Regression and Cross Validation

CISC 271 – Assignment 3

Amaar Jivanji

Abstract

The purpose of this investigation was to determine whether Copper can be used as a proxy when assessing the global economy and commodity prices. The investigation was performed on a data set gathered from the U.S Federal Reserve Bank of St. Louis Economic Data website containing the world prices of a select subset of commonly traded basic commodities. The process involved finding the commodity that when chosen as the y vector, has the lowest RMS error of fit when the remaining variables are gathered into a data matrix. It was observed that the variable that provided the best regression was Copper and the reliability of this conclusion was tested by performing a 5-fold cross-validation on the given data. It was found that the model/regression generated by using Copper as the dependent variable, did precisely well at approximating new data however, the way you interpret the mean and standard deviations of the errors can vary depending on what you expect from the model.

Introduction:

The scientific exploration aimed to discover whether a model can be generated by recognizing a relationship between a chosen independent variable in a linearly correlated set and the remaining independent variables.

The motivation for this exploration was the supposition that the price of Copper is a proxy for the price of other commonly traded basic commodities. That is, can we use the trend of Copper prices as an indicator when looking at the trend of commodity prices in general? Do rising Copper prices indicate that other commodity prices on average are also on the incline?

Therefore, the technical problem of the investigation was to validate whether Copper is the variable that is 'best' explained by the other 15 variables. This process involved recognizing a pattern that would best fit a curve and give the best approximation to the real data. The process to find this pattern is called linear regression. Linear regression aims to solve for a weight vector w, such that when multiplied by a given data matrix A, results in a linear model C that approximates closely to the given data B. The vector B would be the independent variable that we are aiming to fit the model against. For this investigation, each of the 16 commodities were selected as the B vector on 16 different iterations, and the remaining 15 variables were gathered into a data matrix A. The data matrix A had the commodities as variables and the pricing data as observations. The quality of each of the regressions was examined my measuring the overall error, that is the RMS value of the residual error vector. The residual error vector contains the differences between the approximated values and the given values. Hence, the regression with the lowest RMS value suggested that the given model provides the best fit to the data when compared to other models.

Once, the variable that provides the best solution to the regression problem is found, how do we evaluate how well the approximation performs? Until now, we just know which variable is the best among the other variables in creating a model. The conceptual difficulty is that when computing the regression coefficients, we used the entire data to estimate the residual error. To test the goodness of the fit, a 5-fold cross validation was performed on the given data. This process involves dividing the data to two sets, a training and a testing set. The training set contains 4/5 of the data and the testing set would contain the remaining 1/5 of the data. The data from the training set would be used to compute the weight vector of the linear regression and then be applied to the data from the testing set, generating a model. This model is then compared to the observations of the selected independent variable by looking at the RMS value of the residual error. By comparing the differences between the RMS values of training and testing, we can determine how good the model is at predicting and generalizing to new data.

Method:

The data from the csv file is loaded to MATLAB and the first row; names of the commodities, and the first column; recorded dates need to be skipped. The data is real so not all the values were available when the data was gathered. These missing values will be reported as zeros when the data is initially loaded. To fill the missing values, the function F = fillmissing(A,movmethod,window) was used. The movmethod specified was 'movmean' with a window length of 20. This method fills missing entries using a moving window mean which calculates the local average over the elements that fill that window specified length. This method resulted in values that were better fit to the data as compared to filling the values with a constant or filling the entries with the previous non-missing entries.

The data matrix was standardized using the zscore function so that each column has a zero mean and a unit variance. It was observed that the variables have different scales that would not contribute equally to the analysis and end up creating a bias. For example, the ranges of Copper were in thousands whereas the ranges of Coal were in hundreds. Thus, data standardization would help transform the data to comparable scales and make the data internally consistent.

A for loop was then utilized such that on every iteration, a different column representing the observations of a unique variable was chosen as the y vector. The remaining columns/variables were then gathered into a data matrix A. The linsolve(A, B) function was used to solve the regression and calculate the weight vector X. This weight vector was then applied to the data matrix (A*X) to get a model C which was then subtracted from the y vector to give a residual vector ErrorV. The RMS was then calculated by finding the square root of the mean of the squares of the error vector. A variable called minimum is then updated to store the value of RMS if it is lower than the current minimum. If the minimum is updated, then the index of the current y vector is recorded in a variable called lowndx. The rms values of each of the regressions were returned in an array named rmsvars.

Once the variable that provided the best regression is found, we then compute the given regression for it in unstandardized variables. This is to ensure that the computed model has the same scale as the variable for plotting purposes. The y vector (y1) and the predicted model (y2) are then plotted on the same graph using the plot function. The x axis would represent the 120 quarterly recorded dates.

