1.4 Aggregating Algorithm

Implement a function calculating predictions output by the Aggregating Algorithms with equal weights in the binary square loss (Brier) game.

```
function predictions = AA_Brier(expertsPredictions,outcomes)

% AA(expertsPredictions,outcomes) calculates predictions
% produced by the AA given expertsPredictions (this is a matrix
% whose lines are sequences of predictions) and outcomes.
```

The element **expertsPredictions(n,t)** is the prediction produced by expert E_n on step t.

The element **predictions(t)** should be the prediction output by the AA on step t.

In your program use the optimal $\eta = 2$ and equal initial weights. Your program should work with the square loss; you do not need to make provisions to expand it to other loss functions (and doing so may be rather difficult).

[10 marks]

1.5 AA Example

This real-data example about bookmakers' odds is taken from V. Vovk and F. Zhdanov, Prediction with expert advice for the Brier game, in: *Proceedings of the Twenty Fifth International Conference on Machine Learning*, pages 1104-1111. New York: ACM Press, 2008. See http://vovk.net/ICML2008/ for more details.

Bookmakers quote betting odds for a sports event. One may interpret those as probabilities that one side will win. (In fact the situation is a bit more complicated: bookmakers are interested in making money rather than accurate predictions so they always include a margin to guarantee themselves a profit. However we ignore that and interpret their odds as a prediction.) The file **tennis1.txt** contains predictions output by four bookmakers. You can refer to the paper to see how the predictions were calculated from the odds quoted by bookmakers.

The format of the file is as follows. Each line describes a game. The first number is an ID with the technical information about the game; you may ignore it. The second number shows whether player 1 won; it is always 1 because players were re-ordered this way retrospectively. The third column shows whether player 2 won; it is always 0 for the same reason. In your calculations assume that the outcome is always 1 (of course this only makes sense because we apply the AA retrospectively).

The rest of the line contains "predictions" made by four bookmakers. The fourth column is the estimate done by the first bookmaker whether player 1 would win; the fifth is the estimate by the first bookmaker that player 2 would win. They sum up to 1. Ignore the fifth number and treat the fourth as prediction in the square loss game. The sixth and seventh columns contain predictions by the second bookmaker, the eighth and ninth by the third, and the tenth and eleventh by the fourth. (Clearly you do not need odd columns and the second is not very informative either).

Process this file and run your function on the predictions by four bookmakers to get AA predictions.

Calculate the total loss of the AA and quote it in your report and in the `results.txt` file.

Make three graphs.

- The first should show five losses vs. time, $\mathrm{Loss}_{E_1}(t)$, $\mathrm{Loss}_{E_2}(t)$, $\mathrm{Loss}_{E_3}(t)$, $\mathrm{Loss}_{E_4}(t)$ of the experts and $\mathrm{Loss}_L(t)$ of the AA.
- The second should show the following four quantities vs. time: $\mathrm{Loss}_{E_1}(t) - \mathrm{Loss}_L(t)$, $\mathrm{Loss}_{E_2}(t) - \mathrm{Loss}_L(t)$, $\mathrm{Loss}_{E_3}(t) - \mathrm{Loss}_L(t)$, $\mathrm{Loss}_{E_4}(t) - \mathrm{Loss}_L(t)$.
- The third should show the following four quantities vs. time: $\mathrm{Loss}_{E_1}(t) - \mathrm{Loss}_{\mathrm{ave}}(t)$, $\mathrm{Loss}_{E_2}(t) - \mathrm{Loss}_{\mathrm{ave}}(t)$, $\mathrm{Loss}_{E_3}(t) - \mathrm{Loss}_{\mathrm{ave}}(t)$, $\mathrm{Loss}_{E_4}(t) - \mathrm{Loss}_{\mathrm{ave}}(t)$, where $\mathrm{Loss}_{\mathrm{ave}}(t)$ is the loss of the following simple strategy. On every steps it takes the predictions of experts E_1, E_2, E_3, and E_4 and averages them with equal weights $1/4$:

$$\gamma_t = \frac{1}{4}(\gamma_t^1 + \gamma_t^2 + \gamma_t^3 + \gamma_t^4) \ .$$

Include these graphs in your report.

[10 marks]