Homework #1

ECE 461/661: Introduction to Machine Learning for Engineers
Prof. Soummya Kar and Prof. Andrea Zanette

Due: Monday, Sep 9, 2024 11:59 pm ET

Please remember to show your work for all problems and to write down the names of any students that you collaborate with. The full collaboration and grading policies are available on the course website: https://18661.github.io. You are strongly encouraged (but not required) to use Latex to typeset your solutions.

Your solutions should be uploaded to Gradescope (https://www.gradescope.com/) in PDF format by the deadline. We will not accept hardcopies. If you choose to hand-write your solutions, please make sure the uploaded copies are legible. Gradescope will ask you to identify which page(s) contain your solutions to which problems, so make sure you leave enough time to finish this before the deadline. We will give you a 30-minute grace period to upload your solutions in case of technical problems.

1 Warm-up *[15 points]*

1. (5 points) **Multivariable Calculus**: Consider a real function $f(x,z) = \arctan\left(x\sin(z)\right)e^{-(xz-3)^2}$, where $x, z \in \mathbb{R}$. What is the partial derivative of $f(x,z)$ with respect to x?

2. (6 points) Are the following functions convex, concave, or neither? Justify your answer with a proof.

   (a) $h(p) = p\log p + (1-p)\log(1-p)$, where p lies in the range $0 < p < 1$.

   (b) $f(x) = \lambda e^{-x}$, for $x \geq 0$ and a constant $\lambda > 0$.

3. (4 points) Find the critical point(s) p^* of the function $h(p)$ defined above and specify whether they are minima or maxima.

   [***Hint:***] A *critical point* of a differentiable function $f : \mathbb{R} \to \mathbb{R}$ is a value $x \in \mathbb{R}$ at which $f'(x) = 0$.

2 Probability *[10 points]*

1. (4 points) Suppose we obtain samples $X_1, X_2, .., X_n$ which are independent, identically distributed, each with expectation μ and variance σ^2. Define the sample average to be $Y_n = \frac{\sum_{i=1}^{n} X_i}{n}$. What is $\mathbb{E}[Y_n]$ and $\mathrm{Var}(Y_n)$? Using these results, explain why it is better to have a larger number of samples.

2. (6 points) Consider the following joint distribution between X, which takes values T or F, and Y, which takes values a, b, or c.

$P(X,Y)$		Y		
		a	b	c
X	T	0.2	0.3	0.1
	F	0.1	0.1	0.2

(a) What is the marginal distribution P_Y, that is, what are $\Pr(Y = a)$, $\Pr(Y = b)$ and $\Pr(Y = c)$?

(b) What is $\Pr(Y \in \{a, b\}|X = F)$ the probability that Y is either a or b given that X is F?

3 Linear Algebra *[15 points]*

1. (5 points) Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n \times n}$ be two symmetric matrices. Suppose $\boldsymbol{A}$ and $\boldsymbol{B}$ have the exact same set of eigenvectors $u_1, u_2, ..., u_n$ with the corresponding eigenvalues $\alpha_1, \alpha_2, , \alpha_n$ for $\boldsymbol{A}$, and $\beta_1, \beta_2, ..., \beta_n$ for $\boldsymbol{B}$. Assume $\boldsymbol{A}$ is invertible and k is a finite positive integer. Write down the eigenvectors and their corresponding eigenvalues for the following matrices

   (a) $\boldsymbol{C} = \boldsymbol{A}^k - \boldsymbol{B}^k$

   (b) $\boldsymbol{D} = (\boldsymbol{A}^{-1})^k \boldsymbol{B} \boldsymbol{A}^k$

   (c) $\boldsymbol{E} = (\boldsymbol{A}^{-1} \boldsymbol{B} \boldsymbol{A})^k$

2. (5 points) Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$ be given, and $\text{col}(\boldsymbol{A})$ be the column space of $\boldsymbol{A}$. For a given value of m, under what conditions on $\boldsymbol{b}$, $\text{col}(\boldsymbol{A})$, and $\text{rank}(\boldsymbol{A})$ will the equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ have

   (a) no solution?

   (b) exactly one solution?

   (c) infinitely many solutions?

3. (5 points) For a given matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, show that the equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ and $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ share the same set of solutions (i.e., show that $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \Rightarrow \boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ and $\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0} \Rightarrow \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$)

4 Matrix Calculus *[15 points]*

Find the first derivative of the following functions with respect to $\boldsymbol{X}$. Before you attempt the questions below, you are encouraged to review Matrix calculus Wikipedia page(https://en.wikipedia.org/wiki/Matrix_calculus).

