# Linear Regression on Abalone DataSet from scratch-

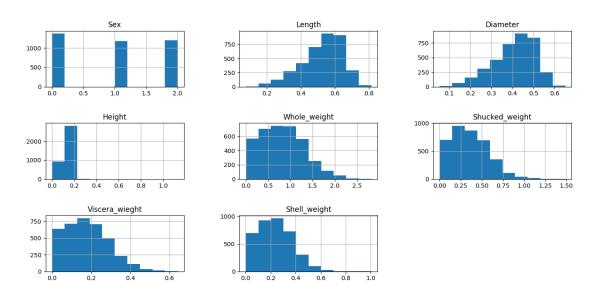
# PROBLEM STATEMENT -

Predicting the age of abalone from physical measurements. The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope -- a boring and time-consuming task. Other measurements, which are easier to obtain, are used to predict the age. Further information, such as weather patterns and location (hence food availability) may be required to solve the problem. So we will be implementing Simple linear Regression, Ridge Regression and Lasso Regression from scratch. Further using K—fold for getting approximate of true error rate.

### Data Set - http://www.cs.toronto.edu/~delve/data/abalone/desc.html

There are 9 columns in dataset. We firstly add the column names in Dataset.data file. We used columns names mentioned in Dataset.spec file. Then using pandas, we have read the dataset. We are using first 8 columns as input variables (X) to predict the last column (y). Out of 8 input features, 1st feature is of type string and rest 7 features are of floating data type. 1st column represents sex, which have only three classes {M,F,I}. We converted this column to float by mapping - M to 0, F to 1 and I to 2.

After this, we have splitted our data set into 90% (for training + validation)[ $X_{train}$ ,  $y_{train}$ ] and 10% (for testing)[ $X_{test}$ ,  $y_{test}$ ] using scikit-learn. We then visualize various attributes of  $X_{train}$  and we get the following graphs -



We can see some of attributes are not scaled properly, so it requires normalization.

### Gradient descent -

We have defined function for gradient descent in which we have implemented its functionality from scratch. We have also customized this function to take into account for L1 and L2 regularization.

### Part a) -

Now, let's move to first part -

- 1. Firstly, we have used KFold implementation of scikit-learn to do 5 splits.
- 2. For each of 5 splits, we are using 4 splits for training and 1 spilt as validation set.
- 3. Now, we will perform following for each of val set -
  - 3.1. In training set we are performing normalization, using formula: X -min/max
  - 3.2. We then store the min, max value of above.
  - 3.3. We initialize our parameters as 9x1 vector equal to 1.(We have also added x0 = 1 in X\_train).
- 3.4. We decide some learning rate and no. of iterations and perform gradient descent to minimize parameter.
  - 3.5. We get some parameters from gradient descent.
  - 3.6. We normalize validation set using values of step 3.2. and formula 3.1.
  - 3.7. We calculate the RMSE on validation set and see the result.

We use step 3 to tune the hyperparametes, to get minimum avg RMSE on val sets.

Finally, we come to conclusion that

learning rate = 0.2

and iterations = 200.

For above iteration vs RMSE graph is as follows -

#### Iterations vs RMSE graph for different folds for linear regression

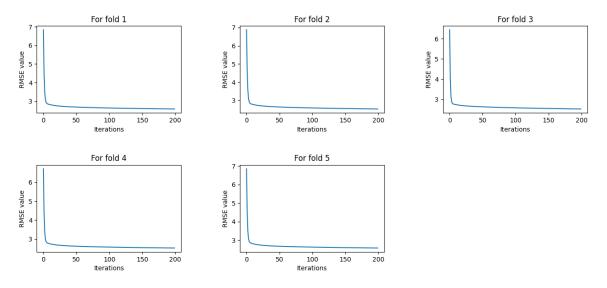


Fig. Iteration vs RMSE graph for linear regression for different folds

RMSE values on validation set is as follows -

```
File Edit Tabs Help

For Linear regression, RMSE value on fold 1 validation set is: 2.4521359054929794

For Linear regression, RMSE value on fold 2 validation set is: 2.668021658014849

For Linear regression, RMSE value on fold 3 validation set is: 2.4802293373337166

For Linear regression, RMSE value on fold 4 validation set is: 2.7190132194062277

For Linear regression, RMSE value on fold 5 validation set is: 2.7190132194062277

For Linear regression, RMSE value on fold 5 validation set is: 2.7190132194062275

For Linear regression, Average RMSE value on fold 1 val set is: 2.4882203631612833

For Linear regression with Li reg, RMSE value on fold 1 val set is: 2.488203631612833

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.488203631612833

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.6616337769790566

For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.7892018038649593

For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.780718038649593

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.566168395794643

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.394288845816699

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.497212336480969

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.497212336480969

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.49721336480969

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.49721336480969

For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.49721336480969

For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.497210336480969

For Linear regression with Li reg, RMSE value on testing set is: 2.77147807470472

By using sklearn Linear Regression on fold 4 very set is: 2.77147867470472

By using sklearn Linear Regression on fold 4 very set RMSE: 2.1259307950766

By using sklearn Linear Regression (Li) on fold 1 we get RMSE: 2.1208459705862
```

Fig. Follow first five lines, which corresponds to Linear regression

# Part b) -

In this we follow the same procedure as part a. In step 3.4 use gradient descent of L1 and L2 respectively.

