Midterm, CSCI 5525 2018

Paul Schrater

October 30, 2018

1 General Knowledge (15 pts)

1. (3 pts) Derive an expression for $expectedLoss$ involving $Bias$, $variance$, and $noise$.
2. (3 pts) Explain how to use cross-validation to estimate each of the terms above.
3. (4 pts) Bias in a classifier means that the probability of classifying a new data point drawn from the same distribution as the training data yields ($x_{new} \propto D_{train}$) will be labeled one category more than another.
   (a) What aspects of the training data affect classfier bias?
   (b) How does the hinge loss function in an SVM handle bias?
   (c) Which parameters of an SVM affect bias on test data? How does increasing or decreasing these parameters affect bias?
4. (5 pts) Consider a naive-Bayes generative model for the problem of classifying samples $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$, $\mathbf{x}_i \in \mathbb{R}^p$ and $y_i \in \{1, \ldots, K\}$, where the marginal distribution of each feature is modeled as a univariate Gaussian, i.e., $p(x_{ij}|y_i = k) \sim \mathcal{N}(\mu_{jk}, \sigma^2_{jk})$, where k represents the class label. Assuming all parameters have been estimated, clearly describe how such a naive-Bayes model will do classification on a test point x_{test}.

2 Experiments (15 pts)

Imagine we are using 10-fold cross-validation to tune a parameter θ of a machine learning algorithm using training set data for parameter estimation, and using the held-out fold to evaluate test performance of different values of θ. This produces 10 models,$\{h_1, ..., h_{10}\}$; each model h_i has its own value θ_i for that parameter, and corresponding error e_i. Let $k = \arg\min_i e_i$ be the index of the model with the lowest error. What is the best procedure for going from these 10 models individual to a single model that we can apply to the test data?

a) Choose the model h_k?
b) weight the predictions of each model by $w_i = \exp(-e_i)$?
c) Set $\theta = \theta_k$, then update by training on the held-out data.
Clearly explain your choice and reasoning.

3 Kernels (15 pts)

Let $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$ be a dataset of n-samples for regression with $\mathbf{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$. Consider a regularized regression approach to the problem:

$$\mathbf{w}^* = \min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{2} \sum_{i=1:n} (y_i - \mathbf{w}^T \mathbf{x_i})^2 + \lambda \mathbf{w}^T \mathbf{w}$$

This problem is known as ridge regression. Using a kernel trick we can write the solution to this problem in a form where we can use weights on either the feature dimensions or the data points. Rewriting the expression in terms of data points allows us to generalize the regression solution to nonlinear forms. To better understand the problem, remember that a data matrix X can be viewed in either row or column form, where rows are data points and columns feature dimensions. A regression solution weighting rows is suitable for a kernel form of a solution.

1. (5 pts) using notation $\mathbf{y} \in \mathbb{R}^n$ for the vector of responses generated by stacking labels into a vector, and $X \in \mathbb{R}^{nxp}$ the matrix of features, rewrite the objective function above in vector matrix form, and find the a closed form solution for $\mathbf{w}^*$. Is the solution valid for $n < p$?

2. (5 pts) Show that the solution can be kernelized (i.e. that $\mathbf{w}^* = \sum_{i=1:n} \alpha_i k(\mathbf{x}_i, \cdot)$) for some function $k(x, \cdot)$ you need to derive. The trick is the derivation is a matrix inverse identity: $(A^{-1} + B^T C^{-1} B)^{-1} B^T C^{-1} = AB^T (BAB^T + C)^{-1}$. In your application, $X = B$, $C = I$ and $A = \lambda I$. The point of using the inverse is to convert your solution from it's standard form into one where you use XX^T, which creates an inner product matrix of size data point by data point. By applying the resulting solution to a new feature vector x, show that $w^T x$ can be written in kernel form as above.

3. (5 pts) Use the kernelization result to derive ridge regression for fitting polynomials to order m using a polynomial kernel function.