(a) $f(\boldsymbol{X}) = tr(\boldsymbol{X}\boldsymbol{X}^T)$, where $\boldsymbol{X} \in \mathbb{R}^{n \times n}$ and tr is the trace of a square matrix.

(b) $f(\boldsymbol{X}) = \boldsymbol{a}^T \boldsymbol{X} \boldsymbol{b}$, where $\boldsymbol{X} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{a} \in \mathbb{R}^m$ and $\boldsymbol{b} \in \mathbb{R}^n$.

(c) $f(\boldsymbol{X}) = \boldsymbol{a}^T \boldsymbol{X}^T \boldsymbol{X} \boldsymbol{b}$, where $\boldsymbol{X} \in \mathbb{R}^{n \times m}$, $\boldsymbol{a} \in \mathbb{R}^m$ and $\boldsymbol{b} \in \mathbb{R}^m$.

5 MLE-MAP *[30 points]*

Suppose that $\mathbf{x} \in \mathbb{R}^d$ is fixed and given. Moreover, assume that $\boldsymbol{\beta} \in \mathbb{R}^d$ is a parameter vector and

$$y = \langle \mathbf{x}, \boldsymbol{\beta} \rangle + \epsilon, \text{ where } \epsilon \sim \text{Laplace}(0, b), \tag{4}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner (i.e., dot) product of two vectors, that is, if $\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1 & \boldsymbol{\beta}_2 & \dots & \boldsymbol{\beta}_d \end{bmatrix}^T$ and $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_d \end{bmatrix}^T$, then $\langle \mathbf{x}, \boldsymbol{\beta} \rangle = \sum_{i=1}^d \boldsymbol{x}_i \boldsymbol{\beta}_i$. Laplace$(0, b)$ denotes the Laplace distribution with mean 0 and parameter b, where the probability distribution function of the Laplace distribution is defined as $f(\epsilon|0, b) = \frac{1}{2b} \exp\left(\frac{-|\epsilon|}{b}\right)$. Therefore, y is a linear function of $\mathbf{x}$ with i.i.d. zero-mean Laplacian noise.

1. (15 points) **Maximum likelihood estimation:**

(a) Write down the probability density function (PDF) of the conditional distribution $f_{y|\boldsymbol{\beta}}$. Your answer can be in terms of the fixed $\boldsymbol{x} \in \mathbb{R}^d$.

(b) Assume that we (independently) draw N pairs $(\mathbf{x}_n, y_n) \in \mathbb{R}^d \times \mathbb{R}$ from the above model, where $\mathbf{x}_n$'s are fixed and then y_n is defined according to (4):

$$y_n = \langle \mathbf{x}_n, \boldsymbol{\beta} \rangle + \epsilon_n, \text{ where } \epsilon_n \sim \text{Laplace}(0, b)\,. \tag{5}$$

What is the PDF of $f_{(y_1,\ldots,y_N|\boldsymbol{\beta})}$?

(c) Write down the associated log-likelihood function for the PDF you found in part (b).

2. (15 points) **Maximum-a-posteriori estimation:** Let $\boldsymbol{\beta} = \begin{bmatrix}\boldsymbol{\beta}_1 & \boldsymbol{\beta}_2 & \ldots & \boldsymbol{\beta}_d\end{bmatrix}^T$, where each $\boldsymbol{\beta}_i \sim \text{Laplace}(0, \sigma)$ for $i = 1, 2, \ldots, d$ and the $\boldsymbol{\beta}_i$ are independent of each other and of ϵ.

(a) After drawing N pairs as above, from Bayes' rule we know the distribution of $\boldsymbol{\beta}|(y_1, \ldots, y_N)$ is

$$P(\boldsymbol{\beta}|y_1, \ldots, y_N) = \frac{P(y_1, \ldots, y_N, \boldsymbol{\beta})}{P(y_1, \ldots, y_N)}.$$

Find the distribution $P(y_1, \ldots, y_N, \boldsymbol{\beta})$.

(b) Note that from the Bayes rule, finding the MAP estimator of $\boldsymbol{\beta}$ is equivalent to maximizing the numerator $P(y_1, \ldots, y_N, \boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$ since the denominator does not depend on $\boldsymbol{\beta}$. Use the expression from part (a) above to formulate a minimization problem whose solution will give the MAP estimator $\hat{\boldsymbol{\beta}}$.