At the end of tuning, we come to conclusion that

# <u>For L1 -</u>

learning rate = 0.2

iterations = 200

regularization parameter(lambda) = 0.01

# For L2 -

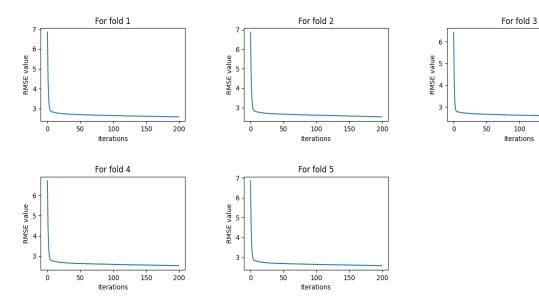
learning rate = 0.2

iteration = 200

regularization parameter(lambda) = 0.005

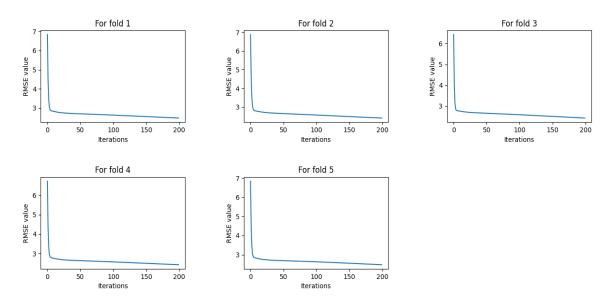
For L1 iteration vs RMSE graph is as follows -

Iterations vs RMSE graph for different folds for linear regression with L1 regularization



# For L2 iteration vs RMSE graph is as follows -

Iterations vs RMSE graph for different folds for linear regression with L2 regularization



### RMSE values on validation set is as follows -

```
For Linear regression with L1 reg, RMSE value on fold 1 validation set is: 2.4521359054929794
For Linear regression, RMSE value on fold 2 validation set is: 2.660021638016889
For Linear regression, RMSE value on fold 3 validation set is: 2.4802293733373796
For Linear regression, RMSE value on fold 4 validation set is: 2.780229373373796
For Linear regression, RMSE value on fold 5 validation set is: 2.790120120450257
For Linear regression, RMSE value on fold 5 validation set is: 2.491281210550225
For Linear regression with L1 reg, RMSE value of all folds is: 2.562513251367399

For Linear regression with L1 reg, RMSE value on fold 1 val set is: 2.4589206361612833
For Linear regression with L1 reg, RMSE value on fold 2 val set is: 2.46892706361612833
For Linear regression with L1 reg, RMSE value on fold 4 val set is: 2.560719391855
For Linear regression with L1 reg, RMSE value on fold 4 val set is: 2.7802018038649593
For Linear regression with L1 reg, RMSE value on fold 4 val set is: 2.506719391855
For Linear regression with L1 reg, RMSE value on fold 5 val set is: 2.506719391855
For Linear regression with L2 reg, RMSE value on fold 1 val set is: 2.497212356480969
For Linear regression with L2 reg, RMSE value on fold 3 val set is: 2.497212356480969
For Linear regression with L2 reg, RMSE value on fold 4 val set is: 2.497212356480969
For Linear regression with L2 reg, RMSE value on fold 3 val set is: 2.49818681869
For Linear regression with L2 reg, RMSE value on fold 4 val set is: 2.49818681869
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.49818681869
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.498186818689
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.4981868186888
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.7988236613901638
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.7988236613901638
For Linear regression with L2 reg, RMSE value on fold 5 val set is: 2.2988236613901638
For Linear regression
```

Fig. Follow linear regression with L1 and L2 on validation set

# Part c)-

Using parameters founded out in part(a) and part(b), we then trained the models on 90% data (train+val) and test them on testing set.

