Exercises

Optimization Methods in Finance

Fall 2010

Sheet 2

Exercises marked with (*) qualify for one bonus point, if correctly presented in the discussion session

Exercise 2.1 (*)

Consider again the simple setting, where we have \( N \) experts that (over a time horizon of \( T \) units) predict a binary event \( (y_t^j \in \{0, 1\}) \) and a forecaster tries to predict the events so that he is not making significantly more mistakes than the best of the experts. Consider the following strategies:

- **Strategy 1:** The forecaster chooses at any time \( t \) the prediction \( \hat{p}_t \) of the expert \( j \) who made the least number of mistakes so far (i.e. \( \hat{p}_t = y_t^j \) where \( j = \text{argmin} \{ m_j \} \) and \( m_j = |\{ t' < t \mid y_{t'}^j \neq z_{t'} \} | \) is the number of mistakes, which were made by expert \( j \) in time \( 1, \ldots, t - 1 \)). If several experts have the same minimal number of mistakes, we choose that one with a smaller index \( j \).

- **Strategy 2:** The forecaster chooses the prediction of expert \( j \) with probability

\[
\frac{t - m_j}{\sum_{j=1}^{N} (t - m_j)}
\]

(i.e. proportional to the number of correct predictions; say in the first iteration, we choose an expert uniformly at random).

Show that both strategies can be much worse (say for \( T \gg N \) and suitable \( \varepsilon \)) than the weighted majority experts algorithm (Algorithm 2 from the lecture).

Exercise 2.2 (*)

Consider again the setting with \( N \) experts and loss vectors \( \ell^t \in [0, 1]^N \). Let \( T \) be the number of iterations, \( \hat{L} \) be the forecasters loss and \( L_j \) be the loss of expert \( j \). In the lecture we saw the bound

\[
E[\hat{L}] \leq \frac{\ln(N)}{\varepsilon} + (1 + \varepsilon)L^j.
\]

Observe that this just bounds the average loss of the forecaster. Can you give a concentration bound statement of the form \( \Pr[\hat{L} \leq (1 + \ldots) \cdot L^j + \ldots] \leq \ldots \). Here the following theorem (a.k.a. Azuma’s Inequality) might be helpful (which you may use without proving it):

Let \( 0 = X_0, X_1, \ldots, X_n \) be a sequence of random variables with increment \( Y_i := X_i - X_{i-1} \). Here \( Y_i := Y_i(X_0, \ldots, X_{i-1}) \) might arbitrarily depend on \( X_0, \ldots, X_{i-1} \), but always \( |Y_i| \leq 1 \) and \( E[Y_i] = 0 \). For \( \lambda \geq 0 \) one has \( \Pr[X_n \geq \lambda \sqrt{n}] \leq e^{-\lambda^2/2} \).
**Exercise 2.3 (**)**
Recall that a function \( f: \mathbb{R}^n \to \mathbb{R} \) is convex, if \( \text{dom}(f) \) is a convex set and for all \( x, y \in \text{dom}(f) \) and \( 0 \leq \lambda \leq 1 \) one has \( f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y) \). Prove that if \( f_1, \ldots, f_n: K \to \mathbb{R} \) are convex, \( \lambda_1, \ldots, \lambda_n \geq 0 \), then also \( \sum_{i=1}^n \lambda_i f_i(x) \) is convex.

**Exercise 2.4 (**)**
Let \( y \in \mathbb{R}^n \) be a vector with \( y_i > 0 \) for all \( i = 1, \ldots, n \) and \( x \in \Sigma^n \). Prove

\[
\|\nabla \left( -\ln(y^T x) \right) \|_\infty \leq \max_{i,j} \left| \frac{y_i}{y_j} \right|
\]

Note: The gradient is w.r.t. \( x \) as variable.

**Exercise 2.5 (one practical bonus point)**
Recall the example from the lecture

<table>
<thead>
<tr>
<th></th>
<th>Stock A</th>
<th>Stock B</th>
<th>Money Market</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>Stable</td>
<td>1.2</td>
<td>1.7</td>
<td>1.3</td>
</tr>
<tr>
<td>Down</td>
<td>0.8</td>
<td>1.2</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Implement the presented algorithm to determine an optimum row strategy. Choose \( \varepsilon := 0.1 \), \( \delta := 0.2 \) and run the algorithm for \( T = 100 \) iterations.

The details for the submission are as follows:

1. You can implement the algorithm in one of the programming languages C/C++/Java/Pascal/Basic/Matlab (you can choose your favourite one).

2. Your submission should contain your (compilable) code together with an output of the algorithm, which states \( t, w^t, p^t, j_t \) for all iterations \( t = 0, \ldots, 100 \).

3. Send the files till 20.10.10 to thomas.rothvoss@epfl.ch

4. You can work in groups up to 3 people (you need only one submission per group).