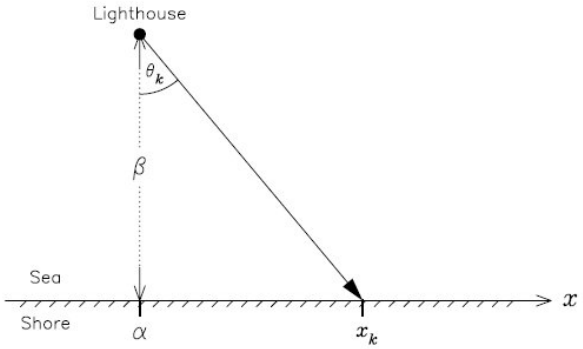
6 Python *[20 points]*

Follow the instructions below and use the included Python code along with your own code to solve the Lighthouse problem.

When uploading to Gradescope, you will need to produce a PDF version of your solutions and code. One way to do this is to use a notebook (`https://jupyter.org`); if you wish to use this, we have provided a Jupyter version of the problem where you can fill in your solutions in hw1.ipynb, which can be downloaded from the Piazza resources page.

6.1 Problem: the lighthouse

(from D. Sivia's book, "Data Analysis - A Bayesian Tutorial"):



A lighthouse is somewhere off a piece of straight coastline at a position α along the shore and a distance β out at sea. It emits a series of short highly collimated flashes at random intervals and hence at random azimuths. These pulses are intercepted on the coast by photo-detectors that record only the fact that a flash has occurred, but not the angle from which it came. N flashes have been recorded so far at positions $\{x_k\}$.

Suppose β is given. Where is the lighthouse?

6.2 Guided solution

We need to estimate the parameter α. Let us start by writing the likelihood for this problem; since the flashes are thrown at random azimuths, we know that:

$$P(\theta_k|\alpha,\beta) = \frac{1}{\pi}.$$

Moreover,

$$\beta\tan(\theta_k) = x_k - \alpha,$$

and by changing variables we get

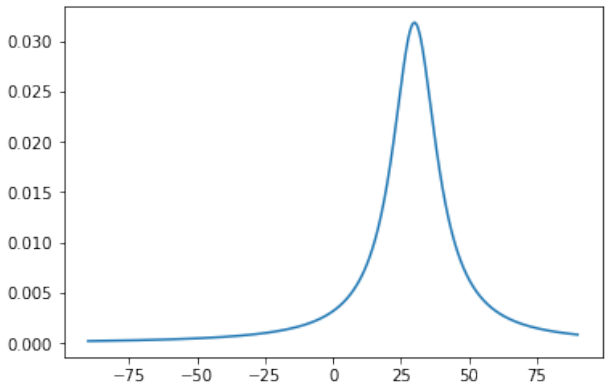
$$P(x_k|\alpha,\beta) = \frac{\beta}{\pi\left[\beta^2 + (x_k-\alpha)^2\right]}.$$

In [2]:
```
# Scientific computing and plotting packages
import numpy as np
import matplotlib.pyplot as plt

# Likelihood definition
def likelihood(x, alpha, beta):
    return beta / (np.pi * (beta ** 2 + (x - alpha) ** 2))

# Parameters
alpha = 30.0 # alpha appears here, only for simmulations purposes, we want to
↪  find the value of this parameter
beta  = 10.0 # beta is given

#Compute and plot the likelihood
x = np.linspace(-90, 90, 1001)
plt.plot(x, likelihood(x, alpha, beta))
plt.show()
```



The above likelihood is a Cauchy or Lorentz distribution. We will sample from it so that we can have some synthetic data to work with.

6.3 Generate synthetic data

```
In [3]: from scipy.stats import cauchy
        samples = cauchy.rvs(loc = alpha, scale = beta, size = 1000)
```

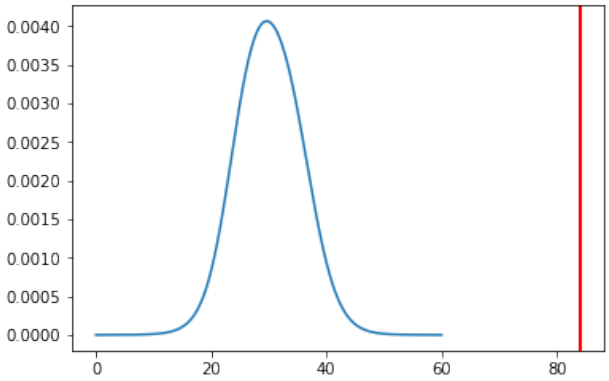
Assuming our prior $P(\alpha)$ is a uniform distribution, the posterior probability is

$$P(\alpha|\{x_k\},\beta) \propto \prod_{k=1}^{N} P(x_k|\alpha,\beta)P(\alpha|\beta) \propto \prod_{k=1}^{N} P(x_k|\alpha,\beta)$$

```
In [4]: # Computes the (unnormalized) posterior for a given set of samples
        def posterior(x, alpha, beta):
            post = np.ones(len(alpha))
            for x_k in x:
                post *= likelihood(x_k, alpha, beta)
                post /= np.sum(post)
            return post

        def plot_posterior(n_samples):
            alphas = np.linspace(0, 60, 1001)
            plt.plot(alphas, posterior(samples[:n_samples], alphas, beta))
            plt.axvline(np.mean(samples[:n_samples]), c = "r", lw = 2)

        plot_posterior(10)
        plt.show()
```



