1 Warm-up [20 points]

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

b. (5 points) **Mean and Variance:** If the variance of a zero-mean random variable $X$ is $\sigma^2$, what is the variance of $2X$? What about the variance of $X + 2$?

c. (5 points) **Probability:** Consider the following joint distribution between $X$ and $Y$.

<table>
<thead>
<tr>
<th>$P(X, Y)$</th>
<th>$Y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X$</td>
<td>$a$</td>
</tr>
<tr>
<td>$T$</td>
<td>0.2</td>
</tr>
<tr>
<td>$F$</td>
<td>0.05</td>
</tr>
</tbody>
</table>

What is $P(X = T | Y = b)$?

d. (5 points) Show that the function $f(x) = |x| + \exp(x)$ over the domain $x \in \mathbb{R}$ is convex.

2 Linear algebra [15 points]

a. (3 points) The covariance matrix $\Sigma$ of a random column vector $X$ is defined as $\Sigma = \mathbb{E}[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top]$, where $\mathbb{E}X$ is the expectation of $X$. Is $\Sigma$ positive-semidefinite? Why? Recall a matrix is positive semi-definite if for any vector $x$, $x^\top Ax \geq 0$.

b. (6 points) Let $A, B \in \mathbb{R}^{n \times n}$ be two symmetric matrices. Suppose $A$ and $B$ have the exact same set of eigenvectors $u_1, u_2, \ldots, u_n$ with the corresponding eigenvalues $\alpha_1, \alpha_2, \ldots, \alpha_n$ for $A$, and $\beta_1, \beta_2, \ldots, \beta_n$ for $B$. Write down the eigenvectors and their corresponding eigenvalues for the following matrices:

(i) $C = A + B$

(ii) $D = A - B$
(iii) \( E = AB \)
(iv) \( F = A^{-1}B \) (assume \( A \) is invertible)

c. (6 points) Let \( A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m \) be given. For a given value of \( m \), under what conditions on \( A \) and \( b \) will the equation \( Ax = b \) have

(i) no solution?
(ii) one solution?
(iii) infinitely many solutions?

3 Vector-valued functions [15 points]

Compute the first and second derivatives of the following functions:

a. \( f(x) = c^\top x \), where \( x, c \in \mathbb{R}^m \).

b. \( f(x) = M^\top Mx \), where \( x \in \mathbb{R}^m \) and \( M \in \mathbb{R}^{m \times m} \). What happens if \( M = M^\top \)?

4 Entropy [10 points]

Let \( X \) be a random variable with the Bernoulli distribution \( B(1, p) \) where \( 0 < p < 1 \). The entropy of \( X \) is defined as \( H(p) = -p \log p - (1 - p) \log(1 - p) \).

a. Derive the second derivative \( H''(p) \) of \( H(p) \). If \( H''(p) < 0 \), \( H(p) \) is called concave. Is \( H(p) \) a concave function of \( p \)?

b. Find the value of \( p \in (0, 1) \) that maximizes \( H(p) \).

5 MLE [25 points]

Suppose that \( x \in \mathbb{R}^d \) is a random variable with probability distribution \( D \), \( i.e., x \sim D \). Moreover, assume that \( w \in \mathbb{R}^d \) is a parameter vector and let

\[
y = \langle x, w \rangle + \epsilon, \text{ where } \epsilon \sim N(0, \sigma^2) .
\]

Therefore, \( y \) is a linear function of \( x \) with Gaussian noise added.

a. (5 points) Suppose we take \( x \) and \( w \) to be fixed and given. We then see that \( y|x, w \) is a random variable: it is the sum of the fixed, given constant \( \langle x, w \rangle \) and the Gaussian random variable \( \epsilon \). What is the conditional distribution of \( y|x, w \)?

b. (6 points) Assume that we (independently) draw \( n \) pairs \((x_i, y_i) \in \mathbb{R}^d \times \mathbb{R} \) from the above model, \( i.e. \) we draw \( x_i \) from \( D \) and then, given \( x_i \), we draw a value for \( y_i \) according to (1). What is the distribution of \((y_1, \ldots, y_n)|\{(x_1, \ldots, x_n), w\}\)?

c. (3 points) Write down the log-likelihood function of \((y_1, \ldots, y_n)|(x_1, \ldots, x_n), w\), using your results from part (b) above.
d. (6 points) We now wish to solve the MLE problem, i.e., to find the parameter $w$ that maximizes the log-likelihood function you derived in part (c). Suppose that $d = 1$, i.e., that $x \in \mathbb{R}$ and $w \in \mathbb{R}$; we can then write $\langle x, w \rangle$ as the product $xw$. Derive the optimal parameter $w$ that maximizes the log-likelihood.

*Hint:* The log-likelihood is a concave function of $w$.

e. (5 points) Now suppose that $d > 1$. Generalize your result from part (d) to derive the parameter $w$ that maximizes the log-likelihood of the conditional model from part (c).

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 https://www.andrew.cmu.edu/courses/18-661/.

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 $\alpha$ along the shore and a distance $\beta$ out at sea. It emits a series of short highly collimated flashes. The flashes are sent out at random azimuths, where the azimuths are chosen uniformly at random between $[0, \pi]$. 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 $\beta$ is given. Where is the lighthouse?
6.2 Guided solution

We need to estimate the parameter $\alpha$. 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

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

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
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 the 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

```python
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 for different values of $N = 2, 5, 20, 100$.

```
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 $\alpha$ that maximizes the posterior (and the likelihood, since our prior here is uniform). The log-likelihood reads:

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

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 $\alpha$ 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, brent
from scipy.optimize import bisect # Bisection method is probably the simpler to understand

# Your solution goes here