Causal State Modeller Toolbox Help File

David Kelly
School of Mathematics and Bristol Centre for Complexity Science,
University of Bristol, Bristol, UK

October 11, 2011

Contents

1 Introduction to CSM

The Causal State Modeller GUI allows causal state models to be inferred for both discrete and (where certain conditions are met) continuous data sets. The code is a reimplementation of the Causal State Splitting Reconstruction (CSSR) algorithm developed and written by Cosma Shalizi and Kristina Shalizi (1). Users are referred to the original paper for description of the algorithm. Briefly though, all subwords in the data up to a maximum length are assigned to a 'causal state', where all words in the same states share next symbol conditional distributions which are deemed equivalent when compared with a statistical test at some given significance level. In other words, histories are grouped which share (roughly) the same futures. This process produces optimal statistical predictors of process under study.

Data can be imported or generated with pre-programmed models or the option to use a custom model. The data can be viewed and histograms plotted. Then continuous data may be discretised using the non-committal discretisation scheme. Finally the CSSR algorithm may be run, chosen metrics calculated and the model analysed and compared to other models.

1.1 Before use

CSM uses the dot.exe application to display inferred models. This must be downloaded from `http://www.graphviz.org/`.

2 Data Source

2.1 Import Data

To import data from a text file (or similar) type the filename and click the 'Import Data' button. The filename (or full path if the file is not in the current directory) should be specified in the (first) textbox labelled 'filename'. Each data set should be on a separate line and white space separated. CSM supports multiline data where separate data sets (from the same process) can be entered on different lines in the data file. (A single dataset may also be entered as a column).

CSM also supports batch processing of files. Ensure all of the files to be processed (and only files to be processed) are present in a single folder and enter the folder path into the filename textbox. Data files for batch processing must be discrete. Parameters are the same for each file. Once the folder is specified and the parameters are chosen, click 'Run batch'. The

outputs will be saved to the same folder by default. In subsequent runs output files will be ignored (hence files for processing must not be named with any suffixes associated with output files otherwise they too will be ignored).

2.2 Generate Data

You have the option to generate data as well. Select one of the pre-programmed processes, the golden mean process or the even process.

Another option is to specify the model in the form of a dot file. Specify the filename (and its path if it is not in the current directory) in the (second) textbox labelled 'filename'. Some example dotfiles are provided which may be edited to produce the desired model. More information is available at `http://www.graphviz.org/`.

Once an option has been selected the length of the data set may be chosen in the input N text box.

You may choose whether the data is to be discrete or continuous by selecting the appropriate option.

Discrete - The data consists of the output symbol at each time step

Continuous - Each output symbol is replaced with a continuous value sampled from a Gaussian distribution. There is a distribution associated with each output symbol. The parameters for each distribution may be changed by selecting the output symbol from the drop down menu and editing the appropriate text box.

Once all the settings have been specified you can generate the data by pressing the 'Generate Data' button. Images of pre-programmed models used to generate the data will be created, the format can be specified by checking or unchecking the boxes in the options panel.

3 Visualise

3.1 Plot Data

Displays the data in a plot. In the case of multiline inputs only the first line will be initially displayed. A control will appear which allows the user to move back and forth through the lines in the imported data file.

3.2 Plot Histogram

Plots a histogram of the data with the specified number of bins. All data is included in the histogram if the input was multiline.

3.3 Output to .pdf

Any graph may be output to .pdf at any time as described in Section 7.0.1.

4 Discretise

4.1 Fit Mixture Model

This function fits a Gaussian Mixture model to the data using maximum likelihood. There is the option to restrict the routine to a specific number of components, if you are sure there are for example only two peaks in the histogram, or specify a maximum number of components. If the latter is selected the function finds the maximum likelihood of all models with number of components less than or equal to the specified maximum, then uses Akaike's information criterion to select which is the best with the minimal number of parameters.

4.2 Discretise

The data may then be discretised using the parameters from the fitted mixture model according to the method described in the paper 'Inferring hidden Markov models from noisy time sequences: a method to alleviate degeneracy in molecular dynamics' available at `http://arxiv.org/abs/1011.2969`. Once the data has been discretised CSSR may be run.

Depending on the degree of clustering in the data and the spacing of the clusters the data may or may not be suitable for analysis. In order to be able to reliably infer a model of the underlying dynamics of the system we may only use data which we may say with (near) certainty was generated by one particular distribution. Thus data which falls in the overlap region of two (or more) distributions cannot be used. If too large a proportion of the data cannot be used it will be very difficult or impossible to infer a meaningful model.

5 CSSR

Several parameters must be specified before running CSSR. Firstly a filename which will be given to the output files generated. Secondly the maximum length of history, L, which will be examined. And finally the significance level for the Kolmogorov-Smirnov statistical test used to compare distributions. These last two parameters are described in more detail in Section 8.