We got the following RMSE values –

```
For Linear regression, RMSE value on fold 1 validation set is: 2.4521359054929794
For Linear regression, RMSE value on fold 2 validation set is: 2.66021658014849
For Linear regression, RMSE value on fold 3 validation set is: 2.6802293373337196
For Linear regression, RMSE value on fold 4 validation set is: 2.71804293773337196
For Linear regression, RMSE value on fold 5 validation set is: 2.71804293773337196
For Linear regression, RMSE value on fold 5 validation set is: 2.718042937733799
For Linear regression, Average RMSE value of all folds is: 2.562513251367399

For Linear regression with Li reg, RMSE value on fold 1 val set is: 2.488203631612833
For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.69870100864001
For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.69870100864001
For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.7892018038649593
For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.7992018038649593
For Linear regression with Li reg, RMSE value on fold 5 val set is: 2.506719394491885
For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.48828981891989
For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.497212356489699
For Linear regression with Li reg, RMSE value on fold 2 val set is: 2.497212356480969
For Linear regression with Li reg, RMSE value on fold 3 val set is: 2.46639981891174344
For Linear regression with Li reg, RMSE value on fold 3 val set is: 2.39823861890183
For Linear regression with Li reg, RMSE value on fold 4 val set is: 2.46639981891174344
For Linear regression with Li reg, RMSE value on fold 5 val set is: 2.79721235648991891774344
For Linear regression with Li reg, RMSE value on fold 5 val set is: 2.798238618901638
For Linear regression on fold 5 value set is: 2.714478525392836
For Linear regression with Li reg, RMSE value on fold 5 value set is: 2.79828618991891749843
For Linear regression with Li reg, RMSE value with regression with Li reg, RMSE value with regression with
```

Fig. Follow RMSE values on testing set for Linear regression, Linear regression +L1, Linear regression+L2

We can see that linear regression + L2 is performing slightly better with these set of parameters.

### Part d) -

In this part, we perform the same steps as part(a) and part(b), but here we will use inbuilt libraries to train the models.

Inbuilt sklearn linear regression does not require any parameters.

By testing on validation set, we set

Parameter for lasso regression as 0.01 and

Parameter for ridge regression as 0.05.

This is result of three models on validation set -

```
File Edit Tabs Help

For Only Linear regression, RMSE value on testing set is: 2.711478525392836

For Linear regression + Li, RMSE value on testing set is: 2.7714476170472

By using sklearn Linear Regression on fold 1 we get RMSE: 2.127930147301517

By using sklearn Linear Regression on fold 2 we get RMSE: 2.125730147301517

By using sklearn Linear Regression on fold 3 we get RMSE: 2.125730147301517

By using sklearn Linear Regression on fold 4 we get RMSE: 2.19253201290417

By using sklearn Linear Regression on fold 4 we get RMSE: 2.19253201290417

By using sklearn Linear Regression on fold 4 we get RMSE: 2.305151064796437

By using sklearn Linear Regression on fold 5 we get RMSE: 2.305151064790437

By using sklearn Linear Regression on fold 5 we get RMSE: 2.1901360247328

By using sklearn Linear Regression (II) on fold 1 we get RMSE: 2.2095653628370892

By using sklearn Lasso Regression(II) on fold 2 we get RMSE: 2.209365922086832

By using sklearn Lasso Regression(II) on fold 2 we get RMSE: 2.209365922086832

By using sklearn Lasso Regression(II) on fold 3 we get RMSE: 2.20836390295

By using sklearn Lasso Regression(II) on fold 5 we get RMSE: 2.208369390295

By using sklearn Ridge Regression(II) on fold 2 we get RMSE: 2.208369390295

By using sklearn Ridge Regression(II) on fold 2 we get RMSE: 2.20836941871522

By using sklearn Ridge Regression(I2) on fold 2 we get RMSE: 2.20365483920812

By using sklearn Ridge Regression(I2) on fold 4 we get RMSE: 2.20365483920812

By using sklearn Ridge Regression(I2) on fold 4 we get RMSE: 2.20365483920812

By using sklearn Ridge Regression(I2) on fold 5 we get RMSE: 2.30437995074076

By using sklearn Ridge Regression(I2) on fold 5 we get RMSE: 2.3063673995074076

By using sklearn Ridge Regression(I2) on fold 5 we get RMSE: 2.3663607790507473

For Linear regression in closed form, RMSE value on fold 1 val set is: 2.19273201473015133

For Linear regression in closed form, RMSE value on fold 5 val set is: 2.1527301473015133

For Linear regression in closed form, RMSE value
```

Fig. Follow Sklearn Linear regression, Sklearn Ridge Regression and Sklearn Lasso Regression on validation set

With parameters describe earlier, we then trained the model on validation set and tested it on testing set.

