CS4780 Midterm

Fall 2018

NAME:	
Net ID:	
Email:	

I promise to abide by Cornell's Code of Academic Integrity.

Signature: ______________________________

1 [??] General Machine Learning

Please identify if these statements are either True or False. Please justify your answer **if false**. Correct "True" questions yield 1 point. Correct "False" questions yield 2 points, one for the answer and one for the justification.

1. **(T/F)** As $n \to \infty$, the 1-NN error is no more than twice the error of the Bayes Optimal classifier.

2. **(T/F)** MLE can overfit the data if n (the number of trainig samples) is small. It tends to work well when n is large.

3. **(T/F)** Both, Gradient descent and Newton's method use only a 1st order approximation of the function to be minimized.

4. **(T/F)** If a data set is linearly separable, the Perceptron guarantees that you find a hyperplane but the SVM finds the maximum margin separating hyperplane.

5. **(T/F)** The best machine learning algorithm make no assumptions about the data.

6. **(T/F)** The k-NN classifier is not a linear classifier.

7. (**T/F**) The k-NN algorithm can be used for classification, but not regression.

8. (**T/F**) The order of the training points can affect the training time of the Perceptron algorithm.

9. (**T/F**) Even on non-linearly-separable datasets, the Perceptron algorithm is guaranteed to converge in finite time.

10. (**T/F**) In MAP, we find the maximizer of the posterior, so we need to find an expression for the posterior.

11. (**T/F**) If you were to use the "true" Bayesian way of machine learning you would put a prior over the possible models and draw several modelsr randomly during training.

12. (**T/F**) If the features are probabilistically dependent on each other, then the naive Bayes assumption cannot hold.

13. (**T/F**) Logistic regression is a generative model.

14. (**T/F**) The order of the training points can affect the convergence of the gradient descent algorithm.

15. (**T/F**) For gradient descent, higher learning rates guarantee faster convergence times.

16. (**T/F**) For Adagrad, we use the same learning rate for all features.

2 [16] K-NN

1. (2 pts) Imagine you apply the kNN classifier with Euclidean distance. Describe what happens if you scale one dimension of the input features by a large positive constant across all examples?

2. (2 pts) What is the modeling assumption of kNN?

3. (4 pts) Consider points, sampled uniformly at random, within a finite volume in d dimensions. How do the pairwise distances change as the dimensionality d increases? How is the distance to a random hyper-plane affected? (No calculation required - just describe what happens.)

4. (8 pts)In class, we have learned that the k-NN algorithm is distance-based.

   Suppose we are provided with the following 2D dataset:

   - Class +1 (red): $\{(2,6)\}$
   - Class −1 (green): $\{(2,2),(4,4)\}$

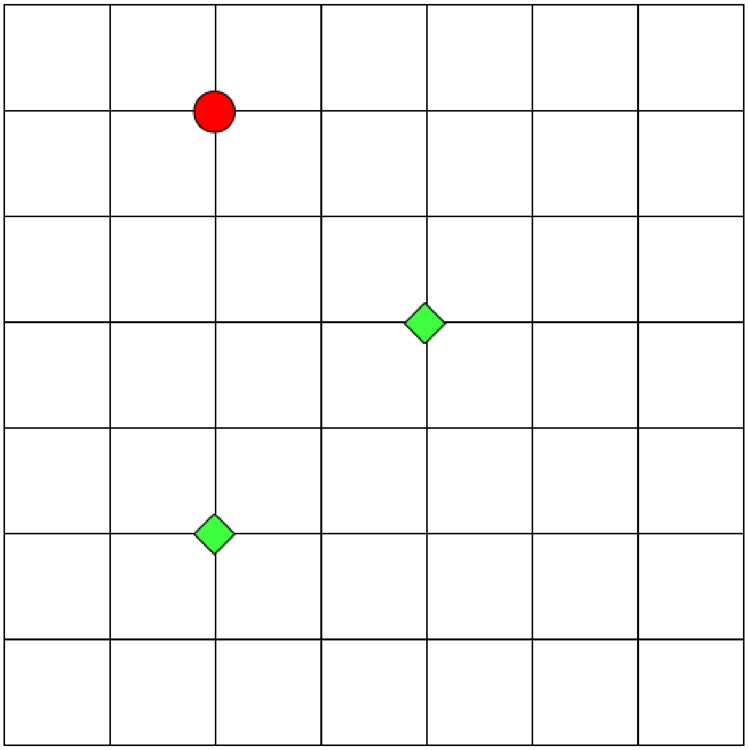
   In this problem, we will study the difference between the l_2 and l_1 distances. For two points $\mathbf{x} = (x_1, x_2)$ and $\mathbf{z} = (z_1, z_2)$, the l_1 distance is defined as

   $$\mathrm{d}_1(\mathbf{x}, \mathbf{z}) = |x_1 - z_1| + |x_2 - z_2|, \tag{1}$$

   and the l_2 distance is defined as

   $$\mathrm{d}_2(\mathbf{x}, \mathbf{z}) = \sqrt{(x_1 - z_1)^2 + (x_2 - z_2)^2}. \tag{2}$$

(a) (4 pts) Draw the decision boundary for the 1-NN classifier with l_2 distance.



