Project 5: Estimating a Hidden Markov Model (HMM)

Objectives

To develop a Hidden Markov Model of the data that will be created using your student id (SI) number. For this analysis you can use any package that provides HMM functions. R, Matlab, and Python have HMM solvers.

Data Set

You will use the data provided to you that is completely randomized based on your student id. The data set consists of 2000 observations. Partition your data set into two parts, the training set and the testing set. For the training data set use the first 1600 observations, and keep the last 400 for the testing data set.

The number of states of the hidden Markov Chain is 3. The set of observations is {1,2,3,4,5}.

Tasks

1. *Develop a Hidden Markov Model*

   Use an HMM solver to estimate the HMM parameters $P=(p_{ij})$, π, and $B=(b_{ik})$. Give your results in one or more tables.

   Run the model to predict the next 400 observations, and compare the HMM results to the training data set and compute the sum of squared errors (SSE), the root mean squared error, and R^2. For this, you have to use a procedure (as explained in class) in order to decide the state the HMM is at the 1600^{th} observation, so that to generate probabilistically the next state. Having done that, you can then run the model for 400 observations. Use existing functions to carry out both tasks. Give all your results in a table.

2. *How much into the future can I forecast?*

   Most likely, you will observe that the HMM model produces good results at the beginning, but then its accuracy declines as we go further away from the 1601^{st} observation. You can observe this by doing a line plot (not scatter diagram!) of the actual vs the estimated results.

   This can be resolved by retraining your HMM model. The question is how well your model can predict the n next estimates. If it can be good for n=10, for instance, then retrain your HMM every 10 observations using the entire set of data up to that point, or 1600 observations shifted to the point.

   Vary n to find a good value. For each n calculate the sum of squared errors (SSE), the root mean squared error, and R^2 obtained over the 400 hundred observations. Give your results in a table.

What to submit

You will submit your report along with the code that you wrote to obtain the results. It is very important that you provide enough results to support your conclusions. Conclusions without insufficient results will make you lose grades. Also, it is important that you develop your own code. Sharing code is not allowed and constitutes cheating, in which case both students (the one

that aids and the one that receives) will get a zero for the project and will be reported to the student conduct office.

Grading
The TA will first verify that your code works and produces the results you submitted. The break down of the grades will be as follows:

Task 1: 40 points
Task 2: 60 points

Extra credit

This task will be graded separately from the above assignment from 0 to 100. The grade will then be transformed to the range [0,3], and it will be added to your final grade for the class. For instance if you get a 3 for the extra credit and you class grade is 83, then your final grade will be 86. This extra credit may push you up one +/- bracket.

Task
Use the forecasting models from project 3, to model the data and compare them against your HMM model. The comparison will be based on the accuracy of predicting the test data, as measured by the statistical metrics used in the forecasting project.

Remember that you will be graded mostly on your ability to interpret the results