Homework 2

March 16, 2017

Homework due: March 28, Tue. 11:59pm. Please submit it online through Blackboard under Assignments section. Each Group Only need to submit 1 homework through Blackboard. Just make sure you put everyone's name on the first page of the slides. Name your homework as "FIN580_HW2_LastName1_LastName2_LastName3.pdf". Name your code (if only 1 file, or package several file into a zip file) in the same way.

1 Data

Data Sets will be posted soon. It contains 9 currencies trade data for every 5 minutes bar. We have Open, High, Low Close, Date and Time for each currency. The same data set as we used in last homework. Note that

- There might some days the price remains constant. You should check the data and delete those days, since the estimate of volatility will be zero.
- Currencies are actually trading 5 days perweek. You should be careful about this.
- You should skip at least first 400 observation as a warm up period for all models.
- When calculate log volatility, first annualize the volatility and then take the log.
- We should fix a look-back window to run rolling window type forecasting. Here we propose to use 100 (almost half year) as look back window.

2 Problem

Usually we use standard deviation of 5-minutes return as the proxy of volatility. We are trying to build volatiltiy forecaster and use the data provided to test the performance of the forecaster. **For this homework:**:

- Consider only daily volatility
- Divide the data into training (first 3.5 years until Oct. 31^{st} 2011) and test (last 1.5 years since Nov. 1^{st} 2011) set. You will fit all the parameters (if any) in the training set. In the Evaluation part of the homework, you only report the metrics for all the models for the test set). Since we dont have enough data, you can use the last x days of training day i
- Use log volatility (log of annualized daily vol) as indicated in the paper instead of volatility to address the negativity problem in linear regression for both LASSO and VAR. For KNN you have the option to use volatility directly if more desirable (something you can check if you want). Compare the two results.
- **Do not use rolling window for VAR and LASSO.** Use expanding window in the test set with all the history up to that day, so you will get new regression parameters for each day in the test set but the λ will stay the same in LASSO.
- For both LASSO and VAR, only consider $p = \{1, 2, 3\}$. If you decide to implement Ordered LASSO, you can also consider $p = 5$.

The following implementations are required:

1. **VAR (vector auto regression):** as described in [1].

2. **LASSO:** described in [1]. You are expected to run one regression with the whole training history (first 3.5 years) for each λ. Run the regressions with different λs and plot the "in sample" Mean Square Error (MSE) with respect to the λ. Use the best λ (corresponds to the lowest MSE) in the test sample to report the final evaluation numbers.

3. **KNN Regression:** we covered during class. k is the parameter to optimize in this part. Plot the MSE for the training set for different choices of $k = \{1, 2, 3, \ldots, 20\}$. Ignore the first 100 days and start from 101 in order to have enough points in your model for KNN to make sense. Use today's vol σ_t as the 1-dimensional feature space, which means $x_t = \sigma_t$. The label is just what you want to forecast, which means $y_t = \sigma_{t+1}$.

4. **Research Problem:** Introduce time series component into KNN. You can either expand the dimension of the model (include time as attributes to your features) or add penalty terms (when selecting the closest point, add a term like $c \|t - t_i\|$, where you need to optimize over c in sample), or any other method that you came up with your own. Remember to justify your choice, such as by comparing the performance. You can also use rolling window, that means, fix the look-back window when training your model, say for each day, and optimize on the look back number of days. (dont go lower than 60 days). You can pick any one of the methods described here or your own way of incorporating time series into the KNN.

Extra Points: The following methods are not required, but you can implement them for extra points.

1. You can also implement ordered LASSO in [1] to get the extra points.

2. You can construct your own feature space for KNN. For example, you can use other 1-dimensional feature, such as return (might be a bad choice), or other multi-dimensional features. Try to improve the single-vol 1-dimensional feature. Then you can get the extra point.

Note that, for LASSO and ordered LASSO, you can use existing libraries. For other methods such as KNN and VAR, apart from optimization alogrithm, you should not use the scikit-learn package or other existing package. You should write your own implementation based on what we've covered in the class and described here.

3 Evaluation

There are two measures we use here for the performance of forecasting. One is the Mean Squared Error (MSE), which is defined by

$$MSE = \frac{1}{T} \sum \left(X_t - \hat{X}_t \right)^2$$

where $\hat{X}_t$ is the forecasted value. Another measure is Quasi-Likelihood, which is defined by

$$QL = \frac{1}{T} \sum_{t=1}^{T} \left(\frac{X_t}{\hat{X}_t} - log \frac{X_t}{\hat{X}_t} - 1 \right).$$

Notice that $QL_t = 0$ when $X_t = \hat{X}_t$.

Presenting your homework in an elegant way is also an important part of the assignment. Please think about what are your conlcusions and what kind of metrics / graphs / tables you could use to present in what kind of format can help you better explain your results and conclusions? For example, you may sum the MSE for all currencies to compare the models.

4 Grading Method

Student should hand in a clear-formated slides (which will also be used in class presentation) together with their code. The slides should describe how you process the data, step by step methods and assumptions behind models, and your results and comments. The code should be able to run directly without modification to generate the results presented in your slides (If there are variables to be mannualy change/input, make it clear at the beginning of the code).

Please write your code in a clear manner and put description of each function you build. See the following link for coding style.

https://google.github.io/styleguide/pyguide.html#Python_Style_Rules

Also, we suggest the students to build functions for volatility proxy, forecasting measures as it will help you build a good infrastructure for future homework assignment and final projects.

The Grades will be assigned based on:

1. The completeness and robustness of the implementation
2. Your code, based on the requirement described above
3. The way you present your results.
4. Any original idea, thinking, additional research/experiment on existing method/results.
5. Extra points indicated in Section 2.

References

[1] Wilms, Ines, Jeroen Rombouts, and Christophe Croux. "Lasso-based forecast combinations for forecasting realized variances." (2016).