CS4780 Midterm

October 2015

NAME:	
Net ID:	
Email:	

GML	
Short	
kNN	
NB	
Perceptron + LR	
SVM	
Linear Regression	
TOTAL	

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 two points, one for the answer and one for the justification.

T/F Naïve Bayes makes the assumption that the individual features are independent, i.e. $p(\mathbf{x}) = \prod_\alpha p([\mathbf{x}]_\alpha)$.
False, it assumes that they are conditionally independent, i.e. $P(\mathbf{x}|y) = \prod_\alpha p([\mathbf{x}]_\alpha|y)$.

T/F A linear regressor is parameterized by parameters $\mathbf{w}$ and b. The parameter b is the *bias*, which, together with *variance* and *noise* combine to the test error.
False, this bias term is very different from the error bias.

T/F If a classifier obtains 0% training error it cannot have 100% testing error.
False, a simple counter example is a classifier that memorizes the training data set.

T/F Any reasonable machine learning algorithm must make assumptions about the data.
True.

T/F If data is drawn uniformly at random within a hypercube, increasing the dimensionality makes the data points more equidistant (i.e. they all have roughly the

same distance from each other).
True.

T/F Although MLE and MAP are different approaches to set model parameters θ, both of them do consider θ to be random variables.
False, only in MAP is θ a random variable.

T/F If a data set is linearly separable, the Perceptron is guaranteed to converge in a finite number of updates. Otherwise, it sometimes converges, but there are no guarantees.
False. If the data is not linearly separable, the Perceptron will not converge.

T/F Assume there exists a vector $\mathbf{w}^*$ that defines a hyperplane that perfectly separates your data. Let the Perceptron vector be $\mathbf{w}$. As the algorithm proceeds, the two vectors must converge ($\mathbf{w} \to \mathbf{w}^*$) and in the limit (after possibly infinitely many updates) we have $\mathbf{w} = \mathbf{w}^*$.
False, the perceptron converges to a separating hyperplane, but it could be quite different from $\mathbf{w}^*$.

T/F The Naïve Bayes classifier is a discriminative classification algorithm.
False, it is generative.

T/F The squared-hinge-loss SVM (with or without bias) can be solved very efficiently with various fast optimization methods. For the Elastic Net this is not the case because the l_1 regularizer (/constraint) is non-differentiable.
False, the Elastic Net can be reduced the squared-hinge-loss SVM (without bias).

2 [22] Short Questions

1. (4) Assume you want to model your data $y_1, \ldots, y_n$ with a Poisson distribution,

$$P(y;\theta) = \frac{\theta^y e^{-\theta}}{y!} \text{ for } y = 0, 1, 2, \ldots$$

derive the log-likelihood of your data as a function of θ. Since

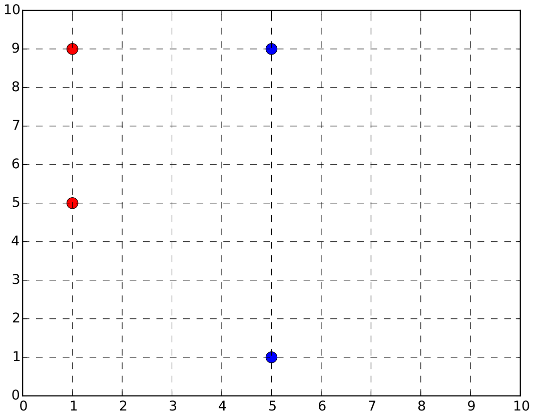
$$log(P(y;\theta) = log(\theta^y e^{-\theta}) - log(y!)) = ylog\theta - \theta - log(y!)$$

we have log likelihood function =

$$log\theta \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} (y_i!) - n\theta$$

2. (4) Briefly describe the Bayes Optimal classifier if $P(\mathbf{x}, y)$ is known. It's the optimal classifier if P(x,y) is known. All the errors are caused by noise.

