582669 Supervised Machine Learning (Spring 2014)
Homework 4, sample solutions

Credit for the solutions goes to mainly to Panu Luosto and Joonas Paalasmaa, with some additional contributions by Jyrki Kivinen

Problem 1

First we bound the drop of potential in one step. Let \( t \in \{1, 2, \ldots, T\} \). If \( \sigma_t = 0 \) then \( P_t - P_{t+1} = 0 \). Assume \( \sigma_t = 1 \). Then,

\[
P_t - P_{t+1} = \frac{1}{2} \| u - w_t \|^2 - \frac{1}{2} \| u - w_{t+1} \|^2
\]

\[
= (w_{t+1} - w_t) \cdot (u - w_t) - \frac{1}{2} \| w_t - w_{t+1} \|^2.
\]

Since the update rule is \( w_{t+1} - w_t = \eta \sigma_t y_t x_t = \eta y_t x_t \), we can write the difference in the form

\[
P_t - P_{t+1} = \eta y_t x_t \cdot (u - w_t) - \frac{1}{2} \| \eta y_t x_t \|^2
\]

\[
= \eta y_t x_t \cdot u - \eta y_t x_t \cdot w_t - \frac{1}{2} \eta^2 \| x_t \|^2.
\]

(1)

It is assumed that \( y_t x_t \cdot u \geq 1 \). Because \( \sigma_t = 1 \), it holds \( -\eta y_t x_t \cdot w_t \geq 0 \). Additionally \( \| x_t \| \leq X \). Plugging all that into (1) yields

\[
P_t - P_{t+1} \geq \eta - \frac{1}{2} \eta^2 X^2.
\]

We can now use the result in the following:

\[
P_1 = \frac{1}{2} \| u - w_{\text{init}} \|^2
\]

\[
\geq P_1 - P_{T+1}
\]

\[
= \sum_{t=1}^{T} (P_t - P_{t+1})
\]

\[
\geq \sum_{t=1}^{T} \left( \eta - \frac{1}{2} \eta^2 X^2 \right) \sigma_t.
\]

If \( \eta - (1/2) \eta^2 X^2 > 0 \), that yields

\[
\sum_{t=1}^{T} \sigma_t \leq \frac{\frac{1}{2} \| u - w_{\text{init}} \|^2}{\eta - \frac{1}{2} \eta^2 X^2}
\]

To get the lowest upper bound for the number of mistakes, we maximize the value in the denominator. Let \( f(\eta) = \eta - (1/2) \eta^2 X^2 \). The function \( f \) is differentiable and concave in the set \([0, \infty]\), and it has its maximum where \( f'(\eta) = 1 - \eta X^2 = 0 \Rightarrow \eta = 1/X^2 \). The maximum value \( f(1/X^2) = 1/(2X^2) \) is positive, and the bound is thus

\[
\sum_{t=1}^{T} \sigma_t \leq \frac{\frac{1}{2} \| u - w_{\text{init}} \|^2 \cdot 2X^2}{\eta - \frac{1}{2} \eta^2 X^2}
\]

\[
= \| u - w_{\text{init}} \|^2 X^2.
\]
Problem 2

Let \( x = (x_1, x_2, \ldots, x_n) \). We use a notational shorthand \( x^m = (x_1, x_2, \ldots, x_m) \) in the following. The definition of the kernel can be then written somewhat explicitly as

\[
k_q(x^m, z^m) = \sum_{A \subseteq \{1, 2, \ldots, m\} : |A| = q} \psi_A(x^m) \psi_A(z^m)
\]

Notice first that if \( q = 1 \), then

\[
k_q(x^m, z^m) = \sum_{i=1}^{m} \psi_{\{i\}}(x^m) \psi_{\{i\}}(z^m)
\]

\[
= \sum_{i=1}^{m} x_i z_i
\]

\[
= x^m \cdot z^m.
\]

Let now \( q \in \{2, 3, \ldots, n\} \), and let \( m \in \{q, q+1, \ldots, n\} \). For further notational convinence, we define \( k_i(x^{i-1}, z^{i-1}) = 0 \) for all \( i \in \{2, \ldots, n\} \). Now we can derive a recursion formula

\[
k_q(x^m, z^m) = \sum_{A \subseteq \{1, 2, \ldots, m\} : |A| = q} \psi_A(x^m) \psi_A(z^m)
\]

\[
= \sum_{A \subseteq \{1, 2, \ldots, m\} : |A| = q} \prod_{i \in A} x_i z_i
\]

\[
= \sum_{A \subseteq \{1, 2, \ldots, m-1\} : |A| = q-1} \prod_{i \in A} x_i z_i \cdot \sum_{A \subseteq \{1, 2, \ldots, m\} : |A| = q} \prod_{i \in A} x_i z_i
\]

\[
= k_{q-1}(x^{m-1}, z^{m-1}) \cdot x_m z_m + k_q(x^{m-1}, z^{m-1}).
\]

