Table of solution

	OLS	Ridge	Lasso Regression
b0	21.9655	29.9095	4.1629
b(1)	0.0250	0.0316	0
b(2)	-1.0836	-1.0248	-1.0241
b(3)	-0.1826	0.2502	0.0000
b(4)	0.0163	0.0219	0.0014
b(5)	-1.8742	-0.8083	-1.7102
b(6)	0.0044	0.0007	0.0024
b(7)	-0.0033	-0.0026	-0.0027
b(8)	-17.8815	-24.8375	0
b(9)	-0.4137	-0.3277	-0.3825
b(10)	0.9163	0.7111	0.8198
b(11)	0.2762	0.1326	0.2852

	Ridge	Lasso Regression
Opt Lambda	51.2563	0.0085
Min error	6.7682	0.4244

Explanation:

For the OLS, I did the optimization of the following formula:

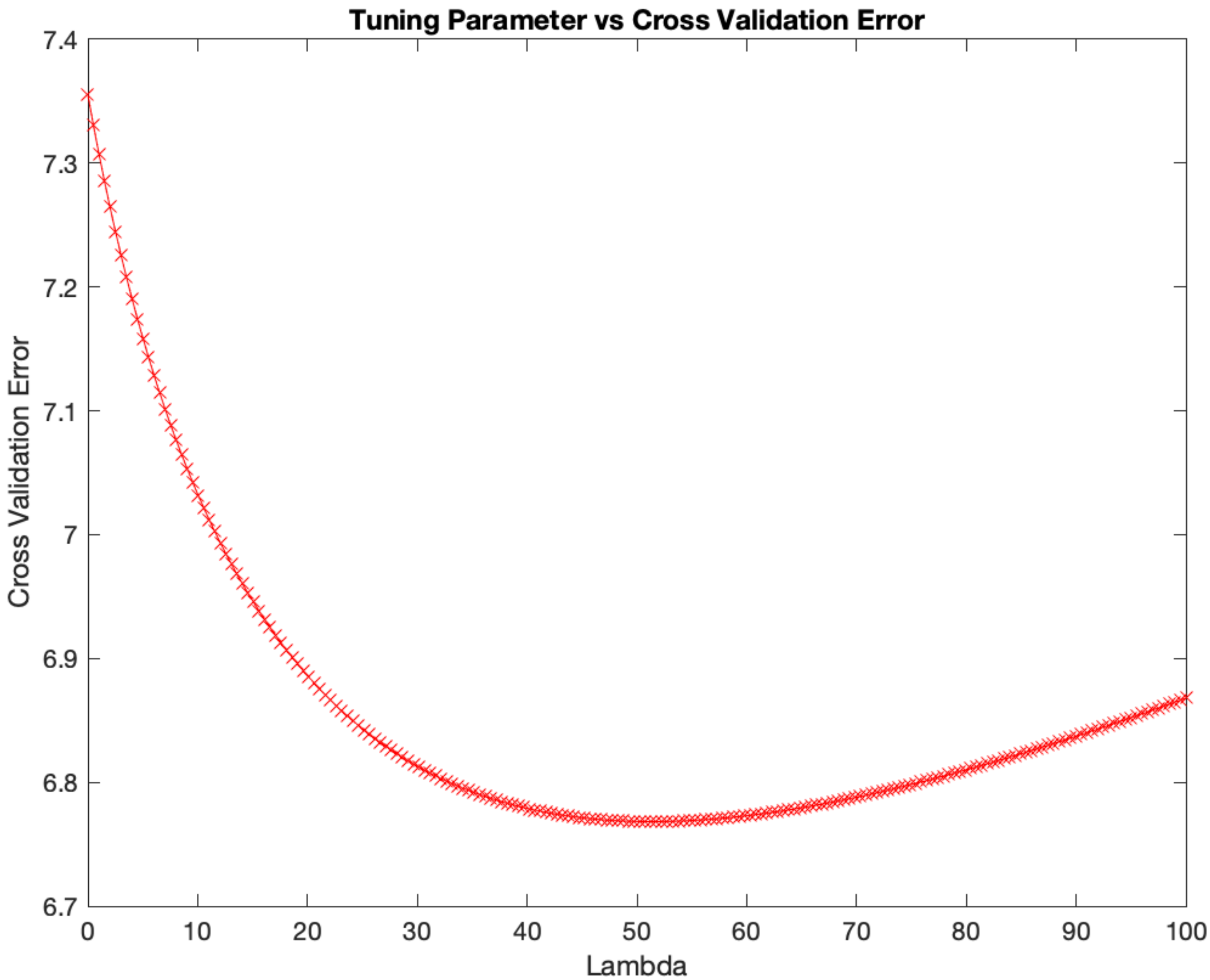
$$\begin{bmatrix} \widehat{\beta_0} \\ \hat{\beta} \end{bmatrix} = arg \min_{\beta_0, \beta} \left\| Y - [\mathbf{1}_n \quad X] \begin{bmatrix} \beta_0 \\ \beta \end{bmatrix} \right\|_2^2$$

