CSCI 1950-F Homework 6: Regularization & Sparsity

Brown University, Spring 2011

Homework due at 11:59pm on March 25, 2011

Question 1:

This problem compares various approaches to regularization and feature selection for binary classification. Let $y \in \{+1, -1\}$ denote the binary class label we want to predict, and x the input features. Consider the following binary logistic regression model:

$$p(y = 1 \mid x, w) = \frac{1}{1 + \exp(-w^T x)}$$

The *Dorothea* dataset (`http://archive.ics.uci.edu/ml/datasets/Dorothea`) contains 100,000 features encoding structural molecular features of chemical compounds, and the label y indicates whether the compound is active (binding) or inactive. We have split the dataset into 400 training, 400 validation, and 350 test instances. Each data instance x is a binary vector of dimension 10^5. The dataset is available at `/course/cs195f/asgn/dorothea`.

We will compare three types of regularizations: L_2 (Gaussian), L_1 (Laplacian), and the Huber loss which smoothly interpolates between quadratic and linear. Given n training examples and m features, the MAP estimator minimizes the following objective:

$$f(w) = -\sum_{i=1}^{n} \log p(y_i \mid x_i, w) + \lambda \sum_{j=1}^{m} L(w_j)$$

For the following questions, we use the validation set to choose among ten logarithmically spaced values of the regularization weight λ:

```
>> lambda = logspace(-8,1,10);
```

For either L_2 or L_1 regularization, we can fit the model via the *logregFit* function from the pmtk package. Here is an example for L_2 regularization:

```
>> modelL2 = logregFit(X_train, Y_train, 'regType', 'L2', 'lambda', lambda);
>> Y_est = logregPredict(modelL2, X_val);
```

The `logregPredict` method can then be used to predict labels for validation or test data.

a) Train logistic regression models with Gaussian regularization ($L_2(w_j) = w_j^2$) on the `train` *data. Create a plot of the* `validation` *error rate as a function of λ.*

b) *Select the λ value which results in the smallest L_2-regularized validation error. Report this optimal λ, the number of nonzero weights in this model (see the Matlab command* `nnz`*), and its error rate on the* `test` *data.*

c) *Train logistic regression models with Laplacian regularization ($L_1(w_j) = |w_j|$) on the* `train` *data. Create a plot of the* `validation` *error rate as a function of λ.*

d) *Select the λ value which results in the smallest L_1-regularized validation error. Report this optimal λ, the number of nonzero weights in this model, and its error rate on the* `test` *data.*

e) *What happens to the L_1-regularized validation error rate when $\lambda = 10$? Provide an explanation for your observation.*

f) *The Huber loss, with "closeness" parameter δ, is defined as follows:*

$$L_H(w, \delta) = \begin{cases} w^2/2 & \text{if } |w| \leq \delta \\ \delta|w| - \delta^2/2 & \text{if } |w| > \delta \end{cases}$$

Derive an expression for the derivative of this loss with respect to w. Is its second derivative defined everywhere?

g) *Implement a gradient-based algorithm for fitting Huber regularized logistic regression, using the* `minFunc` *method, the* `LogisticLossSimmple` *method, and the gradients derived above. The following script provides a starting point.*

```
%Logistic regression loss function from pmtk
nVar = size(X_train,2); % number of variables
loss = @(w) LogisticLossSimple(w, X_train, Y_train, ones(size(X_train,1),1));

%penalizedHuber will be written by you, with inputs:
%  w - weight vector to be optimized,
%  loss - function handle for logistic regression loss function
%  lambdaVec - regularization weights
%  delta - "closeness" parameter of the Huber loss function
lambdaVec = lambda*ones(nVar, 1);
penloss = @(w)penalizedHuber(w, loss, lambdaVec, delta);

% minFunc options
opts.Display     = 'verbose';
opts.verbose     = false;
opts.TolFun      = 1e-3;
opts.MaxIter     = 200;
opts.Method      = 'lbfgs'; % for minFunc
opts.MaxFunEvals = 2000;
opts.TolX        = 1e-3;

w = minFunc(penloss, zeros(nVar,1), opts);
```

h) Train logistic regression models with Huber regularization, and closeness $\delta = 100$, on the `train` *data. Create a plot of the* `validation` *error rate as a function of λ. What happens? Explain your observation.*

i) Now set δ to the median of the absolute value of the weights obtained from the logistic regression model with L_2 regularization and $\lambda = 10^{-8}$. Create a plot of the `validation` *error rate as a function of λ. Explain any differences from the result in part (h).*

j) Select the λ and δ values which result in the smallest Huber-regularized validation error. Report these optimal parameters, the number of nonzero weights in this model, and its error rate on the `test` *data.*

k) What is one advantage of Huber regularization compared to L_1 regularization? What is one disadvantage?

Question 2:

Consider maximum likelihood parameter estimates $\hat{\theta}$ for the following binary classifiers:

GaussI A generative classifier, where the class conditional densities are Gaussian, with both covariance matrices set to identity matrices, i.e. $p(x \mid y = c) = N(x \mid \mu_c, I)$.

GaussX As with GaussI, but with unconstrained covariances $p(x \mid y = c) = N(x \mid \mu_c, \Sigma_c)$.

LinLog A logistic regression model with linear features plus a constant bias feature.

QuadLog A logistic regression model with a constant bias feature, linear features, and quadratic features (encoding the product of all pairs of inputs and their squares).

After training we compute the performance of each model M via evaluating its *conditional* log-likelihood on the training set:

$$L(M) = \frac{1}{n} \sum_{i=1}^{n} \log p(y_i \mid x_i, \hat{\theta}, M)$$

We now want to compare the performance of each model. We will write $L(M) \leq L(M')$ if model M *must* have lower (or equal) conditional log-likelihood (on the training set) than M', for any training set. For each of the following model pairs, state whether $L(M) \leq L(M')$, $L(M) \geq L(M')$, or whether no such statement can be made (i.e., M might sometimes be better than M' and sometimes worse). Also provide 1-2 sentence explanations.

a) GaussI versus LinLog.

b) GaussX versus QuadLog.

c) LinLog versus QuadLog.

d) GaussI versus QuadLog.

e) GaussX versus LinLog.