(b) (4 pts) Show that when $x_1 > 4$ and $x_2 > 6$, 1-NN classifier can't classify the point $\mathbf{x}^* = (x_1, x_2)$ with l_1 distance. (Hint: show that the closest distances from $\mathbf{x}^*$ to two classes' points are the same.)

3 [17] MLE and MAP

1. (5 pts) Recall the coin example in the class. A natural assumption about a coin toss is that the distribution of the observed outcomes is a binomial distribution. If a coin was tossed $n = n_H + n_T$ times and its probability of coming up heads is θ, the probability that we would observe exactly n_H heads and n_T tails is

$$P(\mathcal{D}|\theta) = \binom{n_H}{n_H + n_T}\theta^{n_H} \cdot (1-\theta)^{n_T}.$$

In this model, $\hat{\theta}_{MLE} = \frac{n_H}{n_H+n_T}$. Furthermore, if θ has a prior distribution

$$P(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)},$$

where $B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ is the normalization constant, $\hat{\theta}_{MAP} = \frac{n_H+\alpha-1}{n_H+n_T+\beta+\alpha-2}$. Please answer following questions:

(a) (1 pts)What will happen if the prior of θ is very wrong and sample number n is very small?

(b) (2 pts)Explain why as $n \to \infty$, $\hat{\theta}_{MAP} \to \hat{\theta}_{MLE}$. (No formal proof required.)

(c) (2 pts)What will happen if the prior of θ is very wrong but sample number n is very large?

2. (12 pts) Let $x_1, \cdots, x_n$ be iid random data sampled by the poisson distribution

$$P(x|\theta) = e^{-\theta}\frac{\theta^x}{x!}(x \in \mathbb{N} \cup \{0\}). \tag{3}$$

Here $\theta > 0$ is a hyperparameter of this distribution.

(a) (3 pts) Write the log likelihood function $\log P(x_1, x_2, \cdots, x_n|\theta)$.

(b) (4 pts) Show that the maximum likelihood estimation $\hat{\theta}_{MLE} = \frac{\sum_{i=1}^n x_i}{n}$.

(c) (5 pts) If we have a prior distribution for θ

$$P(\theta) = e^{-\theta}(\theta > 0) \tag{4}$$

Compute the $\hat{\theta}_{MAP} = \arg\max_{\theta} \log P(\theta|x_1, x_2, \cdots, x_n)$. Will $\hat{\theta}_{MAP} <= \hat{\theta}_{MLE}$ always hold?

4 [18] Naive Bayes

1. (2 pts) Write the assumption of Naive Bayes about data.

2. (16 pts)Ronnie is playing a game named Flippin' Extravaganza, he asks if someone can help him win. Can you help him beat the house? This game is easy to undertake, there is a red hat and a blue hat with a weighted penny respectively. The operator secretly flips one of them and asks you to guess under which hat it is. we made the assumption that you somehow know the coins' weights. In fact, in the Coin Flippin' Extravaganza, the operator never reveals the weights. Instead, you've spent the whole day watching people play the game, recording the results below:

game	penny	nickel	dime	hat
1	T	H	T	Red
2	T	T	H	Blue
3	T	H	T	Blue
4	H	H	H	Red
5	H	H	T	Red
6	T	T	H	Blue
7	H	H	T	Red
8	T	T	H	Blue
9	T	H	H	Blue
10	H	H	H	Red
11	T	T	H	Blue
12	T	H	H	Red
13	H	H	T	Red
14	T	T	H	Blue
15	T	H	H	Blue
16	T	T	H	Blue
17	H	T	H	Red
18	H	T	H	Blue

(a) (4 pts) This model could be formalized by a Naive Bayes with Bernoulli distributed features. To see this, define the feature space $\mathcal{X}$ and the label space $\mathcal{Y}$. Is the Naive Bayes assumption valid for this problem? Why?

(b) (2 pts) Compute $P(hat = Red)$ and $P(hat = Blue)$

(c) (6 pts) Estimate the following probabilities with +1 smoothing

hat	$P(penny = H\|hat)$	$P(nickel = T\|hat)$	$P(dime = T\|hat)$
Red			
Blue			

(d) (4 pts)if the coins come up $[H, T, T]$, compute the probability that they came from the blue hat, i.e. $P(H, T, T|hat = Blue)$ by Bayes Rule. Please write the computation formula.

