CS 188 HW 3

Due on Friday, March 6 at 11:59PM

1 Instructions:

You may form small groups (e.g. of up to four people) to work on this assignment, but you must write up all solutions by yourself. List your study partners for the homework on the first page, or "none" if you had no partners.

Keep all responses brief, a few sentences at most. Show all work for full credit.

Start each problem on a new page, and be sure to clearly label where each problem and subproblem begins. All problems must be submitted in order (all of P1 before P2, etc.).

No late homeworks will be accepted. This is not out of a desire to be harsh, but rather out of fairness to all students in this large course.

2 Overfitting

Overfitting is a common problem when doing datascience work.

(a) How can you tell if a model you have trained is overfitting?

(b) Why would it be bad to deploy a model that's been overfitted to your training data?

(c) Briefly describe how overfitting can occur.

(d) In lecture and discussion we've discussed multiple methods for dealing with overfitting. One such technique was regularization. Please describe two different regularization techniques and explain how they help to mitigate overfitting.

3 Overfitting Mitigations

For each of the below strategies, state whether or not it might help to mitigate overfitting and explain why.

(a) Using a smaller training dataset

(b) Restricting the maximum value any parameter can take on

(c) Training your neural network for longer (more iterations)

(d) Training a model with more parameters

(e) Randomly setting the outputs of 50% of the nodes in your neural network to zero

(f) Incorporating additional sparse features into your model

(g) Initializing your parameters randomly instead of to zero

(h) Training your model on a graphics processing unit or specialized accelerator chip instead of a CPU

4 K-Nearest Neighbors

(a) Why is it important to normalize your data when using the k-nearest neighbors algorithm (KNN)?

(b) When doing k-nearest neighbors, an odd value for k is typically used. Why is this?

(c) Say you have the dataset in Table 1, where x and y are features and L is the label. Normalize the features by scaling them so that all values in a column lie in the range $[0, 1]$, so that they can be used with KNN.

(d) Use KNN with K=3 to make predictions for the data points in Table 2.

(e) Use KNN with $k = 1$ to make predictions for the data points in Table 2.

(f) Use KNN with $k = 5$ to make predictions for the data points in Table 2.

(g) What is a potential issue with using a low k for KNN?

(h) What is a potential issue with selecting a k that is too high when doing KNN?

x	y	L
0.2	350	0
0.1	750	0
0.3	700	0
0.1	500	0
0.2	500	0
1.2	400	1
0.9	410	1
1.1	390	1
0.1	760	1

Table 1: Raw KNN data for training

x	y	L_{pred}
.1	760	
.2	700	
.6	200	

Table 2: Raw KNN data for evaluation

(i) Say you had a dataset and wanted to understand how well KNN with a k of 3 performed on it. How could you quantify its performance? Assume you do not have access to any samples beyond those in your dataset.

(j) Say you had a dataset consisting of 200 samples. How might you select the optimal k to use for KNN?

5 Principal Component Analysis

For each of the below situations, state whether or not PCA would work well, and briefly explain why.

(a) Data with a linear structure

(b) Data lying on a hyperbolic plane

(c) A dataset containing non-normalized features

(d) A dataset where each feature is statistically independent of all others

6 Artificial Neural Networks

Consider the following computation graph for a simple neural network for binary classification. Here x_1 and x_2 are input features and y is their associated class. The network has multiple parameters, including weights w and biases b, as well as non-linearity function g. The network will output a value y_{pred}, representing the probability that a sample belongs to class 1. We use a loss function $Loss$ to help train our model. The network is initialized with the parameters in Table 3.

You will first train the model using some sample datapoints and then evaluate its performance. For any questions that ask for performance metrics, generate them using the samples in Table 4.

