EE 5841 Machine Learning

Project 4

Kernel Logistic Regression

Derivation of the dual formulation for linear logistic regression:

Let (x_i, y_i) denote the training set where $x_i \in R^d$ and $y_i \in$ (-1,1)

$$P(y|x) = \frac{1}{1 + \exp(-y_i(w^T.\mathrm{x} + \mathrm{b}))} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots . \dots (1)$$

Taking max log likelihood and adding the regularization term

$$\min E = \frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i \log(1 + e^{\xi}) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (2)$$

Where $\xi = -y_i(w^T\mathrm{x} + \mathrm{b}) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots . (3)$

Taking $g(\xi) = \log(1 + e^{\xi}) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4)$

Taking Lagrangian

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i g(\xi_i) + \sum_i \alpha_i(-\xi_i - y_i(w.x_i - b)) \dots\dots\dots\dots\dots\dots . (5)$$

To obtain the equation in terms of α

Taking Partial derivative of the Lagrangian equation w.r.t. w

$$\nabla_w L = \lambda w - \sum_i \alpha_i y_i x_i = 0 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots . (6)$$

Taking Partial derivative of the Lagrangian equation w.r.t. b

$$\frac{\partial L}{\partial b} = \sum_i \alpha_i y_i = 0 \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (7)$$

Taking Partial derivative of the Lagrangian equation w.r.t. ξ_i

$$\frac{\partial L}{\partial \xi} = g'(\xi_i) - \alpha_i = 0$$

$$\xi_i = {g'}^{-1}(\alpha_i) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (8)$$

From equation (5)

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i g(\xi_i) - \sum_i \alpha_i \xi_i - \sum_i (\alpha_i y_i x_i).w + \sum_i \alpha_i y_i b \ldots\ldots\ldots (9)$$

Taking $G(\alpha_i) = \alpha_i \xi_i - g(\xi_i) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$

Taking partial derivative of Equation (9) w.r.t. α_i

$$\frac{\partial G(\alpha_i)}{\partial \alpha} = (\xi_i \alpha_i \frac{\partial \xi}{\partial \alpha} + (1)\xi_i) - g'(\xi_i)\frac{\partial \xi}{\partial \alpha}$$

$$= \xi_i + \alpha_i \frac{\partial \xi}{\partial \alpha} - g'(\xi_i)\frac{\partial \xi}{\partial \alpha}$$

Substituting equation (8)

$$= \xi_i + \alpha_i \frac{\partial \xi}{\partial \alpha} - \alpha_i \frac{\partial \xi}{\partial \alpha}$$

$$= \xi_i$$

Substituting equation (8)

$$= g'^{-1}(\alpha_i)$$

$g'(\alpha_i)$ is a convex function, therefore $g'^{-1}(\alpha_i)$ is also a convex function

$$\frac{\partial G(\alpha_i)}{\partial \alpha} = G'(\alpha_i) = \log \frac{\alpha_i}{1-\alpha_i} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (11)$$

Therefore,

$$\xi_i = \log \frac{\alpha_i}{1-\alpha_i} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (12)$$

From equation (4)

$$g(\xi) = \log(1 + e^{\xi})$$

Substituting equation (12)

$$g(\xi) = \log\left(1 + e^{\log_e \frac{\alpha_i}{1-\alpha_i}}\right)$$

$$g(\xi) = \log\left(1 + \frac{\alpha_i}{1 - \alpha_i}\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (13)$$

From equation (9)

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i g(\xi_i) - \sum_i \alpha_i \xi_i - \sum_i (\alpha_i y_i x_i).w + \sum_i \alpha_i y_i b$$

Substituting equation (10)

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 - \sum_i G(\alpha_i) - \sum_i (\alpha_i y_i x_i).w + \sum_i \alpha_i y_i b \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (14)$$

Substituting equation (7)

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 - \sum_i G(\alpha_i) - \sum_i (\alpha_i y_i x_i).w$$

Substituting equation (6)

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 - \sum_i G(\alpha_i) - \lambda w^T.w$$

$$\max L = \frac{\lambda}{2}\|\mathrm{w}\|^2 - \sum_i G(\alpha_i) - \lambda\|\mathrm{w}\|^2$$

$$\max L = -\frac{\lambda}{2}\|\mathrm{w}\|^2 - \sum_i G(\alpha_i)$$

To maximize the above equation, we have to minimize the function of α

Therefore, the Dual formulation of the Linear Logistic Regression will be

$$\min f(\alpha) = -\frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i G(\alpha_i) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (15)$$

Subject to $\sum_i \alpha_i y_i = 0 , \forall i$

Kernel Version of the Dual formation of logistic regression:

From equation (15)

$$\min f(\alpha) = \frac{\lambda}{2}\|\mathrm{w}\|^2 + \sum_i G(\alpha_i)$$

Substituting equation (6)

$$\min f(\alpha) = \frac{\lambda}{2}(| \frac{1}{\lambda}\sum_i \alpha_i y_i x_i \ |^T . | \frac{1}{\lambda}\sum_j \alpha_j y_j x_j \ |) + \sum_i G(\alpha_i)$$

$$\min f(\alpha) = \frac{1}{2\lambda}\sum_i \sum_j \alpha_i \alpha_j y_i y_j (x_i).(x_j) + \sum_i G(\alpha_i) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots . (16)$$

