CS 188 Fall 2024 Introduction to Artificial Intelligence Written HW10

Due: Tuesday 11/19 at 11:59pm.

Policy: Can be solved in groups (acknowledge collaborators) but must be submitted individually.

Make sure to show all your work and justify your answers.

Note: This is a typical exam-level question. On the exam, you would be under time pressure, and have to complete this question on your own. We strongly encourage you to first try this on your own to help you understand where you currently stand. Then feel free to have some discussion about the question with other students and/or staff, before independently writing up your solution.

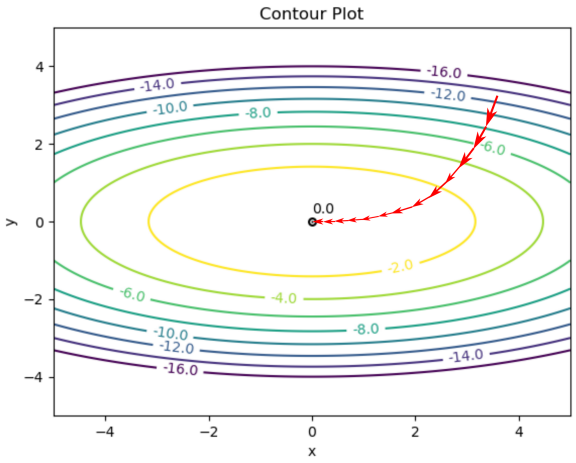
Note: Leave the self-assessment sections blank for the original submission of your homework. After the homework deadline passes, we will release the solutions. At that time, you will review the solutions, self-assess your initial response, and complete the self-assessment sections below. The deadline for the self-assessment is 1 week after the original submission deadline.

Your submission on Gradescope should be a PDF that matches this template. Each page of the PDF should align with the corresponding page of the template (page 1 has name/collaborators, question begins on page 2.). **Do not reorder, split, combine, or add extra pages**. The intention is that you print out the template, write on the page in pen/pencil, and then scan or take pictures of the pages to make your submission. You may also fill out this template digitally (e.g. using a tablet.)

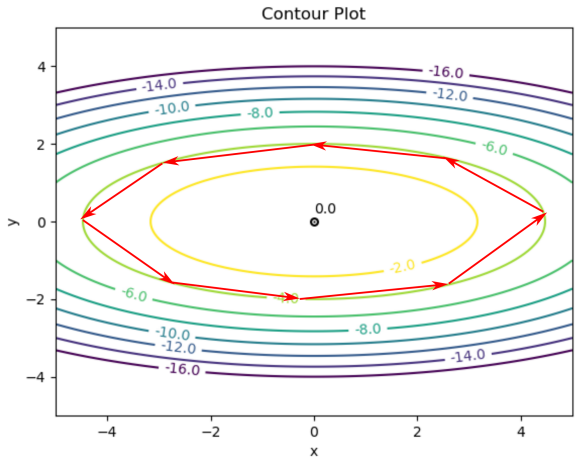
First name	
Last name	
SID	
Collaborators	

Q1. [10 pts] Machine Learning

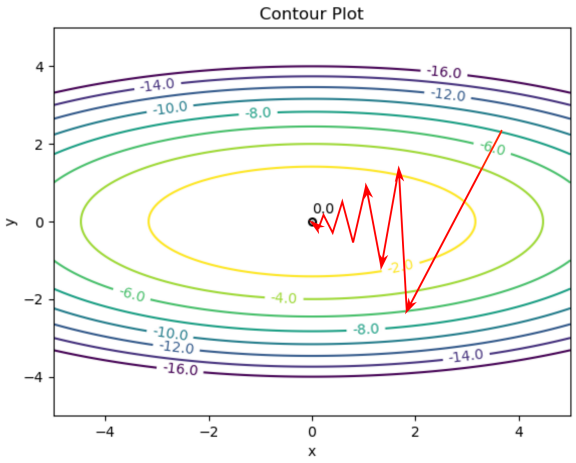
(a) **(i)** [2 pts] The following four figures contain the steps of possible gradient ascent implementations each for some learning rate. Each arrow designates a step of gradient ascent. The numbers in the contour lines denote the value of the function we are maximizing on that contour. Which of the following figures, if any, contain paths that could be generated by gradient ascent?



