1. [16 points] Consider the Aggregating Algorithm for logarithmic loss. It turns out that in this case it is a good idea to choose $\eta = c = 1$. In other words, the weight update is $w_{t+1,i} = w_{t,i} \exp(-L_{\log}(y_t, E_{i,t}))$, where $E_{i,t}$ is the prediction of expert $i$ at time $t$, and the potential we consider is $P_t = \ln W_t$ where $W_t = \sum_{j=1}^{n} w_{t,j}$. As prediction, we take simply the weighted average $\hat{y}_t = \sum_{i=1}^{n} v_{t,i} h_i(x_t)$, where $v_{t,i} = w_{t,i}/W_t$.

(a) Show that at each time step $t$ we have

\[ L_{\log}(y_t, \hat{y}_t) = P_t - P_{t+1}. \]

(b) Using the result from part (a), derive a bound for the total loss of the algorithm in terms of the loss of the best expert.

(c) Suppose we want to get a similar result for absolute loss instead of logarithmic loss. Describe briefly how the algorithm needs to be changed and what kind of a bound we would get. You are not expected to give optimal values for any parameters or do any other technical derivations, a simple explanation is sufficient.

2. [16 points] Let $H$ be a finite set of classifiers with input space $X$.

(a) In the context of agnostic PAC model (our basic statistical learning model), what do we mean by empirical and true risk? State a bound that relates the empirical and true risk to each other. Remember to define all the terms and symbols you use.

(b) Prove the bound you gave in part (a). You may find the Hoeffding Inequalities useful: If $S = \sum_{i=1}^{m} X_i$ where $X_i$ are independent random variables with $a_i \leq X_i \leq b_i$, then

\[
\Pr[S \geq E[S] + t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{m} (b_i - a_i)}\right)
\]

\[
\Pr[S \leq E[S] - t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{m} (b_i - a_i)}\right).
\]

Continues on the reverse side!
3. [16 points] Suppose we are given $m$ examples $(x_i, y_i), i = 1, \ldots, m$, where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$, and wish to learn a linear function mapping $x$ to $y$. One method (somewhat non-traditional but in line with the basic Support Vector Machine approach) is to choose $w \in \mathbb{R}^d$ that minimises

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{m} L_\varepsilon(w \cdot x_i - y_i)$$

where we have used the $\varepsilon$-insensitive loss

$$L_\varepsilon(z) = \begin{cases} 
-z - \varepsilon & \text{if } z < -\varepsilon \\
0 & \text{if } -\varepsilon \leq z \leq \varepsilon \\
z - \varepsilon & \text{if } \varepsilon < z.
\end{cases}$$

Here $C > 0$ and $\varepsilon > 0$ are some constants. However, this formulation of the minimisation problem is inconvenient because $L_\varepsilon$ is not differentiable.

Rewrite the above minimisation problem as a convex optimisation problem where the objective and constraint functions are differentiable. (**Hint:** use slack variables $\xi'_i$ and $\xi''_i$ that have the intuitive meaning $\xi'_i = \max \{ -w \cdot x_i + y_i - \varepsilon, 0 \}$ and $\xi''_i = \max \{ w \cdot x_i - y_i - \varepsilon, 0 \}$.) Obtain the dual of the problem. Show also the kernelised version of the dual, where we assume that $x_i = \psi(z_i)$ is a feature vector and $k$ is the kernel for the feature mapping $\psi$. 