Homework 4

Construct a linear model that predicts the quality of a bottle of wine based on the following features:

1 - fixed acidity
2 - volatile acidity
3 - citric acid
4 - residual sugar
5 - chlorides
6 - free sulfur dioxide
7 - total sulfur dioxide
8 - density
9 - pH
10 - sulphates
11 - alcohol
Output variable (based on sensory data):
12 - quality (score between 0 and 10)

You must turn in a screen shot showing the results of linear regression using the following steps:

1. Download the data from http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/
2. Construct and evaluate a separate model for both red and white wines. Specifically, I want you to report the cross-validation R2 value. To do this you must create a driver program that:
    a. Loads/rearranges the data into the proper format. I suggest using the following command as an example: `ds = dataset('File','winequality-red.csv','delimiter',';');`
    b. This function returns a dataset object that has a lot of useful functionality. Here are some commands that might be useful. If you want to know what variables are available, type `ds.Properties.VarNames`
    c. So if you want to construct a X matrix using two of the variables, you can use the following command:
    ```
    X = [ds.fixedAcidity ds.volatileAcidity];
    ```
    d. Likewise you can construct a y vector:
    ```
    y = ds.quality
    ```
    e. Construct a X matrix using all of the features (except quality of course), and then construct a linear model using LinearModel.fit
    ```
    model = LinearModel.fit(X,y)
    ```
    f. Just like classification, we need to evaluate this on data that is hasn't seen yet, so we need cross-validation. To do this you'll need to use the following commands:
    ```
    cp = cvpartition(length(y),'k',10);
    cvMSE = crossval('mse',X,y,'predfun', @doregression,'partition',cp)
    cvR2 = 1 - cvMSE/mean((y - mean(y)).^2)
    ```
    g. But in order to run this code you'll have to define a function called doregression that looks like the following:
    ```
    function ypredicted = doregression(xtrain, ytrain, xtest)
        model = LinearModel.fit(xtrain,ytrain); % Create the model
        ypredicted = model.predict(xtest); % Run prediction on our training data
    end
    ```
3. I want you to turn in a screen shot(s) showing the code that you created and the results for both types of wine.