Problem Set 1: Statistical Learning Framework & Regression

Due: Tuesday, February 3, 2026 at 11:59pm ET

Instructions: Your answers to the questions below, including plots and mathematical work, should be submitted as a single PDF file. It's preferred that you write your answers using software that typesets mathematics (e.g. LaTeX or MathJax in iPython), though if you need to you may scan handwritten work. You may find the minted package convenient for including source code in your LaTeX document. For the coding problems, the text **Submission:** indicates all you need to submit in your PDF submission.

Problem 1: Squared Risk (15 Points)

First, we will show that if you want to try to predict the value of a random variable, the best you can do (for minimizing expected square loss) is to predict the mean of the distribution. Your expected loss for predicting the mean will be the variance of the distribution.

Problem 1(a) (5 points)

Let y be a random variable with a known distribution, and consider the squared loss function $\ell(a, y) = (a - y)^2$. We want to find the action a^* that has minimal risk. That is, we want to find $a^* = \arg\min_a \mathbb{E}[(a - y)^2]$, where the expectation is with respect to y. Show that $a^* = \mathbb{E}[y]$ and the Bayes risk (i.e., the risk of a^*) is $\mathrm{Var}(y)$.

Now we will introduce an input. Recall that the *expected loss* or *risk* of a hypothesis/decision function $f : \mathcal{X} \to \mathcal{A}$ is

$$R(f) = \mathbb{E}\left[\ell(f(x), y)\right],$$

where $(x, y) \sim P_{\mathcal{X}\times\mathcal{Y}}$ and the *Bayes hypothesis/decision function* $f^* : \mathcal{X} \to \mathcal{A}$ is a function that achieves the minimal risk among all possible functions:

$$R(f^*) = \inf_f R(f).$$

Here we consider the regression setting, in which $\mathcal{A} = \mathcal{Y} = \mathbb{R}$. We will show that for the squared loss $\ell(a, y) = (a - y)^2$, the Bayes decision function is $f^*(x) = \mathbb{E}[y \mid x]$, where the expectation is over y. As before, assume we know the data-generating distribution $P_{\mathcal{X}\times\mathcal{Y}}$.

We'll approach this problem by finding the optimal action for any given x. If somebody tells us x, we know that the corresponding y is coming from the conditional distribution $\mathcal{P}_{\mathcal{Y}|\mathcal{X}}$.

Problem 1(b) (5 points)

For a particular x, what value should we predict (i.e. what action a should we produce) that has minimal expected loss? Express your answer as a decision function $f(x)$, which gives the best action for any given x. Formally, we are looking for

$$f^*(x) = \arg\min_a \mathbb{E}\left[(a-y)^2 \mid x\right],$$

where the expectation is with respect to y. (Hint: You may find the previous question helpful. There is nothing to write except the function $f(x)$, but make sure you understand what is going on with your answer).

In the Problem 1(b), we produced a decision function $f^*(x)$ that minimized the risk for each x. In other words, for any other decision function $f(x)$, $f^*(x)$ is going to be at least as good as $f(x)$ for every single x, as measured by expected squared loss.

Problem 1(c) (5 points)

To show that $f^*(x)$ is the Bayes decision function, we need to show that

$$\mathbb{E}\left[(f^*(x)-y)^2\right] \leq \mathbb{E}\left[(f(x)-y)^2\right]$$

for any f. Prove that this is true. (Hint: Use Problem 1(b) and law of iterated expectation)

Problem 2: Risk Decomposition (30 Points)

For the remaining problems, we will consider a synthetic prediction problem to develop our intuition about risk decomposition. The input space is $\mathcal{X} = [0, 1]$ and the outcome space is $\mathcal{Y} = \mathbb{R}$. Consider the following distribution $P_{\mathcal{X}\times\mathcal{Y}}$. The marginal distribution $P_{\mathcal{X}}$ is the uniform distribution over the unit interval, Unif$[0, 1]$ (i.e. $x \sim \text{Unif}[0, 1]$). Each $y \in \mathcal{Y} = \mathbb{R}$ is defined as a degree-2 polynomial of x. That is, there exists $(a_0, a_1, a_2) \in \mathbb{R}^3$ such that

$$y = p(x) = a_0 + a_1 x + a_2 x^2.$$

In this problem, our action space is the outcome space, so $\mathcal{A} = \mathcal{Y} = \mathbb{R}$. We aim to find a hypothesis $h : \mathcal{X} \to \mathbb{R}$ to predict the outcomes in $\mathcal{Y} = \mathbb{R}$; let $\hat{y} = h(x)$ be the predicted outputs. Let $\mathcal{H}_d$ be the set of polynomial functions on $\mathbb{R}$ of degree d:

$$\mathcal{H}_d = \left\{ x \mapsto w_0 + w_1 x + \cdots + w_d x^d : w_i \in \mathbb{R} \text{ for } 0 \le i \le d \right\}.$$

We will consider hypothesis classes $\mathcal{H}_d$ with varying values of d. We will minimize the squared loss $\ell(\hat{y}, y) = (\hat{y} - y)^2$.