4 Gradient Descent (15 pts)

Consider the following regularized logistic regression problem:

$$\mathbf{w}^* = \min_{\mathbf{w} \in \mathbb{R}^p} \mathcal{L}(\mathbf{w})$$

where

$$\mathcal{L}(\mathbf{w}) = \frac{1}{n}\sum_{i=1:n}\left\{-y_i\mathbf{w}^T\mathbf{x_i} + \log\left(1+\exp\left(-\mathbf{w}^T\mathbf{x_i}\right)\right)\right\} + \lambda G(\mathbf{w})$$

where $G(\mathbf{w})$ is a (possibly non-smooth) regularization function.

1. (5 pts) Derive the subgradient for $\mathcal{L}(\mathbf{w})$ if $G(\mathbf{w}) = \sum_{i=1:p}(a|w_i|^{\frac{1}{2}} + b)^2$
2. (5 pts) Derive a subgradient stochastic gradient descent algorithm for when $G(\mathbf{w}) = \sum_{i=1:p}|w_i|$
3. (5 pts) If $G(\mathbf{w}) = \frac{1}{2}\sum_{i=1:p} w_i^2$, what is the expected runtimes for Gradient descent and Stochastic Gradient Descent in terms of (n, ϵ)

5 Boosting (15 pts)

The problem considers the adaboost algorithm.

1. (5 points) Describe the adaboost algorithm, using pseudocode. Clearly discuss the variables and any assumptions on them.
2. (5 points) For adaboost, let α_t denote the weight on the weak hypothesis G_t in step t. How is α_t selected as the solution to an optimization problem? Clearly explain your answer.
3. (5 points) Recall that adaboost can be viewed as minimizing a suitable upper bound loss function on the "true" loss $[y \neq h(\mathbf{x})]$, using the upper bound function $C(yh(\mathbf{x}) = \exp(-yh(\mathbf{x}))$. Can one choose $C(\cdot)$ to be the hinge loss?, i.e.,

   $$C(yh(\mathbf{x})) = \max(0, 1 - yh(\mathbf{x}))$$

   Explain your answer.

6 Adaboost (25 pts)

At Paul's house, we need help with deciding whether a dog adoption candidate should be brought home. Use following data set to to learn a committee machine via Adaboost to predict whether the doggie are Adoptable (yes) or (no) based on their Price, Potty training preparedness level (Normal or Low), Previous Incidents of Carpet Damage (2, 3, or 4), and hair color (Brown/White or Yellow). Use a sequence of weak learners to predict the target variable Adoptable. Each weak learner can only classify using one dimension. Use up to ten learners, features can be used in any order and you choose how to assign features to learners.

		Item			
Doggie	Potty Training Prep (weeks)	Price ($)	Carpet Damage	Color	Adoptable
Tribi	3	13	0	Brown/White	yes
Ross	1	0.01	1	Brown/White	no
Rachael	1	92.50	2	Yellow	no
Chandler	2	33.33	1	Brown/White	no
Phoebe	3	8.99	3	Yellow	no
Monica	2	8.99	0	Brown/White	yes
Matt	3	13.65	0	Brown/White	yes
David	1	0.01	2	Yellow	no
Jenyfur	1	92.50	2	Brown/White	no
Winona	2	33.33	2	Yellow	no
Boo	3	8.99	1	Brown/White	yes
Beau	2	12.49	2	Brown/White	no

Table 1: Background data for the adaboost problem.

1. (5 points) Using the definition of weak learner, explain which features are eligible for assignment to weak learners?

2. (15 points) Implement adaboost and run for no more than 10 rounds. Using your adaboost committee machine trained on the data above, what should I do with Joey? Joey has 3 weeks potty training, costs 0 dollars, has 0 instances of previous carpet damage and is Brown/White. Adopt or no?

3. (2 points) Describe the first 3 stages of your algorithm as a decision tree.

4. (3 points) How much do you think you could benefit from additional learners? Answer by explaining how you might determine how many learners are needed.