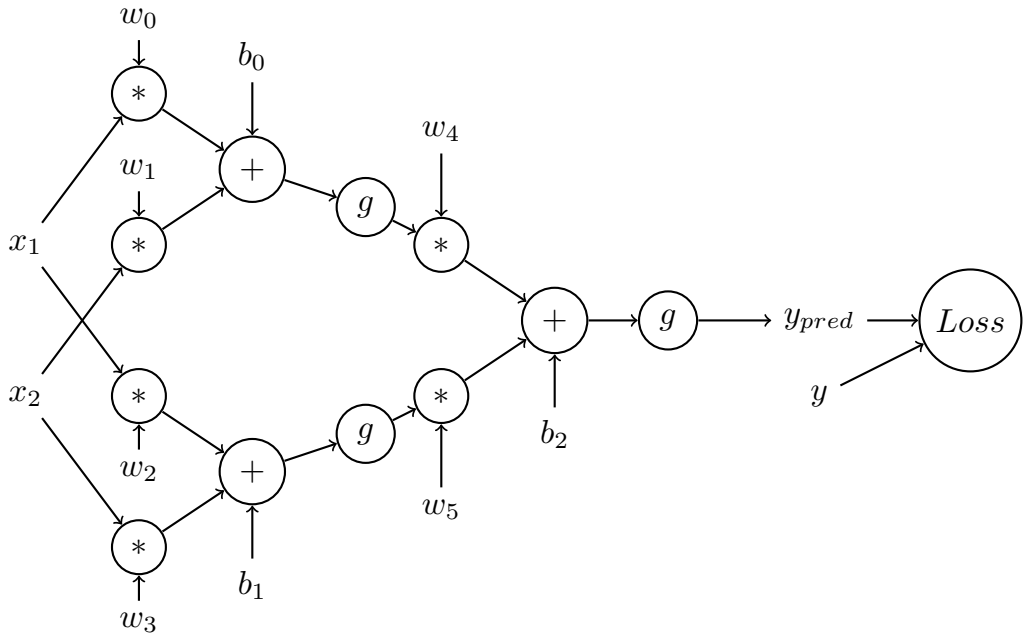

Figure 1: Neural network architecture

(a) Initialize the neural network with the parameters in Table 3 and then train it using the samples in Table 4. Use gradient descent (forward pass followed by backpropagation) to update the parameters.

Suppose the loss function is quadratic, $Loss(y, y_{pred}) = \frac{1}{2}(y - y_{pred})^2$, and g is the sigmoid function $g(z) = \frac{1}{1+e^{-z}}$ (note: it's typically better to use a different type of loss, cross-entropy, for classification problems, but using a quadratic loss function keeps the math easier for this assignment).

Assume that your learning rate $\alpha = .1$.

Pass the samples through the network one at a time in order from top to bottom. Report the final values of all parameters. Show all work.

(b) What is your model's accuracy?

(c) What is your model's precision?

(d) What is your model's recall?

(e) What is your model's F_1 score?

Parameter	Initial value
b_0	1
b_1	-6
b_2	-3.93
w_0	3
w_1	4
w_2	6
w_3	5
w_4	2
w_5	4

Table 3: ANN initial state

x_1	x_2	y
0	0	0
1	1	1
0	1	1

Table 4: ANN training data

(f) Plot the ROC curve.

(g) What is your model's AUC-ROC?

7 Reinforcement Learning

Consider the following instance of reinforcement learning

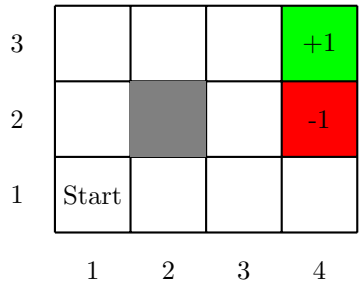

Figure 2: Reinforcement learning grid world

Assume the following:

- The agent lives in a grid
- The agent starts in the bottom left
- Walls (shaded grids) block the agent's path
- The agent's actions do not always go as planned:
  - 80% of the time, the action North takes the agent North (if there is no wall there)
  - 10% of the time, North takes the agent West
  - 10% of the time, North takes the agent East
  - If there is a wall in the direction the agent would have taken, the agent stays put
- $\gamma = .8$
- Big rewards come at the end
- Goal: maximize sum of rewards

Use Q-learning to find an optimal path in the grid. Show each step. State all your assumptions.

8 (Money)ball So Hard

Watch the 2011 movie *Moneyball* (directed by Bennett Miller and written by Steven Zaillian and Aaron Sorkin).

(a) Write a detailed formulation of the problem discussed in the movie and outline, step-by-step, how you would go about solving the problem (data collection, data cleaning, ML, and all other steps).

(b) Discuss what can go wrong when using this method in practice and how it can be improved.