Users may also specify a number of options. If the data to be analysed was continuous and has been discretised then the option 'Noncommittal Discretisation' must be checked (this should be done automatically when the discretisation procedure is run). There is the option to generate images of the resulting machines, in .fig, .png, or .pdf format. Finally the metrics to be calculated are listed and any combination of them may be selected.

When the options have been chosen and the parameters specified, CSSR may be run by clicking 'Run'. The code will generate a number of output files.

- **filenameLx_Log.txt** Text file with the calculated distribution for every string up to the specified maximum length and the details of the process of assigning it to a state.
- **filenameLx_States1.txt** Word frequency table and the final assignment of strings to states prior to the determinisation step.
- **filenameLx_DeterminisationLog.txt** Details of the determinisation procedure. Which strings are not deterministic and their movement to a new state.
- **filenameLx_States2.txt** - Word frequency table and the final assignment of strings to states after the determinisation step.
- **filenameLxdot.txt** .dot file used to generate the images of the machine prior to editing the machine to show only recurrent structure.
- **filenameLxFull.png**

  filenameLxFull.fig

  filenameLxFull.pdf Machine image files for full machine including transient states, generated from the above .dot file.
- **filenameLxFinalNdot.txt** - .dot file used to generate the images of the machine after editing the machine to show only recurrent structure.

In some (rare) cases there may more than one recurrent structure in the machine. In this event all recurrent structures are given and numbered, N.

- **filenameLxFinalN.png**

  filenameLxFinalN.fig

  filenameLxFinaln.pdf - Machine image files for machine showing only recurrent states, generated from the above .dot file(s).

- **filenameLx_StateSeries.txt** File showing the state sequence for each data point. In the case of discretised data this may be the most probable state sequence calculated using the Viterbi algorithm (since for ambiguous data there is an ambiguous state sequence).

- **filenameLx_SymbolSeries.txt** In the case of discretised data and use of the Viterbi algorithm the most probable symbol sequence is also given.

- **filenameLx_Metrics.txt** File containing the metrics of the machine.

In all the above filename is the filename specified and x is the value of L specifed.

6 Machine Analysis

Once a machine has been inferred the programme allows certain analyses to be performed

6.1 Dwell Time Histograms

In order to examine the validity of the assumption of stationarity there is a module for plotting dwell time histograms and fitting exponential distributions to them. Select a state from the drop down menu and click 'Plot' to plot the histogram of dwell times for that state. If the 'Fit Exp' check box is ticked then an exponential distribution will automatically be fitted (using least squares regression) and the mean output to the interface. If the data does not appear to be well fitted by a single exponential decay, then this is indicative of non stationary in the data or that the state in question is actually a combination of a number of states with the same associated outputs but different transition rates.

6.2 Model Distance

If quantitative comparison with another model is required then the model distance can be calculated. This is defined as the difference in the log probabilities of the observed data, $O^{(2)}$, being generated by the generating model and the inferred model, designated λ_2 and λ_1 respectively, normalised for the length of the data, N.

$$D(\lambda_1, \lambda_2) = \frac{1}{N}(\log P(O^{(2)}|\lambda_1) - \log P(O^{(2)}|\lambda_2)) \tag{1}$$

This measure is equal to zero for models with the same statistical properties.

The model to compare the inferred model to can be input in the same way as for the data generating model (Section 2.2) or the option to compare the inferred model to the generating model can also be used.

7 Extra Functions

7.0.1 Plot to pdf

This button outputs the selected figure to pdf. The figure is chosen by entering its number in the 'Fig. Num.' box. The pdf is saved in the current directory as 'FigureN.pdf' where N is the figure number. Existing files of that name will be overwritten.

7.0.2 Clear all

This button erases all data stored by the code and returns the programme to its initial state.

7.0.3 Help

This button opens this file.

8 Guidance on Parameter Choice

When using the CSSR algorithm we must choose values for 2 parameters, the maximum length of history conditioned against, L, and the significance level for the Kolmogorov Smirnov statistical test, α. The following guidelines are useful for choosing these parameters acording to the length of the data and the size of the alphabet (dictated by the data).

The choice of α is essentially arbitrary. It can be interpreted as the willingness to distinguish between conditional distributions and consequently form new states. Typically a low value for α is chosen, around 0.005. This ensures that new states are created only with strong evidence that the distributions really are different and is a stance consistent with the data driven basis of the algorithm.

The amount of data should guide the choice of L. As L increases the expected number of instances of words of that length present in the data decreases rapidly. To obtain reliable conditional distributions with which to determine to which state a string should belong, a minimum sample size is necessary. The word frequencies are given in the **filenameLx_States1.txt** and **filenameLx_States2.txt** output files. These may be checked to ensure that there are sufficient occurances of the longest strings to ensure a reasonable sample size for the statistical test. L should be as great as possible which maintaining sufficient sample size.

References

1. Cosma Rohilla Shalizi and Kristina Lisa Shalizi. Blind construction of optimal nonlinear recursive predictors for discrete sequences. *CoRR*, cs.LG/0406011, 2004.