3. (2) Draw the decision boundary for 1-NN for the following graph :



$(3,10) \rightarrow (3,7) \rightarrow (5,5) \rightarrow (0,0)$

4. (2) What is the relationship between kNN and the Bayes Optimal classifier? err(Bayes Optimal classifier) < err(KNN) < 2 * err(Bayes Optimal classifier)

5. (2) What is a downside of kNN as the data set size gets very large $n \gg 0$? curse of dimensionality & training speed will be slow

6. (2) In what scenario would you want to use the Huber loss? When we need the model to be less sensitive to outliers and differentiable.

7. (2) Name an advantage and a disadvantage of the l_1 regularization over l_2 regularization? l1: ADVANTAGE: Differentiable everywhere; DISADVANTAGE: Somewhat sensitive to outliers/noise l2: ADVANTAGE: less sensitive to noise; DISADVANTAGE: Not differentiable at 0

8. (4) You have a box with two coins. One of them is red, the other blue. We have $P(head|red) = \frac{1}{4}$ and $P(head|blue) = \frac{1}{2}$. You pick a coin uniformly at random and toss it. It comes up heads. What's the probability that it was the red coin?

$$\begin{aligned}P(red|head) &= P(red, head)/P(head) = P(red) * P(head|red)/P(head)\\&= P(red) * P(head|red)/(P(red) * P(head|red) + P(blue) * P(head|blue))\\&= 1/4 * 1/2/(1/2 * (1/4 + 1/2)) = 1/3\end{aligned}$$

3 [12] Naive Bayes

Assume you are provided with the following data:

$$\begin{aligned} \mathbf{x}_1 &= [1,1,2,1]^\top & y_1 &= +1 \\ \mathbf{x}_2 &= [1,2,2,1]^\top & y_2 &= +1 \\ \mathbf{x}_3 &= [2,2,1,1]^\top & y_3 &= -1 \\ \mathbf{x}_4 &= [2,2,2,1]^\top & y_4 &= -1 \\ \mathbf{x}_t &= [1,1,2,2]^\top & y_t &= ? \end{aligned}$$

Throughout this question use *categorical* Naïve Bayes *with +1 smoothing*.

1. (2) What is the probability of $P([\mathbf{x}_t]_2 = 1|y_t = -1)$, where $[\mathbf{x}_t]_2$ corresponds to the second feature value of $\mathbf{x}_t$.
   $P([\mathbf{x}_t]_2 = 1|y_t = -1) = \frac{1+\Sigma_{i=1}^4 I([\mathbf{x}_i]=1 \cap y_i=-1)}{2*1+\Sigma_{i=1}^4 I(y_i=-1)} = \frac{1+0}{2+2} = \frac{1}{4}$

2. (4) What is the probability of $P(\mathbf{x}_t|y_t = -1)$?
   $P(\mathbf{x}_t|y_t = -1) = P([\mathbf{x}_t]_1 = 1|y_t = -1) * P([\mathbf{x}_t]_2 = 1|y_t = -1) * P([\mathbf{x}_t]_3 = 2|y_t = -1) * P([\mathbf{x}_t]_4 = 2|y_t = -1)$
   $P(\mathbf{x}_t|y_t = -1) = \frac{1+0}{2+2} * \frac{1}{4} * \frac{1+1}{2+2} * \frac{1+0}{2+2} = \frac{1}{4} * \frac{1}{4} * \frac{1}{2} * \frac{1}{4} = \frac{1}{128}$

3. (6) What is the probability of $P(y_t = +1|\mathbf{x}_t)$?
   $P(y_t = +1|\mathbf{x}_t) = \frac{P(\mathbf{x}_t|y_t=+1)*P(y_t=+1)}{P(\mathbf{x}_t)}$
   $P(y_t = +1) = P(y_t = -1) = \frac{1}{2}$
   $P(\mathbf{x}_t) = P(\mathbf{x}_t|y_t = +1) * P(y_t = +1) + P(\mathbf{x}_t|y_t = -1) * P(y_t = -1)$
   $P(\mathbf{x}_t|y_t = +1) = P([\mathbf{x}_t]_1 = 1|y_t = +1) * P([\mathbf{x}_t]_2 = 1|y_t = +1) * P([\mathbf{x}_t]_3 = 2|y_t = +1) * P([\mathbf{x}_t]_4 = 2|y_t = +1)$
   $P(\mathbf{x}_t|y_t = +1) = \frac{3}{4} * \frac{1}{2} * \frac{3}{4} * \frac{1}{4} = \frac{9}{128}$
   $P(y_t = +1|\mathbf{x}_t) = \frac{\frac{9}{128}*\frac{1}{2}}{\frac{9}{128}*\frac{1}{2}+\frac{1}{128}*\frac{1}{2}} = \frac{9}{10}$