Subject to $\sum_i \alpha_i y_i = 0 , \forall i$

Where $(x_i).(x_j)$ is the Kernel Term

Kernel Version of Pr(y|x) used the obtained dual variables:

From equation (1)

$$P(y|x) = \frac{1}{1 + \exp\left(-y_i\left(w^T.x_j + \mathrm{b}\right)\right)}$$

Substituting equation (6)

$$P(y|x) = \frac{1}{1 + \exp\left(-\mathrm{y}\left(\frac{1}{\lambda}\sum_i \alpha_i y_i (x_i).(x_j) + \mathrm{b}\right)\right)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

Subject to $\sum_i \alpha_i y_i = 0 , \forall i$

Where $(x_i).(x_j)$ is the Kernel Term

Code:

```
clear all;
data = load('heartstatlog_trainSet.txt');
labels = load('heartstatlog_trainLabels.txt');
dataTest = load('heartstatlog_testSet.txt');
labelTest = load('heartstatlog_testLabels.txt');
data = bsxfun(@rdivide,bsxfun(@minus,data,mean(data)),std(data));

labels = 2*(labels - 1.5);
labelTest = 2*(labelTest - 1.5);

%Formulating Linear functions
kernel = data*data';
kernelTest = data*dataTest';
C = [0.01 0.1 0.25 1 5 25 100];

for j = 1:size(C,2)

k = 5;
Ntrain=length(data);
ind = randperm(Ntrain);

testInd=ind(1:floor(Ntrain/k))';
trainData = data;
trainLabels = labels;

trainData(testInd,:) = [];
trainLabels(testInd,:) = [];

testData = data(testInd,:);
testLabels = labels(testInd,:);

kernelCVTrain = kernel;
kernelCVTrain(testInd,:) = [];
kernelCVTrain(:,testInd) = [];

kernelCVTest = kernel(1:size(trainData,1),testInd);
kernelCVVal = kernel(1:size(testData,1),testInd);

kernelTrain = data(1:size(trainData,1),:)*data(1:size(trainData,1),:)';

kernelTestCV = kernelTest(1:size(trainData,1),:);

alpha = ones(size(kernelCVTrain,1),1)*(0.5)*(1/C(j));
fun3 = @(alpha)objFun3(alpha,trainLabels,kernelCVTrain,C(j));

A = [];
b = [];
Aeq = trainLabels';
```

```
beq = 0;
lb = zeros(size(kernelCVTrain,1),1);
ub = C(j)*ones(size(trainLabels,1),1);

opAlpha = fmincon(fun3,alpha,A,b,Aeq,beq,lb,ub);

supporters = find(opAlpha > 1e-5);
b = 0;
bias = @(b)objBias3(b,opAlpha,trainLabels,kernelCVTrain,C(j));

opB(j) = fminunc(bias,b);

predVal = sign((((opAlpha.*trainLabels)/C(j))'*kernelCVTest)' + opB(j));
errorsCVValid(:,j) = sum(predVal ~= testLabels)/length(testLabels);

predTrain = sign((((opAlpha.*trainLabels)/C(j))'*kernelTrain)' + opB(j));
errorsCVTrain(:,j) = sum(predTrain ~= labels(1:length(predTrain)))/length(predTrain);

predTest = sign((((opAlpha.*trainLabels)/C(j))'*kernelTestCV)' + opB(j));
errorsTestCV(:,j) = sum(predTest ~= labelTest)/length(labelTest);

end

[minErrorCV,lambdaInd] = min(errorsCVValid);
bestLambda = C(lambdaInd);
bestB = opB(lambdaInd);

%running on entire training data
A = [];
b = [];
Aeq = labels';
beq = 0;
lb = zeros(size(kernel,1),1);
ub = bestLambda*ones(size(labels,1),1);
alpha = ones(size(kernel,1),1)*(0.5)*(1/bestLambda);
fun3 = @(alpha)objFun3(alpha,labels,kernel,bestLambda);

trainAlpha = fmincon(fun3,alpha,A,b,Aeq,beq,lb,ub);

supporters = find(trainAlpha > 1e-5);
b = 0;
bias = @(b)objBias3(b,trainAlpha,labels,kernel,bestLambda);

trainB = fminunc(bias,b);

pred = sign((((trainAlpha.*labels)/bestLambda)'*kernel)' + trainB);
errorsTrain = sum(pred ~= labels)/length(labels);

%running on testing data
```

```
pred = sign((((trainAlpha.*labels)/bestLambda)'*kernelTest)' + bestB);
errorsTest = sum(pred ~= labelTest)/length(labelTest);

figure
plot(log10(C),errorsCVTrain,'-x');
hold on
plot(log10(C),errorsCVValid,'-o');
hold on

plot(log10(C),errorsTestCV,'--
gs','LineWidth',2,'MarkerSize',10,'MarkerEdgeColor','b');
legend('Train','Validation','Test');
xlabel('Lambda');
ylabel('Error');
```

Plot: