Forecasting Jeep Wrangler Sales

Introduction

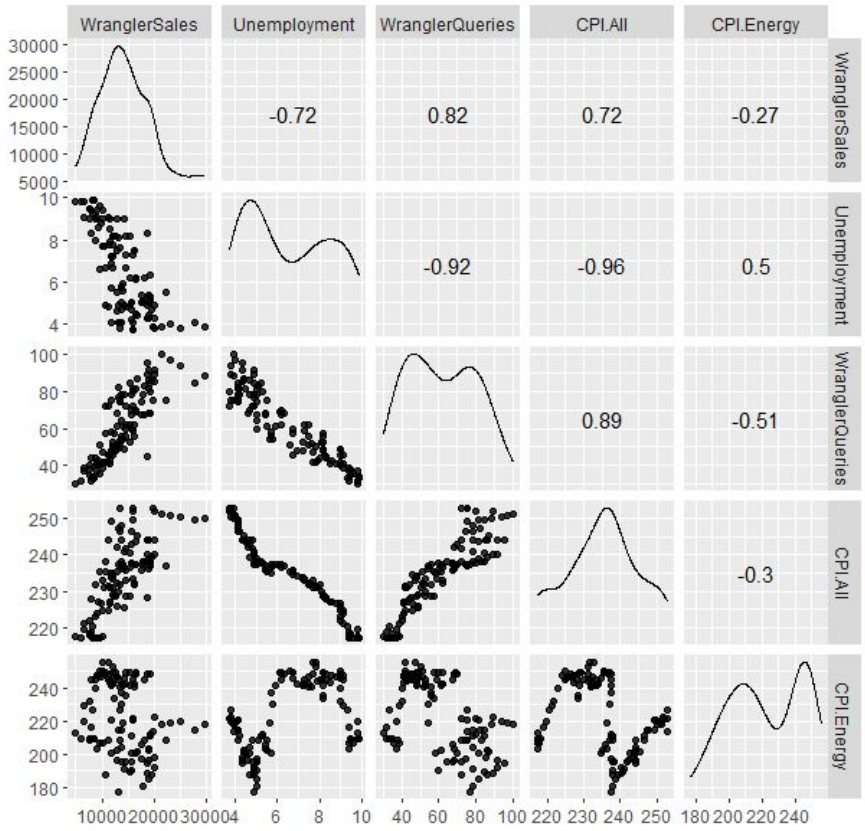
Nearly all companies seek to accurately predict future sales of their product(s). If the company can accurately predict sales before producing the product, then they can better match production with customer demand, thus reducing unnecessary inventory costs while being able to satisfy demand for their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

In this report, we try to predict the monthly sales in the United States of the Jeep Wrangler automobiles. Jeep is brand of American automobiles that is a division of the ItalianAmerican corporation Fiat Chrysler Automobiles (FCA). The Wrangler is a car model that has been produced since 1986, with most of its sales in the United States. We will use linear regression to predict monthly sales of the Wrangler 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 Wrangler242-Spring2019.csv. Each observation in the file is for a single month, from January 2010 through December 2018.

Exploratory Data Analysis

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.
`WranglerSales`	The number of units of the Hyundai 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.
`WranglerQueries`	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.

We first check the VIF to check for multicollinearity. We realise that the VIF of the variable CPI.ALL is 69 and is way too high, so we remove it. Upon removing it, the highest VIF falls to 4, which is okay. After checking the performance of the model again (model2), we obtain an R2 of 0.7944, still good. We also notice that the parameter for the variable CPI.Energy is now positive, which might be counter-intuitive, given that WranglerSales and CPI.Energy are negatively correlated. We still have only one significant variable, WranglerQuieries. It seems like the variable Unemployment is the least significant. We try plotting the confidence intervals and see that Unemployment is clearly insignificant, while the confidence interval for CPI.Energy contains 0, but seems to be much more significant.