Our algorithm is simply the following. For \( i \in \{1, 2, \ldots, q\} \) in ascending order, compute and store the values \( k_i(x^i, z^i), k_i(x^{i+1}, z^{i+1}), \ldots, k_i(x^n, z^n). \)

For computing the values for \( k_i(\cdot, \cdot) \), time \( O(n) \) is enough, because \( k_1(x^i, z^i) = k_1(x^{i-1}, z^{i-1}) + x_i z_i \) for all \( i \in \{2, 3, \ldots, n\} \). After that, every single value can be computed using the recursion formula in constant time, so the overall time requirement is \( O(nq) \).

Problem 3

We use a similar potential function as in the first exercise, so the drop of the potential is again for all \( t \in \{1, 2, \ldots, T\} \)

\[
P_t - P_{t+1} = \frac{1}{2} \| u - w_t \|^2 - \frac{1}{2} \| u - w_{t+1} \|^2
\]

\[
= (w_{t+1} - w_t) \cdot (u - w_t) - \frac{1}{2} \| w_t - w_{t+1} \|^2.
\]

The update rule is this time \( w_{t+1} - w_t = \eta(y_t - \hat{y}_t)x_t \), and thus \( (w_{t+1} - w_t) \cdot (u - w_t) = \eta(y_t - \hat{y}_t)x_t \cdot (u - w_t) = \eta(y_t - \hat{y}_t)(u \cdot x_t - w_t \cdot x_t) = \eta(y_t - \hat{y}_t)^2 \). Using the assumption that \( \| x_t \|_2 \leq X \) for all \( t \), we have a bound

\[
P_t - P_{t+1} = \eta(y_t - \hat{y}_t)^2 - \frac{1}{2} \eta(y_t - \hat{y}_t)^2 \| x_t \|^2
\]

\[
\geq \eta(y_t - \hat{y}_t)^2 - \frac{1}{2} \eta^2 X^2 (y_t - \hat{y}_t)^2
\]

\[
= \left( \eta - \frac{1}{2} \eta^2 X^2 \right) (y_t - \hat{y}_t)^2.
\]
The potentials are all non-negative, and we can estimate that
\[ P_1 = \frac{1}{2} \|u - 0\|^2_2 \]
\[ = \frac{1}{2} \|u\|^2_2 \]
\[ \geq P_1 - P_{T+1} \]
\[ = \sum_{t=1}^T (P_t - P_{t+1}) \]
\[ \geq \sum_{t=1}^T \left( \eta - \frac{1}{2} \eta^2 X^2 \right) (y_t - \hat{y}_t)^2. \]

From the first exercise we know that \( \eta - (1/2) \eta^2 X^2 = 1/(2X^2) > 0 \) when \( \eta = 1/X^2 \). That yields
\[ \sum_{t=1}^T (y_t - \hat{y}_t)^2 \leq \frac{1}{2} \|u\|^2_2 \cdot 2X^2 \]
\[ = \|u\|^2_2 X^2. \]

**Extra credit.** In the more general case, we do not assume any more \( u \cdot x_t = y_t \). For all \( t \), we can still write
\[ P_t - P_{t+1} = \eta(y_t - \hat{y}_t)x_t \cdot (u - w_t) - \frac{1}{2} \|w_t - w_{t+1}\|^2_2 \]
\[ \geq \eta(y_t - \hat{y}_t)(u \cdot x_t - \hat{y}_t) - \frac{1}{2} \eta^2 (y_t - \hat{y}_t)X^2 \]
\[ = \eta(y_t - \hat{y}_t)((u \cdot x_t - y_t) + (y_t - \hat{y}_t)) - \frac{1}{2} \eta^2 (y_t - \hat{y}_t)^2 X^2 \]
\[ = \left( \eta - \frac{1}{2} \eta^2 X^2 \right) (y_t - \hat{y}_t)^2 - \eta(y_t - \hat{y}_t)(u \cdot x_t - y_t). \]

Because it holds for any \( a, b \in \mathbb{R} \) that \( (a - b)^2 = a^2 - 2ab + b^2 \geq 0 \Rightarrow ab \leq (1/2)(a^2 + b^2) \), we can estimate that
\[ P_t - P_{t+1} \geq \left( \eta - \frac{1}{2} \eta^2 X^2 \right) (y_t - \hat{y}_t)^2 - \eta(y_t - \hat{y}_t)(u \cdot x_t - y_t) \]
\[ \geq \left( \eta - \frac{1}{2} \eta^2 X^2 \right) (y_t - \hat{y}_t)^2 - \frac{1}{2} \eta^2 ((y_t - \hat{y}_t)^2 + (u \cdot x_t - y_t)^2) \]
\[ = \frac{1}{2} \left( \eta - \eta^2 X^2 \right) (y_t - \hat{y}_t)^2 - \frac{1}{2} \eta(y_t - u \cdot x_t)^2. \]

