Work Summary 2015\_11\_8

KangRong

# Column selection

I extract the first 100 line of one table from JinFeng dataset, then show all columns in excel chart. As follow:

As the chart showing, the fluctuation of column “visu\_converter\_active\_power” is more obvious than other items. Thus I take the relation between “visu\_converter\_active\_power”([abbreviated](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E7%BC%A9%E5%86%99" \t "_blank) [to](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E7%BC%A9%E5%86%99" \t "_blank) ["](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E7%BC%A9%E5%86%99" \t "_blank)[c1](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E7%BC%A9%E5%86%99" \t "_blank)" [.](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E7%BC%A9%E5%86%99" \t "_blank)) and time as research data.

To obtain the legends, we select randomly substring whose length in the range of 200 to 300 with a python script like these:



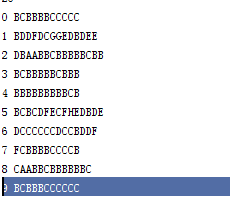
Figure: **legend id** is an integer range from 0 to legend amount – 1.

# SAX representation

I cut those legend time series into some segment with same length(I set to 20). Then I use PAA approximation to reduce its dimensions.

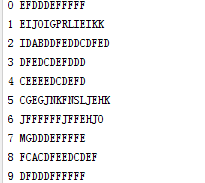
Comparing Gauss distribution, I choose the linear cutting strategy. It take 300 as lower bound. SAX symbol add 1 while point value adding interval.

When I set interval value to be 50, the result is showed as follow:



Some string is so flat that it’s hard to be regard as a “legend”,like time string 4: BBBBBBBBBCB.

I low the interval to 20:



Thus I find **a problem**: while the interval is too large, time series would went hard-distinguished. On the contrary, the performance of suffix tree would deteriorate while the amount of legend get larger.

Another problem is definition of distance in suffix-tree. In my primary opinion, The similarity measure between two time series string is simply LCS. However, how to measure the differences between two series with same suffix branch depth?

Anyway, I push these string into suffix tree (I get it from a public GIT repository), and take one sequence as standard. The search result is not satisfying. Most of similarity is the same symbol like AAAAA comparing BBBBB. I think I should focus on the variation but not similar smooth partition.

I choose ten series:

DFEDCDEFD

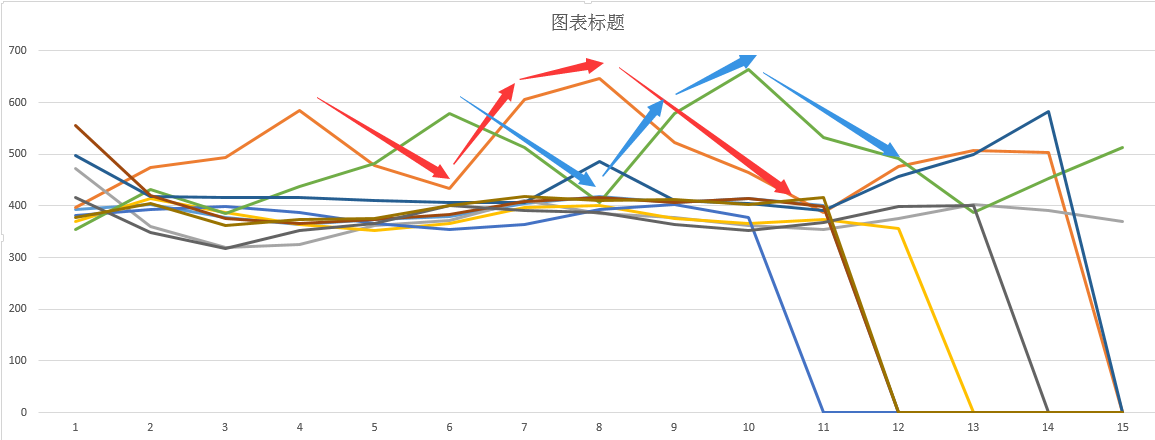
As red and blue arrows indicating, these two series might be similar. Their SAX Strings are:

EIJOIGPRLIEIKK

CGEGJNKFNSLJEHK

First of all, the similar tendency is covered by SAX.

Secondly, their LCS is “LIE”. Note the last symbol“K” is close to LIE. I think the **distance between common two subsequence** (may be not longest) should take into account.



# Future:

I will consider variation tendency in one sequence. For example, these two series might be similar: ACEDDD with ABCDED. Traditional SAX ignore the variation tendency. Some information might lost.