4 [17] Perceptron / LR

Assume you are provided with the following data:

$$\mathbf{x}_1 = [1, 1, 0, 1]^\top \qquad y_1 = +1$$
$$\mathbf{x}_2 = [1, 0, 0, 1]^\top \qquad y_2 = +1$$
$$\mathbf{x}_3 = [0, 0, 1, 1]^\top \qquad y_3 = -1$$
$$\mathbf{x}_4 = [0, 0, 0, 1]^\top \qquad y_4 = -1$$
$$\mathbf{x}_t = [1, 1, 0, 0]^\top \qquad y_t = ?$$

1. (8) Assume you train the *Perceptron* classifier on this data set and you visit the training data in the order $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_1, \mathbf{x}_2, \ldots$. How many updates do you need to make? What is the weight vector after each update?

$$\mathbf{w}_0 = [0, 0, 0, 0] \tag{1}$$
$$\mathbf{w}_1 = \mathbf{w}_0 + \mathbf{x}_1 = [1, 1, 0, 1] \tag{2}$$
$$\mathbf{w}_2 = \mathbf{w}_1 - \mathbf{x}_3 = [1, 1, -1, 0] \tag{3}$$
$$\mathbf{w}_3 = \mathbf{w}_2 - \mathbf{x}_4 = [1, 1, -1, -1] \tag{4}$$
$$\mathbf{w}_4 = \mathbf{w}_3 + \mathbf{x}_2 = [2, 1, -1, 0] \tag{5}$$
$$\mathbf{w}_5 = \mathbf{w}_4 - \mathbf{x}_4 = [2, 1, -1, -1] \tag{6}$$

2. (2) Name one advantage of Logistic Regression over Naïve Bayes, and one advantage vice versa.
Logistic over Naive Bayes: Logistic Regression do not need naive assumption.
Naive Bayes over Logistic: Faster than Logistic Regression in trainning
(There are other reasonable answers)

3. (7) Consider the following two data sets (Left and Right).

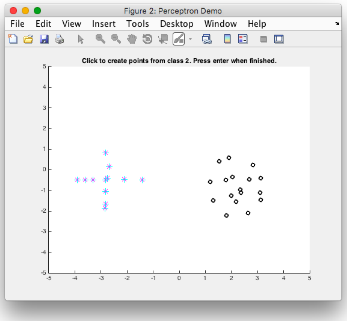


Figure 2: Perceptron Demo
File Edit View Insert Tools Desktop Window Help
5
4
3
2
1
0
-1
-2
-3
-4
-5
-5 -4 -3 -2 -1 0 1 2 3 4 5

Click to create points from class 2. Press enter when finished.

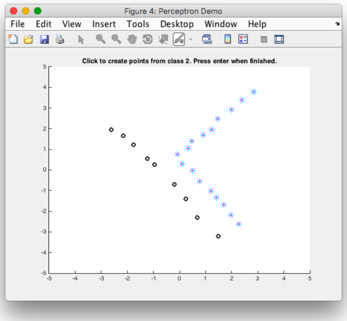


Figure 4: Perceptron Demo
File Edit View Insert Tools Desktop Window Help
5
4
3
2
1
0
-1
-2
-3
-4
-5
-5 -4 -3 -2 -1 0 1 2 3 4 5

Click to create points from class 2. Press enter when finished.

For both data sets state if

- . . . Gaussian Naïve Bayes will obtain zero training loss;
- . . . Logistic Regression will classify obtain zero training loss;
- . . . LR and NB will yield (almost) identical hyperplanes.