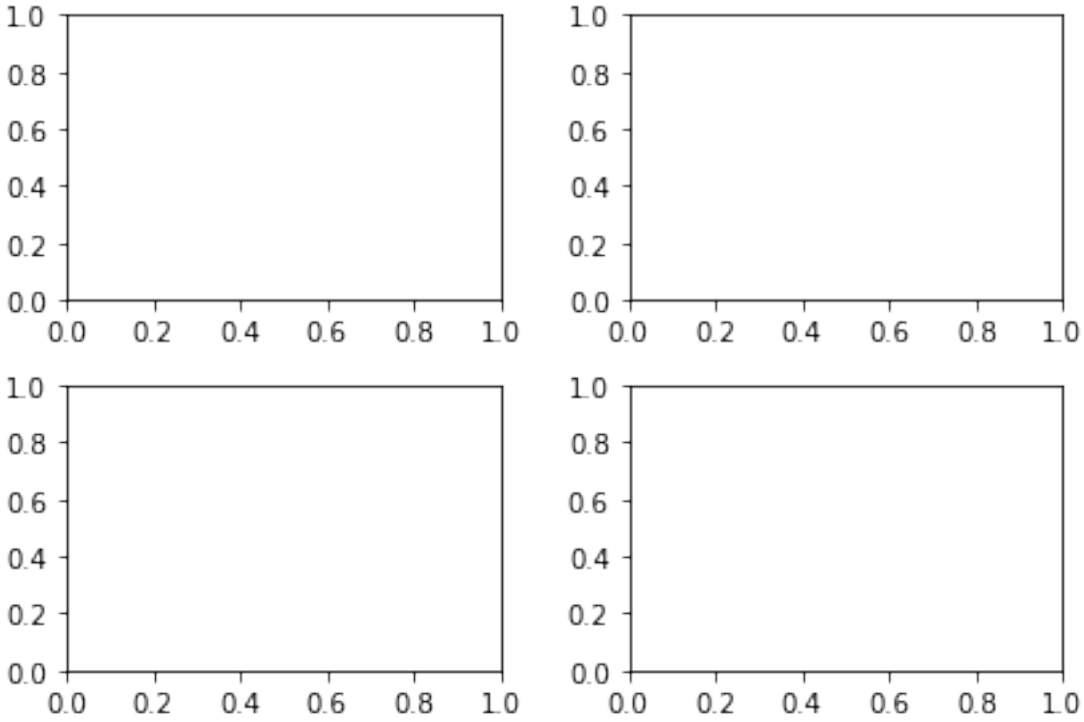
Exercise 1: Create 4 subplots of the posterior distribution for different values of $N = 2, 5, 20, 100$, and include in each one a red line representing the sample mean.

```
In [5]: fig, axs = plt.subplots(2, 2)
        fig.tight_layout()
```

```
alphas = np.linspace(0, 60, 1001)

# Your solution goes here

plt.show()
```



Note the mean does not coincide with the mode of the posterior!

Why is that? Will they coincide in the $N \to \infty$ limit?

Now compute the value of α that maximizes the posterior (and the likelihood, since our prior here is uniform). The log-likelihood reads:

$$\mathcal{L}(\alpha) = \sum_k \log P(x_k|\alpha, \beta) = -\sum_k \log[\beta^2 + (x_k - \alpha)^2)] + \text{c},$$

where c is a constant.

Hence the maximum is obtained at

$$2\sum_k \frac{x_k - \alpha^*}{\beta^2 + (x_k - \alpha^*)^2} = 0.$$

Now let's solve this numerically for different values of N.

Exercise 2: Plot the ML estimate of α for N between 10 and 1000.

```
In [6]: # Use a off the shelf method to find a root of a function on an interval
        # - ex: bisect, brentq, brenth, ridder
```

```python
from scipy.optimize import bisect # Bisection method is probably the simpler to
↪  understand

# Your solution goes here
```