Concept Change Detection (Ver. 9/7/2015)

Streaming data is not a new concept as all markets and real-time data have been around for many years. However, the price drop of sensors ranging from video, wearable devices, etc. has provided many new opportunities for data analysis. In order to fully utilize the value, machine learning systems often have to recognize patterns within the time series data and divide up the data into separate chunks. For instance, my FitBit analysis should differentiate general movement, rigorous exercise, and sleep from streaming data with a limited memory. Advertisers might want to tailor ads based on the types of web sites visited from a computer shared by husband, wife, and children.

The change detection process has the goal of recognizing the transition based on univariate or multivariate data streaming in. In my examples, I've only created univariate data but you are welcome to include multivariate patterns. Since it is streaming, we cannot store it all and decide later, but can have a limited window to compare the present and recent past, perhaps. To simulate the streams, we have provided a series of files with labels "positive" and "negative" indicating that we believe a concept change occurred or did not, respectively. You will want to create and submit additional files labeled "positive" and "negative" with your team name included. We will assume only one concept change is present per file and the change is not in the first 50 entries so as to establish a baseline. The change could occur anywhere after this 50^{th} position.

This is a team project with groups of 2. We plan to be strict about this so groups of 1 and 3+ are not acceptable. Use the bulletin board to help identify partners. We will have multiple partnering projects throughout the semester and you are welcome to change partners.

You are welcome to use any method you like. We encourage you to implement and try many methods and tell us what you tried out.

What to submit:

1) Python code such that when we type:
   runme.py <dirname>

   it will generate a single ouput[teamname].txt file with two columns separated by a tab. The first column contains the name of each test file in dirname (just the file name rather than the full path). The second number is -1 if a change was not detected, and the position where you believe the change occurred for a detected change. Be sure to include your name and your partner's name in the headers. Teamname is just a unique output name and up to you. To avoid issues with directory structures on different operating systems, let's assume the directory of test files is a folder in the same folder as the runme.py.

2) Your sample output file on the combination of your test data and our provided files.

3) Sample files in the same format as ours beginning with pos*_<loc>.txt or neg*.txt without headers and tab-delimited columns (if more than one). Be sure that the change does not occur within the first 50 samples.

4) A 1-2 page write-up describing what you tried and how it worked. Be sure to include your name and your partner's name on this. Include your assumptions about the types of patterns that you designed your system to recognize. Also describe how you decided upon the location.

5) To ensure fairness of the partnership, each member will provide a separate .csv file in the provided format. It will list your partner's name and how you would rank their work on a 1-10 scale. Your score is a 5 default. Providing a 10 suggests they were awesome and did most of the work. Providing a 1 suggests you carried the team all the way. Include a few lines on your contribution and your partners. Each partner should submit this separately.

FAQ:

If we have teams of 2, should we both submit the project?

No, the person whose last name comes first should submit the project to their drop box along with their partner evaluation. The person whose last name comes last should submit just their partner evaluation. So, Ms. Jones should submit the project and partner evaluation; Mr. Smith should submit just their partner evaluation.

Will we get a team grade or an individual grade?

We will assign team grades but make notes of the contributions of each member that we will factor into the final course grade. In cases of a consistent discrepancy in perceived contributions, we will communicate with you to get a better picture.

Can I use any existing software or language?

We're asking for a common python interface and would prefer that you develop your python machine leaning and statistic skills by coding in python. You are welcome to use other sources for the project and you would need to cite these both in the code and the write-up. That said, if you find a concept change detection python package, don't use it as the goal is to use the statistical and other concepts to come up with a solution. You may of course use the literature to come up with ideas.

How to get started?

Since we have just started covering statistical concepts, these would be a natural place to start. The concept of a sliding 'window' where you can model data at time[t-window_length , t] and compare it to new data or other data windows. We'll cover some comparative statistics in Lecture 2.

Does it matter if it's runs slowly?

The teaching staff will be running all of these projects against a battery of test runs. Brownie points for very efficient computations. Our computers have other uses so I don't want to say unlimited time, but we hope any single file run can finish in seconds rather than minutes.

Do you take off points if there are no comments?

The teaching staff will be looking to understand what your code does and how that relates to your document. Clear explanations, clean code, file- and method-based comments can all help achieve this. No, we will not directly take off points for the absence of comments, but are looking to quickly understand what you have done.

What sorts of patterns should we be able to recognize?

We have provided a ***few*** test cases but there is a wide range of patterns that you will have to consider and anticipate. I have generated a much wider set of test cases and will test many. I would approach the problem as a set of detectors that recognize different kinds of changes. Start by brainstorming what could constitute a pattern and vary them in terms of signal magnitude, noise, direction, and other attributes that I'll let you think about.