Justify your answers *briefly*.

	left	right
Naïve Bayes	T	F
Logistic Regression	T	T
identical hyperplanes	T	F

Logistic Regression can obtain zero training loss when linear separable data. But Naive Bayes cannot obtain zero training loss in the right graph as the assumption of Naive Bayes will not hold. They will get (almost) identical hyperplanes for left dataset.

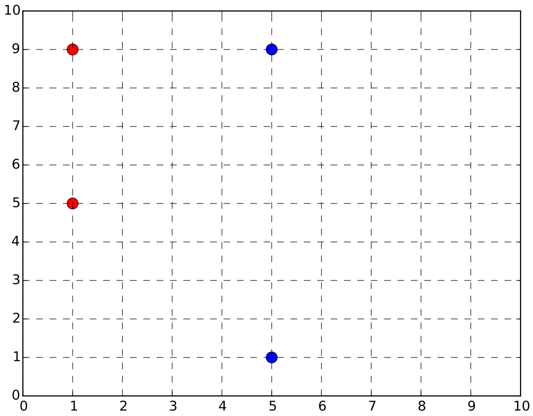
5 [7] SVM

1. (2) Name one advantage of Logistic Regression over the SVM classifier, and one advantage vice versa.
   Out of the many valid advantages and disadvantages, one for each of them is:

   - Logistic regression models are much more interpretable than SVMs i.e. there is a probabilistic interpretation possible to the class of a given test point.
   - SVMs are much more robust and generalize well on testing data, especially in case of text (high dimensional spaces).

   Some other acceptable answers are that we can kernelize SVMs, so they support non-linear boundaries. Please note that we have given points for such an answer but it is not entirely correct because logistic regression can also be kernelized.

2. (3) Assume you train an SVM (without slack variables) on the following data set. Draw in the decision boundary.



   A straight line at $x = 3$

3. (2) Now you need to add two data points such that the problem still *yields a feasible solution* (without slack) (i.e. the SVM must still find a valid solution after the point is added). First add a ”blue” data point (big X) that *would not affect* the decision boundary. Then add another ”blue” data point (big circle) that *would* affect the decision boundary.

Put the first blue point anywhere in the region $x \geq 5$ and the second blue point left to the margin $x = 5$, say at $(4, 5)$

6 [11] Linear Regression

Remember the loss function of the ordinary least squares (OLS)

$$\ell(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i^\top \mathbf{w} - y_i)^2 \tag{7}$$

for the data $\{\mathbf{x}_i, y_i\}$ where $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$, $i \in \{1...n\}$ in terms of the matrix $\mathbf{X} := [x_1\ x_2\ x_3\ ...\ x_n]$, the vector $\mathbf{y} := [y_1\ y_2\ ...\ y_n]^T$ and the weight vector $\mathbf{w} \in \mathbb{R}^d$

1. (3) State a model assumption for $P(y|\mathbf{x};\mathbf{w})$, under which the MLE estimate leads to the loss function $\ell(\mathbf{w})$ as stated above.

   $y_i \in \mathbb{R}$
   $y_i = w^T x_i + \epsilon_i$ where $\epsilon_i\ \ N(0, \sigma^2)$

2. (2) Write down the gradient descent update for the OLS problem. $\ell(\mathbf{w}) = \frac{1}{n}||X^T W - Y||$ therefore $g = 2X(X^T - Y)$. Gradient descent update:

   $$w_{t+1} \leftarrow w_t - cg$$

3. (4) If you were to optimize the OLS objective with Newton's Method, how many steps would you need until convergence (you can assume the Hessian is invertible)? (You can either derive the answer or state it clearly with justification.)

   A step of Newton's method minimizes the second order approximation of a function at a point w. OLS is quadratic, therefore its second order approximation at any point is equal to the function itself, hence one step of Newton minimizes the OLS.

4. (2) Why can't you use Newton's Method to optimize the (standard) SVM hinge loss?

   Not twice differentiable

This page was intentionally left blank.

This page was unintentionally left blank.