In model3, we remove it and check the model's performance again. The results did not really improve and are very similar, with an R2 of 0.7933. This will be our chosen model for the Training Set. It is important to note that while some of the earlier models performed slightly better, we are only assessing performance on a training set so far.

Training & Models

The equation for the linear regression model is:

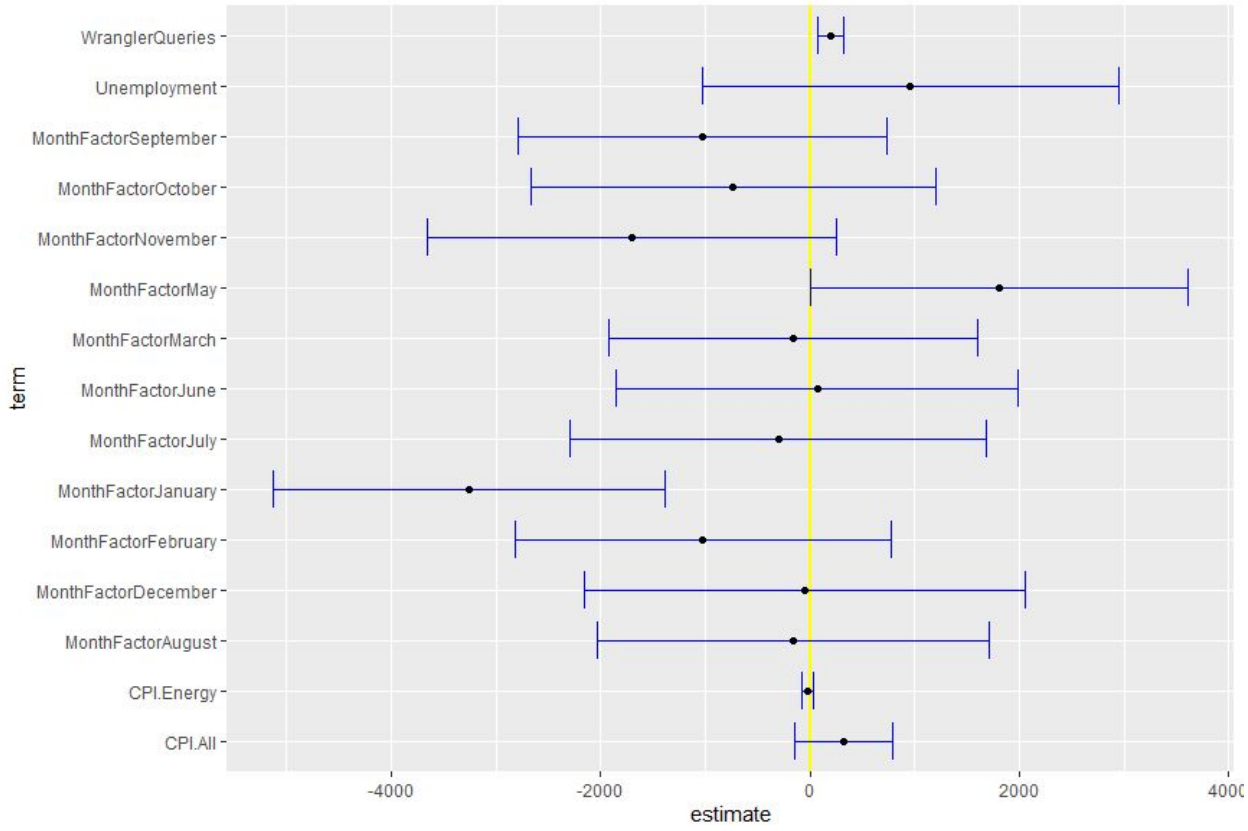
$$WranglerSales = -4463.50 + WranglerQueries*262.34 + CPI.Energy*14.95$$

The coefficients of each variable represent the incremental change in WranglerSales if we increase that variable by one unit. We can see that increasing one unit increase of WranglerQueries would affect a lot more our sales, in a positive way, than one unit increase of CPI.Energy. Thus, the number of WranglerQueries and Sales of Wrangler are highly related.