Problem 2(a) (5 points)

Recall the definition $R(f)$ of a predictor f. While this cannot be computed in general, note that we know $P_{\mathcal{X}\times\mathcal{Y}}$ exactly in this problem. With this knowledge, state which function $f^* : \mathcal{X} \to \mathcal{Y}$ is the obvious Bayes predictor and justify why $R(f^*)$ is minimum at f^*.

Problem 2(b) (5 points)

Using $\mathcal{H}_2$ as your hypothesis class, which function $f^*_{\mathcal{H}_2}$ is the risk minimizer in $\mathcal{H}_2$? What is the approximation error achieved by $f^*_{\mathcal{H}_2}$?

Problem 2(c) (5 points)

Consider now $\mathcal{H}_d$ with $d > 2$. For any statistical learning problem (not just the one described above) what inequality direction (i.e. $\geq$ or $\leq$) should fill in

$$R(f^*_{\mathcal{H}_2}) \;\square\; R(f^*_{\mathcal{H}_d})?$$

Make sure you justify your decision. For this problem, which function $f^*_{\mathcal{H}_d}$ is a risk minimizer in $\mathcal{H}_d$? What is the approximation error achieved by $f^*_{\mathcal{H}_d}$?

Problem 2(d) (10 points)

For this question, assume $a_0 = 0$. Now, consider the hypothesis class

$$\mathcal{H} = \{x \mapsto w_1 x : w_1 \in \mathbb{R}\}.$$

Which function $f^*_{\mathcal{H}}$ is a risk minimizer in $\mathcal{H}$? What is the approximation error achieved by $f^*_{\mathcal{H}}$? Suppose, further, that $a_2 = 0$ as well. Then, what is the approximation error of $f^*_{\mathcal{H}}$?

Problem 2(e) (5 points)

In the previous questions, we assumed that the conditional distribution $y \mid x$ was completely determined by the function $p(x) = a_0 + a_1 x + a_2 x^2$. In some real-world scenarios, however, there may be some random noise in the relationship between y and x. Now, suppose that

$$y = p(x) + \epsilon = a_0 + a_1 x + a_2 x^2 + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, 1)$ is distributed as an independent standard Gaussian random variable. In this case, what is the marginal distribution $P_{\mathcal{X}}$ and what is the conditional distribution $P_{\mathcal{Y}|\mathcal{X}}$? State (and justify) which function $f^* : \mathcal{X} \to \mathcal{Y}$ is the Bayes predictor and state the Bayes risk $R(f^*)$ (Hint: Problem 1 may be helpful).

Problem 3: Polynomial Regression & Least Squares (25 Points)

This problem is a continuation of the setup of Problem 2. In practice, $P_{\mathcal{X}\times\mathcal{Y}}$ is usually unknown and we use the empirical risk minimizer (ERM). We will reformulate the setup in Problem 2 as a d-dimensional linear regression problem. First, note that functions in $\mathcal{H}_d$ are parameterized by a vector

$$w = (w_0, w_1, \ldots, w_d) \in \mathbb{R}^{d+1}.$$

We will use the notation $f_w : \mathcal{X} \to \mathbb{R}$ to denote the hypothesis f_w parameterized by w. Similarly, we use $p_a : \mathcal{X} \to \mathbb{R}$ to denote the true polynomial parameterized by $a = (a_0, a_1, a_2) \in \mathbb{R}^3$. We can assume that we are back in the zero noise setting for the conditional distribution $P_{\mathcal{Y}|\mathcal{X}}$, where

$$y = p_a(x) = a_0 + a_1 x + a_2 x^2.$$

As is typical in the statistical learning framework, we receive a sample $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$ drawn i.i.d. from $P_{\mathcal{X}\times\mathcal{Y}}$. In this problem, our goal will be to show that polynomial regression reduces to performing linear least squares regression on a transformed dataset.