We got following RMSE values -

```
File Edit Tabs Help
For Only Lihear regression, RMSE value on testing set is: 2.711478525392836
For Linear regression * 1.1, RMSE value on testing set is: 2.17143461548934548
For Linear regression * 1.2, RMSE value on testing set is: 2.17143461548934548
For Linear regression * 1.2, RMSE value on testing set is: 2.17143461548934548
For Linear regression * 1.2, RMSE value on testing set is: 2.17143461548934548
For Linear regression * 1.2, RMSE value on testing set is: 2.1914361549347
By using sklearn Linear Regression on fold 1 we get RMSE: 2.1293014770117
By using sklearn Linear Regression on fold 3 we get RMSE: 2.150151064799437
By using sklearn Linear Regression on fold 5 we get RMSE: 2.150151064799437
By using sklearn Linear Regression on fold 5 we get RMSE: 2.150151064799437
By using sklearn Lasso Regression(L1) on fold 2 we get RMSE: 2.150365028
By using sklearn Lasso Regression(L1) on fold 2 we get RMSE: 2.2350459922086832
By using sklearn Lasso Regression(L1) on fold 3 we get RMSE: 2.3050459922086832
By using sklearn Lasso Regression(L1) on fold 3 we get RMSE: 2.305046639425085
By using sklearn Lasso Regression(L2) on fold 5 we get RMSE: 2.20146439825085
By using sklearn Ridge Regression(L2) on fold 5 we get RMSE: 2.20146439825085
By using sklearn Ridge Regression(L2) on fold 2 we get RMSE: 2.201464398250812
By using sklearn Ridge Regression(L2) on fold 2 we get RMSE: 2.2036568483920812
By using sklearn Ridge Regression(L2) on fold 4 we get RMSE: 2.2036568483920812
By using sklearn Ridge Regression(L2) on fold 4 we get RMSE: 2.203656884329
By using sklearn Ridge Regression(L2) on fold 4 we get RMSE: 2.203656888329
By using sklearn Ridge Regression(L2) on fold 5 we get RMSE: 2.30510493989
By using sklearn Ridge Regression(L2) on fold 5 we get RMSE: 2.30510493989
By using sklearn Ridge Regression on testing set, we get RMSE: 2.3051049389
By using sklearn Ridge Regression on testing set, we get RMSE: 2.3051049389
By using sklearn Ridge Regression on testing set, we get RMSE: 2.3051049389
By using sklearn Rid
```

Fig. Follow Testing result of Sklearn Linear regression, Sklearn Ridge Regression and Sklearn Lasso Regression

We can see that only linear regression and linear regression+L2 is almost giving same RMSE values.

Comparing it with our above result part(d), we can see that there is slight difference in decimals. And inbuilt functions are slightly giving better RMSE values. This could be due to implementation of inbuilt functions. As they implement closed form solution and we are training with gradient descent. Also we are only doing 200 iterations in gradient descent.

#### Part e)-

Now for implementing closed form solution, we used formula derived in theory part of this assignment.

We get following RMSE value on validation sets -

```
File Edit Tabs Help

For Only Linear regression, RMSE value on testing set is: 2.711478525392836

For Linear regression * L1, RMSE value on testing set is: 2.7114786153934548

For Linear regression * L1, RMSE value on testing set is: 2.5777440746170472

By using sklearn Linear Regression on fold 1 we get RMSE: 2.125930147301517

By using sklearn Linear Regression on fold 2 we get RMSE: 2.265360779050746

By using sklearn Linear Regression on fold 3 we get RMSE: 2.199233201290417

By using sklearn Linear Regression on fold 4 we get RMSE: 2.199233201290417

By using sklearn Linear Regression on fold 5 we get RMSE: 2.190131620247328

By using sklearn Linear Regression on fold 5 we get RMSE: 2.150131620247328

By using sklearn Linear Regression on fold 5 we get RMSE: 2.1806497005386228

By using sklearn Linear Regression (L1) on fold 1 we get RMSE: 2.1836499705586228

By using sklearn Lasso Regression (L1) on fold 2 we get RMSE: 2.1200463920086802

By using sklearn Lasso Regression (L1) on fold 4 we get RMSE: 2.19084015771

By using sklearn Lasso Regression (L1) on fold 4 we get RMSE: 2.210469910375

By using sklearn Lasso Regression (L2) on fold 4 we get RMSE: 2.210469910935

By using sklearn Lasso Regression (L2) on fold 4 we get RMSE: 2.210469910935

By using sklearn Ridge Regression (L2) on fold 4 we get RMSE: 2.2210469910935

By using sklearn Ridge Regression (L2) on fold 4 we get RMSE: 2.22104991645

By using sklearn Ridge Regression (L2) on fold 3 we get RMSE: 2.203643940076

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.203643940076

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.103937996109908

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.103937996109908

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.103937996109908

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.103937996109908

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE: 2.10393799710776

By using sklearn Ridge Regression (L2) on fold 5 we get RMSE
```

Fig. Follow the last set of results for closed form solution

Note that, result we got almost same result for this as we got by using scikit-learn linear regression.

This is because of the fact that scikit-learn implements closed form solution in its implementations.

# For RUNNING python code -

Code.py file should be executed for this part.

Dataset - should be manipulated as discussed at start of this question and should be placed in same folder as code.

There are functions calls at the end of the file Code.py, for each part. Uncommenting any part will not run that part.