Note: All coding problems are to be implemented in a single jupyter notebook.
Note: this is the distribution of questions:

(a) Question 1 to Question 8: Required for everyone.

(b) Question 9 - Question 10: Required only for Graduate Students

(c) Question 11 - Question 12: Bonus marks

Problem 1 (5 points)

What is the difference between Root Mean Squared Error (RMSE) and Mean Squared Error(MSE). Describe the context in which they will be most useful? Which of these losses penalizes large differences between the predicted and expected results and why?

Problem 2 (5 points)

Which of the following statements is true about the relationship between features and dataset sizes? Choose one of the following and explain the chosen answer

(a) When training a model, as you add more features to the dataset, you often need to increase the dataset's size to ensure the model learns reliably.

(b) When training a model, adding more features to the dataset increases the amount of information you can extract from the dataset. This allows you to use smaller datasets to train the model and still achieve good generalization performances from the training data.

(c) When training a learning algorithm, as you decrease the number of features in the dataset, you need to increase the training sample size to make up the difference.

(d) When training a learning algorithm, the number of features in your dataset is entirely dependent on the amount of information you can extract from the dataset.

Problem 3 (4 points)

You are building a deep learning model and after X epochs you see that the training loss is decreasing, while your validation loss is either constant or increasing. What would be the most likely cause of this and suggestion to rectify it

(a) Learning rate is too high

(b) Gradient descent does not converge

(c) Insufficient training data size

(d) Gradient descent is stuck in a local minimum

Problem 4 (4 points)

Perceptron

(a) is a linear classifier. A. True B. False

(b) and cannot be trained on linearly unseperable data. A. True B. False

Problem 5 (4 points)

Logistic Regression

(a) is a linear classifier. A. True B. False

(b) and always has a unique solution A. True B. False

Problem 6 (10 points)

Which of the following is true, given the optimal learning rate?

(a) Batch gradient descent is always guaranteed to converge to the global optimum of a loss function.

(b) Stochastic gradient descent is always guaranteed to converge to the global optimum of a loss function.

(c) For convex loss functions (i.e. with a bowl shape[1]), batch gradient descent is guaranteed to eventually converge to the global optimum while stochastic gradient descent is not.

(d) For convex loss functions (i.e. with a bowl shape), stochastic gradient descent is guaranteed to eventually converge to the global optimum while batch gradient descent is not.

(e) For convex loss functions (i.e. with a bowl shape), both stochastic gradient descent and batch gradient descent will eventually converge to the global optimum.

(f) For convex loss functions (i.e. with a bowl shape), neither stochastic gradient ndescent nor batch gradient descent are guaranteed to converge to the global optimum.

Problem 7 (5 points)

Given the following data on a 2D plane :

x	y
-1	-2
1	-1
2	3

Fit a linear regression model to the data without the constant term: $y_i = \beta x_i + \epsilon_i$. Give an expression of the minimization problem for finding β, Show how to compute it's value and show the value

Problem 8 (20 points)

Perform a polynomial fitting to compute a design matrix X such that:

$$X_{ij} = y_i^j \tag{1}$$

You should implement this **without a single for loop** by using vectorization and broadcast. Here $(1 \leq j \leq 50)$ and $y = \{-20, -19.9, ..., 20\}$. Implement code that generates such a matrix.

[1]More formally f is convex on $[a, b]$ if $\forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in [a, b]$ if $f(\lambda \boldsymbol{x}_1 + (1 - \lambda)\boldsymbol{x}_2) \leq \lambda f(\boldsymbol{x}_1) + (1 - \lambda) f(\boldsymbol{x}_2)$

Bonus for undergraduates beyond this line

Problem 9 (15 points)

Is this a good idea to initialize parameters with zeros when training a logistic regression model? Explain your answer.

Problem 10 (15 points)

As discussed in the class, the logistic regression likelihood simplifies to the following form:

$$\text{cost}(f(x), y) = -y \log(f(x)) - (1 - y) \log(1 - f(x))$$
$$y \in \{0, 1\} \text{ class label}$$

What will happen when $y = 1$ and $f(x) = 1$? Will this work in actual implementations?

Extra credit problems

Problem 11 (extra credit 20 points)

In the three examples of Figure , find the perceptron weights w_0, w_1, w_2 for each of them. The line that separates the two classes is given by the decision boundary that divied into positive and negative classes. (No calculations required, you can simply note the answer.)

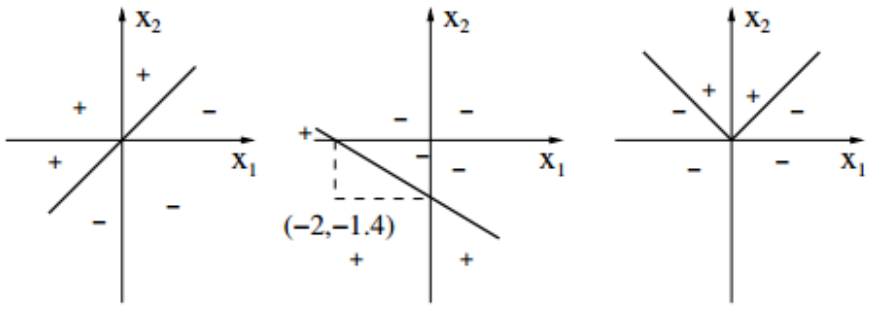

Figure 1: Binary class data and separating decision boundaries.

Problem 12 (extra credit 20 points)

Consider the evaluation of E as given below:

$$E = g(x, y, z) = \sigma(c(ax + by) + dz) \tag{2}$$

Here x, y, z are inputs and a, b, c, d are parameters. σ is logistic sigmoid function defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3}$$

Note that E is the function of a, b, c, d Compute the partial derivative of E with respect to parameters a, b, d i.e $\frac{\partial E}{\partial a}, \frac{\partial E}{\partial b}, \frac{\partial E}{\partial d}$