As stated above, our samples $(x^{(i)}, y^{(i)})$ are pairs of real numbers, $x^{(i)} \in \mathcal{X} = [0, 1]$ and $y^{(i)} \in \mathbb{R}$. We can imagine transforming each one-dimensional $x^{(i)}$ into a $d + 1$ dimensional feature vector

$$\overline{x^{(i)}} = \begin{pmatrix} 1 & x^{(i)} & (x^{(i)})^2 & \ldots & (x^{(i)})^d \end{pmatrix} \in \mathbb{R}^{d+1}$$

and stacking these feature vectors row-by-row into a matrix $X \in \mathbb{R}^{n\times(d+1)}$. We can also arrange all the $y^{(1)}, \ldots, y^{(n)}$ into a vector in $\mathbb{R}^n$ as well. This results in the following matrix and vector:

$$X = \begin{bmatrix} 1 & x^{(1)} & \ldots & (x^{(1)})^d \\ 1 & x^{(2)} & \ldots & (x^{(2)})^d \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x^{(n)} & \ldots & (x^{(n)})^d \end{bmatrix} \qquad y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix} \tag{1}$$

We refer to $X \in \mathbb{R}^{n\times(d+1)}$ as the *design matrix* or *input matrix* and $y \in \mathbb{R}^n$ as the *label vector*. Writing these objects this way allows us to take advantage of the tools of linear algebra to solve the empirical risk minimization problem.

Problem 3(a) (10 points)

Recall the definition of empirical risk minimization from class. Show that solving the minimization problem with the above design matrix and label vector

$$\hat{w} \in \operatorname*{arg\,min}_{w \in \mathbb{R}^{d+1}} \|Xw - y\|_2^2$$

yields an empirical risk minimizer $f_{\hat{w}} \in \mathcal{H}_{d+1}$ for squared loss $\ell(\hat{y}, y) = (\hat{y} - y)^2$ on the dataset $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$.

Problem 3(b) (5 points)

Prove the following useful lemma from linear algebra: for any real matrix X, $\operatorname{rank}(X) = \operatorname{rank}(X^\top X)$.

One way to solve the following problem is via standard optimization and multivariable calculus. You may find the following identities from matrix calculus useful:

- For a fixed symmetric $A \in \mathbb{R}^{d \times d}$, $\nabla_v v^\top A v = 2Av$.
- For a fixed symmetric $A \in \mathbb{R}^{d \times d}$, $\nabla_v^2 v^\top A v = 2A$.
- For a fixed vector $a \in \mathbb{R}^d$, $\nabla_v a^\top v = a$.

Above, ∇_v denotes the gradient with respect to v and ∇_v^2 denotes the Hessian with respect to v. You may also freely use the following fact: if $F : \mathbb{R}^d \to \mathbb{R}$ is twice-differentiable and has a positive semidefinite Hessian $\nabla_v^2 F(v) \in \mathbb{R}^{d \times d}$, then any critical point (i.e., v where $\nabla_v F(v) = 0$) of F is a global minimizer.

Problem 3(c) (10 points)

Using the Lemma in 3(b), show that if $n > d$ and X is full rank, the solution to the minimization problem in (1)

$$\hat{w} \in \operatorname*{arg\,min}_{w \in \mathbb{R}^{d+1}} \|Xw - y\|_2^2$$

is given by $\hat{w} = (X^\top X)^{-1} X^\top y$. In this problem, be sure to divide by n. (Hint: solve the optimization problem by taking gradients with respect to w).

In general, the empirical risk minimization problem for least squares regression with a linear hypothesis class can be solved as in Question 4 above. That is, if we want to find

$$w \in \operatorname*{arg\,min}_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \left(w^\top x^{(i)} - y^{(i)} \right)^2$$

for $x^{(i)} \in \mathbb{R}^d$, we can construct a design matrix $X \in \mathbb{R}^{n\times d}$ and outcome vector $y \in \mathbb{R}^n$

$$X = \begin{bmatrix} \leftarrow x^{(1)} \rightarrow \\ \leftarrow x^{(2)} \rightarrow \\ \vdots \\ \leftarrow x^{(n)} \rightarrow \end{bmatrix} \quad y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$$

and solve the linear algebraic formulation:

$$w \in \operatorname*{arg\,min}_{w\in\mathbb{R}^d} \|Xw - y\|^2.$$

As Question 4 shows, the closed-form solution is $w = (X^\top X)^{-1}X^\top y$. For polynomial regression, we encoded each $x^{(i)}$ into a $d+1$-dimensional vector of *features* $\overline{x^{(i)}}$ to convert our originally nonlinear problem of solving for a polynomial into a linear problem where we only needed to solve for the *coefficients* of the polynomial.