The new model takes into account the month in which the values were recorded. This lets the regression model have more flexibility and be able to represent seasonality, with the sales really varying between periods of the year. Thus, the equation obtained is

$$\begin{aligned} WranglerSales = &- 71969.75 - 158.57*MonthFactorAugust - 50.31*MonthFactorDecember \\ &- 1021.34*MonthFactorFebruary - 3256.91*MonthFactorJanuary - 303.87*MonthFactorJuly \\ &+ 67.03*MonthFactorJune - 158.79*MonthFactorMarch + 1806.86*MonthFactorMay \\ &- 1701.16*MonthFactorNovember - 731.70*MonthFactorOctober - 1028.08*MonthFactorSeptember \\ &+ 959.91*Unemployment + 192.08*WranglerQueries - 22.78*CPI.Energy \\ &+ 318.26*CPI.All \end{aligned}$$

In this model, the parameters associated with the new dummy variables let us directly know how much is the effect of the observation month on the total sales. It also lets us disaggregate this data and refine the WranglerSales, such that monthly observations have different weights, and can be turned off/on using the binary variables representing the months. This is very powerful since we now have a model that kind of takes into account 12 models (one for each month), such that the data may be linear in similar periods (seasons) of the year, but may really vary otherwise in the same year in between seasons, in a non-linear fashion.



The introduction of categorical variables has thus greatly improved the model's performance on the Training Set for the reasons mentioned above, and this can be seen from R2=0.873, which is excellent. We can see from the corresponding p-values that the January Month Factor and the WranglerQueries is significant, while variables such as MonthFactorMay and MonthFactorNovember are slightly insignificant, and would be significant if we had chosen a slightly higher p-value threshold of 10 percent (with p-values of 5.02 percent and 8.6 percent respectively).

Further Iterations

Another method could involve introducing an actual time variable (not categorical, say August 2010 would be represented as 08-01-2010) such that we could use the same regression analysis to forecast the WranglerSales but using the time period itself as an independent variable. I think this method might be able to work slightly better than the current one, although the performance of the current model on the training set is good. Other methods, could involve trying to remove or model the trend and seasonality by weighted moving average or exponential smoothing. I think these models could perform well if the future WranglerSales doesn't deviate too much, but would perform worse than the first method, if just implemented alone (if we don't use any features, just work on the data itself). Also, another method that is used for time series forecasting is introducing a time lag in the features data, such that we can train on the shifted data and predict on the remaining periods to get a value for the dependent variable in the future, usually done in parallel with other models like the moving average or exponential smoothing.

Evaluation

In the previous section, we chose all four of the so-called independent variables and all categorical MonthFactors. Since it is not the goal of this exercise to tweak the categorical variables, and because we know already from the previous analysis that some of the variables have multicollinearity, we will remove the most critical again. Since we do not have a measure of VIF (but GVIF instead) to check for multicollinearity, we just remove the variables previously heavily correlated with other variables, such as Unemployment and CPI.All, which had a really high VIF before (if they are already correlated with other variables, the introduction of categorical variables will not change that fact). The final model has the following variables, chosen for the reasons mentioned above:

Month Factor (all 12 months) Wrangler Queries CPI.Energy

The performance of the model was good on the Training Set with an R2=0.868. The performance of the model was mediocre for the Test Set, with an OSR2=0.24. However, we computed the OSR2 of the previous model created in c), which, even though has higher R2, has an OSR2 of near 0! This shows that the first model in c) is subject to severe overfitting, and that this overfitting is being addressed in the current final model by removing some correlated variables, although still massively present (OSR2 with bad performance << R2 with good performance). The model might be further improved by tweaking the individual MonthlyFactors.