Summing over \( t \in \{1, 2, \ldots, T\} \) yields
\[ P_1 = \frac{1}{2} \|u\|^2_2 \]
\[ \geq \sum_{t=1}^T (P_t - P_{t+1}) \]
\[ \geq \sum_{t=1}^T \left( \frac{1}{2} (\eta - \eta^2 X^2)(y_t - \hat{y}_t)^2 - \frac{1}{2} \eta(y_t - u \cdot x_t)^2 \right) \]
or
\[ (\eta - \eta^2 X^2) \sum_{t=1}^T (y_t - \hat{y}_t)^2 \leq \|u\|^2_2 + \eta \sum_{t=1}^T (y_t - u \cdot x_t)^2. \]
When $\eta - \eta^2 X^2 > 0 \Rightarrow \eta < 1/X^2$, we have thus the bound
\[
\sum_{t=1}^{T} (y_t - \hat{y}_t)^2 \leq \frac{1}{\eta - \eta^2 X^2} \left( \|u\|_2^2 + \eta \sum_{t=1}^{T} (y_t - u \cdot x_t)^2 \right)
\]
\[
= \frac{\|u\|_2^2}{\eta - \eta^2 X^2} + \frac{1}{1 - \eta X^2} \sum_{t=1}^{T} (y_t - u \cdot x_t)^2.
\]
This bound has an additional penalty depending on the sum of squared mistakes that the classifier $u$ makes.

**Problem 4**

The Perceptron Convergence Theorem guarantees that the Perceptron Algorithm makes at most $B^2/\gamma^2$ mistakes, where $B$ is an upper bound for the Euclidean norm of any $x$ in the sample. Since the data was sampled from the $[-1, 1]^d$ hypercube, we know that $B^2 \leq d$. However, this bound was very loose for the following experiment.

```octave
#!/usr/bin/octave -qc

function [xs, ys] = make_sample (n, d, gam)
    xs = zeros (n, d);
    ys = zeros (n, 1);
    cnt = 0;
    while (cnt < n)
        x = 2 * (rand (1, d) - 0.5);
        t = (x(1) + x(2)) / sqrt (2);
        if (abs (t) > gam)
            xs(++cnt, :) = x;
            ys(cnt) = sign (t);
        endif
    endwhile
endfunction

function mistakes = perceptron (xs, ys)
    mistakes = 0;
    right_predictions = 0;
    n = rows (xs);
    xs = cat (2, xs, ones (n, 1));
    w = zeros (1, columns (xs));
    while (right_predictions < n)
        for i = [1:n]
            hat_y = sign (w * xs(i, :));
            if (hat_y == 0)
                hat_y = 1;
            endif
            if (hat_y != ys(i))
                mistakes++;
                right_predictions = 0;
                w = w + ys(i) * xs(i, :);
            elseif (++right_predictions == n)
                break;
            endif
        endfor
    endfor
```
endwhile
endfunction

figure (1, "visible", "off");
n = 1000;

ds = [5 10 100 500 1000];
gammas = [0.05 : 0.05 : 1];
mistakes = zeros ( length (ds), length (gammas));
for i = [1 : (length (ds))]
  for j = [1 : (length (gammas))]
    [xs ys] = make_sample (n, ds(i), gammas(j));
    mistakes(i, j) = perceptron (xs, ys);
  endfor
  plot (gammas, mistakes(i, :) ,"+");
  print ( sprintf ("dim_%d.eps", ds(i)), "-deps");
endfor
plot (gammas, mistakes', "o-");
print ( "fixed_dim.eps", "-deps");

gammas = [0.05 0.1 0.4 0.7 1];
ds = [5 : 50 : 1000];
mistakes = zeros ( length (gammas), length (ds));
for i = [1 : (length (gammas))]
  for j = [1 : (length (ds))]
    [xs ys] = make_sample (n, ds(j), gammas(i));
    mistakes(i, j) = perceptron (xs, ys);
  endfor
  plot (ds, mistakes(i, :) ,"+");
  print ( sprintf ("gamma_%.2f.eps", gammas(i)), "-deps");
endfor
plot (ds, mistakes', "o-");
print ( "fixed_gamma.eps", "-deps");
Figure 1: Number of mistakes as a function of $\gamma$. Number of dimensions from bottom to top: 5, 10, 100, 500, 1000. The size of the sample was $n = 1000$.

Figure 2: Number of mistakes as a function of $d$. $\gamma$ from top to bottom: 0.05, 0.1, 0.4, 0.7, 1. The size of the sample was $n = 1000$. 