Problem 1.

Consider the training objective $J = ||Xw - t||^2$ subject to $||w||^2 \leq C$ for some constant C.

How would the hypothesis class capacity, overfitting/underfitting, and bias/variance vary according to C?

	Larger C	Smaller C
Model capacity (large/small?)	______	______
Overfitting/Underfitting?	__fitting	__fitting
Bias variance (high/low?)	__ bias / __ variance	__ bias / __ variance

Solution 1.

	Larger C	Smaller C
Model capacity (large/small?)	**Large**	**Small**
Overfitting/Underfitting?	**Over**fitting	**Under**fitting
Bias variance (high/low?)	**Low** bias / **High** variance	**High** bias / **Low** variance

Problem 2. (cross-ref: W3)
Consider the l_2-penalized square error as the training objective, given by

$$J = \frac{1}{2M} \sum_{m=1}^{M} \Big(\sum_{i=0}^{d} w_i x_i^{(m)} - t^{(m)} \Big)^2 + \lambda \sum_{i=0}^{d} w_i^2$$

where λ is a positive constant.

a) Give a closed-form solution to the problem.
b) Give a gradient-based solution to the problem. Requirement: Write pseudocode and calculate the gradient.

Solution 2.

(See Problem 3, W2-sol for additional details.) It is quite convenient to work with the vector notation here. Let $x^{(m)} \in \mathbb{R}^{d+1}$ be the vector defined as $(x^{(m)})_i = x_i^{(m)}$. Similarly, define $w \in \mathbb{R}^{d+1}$, $t \in \mathbb{R}^M$, and $X \in \mathbb{R}^{M\times(d+1)}$, where the rows of X are made up of $x^{(m)}$s, i.e. $X_{i,j} = x_j^{(i)}$ for $1 \le i \le M$ and $1 \le j \le d + 1$. Next, note that the above loss can be written in the vector notation as:

$$J = \frac{1}{2M}||Xw - t||_2^2 + \lambda||w||_2^2.$$

Then we directly obtain the following gradient:

$$\nabla J = \frac{1}{M}X^\top(Xw - t) + 2\lambda w.$$

a) The problem to solve is $min_w J$. To do this, we set the gradient of the objective to zero:

$$\nabla J = 0 \qquad \Rightarrow \qquad (X^\top X + 2M\lambda I)w^* = X^\top t,$$

and then solve for the closed form expression of w:

$$w^* = (X^\top X + 2M\lambda I)^{-1}X^\top t.$$

Note that since the Hessian $\nabla^2 J = \frac{\partial}{\partial w}\nabla J = \frac{1}{M}X^\top X + 2\lambda I$ is PSD (why?), MSE is a convex function (also see the notes for Lecture 5). This means that the above solution w^* is a global minimum of the MSE problem. (Note that, for all $M > 0$ and $\lambda > 0$, the matrix $X^\top X + 2M\lambda I$ is guaranteed to be non-singular (why?)).

b) The pseudo-code for the gradient based solution is:

1. Input initial weights $w^{(0)} \in \mathbb{R}^{d+1}$, stepsize α, and the number of iterations T
2. For $t = 1, 2, ..., T$:
3. Compute the gradient:
$$\nabla J_{w^{(t-1)}} = \frac{1}{M}X^\top(Xw^{(t-1)} - t) + 2\lambda w^{(t-1)}.$$
4. Update the weights:
$$w^{(t)} = w^{(t-1)} - \alpha \nabla J_{w^{(t-1)}}$$
5. Return the weights $w^{(T)}$

Problem 3.

Give the prior distribution of w for linear regression, such that the max a posteriori estimation is equivalent to l_1-penalized mean square loss.

Note: Such a prior is known as the *Laplace distribution*. Also, getting the normalization factor in the distribution is not required.

Solution 3.

In this solution, we will assume that the conditional distribution of the output labels given the input data is normally distributed. Furthermore, we will put a Laplacian prior on the weights themselves. Then the solution is apparent.

