Episode Mining
Autonomous work

Julien Blanchard

November 2017

The dataset `YahooFinance.data` describes 1255 days of Dow Jones indexes. The values are summed up as increases (1 or 2), drops (-1 or -2) or constant index (0). The goal is to extract episodes from this sequence and assess them.

1. Write a Python function which :
   - extracts the frequent episodes of size 2 from the sequence, by considering only the increase and drop events (as the 0s are too frequent and not interesting),
   - for each frequent episode (X, Y) ("X folowwed by Y"), computes the cardinalities n_X, n_Y and $n_{X \to Y}$ (number of Xs followed by at least one Y in ω) as seen during the last course [1],
   - for each frequent episode, computes the measures frequency, confidence, recall and j-measure.

   The parameters of your function will be : the frequency threshold min_freq (as a count), the size $\omega > 1$ of the window, and the path towards the dataset file. You will use a $min_freq = 50$ by default, and $\omega = 2$.

   You can draw your inspiration from the Winepi algorithm (state automaton) or the epiSPADE algorithm (vertical format).
2. In your opinion, what are the "best" episodes of size 2 in the sequence ? (remember that the higher the measures, the better the episodes).
3. Describe the distributions of the four measures on the episode population with histograms.

1. by applying your function on the inverse sequence (from right to left), you will be able to compute $n_{Y \leftarrow X}$ (number of Ys preceed by at least one X in ω)