Indian Institute of Technology Ropar

CSL-603

Linear and Logistic Regression

Submitted To:
Narayanan C Krishnan
Computer Science
Department

Submitted By :
Siddharth Nahar
2016CSB1043
Group-G1
5th Semester

Contents

1 Linear regression Model and Analysis

This is Report for Linear Regression model through Analytical Solution.In this Assignment we have to predict Age of Albalone.Features used in this Dataset are taken from http://archive.ics.uci.edu/ml/datasets/Abalone

From the Training Examples I have samples 20% as Test Set and Remaining is partitoned as necessary in Training and Validation Set.Training set is saved in linregdata.All Results and Analysis is mentioned below.

1.1 Basic Analysis of Linear Regression

This Analysis is based on Convergence of Cost Function by analytical Normal Equation.Cost Function used is Mean Square Error.

Let

$$J = CostFunction$$

$$X = FeatureMatrix(N * D), Y = LabeledData = (N * 1)$$

$$\lambda = RegularizationParameter$$

$$J = \sum_{i=1}^{N} (XW - Y)^2 + \lambda ||W||^2$$

$$W_{new} = W_{prev} - \alpha(X^T(XW_{prev} - Y) + \lambda W_{prev})$$

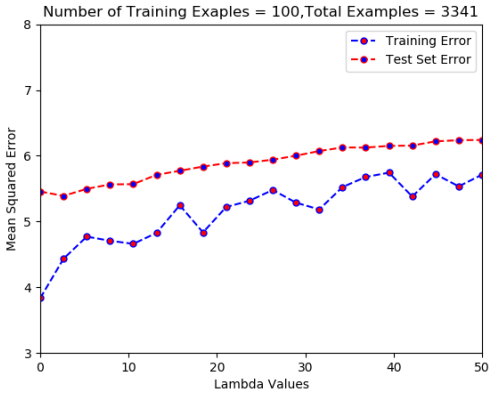
Analytical Solution :-

$$W = (X^T X + \lambda I)^{-1} X^T Y$$

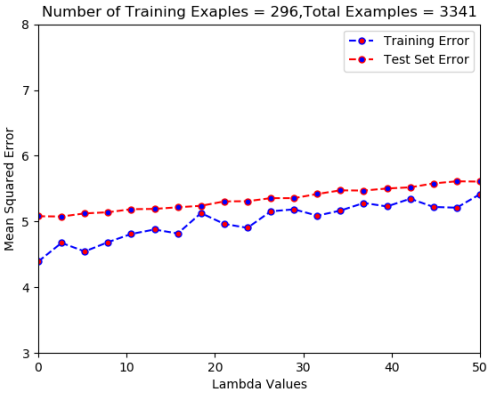
1.2 Approach Applied and Graphs Plotted

First I have created 100 fractions of Training Set and each fraction is trained 50 times for 20 λ values.

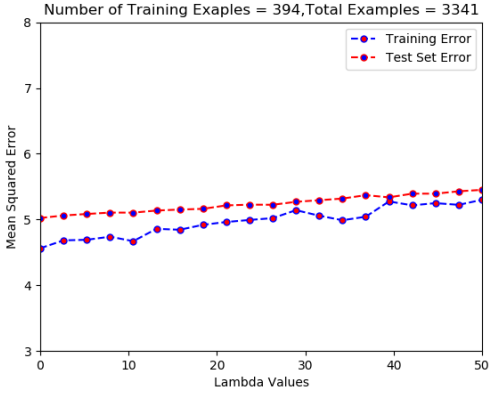
Graphs of particular Fraction Mean Square Error vs Lambda values :-