For a linear regression problem with d-dimensional input $\mathbf{x} = [x_1, \ldots, x_d]^\intercal \in \mathbb{R}^d$ and outputs $y \in \mathbb{R}$. Now, let $\hat{\mathbf{x}}$ be the augmented input by padding $\mathbf{x}$ with a one (i.e. $\hat{\mathbf{x}} = [1, x_1, \ldots, x_d]^\intercal$). Then we can define our model as a linear function $f(\hat{\mathbf{x}}) = \mathbf{w}^\intercal \hat{\mathbf{x}}$ with parameters $\mathbf{w} = [w_0, w_1, \ldots, w_d]^\intercal \in \mathbb{R}^{d+1}$ such that its first entry is the bias term.

To obtain a maximum a posteriori (MAP) estimate, we treat the parameters as random variables. To this end, we define a prior over $\mathbf{w}$ such that all random variables are independent, with w_0 following $\mathcal{U}(-1/(2c), 1/(2c))$ (this is an improper prior) and w_i following Laplace$(0, b), b > 0$ for $i = 1, \ldots, d$. Then, we can write the prior as

$$
\begin{aligned}
p(\mathbf{w}) &= \prod_{i=0}^{d} p(w_i) \\
&= c \prod_{i=1}^{d} \frac{1}{2b} \exp\left(-\frac{|w_i|}{b}\right).
\end{aligned}
$$

Now, given dataset $D = \{(\hat{\mathbf{x}}^{(i)}, y^{(i)})\}_{i=1}^{N}$, where the input/output pairs are i.i.d., we assume that $y \sim \mathcal{N}(\mathbf{w}^\intercal \hat{\mathbf{x}}, \sigma^2)$, where $\sigma^2 > 0$ is a constant. Thus, we can write the MAP estimator as follows:

$$
\begin{aligned}
\mathbf{w}^{\mathrm{MAP}} &= \underset{\mathbf{w}}{\operatorname{argmax}} P(\mathbf{w}|D) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \frac{P(D|\mathbf{w})P(\mathbf{w})}{P(D)} \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} P(D|\mathbf{w})P(\mathbf{w}) \\
&= \underset{\mathbf{w}}{\operatorname{argmax}} \left(\prod_{i=1}^{N} P(y^{(i)}|\mathbf{w}, \hat{\mathbf{x}}^{(i)})\right) \left(c \prod_{i=1}^{d} \frac{1}{2b} \exp\left(-\frac{|w_i|}{b}\right)\right).
\end{aligned}
$$

This optimization problem is exactly the same as solving for the negative log posterior:

$$
\begin{aligned}
\mathbf{w}^{\text{MAP}} &= \underset{\mathbf{w}}{\operatorname{argmax}} \left(\prod_{i=1}^{N} P(y^{(i)}|\mathbf{w}, \hat{\mathbf{x}}^{(i)})\right) \left(c\prod_{i=1}^{d} \frac{1}{2b} \exp\left(-\frac{|w_i|}{b}\right)\right) \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} - \log\left(\left(\prod_{i=1}^{N} P(y^{(i)}|\mathbf{w}, \hat{\mathbf{x}}^{(i)})\right) \left(c\prod_{i=1}^{d} \frac{1}{2b} \exp\left(-\frac{|w_i|}{b}\right)\right)\right) \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} - \left(\sum_{i=1}^{N} \log\exp\left(-\frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^\intercal\hat{\mathbf{x}})^2\right)\right) - \left(\sum_{i=1}^{d} \log\exp\left(-\frac{|w_i|}{b}\right)\right) - d\log\frac{1}{2b} - N\log\frac{1}{\sqrt{2\pi\sigma^2}} - \log c \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} \left(\sum_{i=1}^{N}\left(\frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^\intercal\hat{\mathbf{x}})^2\right)\right) + \left(\sum_{i=1}^{d}\frac{|w_i|}{b}\right) \\
&= \underset{\mathbf{w}}{\operatorname{argmin}} \left(\sum_{i=1}^{N}\left(\frac{1}{2}(y^{(i)} - \mathbf{w}^\intercal\hat{\mathbf{x}})^2\right)\right) + \frac{\sigma^2}{b}\left(\sum_{i=1}^{d}|w_i|\right).
\end{aligned}
$$

The first term is from resembles the ordinary least squares loss and the second term is the L1-penalty term, where $\frac{\sigma^2}{b}$ can be seen as a "hyperparameter" controlling the amount of regularization over the weights.

Note: In this question, we added a bias term as well. If you prefer, you can avoid introducing the w_0 term.

END