5 [14] Gradient Descent

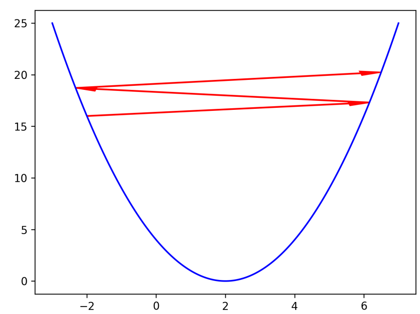
You wish to use gradient descent to minimize

$$l(w) = (w-2)^2 \tag{5}$$

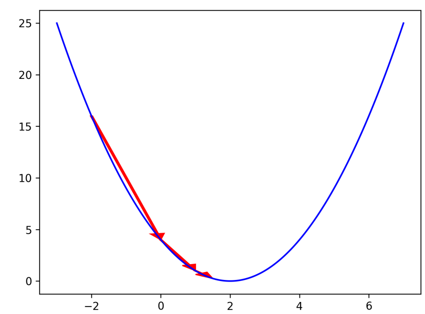
with learning rate $\alpha > 0$. Solving these following problems will help you choose a appropriate α for this particular loss function.

1. (3 pts) Suppose at time t, we have w_t. Write the $l'(w)$ and update formula for w_{t+1} using Gradient Descent.

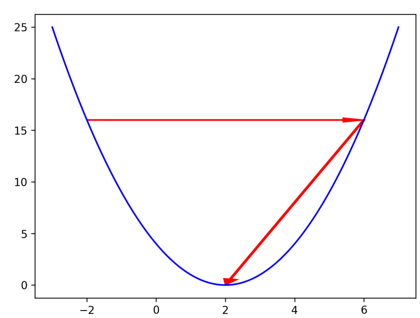
2. (4 pts) Starting with $w_0 = -2$, we use gradient descent to update w. The following three figures are the first 3 updates for different learning rate $\alpha = 0.25, 0.75,\ 1.02$ and $\frac{1}{t+1}$ (when we update w_t to w_{t+1}). Please match different figures to their corresponding learning rates.



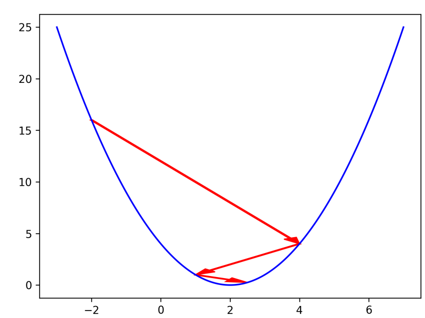
(a)



(b)



(c)



(d)

3. (4 pts) Show that $\forall t, l(w_{t+1}) < l(w_t)$ if and only if $0 < \alpha < 1$.

4. (3 pts) Show that Newton's method will help $l(w)$ converge to minimum in one step update with arbitrary w_0.

6 [13] Linear Classifiers

1. (10 pts) Consider the following 2D dataset $\mathcal{D}$:

   - Class $+1$: $\{(1,3),(3,3)\}$
   - Class -1: $\{(5,1)\}$

   (a) (4 pts)To find a $\mathbf{w}$ and b s.t. $\forall(\mathbf{x},y)\in\mathcal{D}, y(\mathbf{w}^\top\mathbf{x}+b)>0$, please first transform $\mathcal{D}$ to $\mathcal{D}'=\{([\mathbf{x},1],y)|(\mathbf{x},y)\in\mathcal{D}\}$ and consider a new $\mathbf{w}'=[\mathbf{w},b]$. Next, apply perception algorithm in this new dataset. Write down the sequence of each updates in the perception algorithm $[\mathbf{w}'_0,\mathbf{w}'_1,\cdots]$ starting with $\mathbf{w}'_0=(0,0,0)$. Notice that different sequence of points will lead to different sequence of updates, please consider the points in this fixed order $[(3,3),(1,3),(5,0)]$ in each iteration of perception.

   (b) (2 pts)If one more point $(2,3)$ labeled -1 is added to $\mathcal{D}$, what will happen to the perception algorithm?

   (c) (4 pts) In the lecture we learned that SVM will help us find the maximum margin separating hyperplane. Let's guess what this hyperplane is without the proof of it's "maximum". First, let's delete the point $(1,3)$ from our dataset. What is the maximum margin separating hyperplane for the remaining two points dataset? After adding the point $(1,3)$ back to the dataset, will the maximum margin separating hyperplane change? If we have one more point $(3,2)$ labeled $+1$ to the dataset, will the maximum margin separating hyperplane change?

2. (2 pts) Write two commonly used binary classification loss functions $l(h_{\mathbf{w}}(\mathbf{x}_i,y_i))$ (For example zero-one loss or exponential loss).

3. (1 pts) Write one commonly used regression loss function $l(h_w(\mathbf{x}_i, y_i))$ (For example squared loss or absolute loss).

True/False	
kNN	
MLE and MAP	
NB	
Gradient Descent	
Linear Classifiers	
TOTAL	