional feature selected by the user based on intuition. In particular, I chose to include Real Domestic Gross Product (RGDP) as my new variable. I did this under the assumption that since RGDP is an inflation-adjusted measure of the total spending (consumers, industry, government and excess of exports over imports), we would see automobile purchases be strongly linked to this metric.

```
lm(formula = ElantraSales ~ MonthFactor + ElantraQueries + CPI.Energy +
    RGDP, data = elantra3.train)

Residuals:
    Min      1Q  Median      3Q     Max
-6336.2 -1323.1   -78.5  1395.5  6077.4

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -27975.97    8606.37  -3.251  0.00218 **
MonthFactorAugust    7668.23    4467.48   1.716  0.09296 .
MonthFactorDecember  2542.66    2874.98   0.884  0.38118
MonthFactorFebruary -6381.54    2036.67  -3.133  0.00304 **
MonthFactorJanuary  -2478.41    3333.07  -0.744  0.46100
MonthFactorJuly      7315.25    3848.12   1.901  0.06372 .
MonthFactorJune      5462.24    3555.66   1.536  0.13149
MonthFactorMarch      874.17    1965.74   0.445  0.65867
MonthFactorMay       3698.85    2574.41   1.437  0.15770
MonthFactorNovember -3904.16    2094.15  -1.864  0.06881 .
MonthFactorOctober  -7787.71    2287.38  -3.405  0.00140 **
MonthFactorSeptember  -896.45    2150.93  -0.417  0.67882
ElantraQueries          13.20      44.00   0.300  0.76552
CPI.Energy             150.92      48.13   3.136  0.00302 **
RGDP                   140.54      82.95   1.694  0.09710 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3071 on 45 degrees of freedom
Multiple R-squared:  0.6868,    Adjusted R-squared:  0.5894
F-statistic: 7.048 on 14 and 45 DF,  p-value: 2.271e-07
```



It should be noted that after the inclusion of RGDP into the data, we see OSR^2, R^2 and F-statistic decrease. VIF analysis per feature also worsened compared to every previous model.

	GVIF	Df	GVIF^(1/(2*Df))
MonthFactor	23.132495	11	1.153480
ElantraQueries	3.230337	1	1.797314
CPI.Energy	3.168864	1	1.780130
RGDP	17.932650	1	4.234696

Through backward feature selection, my model quality metrics quickly showed that RGDP as an unreliable predictor.

Knowing that RGDP does not improve the predictive capacity of our model, I chose to use the combined model for forecasting purposes -- since it considers seasonality, and also had the best model accuracy metrics of all models tested.

Forecasting Unit Sales

In order to compute the predicted sales for August 2017, a linear model was constructed using the estimated coefficients from the combined model, ran on the training set. All *MonthFactors*, with the exception of August (times 1) were multiplied times zero. Finally, all estimates for relevant features (*CPI.All*, and *ElantraQueries*) were taken as averages, due to the relatively stable (low variance) numbers over time. Following is the prediction:

```
# Average number of elantra queries for Jan/17-Jul/17
i <- 92
# CPI average for Jan/17-Jul/17
j <- 244.03
# Actual sales
act <- 15127

Aug2017_ES_combined <- ((-84309.21) - 7011.47*(0) - 5063.52*(0) + 446.33*(0) + 416.636*(0) - 496.86*(0) + 696.16*(0)
                                    - 2.97*(1) - 3122.47*(0) - 6600.71*(0) - 6039.35*(0) - 2505.80*(0) + 452.00*(j))
```

I. Evaluating the Quality of Forecasting Model

The predicted sales and absolute error for the model follows:

```
> Aug2017_ES_combined
[1] 25989.38
> Error <- abs(Aug2017_ES_combined - act)
> Error
[1] 10862.38
```

Based on the given, our model has odds of predicting the right number approximately equal to 30%, and is thus not a very good model. However, this was expected.

Please recall from our evaluation of the combined model that:

$$R^2 = \frac{TSS-RSS}{TSS} = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2}$$

By observing OSR^2 through the given relationship, we see that all of our models are flawed. $RSS > TSS$ shows us that our regression does not explain much of the variability in the response. But perhaps most importantly, since OSR^2 has been negative for all of our models, and backward feature selection has only served to make that negative number smaller, we can assume one or both of the following:

1) Our linear model is wrong
2) There is an inherent error in our data, meaning variance will always be high

As the saying goes, “garbage in, garbage out”.

Conclusion

As this project shows, one cannot overstate the importance of one's quality-of-data. In our case, we began with insufficient data, which meant that our predictors where too few in number to actually provide any predictive power to our linear regression model. In future iterations of this project, larger number of variables, as well as more observations should be included in our data set. Doing so will increase our R^2 metrics, as well as increase the likelihood of our model predicting the correct number of vehicles sold at a particular point in time.