MATH 60210: Homework #2

Professor Anthony Sanford

Due: March 25, 2024 by 11:55pm on ZoneCours

Instructions: For this assignment, you are to work in groups of **three**. I will **not** make the groups. It is your responsibility to find team members for your homework. If you have trouble finding a team, you can reach out to me via email and I will try to match you with other classmates. This assignment includes two deliverables: 1) your write-up and 2) your code. Failure to submit one of these **two** components will be considered an incomplete submission. Your code must by written in Python. Your code should run smoothly and be clearly annotated. If you used resources outside of what is assigned for this course, you **must** give credit to the source. For example, if you googled code and used whatever you found, reference the source. Otherwise, you are cheating. Your write-up is meant to analyze, discuss, and interpret your results. Simply submitting a table without clearly discussing that table is not an answer to the question. Consider what I do in class – I present something and discuss it. Think of your write-up in this exact way. You are interpreting your findings for someone who is reading your report. Recall that HEC Montréal has rules regarding plagiarism apply to both your written answers and your computer code.

There are 4 multi-part questions, points for each question/part is written below. For each part, there are four possible grades:

- E: This is the grade if you literally write nothing. Worth 0%
- C: This is the grade if you really don't understand what you're doing...but you wrote something. Worth 50%.
- B: This is the grade if you got the answer mostly correct. Worth 80%. This will likely be the most common grade.
- A: This is the grade if your answer is as good (or better!) than mine. Worth 100%.

Note: Comments in your code that allow me to easily understand what you were trying to do may improve your grade. Particularly ugly or inelegant python code may reduce your grade by 10% (e.g. if you're cutting and pasting a lot because you don't know how to use loops or functions or array operations effectively). You may also lose points if your code is not running correctly or does not provide the same solution as your write-up answer.

Problem 1 [30 Pts], Machine Learning

For this question, use the same data that you collected in homework 1.

a. Write a function that accepts [5 pts]:

- a pandas series *ser* (i.e. a dataframe with one column)
- a scalar positive integer n

and returns

- a dataframe *df* with $1 + 2 \cdot n$ columns

The first column of *df* is a copy of the values in *ser*. The next n columns of *df* contain the values of *ser* lagged from 1 to n periods. The last n columns of *df* contain the squared values of *ser* lagged from 1 to n periods. Your function should return a dataframe with proper column names and should drop all rows with NaNs.Your function should not alter *ser*. **For this question, report the first few rows of the resulting dataframe.**

b. Create a function that accepts [5 pts]:

- a one-column dataframe *df*
- a positive integer n

and runs an OLS regression of the values in *df* on a constant and the $2n$ lagged values and lagged squared values. The function returns:

- a *RegresssionResults* object

from your OLS regression. SUGGESTION: Use "Q3_Factors(df, n)" to get lagged and squared lagged values. **Report and interpret your results.**

c. Create a function that accepts [10 pts]:

- a DataFrame *df*
- a positive real scalar s
- a positive integer n

and produces a plot of Ridge regression coefficients for various degrees of shrinkage.

- the dependent variable in the Ridge regression is the first column of *df*
- all the remaining variables in *df* are used as independent variables in the regression.
- the dataframe to use is the one with the lags.
- the regression includes a constant.
- the plot should show results for n Ridge regressions that use n values of regularization parameter λ evenly spaced between 0 and s. ("np.linspace()" might be useful for that.)

Report and interpret both your results and the plot.

d. Create a function that accepts [10 pts]:

- a DataFrame *df*
- a positive real scalar s
- a positive integer n
- a Boolean *LOOCV*

and returns

- the optimal shrinkage parameter λ
- the estimated Ridge Regression coefficients *beta* at the optimal shrinkage parameter

The arguments *df*, s and n the same as in part c above.

- the dependent variable in the Ridge regression is the first column of *df*
- all the remaining variables in *df* are used as independent variables in the regression.
- the regression includes a constant.
- peformance is compared for the n values of regularization parameter λ evenly spaced between $\lambda = 10^{-8}$ and s.

