UNIVERSITY OF VICTORIA
Department of Electrical and Computer Engineering

ECE 403/503 Optimization for Machine Learning

LABORATORY REPORT

Experiment No: 3

Title: Predicting Energy Efficiency for Residential Buildings

Date of Experiment: 10/31/2019

Report Submitted on:11/7/2019

To: TA

Laboratory Group No.:

Name(s): Kurt Elliott & Raj Deepak S. N.

Objectives:

In this experiment, we investigate a technique for multi-category classification based on the multi-output linear model. The technique is then applied to train a model with 8 features from a building dataset with corresponding heating and cooling loads. The model will then be tested on a test dataset of building to demonstrate its effectiveness.

Introduction:

The dataset of the heating and cooling loads is used in this experiment was created by A. Tsanas and A. Xifara in 2012, which includes 8 features, namely:

- Relative compactness
- Surface area
- Wall area
- Roof area
- Overall height
- Orientation
- Glazing area
- Glazing area distribution

These 8 features are used to evaluate the heating and cooling loads of each building. They will be used in two separate datasets, one each for training and testing the model. The dataset set contains 768 samples, each sample is characterized by an input vector **x,** respectively, with 8 components that are numerical values of the eight features mentioned. Each sample also has a 2 component output vector **y,** which represents the heating and cooling load of the building respectively.

Results:

The following is the MATLAB result from the script used in this lab, and a brief description of how the program is implemented. The script was sliced into sections for readability, therefore refer to appendix for full script.

The script used in this lab consists of three primary sections. The first section of the code is used to load and extract the training and testing datasets. The datasets consists of input vectors **x**, which consist of 8 components (features), and **y** output vectors, which consist of 2 components (heating & cooling loads).

```
% Load Datasets into workspace
load('D_build_tr');
load('D_build_te');

%% Format dataset into X and Y for training and testing
Xtr = D_build_tr(1:8,:);
Ytr = D_build_tr(9:10,:);

Xte = D_build_te(1:8,:);
Yte = D_build_te(9:10,:);
```

Next is the model training section. A variable lambda is used to ensure that the inverse matrix exists during evaluation. Then the data is formatted into ‘Xhat’ and ‘Yhat’ matrices to be easily evaluated into ‘wb’, which consists of ‘HAT’ matrix multiplied by ‘Yhat’ matrix. Lastly the ‘wb’ is broken down into its corresponding components; ‘h’ is used to represent heating components and ‘c’ is used to represent cooling components.

```
%% Training the model
% Paramaters:
Lamda = 0.001;

% Make matrix with transposed Xtr and add ones Column
Xhat = [Xtr' ones(640,1)];
% make matrix with transposed Ytr
Yhat = [Ytr'];
% wb is the HAT matrix multiplied by Yhat
wb = ((Xhat'*Xhat+Lamda*eye())^-1)*Xhat'*Yhat;
% h and c indicates heating and cooling load elements
wh = wb(1:8,1);
wc = wb(1:8,2);
bh = wb(9,1);
bc = wb(9,2);
```