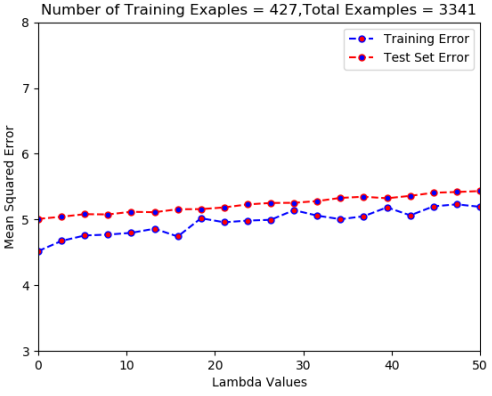
(a) Fraction : 0.03



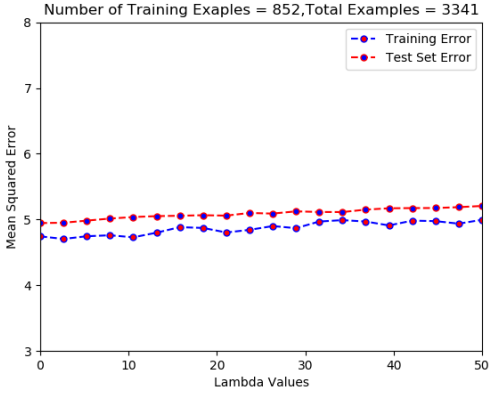
(b) Fraction : 0.0886



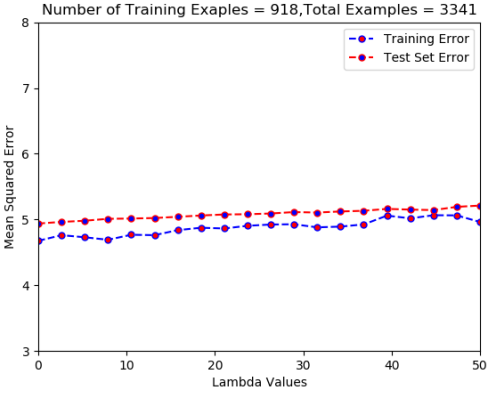
(c) Fraction : 0.118



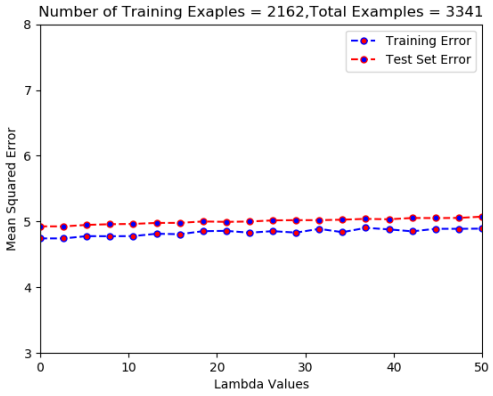
(d) Fraction : 0.128



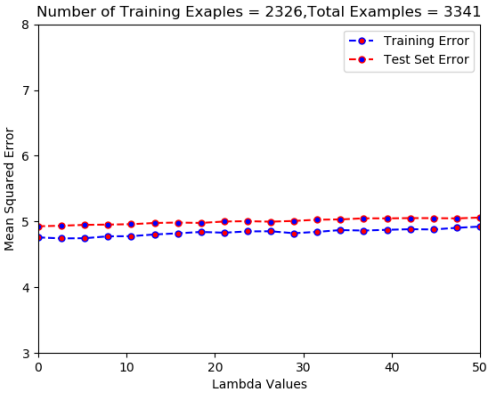
(e) Fraction : 0.255



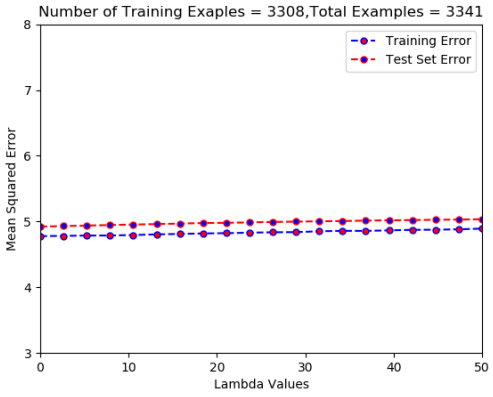
(f) Fraction : 0.2747



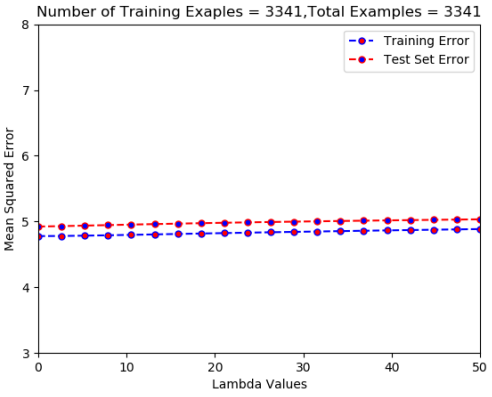
(g) Fraction : 0.6471



(h) Fraction : 0.707



(i) Fraction : 0.99



(j) Fraction : 1

Observations :

- We Observe that for Low fractions lambda change, change Error drastically.This is direct correlation of High Variance so Lambda tries to Regularize.
