Problem Set 5

(Due March 10, 1:00 PM)

Instructions

1. The following questions should each be answered within an R script. Be sure to provide many comments in the script to facilitate grading. Undocumented code will not be graded.

2. Work on git. Fork the repository found at `https://github.com/jhomola/PS5`. There is already code to get you started. As you add your code, commit and push frequently. Use meaningful commit messages – these may affect your grade.

3. You may work in teams, but each student should develop their own R script. To be clear, there should be no copy and paste. Each keystroke in the assignment should be your own.

4. If you have any questions regarding the Problem Set, contact the TAs or use their office hours.

5. For students new to programming, this may take a while. Get started.

Calculating fit statistics

In prediction tasks, there are a number of alternative methods for evaluating the fit of various statistical models. Some indicators (e.g., root mean squared error (RMSE) and median absolute deviation (MAD)) provide summaries for the error of each model. Others, (e.g., median absolute percentage error (MEAPE)) measures error as a proportion of the dependent variable. Finally, to aid interpretation, we also might evaluate models relative to some meaningful baseline (e.g., median relative absolute error).

Denote the prediction for some observation i as p_i and the observed outcome as y_i. We define the *absolute error* as $e_i \equiv |p_i - y_i|$ and the *absolute percentage error* as $a_i \equiv e_i/|y_i| \times 100$.

Denoting the median of some vector $\mathbf{x}$ as $med(\mathbf{x})$, we define the following statistics:

$$\begin{aligned}
\text{RMSE} &= \sqrt{\frac{\sum_1^n e_i^2}{n}} \\
\text{MAD} &= \text{med}(\mathbf{e}) \\
\text{RMSLE} &= \sqrt{\frac{\sum_1^n \left(\ln(p_i+1) - \ln(y_i+1)\right)^2}{n}} \\
\text{MAPE} &= \frac{\sum_1^n a_i}{n} \\
\text{MEAPE} &= \text{med}(\mathbf{a})
\end{aligned}$$

1. In the github repository, you will find a slimmed down version of the American National Election Study (ANES) 2012 Time Series Study (`anes_timeseries_2012_stata12.dta`) and the codebook for the study. Based on this dataset, randomly subset the data into two partitions. Use one partition (your "training set") to build at least three statistical models where respondents' feeling thermometer score for President Obama (`ft_dpc`) is the dependent variable. Your statistical models can be anything you like, but have some fun with it. R has lots of machine learning packages etc. Also, make sure to consult the codebook and document carefully how you deal with missingness.

2. Using these models, make "predictions" for the partition of the data you did NOT use to fit the model (hint: `predict()`). This is your "test" set.

3. Write a function that takes as arguments (1) a vector of "true" observed outcomes ($\mathbf{y}$), and (2) a matrix of predictions ($\mathbf{P}$). The matrix should be organized so that each column represents a single forecasting model and the rows correspond with each observation being predicted.

   The function should output a matrix where each column corresponds with one of the above fit statistics, and each row corresponds to a model.

4. Allow the user to choose which fit statistics are calculated.

5. Evaluate the accuracy of the models you fit above using the test set.

6. FOR GRAD STUDENTS ONLY: For each observation we can also have predictions from naive forecast, r_i, that serve as a baseline for comparison. Based on these, we can define $b_i \equiv |r_i - y_i|$. We can now also define another fit statistic:

$$\text{MRAE} = \text{med}\left(\frac{e_1}{b_1}, \ldots, \frac{e_n}{b_n}\right)$$

   Add a vector of naive forecasts ($\mathbf{r}$) as a third argument to your function, and allow the user to also calculate the MRAE with your function. Make sure the function still works if the vector of naive forecasts is not provided (just drop the MRAE).