To carry out the 5-fold cross validation, the cvpartition constructor from the cvpartition class was used to define a random partition on the set of data of specified size 5. A for loop was used to iterate from 1 to NumTestSets(I.e. the number of folds). On each iteration, the methods test and train from the constructor class were used to get the randomly generated indices of the observations to be used for the training data set and

the testing data set. The training data set for each fold should have approximately 90 observations (4/5 of data) and the testing data set for each fold should have approximately 22 observations (1/5 of the data). The regression was then solved to compute the weight vector once with the training set and then applied to the testing set. The residual error ErrorV is then computed by finding the difference between the y vector and the generated model. Thereafter, the RMS value of the residual error of the training set and the testing set is calculated. The RMS values of the 5 folds for the training sets and the testing sets were returned in rmstrain and rmstest respectively.

Results:

Table1: *RMS errors of the 16 regressions. The index of the commodity that has the lowest RMS error (variable of best regression) is **2** and the corresponding name of the commodity is **Copper.***

1	**2**	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0.2891	**0.1142**	0.3875	0.3049	0.5556	0.6405	0.2415	0.2291	0.2959	0.2014	0.4981	0.6103	0.1619	0.2898	0.2332	0.2650

Figure 1: *Plot of the pricing data of Copper and the linear regression to the dependent data over 120 quarters.*

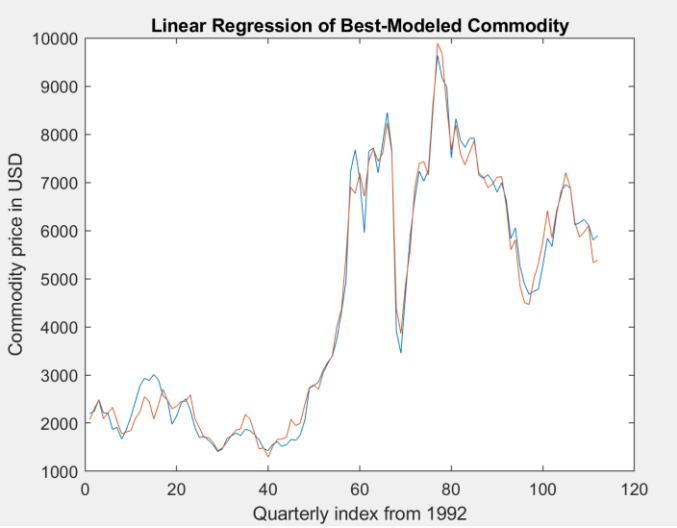

Table 2: *Root mean square errors of training and testing of the 5 folds. The index of the variable that is best explained by the other variables is 2 and the corresponding name of the basic commodity is Copper. Values in US dollars.*

	1	**2**	**3**	**4**	**5**
rmstrain	274.9089	277.8501	277.2443	287.5484	300.6856
rmstest	371.9608	339.7504	351.5963	329.9520	253.6804

Discussion:

We can conclude that Copper, when chosen as the dependent variable gave the best regression, when the remaining variables are gathered into a data matrix A. This is evident from the observation that the second column of the data matrix gave the lowest RMS error. That is to say, the difference between the approximated values and the given values was lowest with Copper as the y vector. This is also clearly visible from the graph plotted where the regression appears to tightly fit the Copper data. From a distant, it appears that both plots are almost identical. However, when taking a closer look, the difference between both the plots are in tens and hundreds of dollars. These are significant values given that we are looking at price. A model that predicts the price to be hundred dollars more than the actual value can be very unreliable based on different interpretations and expectations from the model.

Observing that Copper was the variable of best regression only tells us its relative performance against the other variables, but it doesn't tell us how well it performs at predicting new data. Therefore, to validate the supposition that Copper can indeed be used as a proxy for the price of other commodities, we look at the rms errors of training and testing after performing the 5-fold cross validation. The means of the rms errors of training and testing were 283.6475 and 329.388 respectively. The mean of the error from training will always be lower given that we used more data in the training stage to solve the regression, thus improving the accuracy of the prediction. The means are fairly low with respect to the actual prices of Copper that ranged in thousands. For example, if the price of Copper at a given time is $9639 and the model predicts a value that is plus or minus 329, your prediction would be off by only (329/9639)*100 which is 3.41%. This is very low. However, at lower prices, the value of the error becomes more significant. For example, if the price of Copper at a given time is $1176, your prediction would be on average off by (329/1176)*100 which is around 28%. As a result, using a model in these scenarios would be a serious gamble.

Furthermore, the standard deviations of the training and testing errors were also calculated to better understand the amount of variation and dispersion in the errors. These were 10.6828 and 45.1248 respectively. These standard deviations are fairly low and show that the errors were fairly consistent and not too far off the mean on every fold. Therefore, we can be confident to assume that we would get roughly similar errors on every prediction.

Hence, I do believe that the regression line generalizes well to the data and new data that it receives. The differences between the approximated values are low and the errors are also not too spread out. Using the model for exact predictions would be a gamble or not depending on your needs and the current price, however, using the model to forecast the general momentum of global prices would be reliable.

One way this model can be improved is by increasing the size of the data. Instead of quarterly data, having data recorded every month, would give us 324 observations. 3 times more observations than the current data. This would significantly improve the training of the model and thus, the results of the testing stage.