Problem 4: Polynomial Regression Implementation (30 Points)

We now continue this polynomial regression exploration with a hands-on coding problem. Open the source code file `ps1_skeleton.py` from the `ps1-regression.zip` folder.

Problem 4(a) (5 points)

Write a function called `least_squares_estimator` taking as input a design matrix $X \in \mathbb{R}^{n\times(d+1)}$ and a corresponding vector of labels $y \in \mathbb{R}^n$, returning $\hat{w} \in \mathbb{R}^{d+1}$. Your function should handle any value of n and d and it should return an error if $n \leq d$. You can assume that the input design matrix X is full rank, as drawing the x at random from the uniform distribution makes it almost certain that any design matrix X is full rank.
Submission: Include the code you write in the your submission.

Problem 4(b) (3 points)

Recall the definition of empirical risk $\hat{R}_n(h)$ of some hypothesis h for a sample $(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)})$. Write a function `empirical_risk` that takes as input a design matrix $X \in \mathbb{R}^{n\times(d+1)}$, a vector $y \in \mathbb{R}^n$, and a $w \in \mathbb{R}^{d+1}$ and outputs the empirical risk of the hypothesis f_w parameterized by w (see Problem 3). In this problem, be sure to divide by n.
Submission: Include the code you write in your submission.

Problem 4(c) (5 points)

Using the function `get_a` with $d = 5$, get a value for a and draw `x_train`, `y_train` of size $n = 10$. Use the code you wrote to estimate $\hat{w}$ from `x_train` and `y_train`. Compare $\hat{w}$ and a. Make a plot (Plot 1) with x on the x-axis and y on the y-axis, displaying the points in your training set and the true underlying function $p(x)$ in $[0, 1]$. Make a second plot (Plot 2) with x on the x-axis and y on the y-axis, displaying the points in your training set and your estimated function $f_{\hat{w}}(x)$ in $[0, 1]$. You should use `np.linspace` from NumPy to ensure that $p(x)$ and $f_{\hat{w}}(x)$ are sufficiently smooth.
Submission: Submit Plot 1 and Plot 2 (no code). Update (01/25): A previous version of this problem mentioned drawing a test set, but there was nothing to do with that test set. We have removed that text from the problem.

Probelm 4(d) (2 points)

Now, adjust d. What values of d can we get a "perfect fit?" How does this result relate to the conclusions you made about approximation error above?
Submission: Submit a written answer to these questions (no code).

Now we will modify the true underlying $P_{\mathcal{X}\times\mathcal{Y}}$ by adding some noise. In particular, the `draw_sample_with_noise` function generates

$$y = p(x) + \epsilon$$

where $\epsilon \sim \mathcal{N}(0,1)$ is distributed as a standard Gaussian random variable.

Problem 4(e) (6 points)

Now using `draw_sample_with_noise` to draw samples, plot the empirical risk of $f_{\hat{w}}$ on the training set that $\hat{w}$ was obtained from as a function of n for $d < n < 150$ for $d = 2$ (Plot 3). On the same plot, plot the empirical risk of $f_{\hat{w}}$ on an independent test set of size $n_{\text{test}} = 1000$ using a separate call to `draw_sample` (with the same underlying $a \in \mathbb{R}^d$ for each d) for each of the same values. The x-axis of this plot should be n, and the y-axis should be the values for risk. You should use a logarithmic scale for the plot's y-axis. Repeat this for $d = 5$ and $d = 10$ (Plots 4 and 5). There should be a total of 3 different plots (Plot 3, Plot 4, Plot 5) for $d = 2$, $d = 5$, and $d = 10$, each with two curves. **Submission:** Submit Plots 3, 4, 5 (no code). Update (01/25): Added guidance on drawing an independent test set in this problem for clarity.

Problem 4(f) (9 points)

Using `draw_sample_with_noise` for the values $n = 15$, $n = 30$, and $n = 100$ for each of $d = 2$, $d = 5$, and $d = 10$, fit $f_{\hat{w}}$ on these training points. Make a single plot for each of these pairs of (n, d) of the training points, the underlying true function $p(x)$ in $[0, 1]$, and the estimated function $f_{\hat{w}}$ in $[0, 1]$. There should be a total of 9 plots (Plots 6 - 14). You should use `np.linspace` from NumPy to ensure that $p(x)$ and $f_{\hat{w}}(x)$ are sufficiently smooth. Comment on the effect of increasing n. Comment on the effect of increasing d. **Submission:** Comment on n and d and Plots 6 - 14 (no code).