Homework Assignment 3

Due in class, Thursday October 21

SDS 383C Statistical Modeling I

1 Ridge regression and Lasso

1. Get the Prostrate cancer data from `http://statweb.stanford.edu/~tibs/ElemStatLearn/datasets/prostate.data`. More information about this dataset can be found in `http://statweb.stanford.edu/~tibs/ElemStatLearn/datasets/prostate.info.txt`.

   (a) In class we learned about Ridge regression with tuning parameter λ. Define

   $$\mathrm{df}(\lambda) = \mathrm{tr}\left(\boldsymbol{X}(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X} + \lambda \boldsymbol{I})^{-\lambda}\boldsymbol{X}^{T}\right).$$

   Plot the coefficients of the covariates as λ is increased from 0 to 1000. A similar plot can be found in figure 3.8 in H-T-F. This figure essentially plots the ridge regression coefficients of the covariates as dfλ is increased.

   (b) Now plot the coefficients learned by Lasso as λ is increased from 0 to 100. For this you can use the LARS package.

   (c) Finally reproduce columns 4 and 5 for Ridge regression and Lasso in Table 3.3. Remember to reproduce the test set prediction errors as well.

2 Discriminative vs. Generative Classifiers

A very common debate in statistical learning has been over generative versus discriminative models for classification. In this question we will explore this issue, both theoretically and practically. We will consider Naive Bayes and logistic regression classification algorithms.

To answer this question, you might want to read: *On Discriminative vs. Generative Classifiers: A comparison of logistic regression and Naive Bayes*, Andrew Y. Ng and Michael Jordan. In NIPS 14, 2002. `http://www.robotics.stanford.edu/~ang/papers/nips01-discriminativeg` `pdf`

2.1 Logistic regression and Naive Bayes

(a) `The discriminative analog of naive Bayes is logistic regression`. This means that the parametric form of $P(Y|X)$ used by Logistic regression is implied by the assumptions of a Naive Bayes classifier, for some specific class-conditional densities. In the reading you will see how to prove this for a Gaussian naive bayes classifier for continuous input values. Can you prove the same for binary inputs ?Assume X_i and Y are both binary. Assume that $X_i|Y = j$ is Bernoulli(θ_{ij}), where $j \in \{0, 1\}$, and Y is Bernoulli(π).

2.2 Double counting the evidence

(a) Consider the two class problem where class label $y \in \{T, F\}$ and each training example X has 2 binary attributes $X_1, X_2 \in \{T, F\}$. How many parameters will you need to know/evaluate if you are to classify an example using the Naive Bayes classifier?

Let the class prior be $P(Y = T) = 0.5$ and also let $P(X_1 = T|Y = T) = 0.8$ and $P(X_1 = F|Y = F) = 0.7$. , $P(X_2 = T|Y = T) = 0.5$ and $P(X_2 = F|Y = F) = 0.9$. So, attribute X_1 provides a slightly stronger evidence about the class label than X_2.

i. Assume X_1 and X_2 are truly independent given Y. Write down the Naive Bayes decision rule.

ii. Show that if Naive Bayes uses both attributes, X_1 and X_2, the error rate is 0.235. Is it better than using only a single attribute (X_1 or X_2)? Why ? The error rate is defined as the probability that each class generates an observation where the decision rule is incorrect.

iii. Now, suppose that we create new attribute X_3, which is an exact copy of X_2. So, for every training example, attributes X_2 and X_3 have the same value, $X_2 = X_3$. What is the error rate of Naive Bayes now?

iv. Explain what is happening with Naive Bayes?

v. (extra credit) In spite of the above fact we will see that in some examples Naive Bayes doesn't do too badly. Consider the above example i.e. your features are X_1, X_2 which are truely independent given Y and a third feature $X_3 = X_2$. Suppose you are now given an example with $X_1 = T$ and $X_2 = F$. You are also given the probabilities $P(Y = T|X_1 = T) = p$ and $P(Y = T|X_2 = F) = q$, and $P(Y = T) = .5$. Prove that the decision rule is $p \geq \frac{(1-q)^2}{q^2+(1-q)^2}$ (Hint : use Bayes rule again). What is the true decision rule ? Plot the two decision boundaries (vary q between $0 - 1$) and show where Naive Bayes makes mistakes.

2.3 Learning Curves of Naive Bayes and Logistic Regression

Compare the two approaches on the Breast Cancer dataset you can download from course webpage. You can find the description of this dataset from the course webpage. We have removed the records with missing values for you. Obtain the learning curves similar to Figure 1 in the paper.

Implement a Naive Bayes classifier and a logistic regression classifier with the assumption that each attribute value for a particular record is independently generated.

For the NB classifier, assume that $P(x_i|y)$, where x_i is a feature in the breast cancer data (that is, i is the number of column in the data file) and y is the label, of the following multinomial distribution form:

For $x_i \in \{v_1, v_2, ..., v_n\}$,

$$p(x_i = v_k|y = j) = \theta^i_{jk} \; s.t. \; \forall i, j : \sum_{k=1}^{n} \theta^i_{jk} = 1$$

where $0 \leq \theta_{jk} \leq 1$ It may be easier to think of this as a normalized histogram or as a multi-value extension of the Bernoulli.

Use 2/3 of the examples for training and the remaining 1/3 for testing. Be sure to use 2/3 of each class, not just the first 2/3 of data points.

For each algorithm:

(a) Implement the IRLS algorithm for Logistic regression.

(b) Plot a learning curve: the accuracy vs. the size of the training data. Generate six points on the curve, using [.01 .02 .03 .125 .625 1] fractions of your training set and testing on the full test set each time. Average your results over 5 random splits of the data into a training and test set (always keep 2/3 of the data for training and 1/3 for testing, but randomize over which points go to training set and which to testing). This averaging will make your results less dependent on the order of records in the file. Plot both the Naive Bayes and Logistic Regression, learning curves on the same plot. Use the `plot(x,y)` function in Matlab since the training data fractions are not equally spaced.

Specify your choice of prior/regularization parameters and keep those parameters constant for these tests. A typical choice of constants would be to add 1 to each bin before normalizing (for NB) and $\lambda = 0$ (for LR).

(c) What conclusions can you draw from your experiments? Specifically, what can you say about speed of convergence of the classifiers? Are these consistent with the results in the NIPS paper that we have mentioned? If yes, state that. If no, explain why not.