I used CVX software in MATLAB to solve the minimization problem. As a result, the $\widehat{\beta_0}$ and $\hat{\vec{\beta}}$ can be obtained. In this case, both the $\widehat{\beta_0}$ and $\hat{\vec{\beta}}$ are fixed values, no variability occurs since there is no tuning parameter.

For the Ridge Regression, I used leave k-out cross validation to obtain the tuning parameter (λ), where k=100. Also, I used ridge function in MATLAB to compute the optimal lambda. I did Ridge Regression many times and finding different optimal lambdas, there are huge amount of variability on the optimal lambda. So, I decided to pick the optimal lambda that is closest to the OLS result.

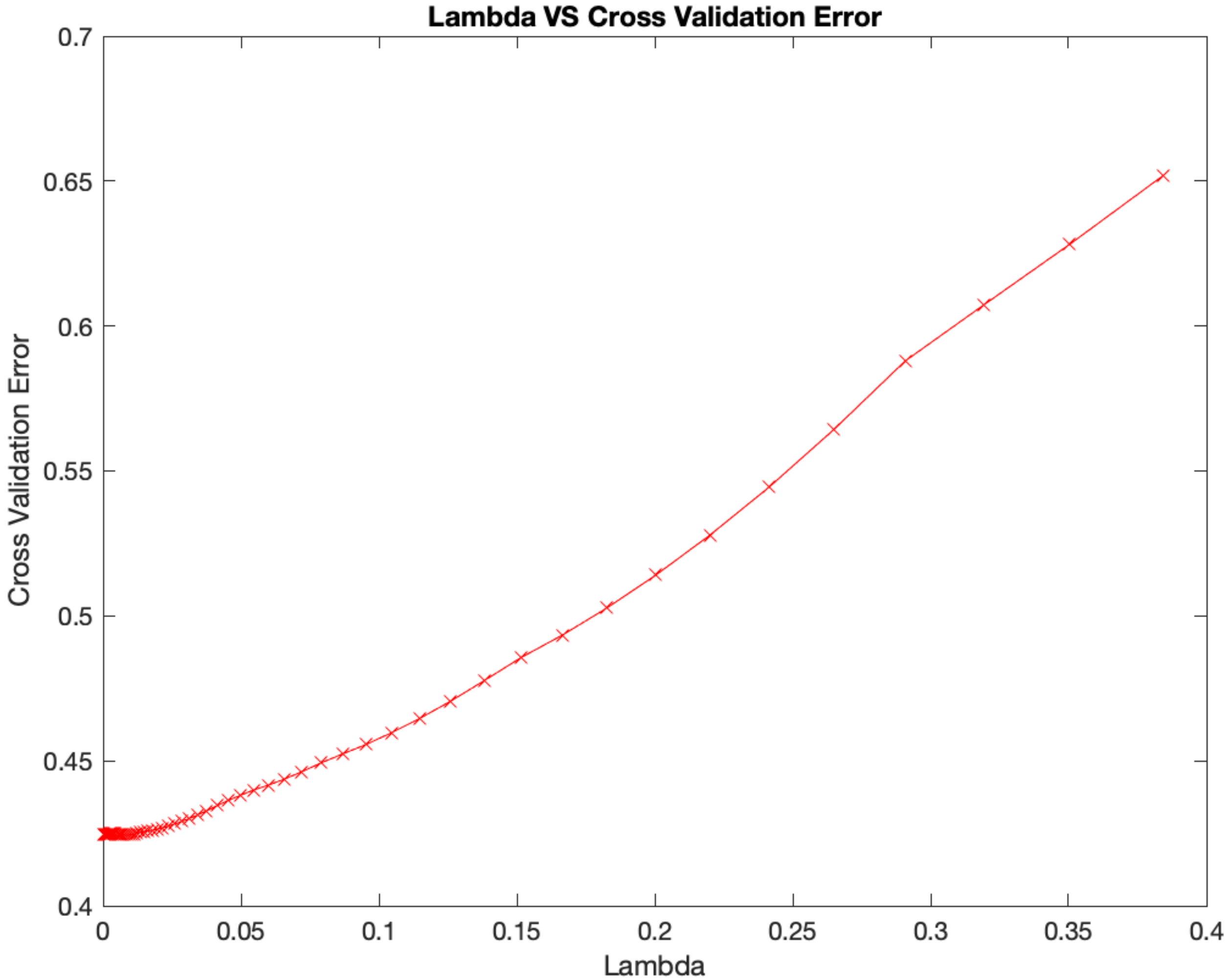
For the lasso regression, I used the lasso function in MATLAB. Lasso function in MATLAB used k-fold cross validation, where k is a factor of data size of X (k=39, 39 is a factor of 1599). I tried to do lasso regression using the function in MATLAB many times, and there is only little to no variability on the optimal λ. So, I decided to pick the optimal λ and betas as indicated in the table. My interpretation is that the result of betas from Lasso regression is very far compared to OLS and Ridge Regression. This is because the feasible set of the Lasso regression is pointy, meaning the beta for Lasso is located at the corner of the feasible set. This explains why there is a significance difference on the betas between Lasso regression and other type of regression in this project.