When "LOOCV == True" use LOOCV and otherwise use 10-fold CV. **Report and interpret your results.**

Problem 2 [40 Pts], AR Models

Use the PJM.csv dataset for this question.

a. Provide descriptive statistics and time-series graphs for your data. Make sure to label your graphs correctly and discuss the results (both the descriptive statistics and the graphs).[5 pts]

b. Write a function that accepts [5 pts]:

- a series (y)
- *lag* : an integer or one of following strings: 'aic', 'bic', 'hqic'
- *max_lag* : an integer or None. (Default None)

and returns an AutoRegResults object. If *lag* is an integer, use it as lag number of model, otherwise use that method to find suitable lag length, then estimate the model. If *lag* is a string, *max_lag* should be provided. **Report and interpret the results**.

c. Write a function that accepts [5 pts]:

- an AutoRegResults object (res)
- an integer $hmax$

and returns:

- a 1-D array containing out-of-sample forecasts for the next $hmax$ periods.

Report and interpret the results.

d. Write a function that accepts [5 pts]:

- an 1D arraylike object s
- 3 integers: T_0, p, h

The function will then:

- Estimate an $AR(p)$ on the first T_0 observations of s and stores out-of-sample forecasts for the next h periods.
- Drop the first observation, adds a new observation at $T_0 + 1$, and repeats the previous step.
- Continues dropping the first observation, adding a new observation at the end, reestimating the forecasting model, and storing the new forecasts.

Once the last observation in s is used to estimate a model and the forecasts are made, the function returns a 2D array of forecasts with h columns containing the forecasts from 1 to h periods ahead. **Report and interpret the results**.

e. The DM_statistic() function that accepts [10 pts]:

- a 1-D array d
- an integer h indicating the forecast horizon

and returns Diebold-Mariano statistic (DM). Use this function to test the H_0 that forecasts from $AR(3)$ model perform as well as and $AR(6)$ in terms of MSFE for 'price'. Construct the forecasts using rolling estimation of the AR model and an estimation window with a constant size of 300 observations. Compare all forecast horizons from 1 to 24 months and show the resulting test statistics. At which forecast horizons can you conclude (with at least 90% confidence) that the $AR(3)$ forecasts better? At which forecast horizons can you conclude (with at least 90% confidence) that the $AR(6)$ forecasts better? **Report and interpret the results**.

f. Use DM_statistic(d,h) function to test the H_0 that the $AR(3)$ forecast-encompasses the $AR(6)$ model. Estimate the models as you did for part (d) above and for the same forecast horizons. Return a dataframe whose rows are 1) the DM test statistic and 2) its p-value and whose columns are the forecast horizons. At which forecast horizons can you conclude (with at least 99% confidence) that the $AR(6)$ model is not forecast-encompassed by the $AR(3)$ model? **Report and interpret the results**. [10 pts]

Problem 3 [30 Pts], VAR Models

Use the PJM.csv dataset for this question.

a. Write a function that accepts [10 pts]:

- a DataFrame (df)
- a list of integers ($maxlags$)

and returns:

- a $len(maxlags) \times 3$ DataFrame IC_{lags}

The function estimates the optimal length for a VAR for the variables in df using the $AIC, BIC, HQIC$ respectively and a maximum lag length of $maxlags$. **Report and interpret the results.**

b. Write a function that accepts [10 pts]:

- a VARResults object (res)
- an integer lag

and returns:

- a $N \times N$ dataframe of Ftest statistics testing the H_0 of no Granger-Causality for each pair of variables.

Report and interpret the results.

c. Estimate a $VAR(3)$ for our 3 variables $[prices, Z_{load}, S_{load}]$. [10 pts]

- Plot response of each variable to a shock in Z_{load} for 12 periods ahead.
- Which variables appear to have a short-lived response to the shock?

Report and interpret the results.