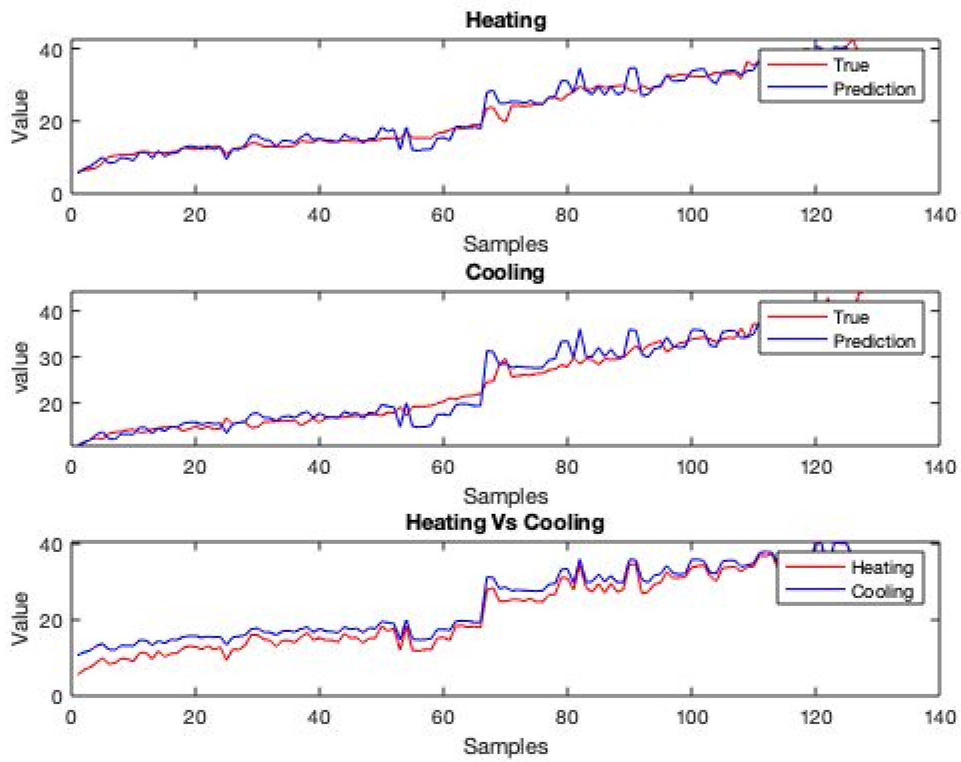
The final part was building a linear regression model from the ‘w’ and ‘b’ components and testing the models prediction with the test dataset. This was done by taking the difference between the test ‘Yte’ and the models ‘Y’ than taking the norm of the difference and normalizing it by dividing by the norm of ‘Yte’. The final error achieved was 10.87% .

```
%% Testing the model with given test files.
% h and c indicates heating and cooling load elements
Yh = wh'*Xte +bh;
Yc = wc'*Xte +bc;
Y = [Yh;Yc];

DeltaY = Yte - Y;
% Calculate the overall relative prediction error
Eps = norm(DeltaY,'fro')/norm(Yte,'fro')
```

Discussion:

After training the model, analysis was done on the testing dataset and the results from the true values and the model predicted values were plotted. While the true values seem to be smoothly increasing, the predicted values have more variations as we increase the sample size. This trend is seen for both heating and cooling, with comparatively greater variations in the predicted values for cooling. The model we have derived has a good performance for 89.13%. A comparison between the predicted heating and cooling values were plotted and the 2 lines look similar in variations. This is not a case in the true values, as most of the peaks in heating values correspond to troughs in the cooling values and vice-versa. Our model might perform well if this problem is addressed.



Conclusions:

Multi-category classification based on multi-output linear model was investigated. With this technique, the heating and cooling efficiencies of a residential building were computed. These computed values were compared with the true values. The values differed by an error of 10.87%.

Appendix:

Table of Contents

.... 1
Format dataset into X and Y for training and testing 1
Training the model 1
Testing the model with given test files. 1
Plotting 2


```
clear all, clc;

% Load Datasets into workspace
load('D_build_tr');
load('D_build_te');
```

Format dataset into X and Y for training and testing

```
Xtr = D_build_tr(1:8,:);
Ytr = D_build_tr(9:10,:);

Xte = D_build_te(1:8,:);
Yte = D_build_te(9:10,:);
```

Training the model

Paramaters:

```
Lamda = 0.001;

% Make matrix with transposed Xtr and add ones Column
Xhat = [Xtr' ones(640,1)];
% make matrix with transposed Ytr
Yhat = [Ytr'];
% wb is the HAT matrix multiplied by Yhat
wb = ((Xhat'*Xhat+Lamda*eye())^-1)*Xhat'*Yhat;
% h and c indicates heating and cooling load elements
wh = wb(1:8,1);
wc = wb(1:8,2);
bh = wb(9,1);
bc = wb(9,2);
```

Testing the model with given test files.

h and c indicates heating and cooling load elements

```
Yh = wh'*Xte +bh;
```

```
Yc = wc'*Xte +bc;
Y = [Yh;Yc];

DeltaY = Yte - Y;
% Calculate the overall relative prediction error
Eps = norm(DeltaY,'fro')/norm(Yte,'fro')
```

```
Eps =

    0.1087
```

Plotting

```
figure(1)
subplot(3,1,1)
plot(Yte(1,:),'r-');
hold on;
plot(Yh,'b-');
xlabel('Samples');
ylabel('Value');
title('Heating');
legend('True','Prediction');


subplot(3,1,2)
plot(Yte(2,:),'r-');
hold on;
plot(Yc,'-b');
xlabel('Samples');
ylabel('value');
title('Cooling');
legend('True','Prediction');

subplot(3,1,3)
plot(Yh,'r-');
hold on;
plot(Yc,'-b');
xlabel('Samples');
ylabel('Value');
title('Heating Vs Cooling');
legend('Heating','Cooling');
```