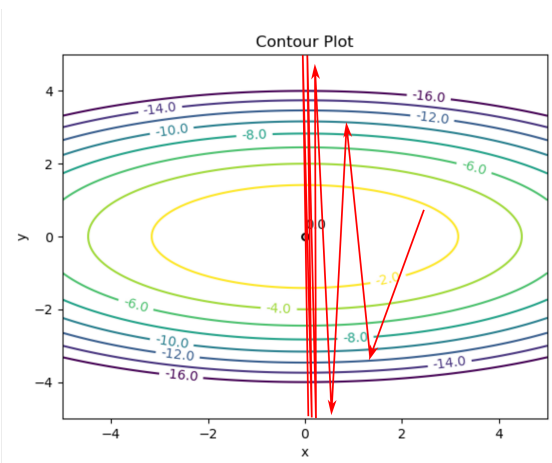
A.



B.



C.



D.

■ A. □ B. ■ C. ■ D.

A and C show gradient ascent steps in which C has a larger learning rate. D also shows gradient ascent steps with a learning rate large enough to make the method diverge. In B on the other hand the gradient directions are not perpendicular to the contour lines as they should be, since gradients point towards the steepest ascent-descent direction.

(ii) [2 pts] Assume that we are trying to maximize a function that has many local maxima using gradient ascent. Which of the following variations of gradient ascent are likely to help obtain a better solution than what would be achieved with basic gradient ascent that uses a fixed step size α?

- [x] Once gradient ascent has converged, store the solution, randomly perturb the solution, and run gradient ascent again until convergence. Repeat this process K times and return the best solution.
- [x] Run N independent gradient ascent methods starting from N different initial solutions. Out of the ones that converge choose the best solution.
- [ ] For the chosen magnitude of the learning rate $|\alpha|$, at each step choose its sign with equal probability.
- [ ] After each iteration, increase the learning rate linearly.

Gradient ascent is not guaranteed to converge to the global minimum when the function has many local maxima. If we store the solution after convergence, perturb the solution and then rerun gradient ascent from the perturbed solution it is possible that the "new" gradient ascent will return a better solution. It is quite likely that at least one of these "new" gradient descents will return a solution better than the first one.
Running N independent gradient ascent methods and choosing the best solution is clearly beneficial as compared to a single gradient ascent implementation.
Randomly choosing the sign of the learning rate would switch between performing gradient descent and gradient ascent updates which is not what we want given that we are maximizing a function.
Finally, increasing the learning rate at each iteration is obviously not likely to help as in that case the method would most likely diverge after some point.

(b) **(i)** [1 pt] Assume that we observe some data features $x_1^{(i)} \in \mathbb{R},\ i = 1, \ldots, N$ and outputs $y^{(i)} \in \mathbb{R},\ i = 1, \ldots, N$. We will use a linear model of the form $\hat{y}^{(i)} = c + bx_1^{(i)}$ to predict the output y. If we use the squared euclidean norm as an objective, i.e. $loss = \frac{1}{2}\sum_{i=1}^{N}\left(\hat{y}^{(i)} - y^{(i)}\right)^2$ and a learning rate α, which of the following is the correct gradient descent rule for finding the optimal value of b?

- ○ $b \leftarrow b + \alpha \sum_{i=1}^{N}(c + bx_1^{(i)} - y^{(i)})x_1^{(i)}$
- ● $b \leftarrow b - \alpha \sum_{i=1}^{N}(c + bx_1^{(i)} - y^{(i)})x_1^{(i)}$
- ○ $b \leftarrow b + \alpha \sum_{i=1}^{N}(c + bx_1^{(i)} - y^{(i)})$
- ○ $b \leftarrow b - \alpha \sum_{i=1}^{N} y^{(i)}x_1^{(i)}$

The gradient of the objective function with respect to b is $\frac{\partial loss}{\partial b} = \sum_{i=1}^{N}(a + bx_1^{(i)} - y^{(i)})\frac{\partial a + bx_1^{(i)}}{\partial b} = \sum_{i=1}^{N}(a + bx_1^{(i)} - y^{(i)})x_1^{(i)}$. Given that we perform gradient descent we need a negative sign for the gradient.

Now we observe the following dataset in which the outputs y are binary. Furthermore, we are given three features x_1, x_2, and x_3. The training dataset is as follows:

y	1	1	0	0
x_1	1	0	1	0
x_2	0	1	1	0
x_3	0	0	0	0

(ii) [1 pt] We decide to use logistic regression for this classification task. Using only these features, what is the shape of the decision boundary we obtain from logistic regression?

- ● Linear
- ○ Nonlinear in general
- ○ Sigmoid
- ○ None of the above

The decision boundary we obtain from logistic regression is always linear. This is a bit of a trick question, because with the given data points logistic regression would not actually be able to learn a linear decision boundary which correctly classifies all training points.

(iii) [2 pts] We decide to train a neural network to perform classification on the whole dataset shown above. We propose using three hidden layers each of dimension 10 with a ReLu activation function except for the output layer which uses a sigmoid activation function. We will test the classifier on new unseen test data. Which of the following scenarios are likely to occur?

- [x] The network will achieve zero classification error on the training set.
- [ ] The network will underfit the data.
- [x] The network will overfit the data.
- [ ] The network will achieve zero classification error on the test set.

The network has many parameters and our training dataset is very small. This would lead to overfitting the data and achieving most likely zero training error while it will most likely perform poorly on the test set.

(iv) [2 pts] This is a more general question independent of the aforementioned datasets. Assume that we have trained a logistic regression classifier on the last dataset obtaining the weight vector $\mathbf{w} = [w_1, w_2, w_3]$ and we observe a new feature vector $\mathbf{x}' = [x'_1, x'_2, x'_3]$. Let $P(y = 1|\mathbf{x}; \mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}}$, with $\cdot$ denoting the inner product. Which of the following formulas describe the correct classification rule for the label y' of this data point?

- [x] $y' = \underset{y' \in \{-1,1\}}{\text{argmax}}\, P(y'|\mathbf{x}'; \mathbf{w})$
- [ ] $y' = \begin{cases} 1, & \text{if } P(y' = 1|\mathbf{x}'; \mathbf{w}) \geq 0 \\ -1, & \text{otherwise} \end{cases}$
- [x] $y' = \begin{cases} 1, & \text{if } P(y' = 1|\mathbf{x}'; \mathbf{w}) \geq 0.5 \\ -1, & \text{otherwise} \end{cases}$
- [ ] $y' = \begin{cases} 1, & \text{if } P(y' = 1|\mathbf{x}'; \mathbf{w})(1 - P(y' = 1|\mathbf{x}'; \mathbf{w})) \geq 0 \\ -1, & \text{otherwise} \end{cases}$

In logistic regression the logistic function $P(y|\mathbf{x}; \mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}}$ is used to model the probability of of $y = 1$. Hence we use 0.5 as a cutoff value and any feature $\mathbf{x}$ with that value being higher than 0.5 is classified as $y' = 1$ and for any value less than 0.5 we classify it as $y = 0$.

Q1 Self-Assessment - leave this section blank for your original submission. We will release the solutions to this problem after the deadline for this assignment has passed. After reviewing the solutions for this problem, assess your initial response by checking one of the following options:

- ◯ I fully solved the problem correctly, including fully correct logic and sufficient work (if applicable).
- ◯ I got part or all of the question incorrect.

If you selected the second option, explain the mistake(s) you made and why your initial reasoning was incorrect (do not re-iterate the solution. Instead, reflect on the errors in your original submission). Approximately 2-3 sentences for *each* incorrect sub-question.