- Ridge Regression plot



From this graph, the optimal lambda is clear since there is a clear minimum value. So, the optimal lambda in this case is not too big nor too small. The optimal lambda in the Ridge Regression should fit the data that we have.

- Lasso plot



In this Lasso plot, the location of the minimum (the optimal lambda) is not clear. The optimal lambda in this case is almost zero. I tried to repeat the lasso regression many times using the Lasso function in MATLAB, and it only generates little variability. On the left side of the graph, the graph is flat (plateau) then the graph increases significantly onwards. This suggest that smaller lambda will not help, and optimal lambda should be bigger than the optimal lambda generated using Lasso regression.

Source code (MATLAB):

Only one MATLAB file required for my MATLAB code.

```
%% Project IEOR 165, William Wijaya

%% Begin OLS
disp('Press enter to start OLS:');pause;clc;clear all;

data=readtable('winequality-red.csv');

bt=[];
b0t=[];

X=data{:,1:11}; %X Matrix
Y= data.quality; %Y matrix
bold_one=ones(length(Y),1); %bold one matrix
Xm=[bold_one,X]; % Concatenating bold_one and X matrix

cvx_setup % Use cvx to solve, since objective function is convex
cvx_begin
variables b(11) b0;
minimize (sum_square(Y-Xm*[b0;b]))
cvx_end

fprintf('\nFinal answer:\n')
display(b)
display(b0)

%% Begin RR

disp('Press enter to start RR:');pause;clc;clear all;

data=readtable('winequality-red.csv');

k=100; %Use Leave k-out cross validation

lambda=linspace(0,100,200);
error=zeros(1,length(lambda)); %Setting ej=0 for all j

random=randperm(1599,k);
randomC=setdiff(1:1599,random);

data_1=data(random,:); % k data points
data_2=data(randomC,:); % n-k data points

X=data_1{:,1:11};
Y=data_1{:,12};

for j=1:length(error)
    beta=ridge(Y,X,lambda(j),0);
    Ypredict=[ones(length(randomC),1),data_2{:,1:11}]*beta;
```

```
    error(j)=error(j)+(1/k)*(sum((data_2{:,12}-Ypredict).^2));
end

[value,idx]=min(error);
mu=lambda(idx); % Optimal lambda
plot(lambda,error,'R-x')
xlabel('Lambda');
ylabel('Cross Validation Error');
title('Tuning Parameter vs Cross Validation Error');

X=data{:,1:11}; %X Matrix
Y= data.quality; %Y matrix
optbeta=ridge(Y,X,lambda(j),0); %Optimal beta

%% Begin Lasso

disp('Press enter to start Lasso:');pause;clc;clear all;

data=readtable('winequality-red.csv');
X=data{:,1:11}; %X Matrix
Y= data.quality; %Y matrix

k=39; %k fold cross validation,k is a factor of 1599

[B,Info] = lasso(X,Y,'CV',k);

lambda = Info.LambdaMinMSE; % Optimal lambda with minimum MSE
min_error = Info.MSE(Info.IndexMinMSE); % Minimal error
b = B(:,Info.IndexMinMSE); % Beta value

% Including b0 in the solution of beta
b_sol=zeros(1,12)';
b0 = b(1) + Info.Intercept(Info.IndexMinMSE);
b_sol=[b0;b];

%plot
plot(Info.Lambda,Info.MSE,'R-x');
title('Lambda VS Cross Validation Error');
xlabel('Lambda');
ylabel('Cross Validation Error');

fprintf('\nIEOR 165 project, finished!\n')
```