PRACTICE MIDTERM EXAM

CSC311 Fall 2019

University of Toronto

1. kNN.

(a) When do we expect k-NN to be better than logistic regression?
(b) Describe a sensible method for setting k in a k-nearest neighbor classifier.
(c) Contrast the decision boundaries for logistic regression and kNN.

2. Entropy and Information Gain. Recall the definitions of information gain and entropy:

$$\begin{aligned} Entropy(C) &\equiv H(C) = \sum_c -P(C=c)\log_2 P(C=c) \\ Gain(C,A) &= H(C) - \sum_{v\in Values(A)} P(A=v)H(C|A=v) \end{aligned}$$

(a) Suppose that in a set of examples there are two classes, with 150 examples in the + class and 50 examples in the − class. What is the entropy of the class variable (you can leave this in terms of logs)?
(b) For this data, suppose the *Color* attribute takes on one of 3 values (red, green, and blue), and the split into the two classes across *red*/*green*/*blue* is + : (120/10/20) and − : (0/10/40). Write down an expression for the class entropy in the subset containing all *green* examples. Is this entropy greater or less than the entropy in the previous question?
(c) Is *Color* a good attribute to add to the tree? Explain your answer.
(d) What is the information gain for a particular attribute if every value of the attribute has the same ratio between the number of + examples and the total number of examples?

3. Linear Classifiers.

3.1. *Logistic regression.* In class, we encoded the target values for logistic regression with $t^{(i)} \in \{0, +1\}$. In this problem, you will derive an equal formulation when targets are encoded with $\tilde{t}^{(i)}\{-1, +1\}$.

For a dataset $\mathcal{D}_N = \{(\mathbf{x}^{(i)}, t^{(i)})\}$ with $t^{(i)} \in \{0, +1\}$, logistic regression is defined using the following steps:

$$\begin{aligned} &z = \mathbf{w}^\top \mathbf{x} + b \\ &y = \sigma(z) \\ &\mathcal{L}(y, z) = -t\log(y) - (1-t)\log(1-y). \end{aligned}$$

(a) Write the equivalent cost minimization problem over training data by eliminating the intermediate variables y and z. Your cost function should only depend on variables $\mathbf{w}$ and b, and dataset $\mathcal{D}$.

(b) Show that if $\tilde{t}^{(i)} \in \{-1, +1\}$, the minimization problem takes the following form.

$$\text{minimize}_{\mathbf{w},b} \sum_{i=1}^{N} \log\Big(1 + \exp\{-\tilde{t}^{(i)}(\mathbf{w}^\top \mathbf{x}^{(i)} + b)\}\Big)$$

3.2. *Linear decision boundary.* Assume that we trained a logistic regression model and our class probabilities can be found by

$$z(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + b)$$

where $(\mathbf{w}_k, w_{k,0})$ are the parameters, and we classify using the rule

$$y(\mathbf{x}) = \mathbb{1}[z(\mathbf{x}) > 0.5].$$

Show that this corresponds to a linear decision boundary in the input space.

4. Optimization.

4.1. *Minimizing training error - 5pts.* Assume that you are minimizing a cost function which can be written as

$$\mathcal{J}(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\mathbf{w}, \mathbf{x}_i, t_i),$$

where $N = 1,000,000$.

(a) Write the one-step update rules for gradient descent (GD), stochastic GD (SGD), and mini-batch SGD (mSGD) with batch size 100. You can denote the gradient of the loss with respect to $\mathbf{w}$ for each sample with $\mathbf{g}_i = \nabla \mathcal{L}(\mathbf{w}, \mathbf{x}_i, t_i)$, and your learning rate with η.
(b) Rank the computational cost of each iteration for GD, SGD, and mini-batch SGD (with batch size 100) from smallest the largest.

5. Neural networks.

5.1. *NN-1.* Consider the following learning rule:

$$w_{ji}^{\texttt{new}} = w_{ji}^{\texttt{old}} - \eta \sum_n (y_j^{(n)} - t_j^{(n)}) x_i^{(n)}$$

(a) Define each of the five terms on the right-hand side of the learning rule.
(b) Imagine that another term is added, producing this new learning rule:

$$w_{ji}^{\texttt{new}} = w_{ji}^{\texttt{old}} - \eta \sum_n (y_j^{(n)} - t_j^{(n)}) x_i^{(n)} - 2\alpha w_{ji}^{\texttt{old}}$$

What is the main aim of such a term? What effect does this term have on the network weights?

5.2. *NN-3.* The "flexibility" of a neural network, it's ability to model different functions, is given by the number of hidden units. If we wanted to, we could simply use millions (i.e., a lot) of hidden units in order to model any kind of function we wanted. Why is this a bad idea in general? How could we avoid this problem?

6. True or False questions. Circle either True or False. Each correct answer is worth 2 points. To discourage random guessing, 2 points will be deducted for a wrong answer.

1. (True or False) Assume that you are using cross validation to choose the penalty parameter λ in L^2 regularized linear regression. As the number of training samples increases, we expect that the value of λ chosen by cross validation becomes larger.
2. (True or False) In the K-fold cross-validation procedure for selecting a model parameter λ out of m values, you fit your model $K \times m$ times.
3. (True or False) Assume that you have a dataset composed of N observations: the target $\mathbf{t}$ and features $\mathbf{X}$. You want to fit a linear regression model and find the weights $\mathbf{w}$, but you also know that more data is always helpful. Instead of fitting a model with $\mathbf{t} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{n \times d}$, you concatenate the data and fit a model using $\begin{bmatrix} \mathbf{t} \\ \mathbf{t} \end{bmatrix} \in \mathbb{R}^{2n}$ and $\begin{bmatrix} \mathbf{X} \\ \mathbf{X} \end{bmatrix} \in \mathbb{R}^{2n \times d}$. Running linear regression on this new dataset will give the same weights as on the original dataset.
4. (True or False) The decision boundaries resulting from linear regression with 1-of-K encoded targets are always the same those resulting from logistic regression.
5. (True or False) We use stochastic gradient descent (SGD) with very small constant step size to minimize a loss function. Assuming that we can run SGD for a very long time, eventually it will converge to a minimum of the loss function.