- As Fraction size increases Error almost becomes constant showing variance decreases with more training examples.
- We can also See that as lambda increases bias increases so training error increases for lambda

Graph for Minimum Squared Error and Lambda vs Fraction Values and Graph for Predicted vs Actual Values:

Observations :

- We Observe if we Train on higher fraction Sqaure Error Decreases, As Variance decreases with more Training examples.
- We can See Graph is Decreasing with Increasing Fractions.
- This follows that keeping Bias Constant and decreasing Variance increases Accuracy of our Model.
- We learn good model if Predicted vs Actual lies around 45 deg line

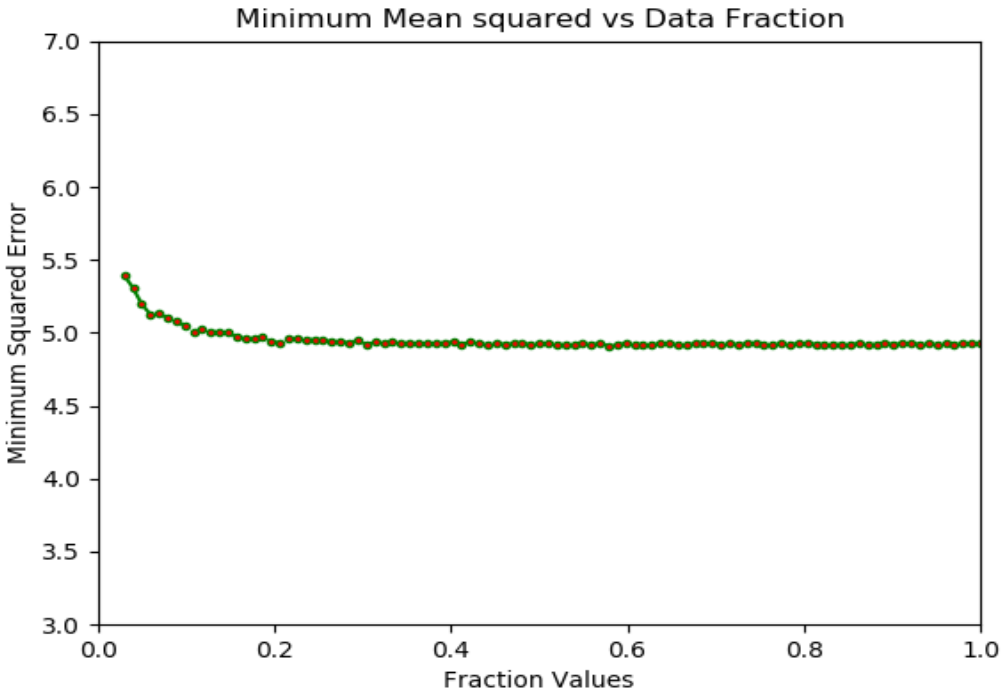


Figure 1: Part 8A : Mean Squared Error vs Fraction Values

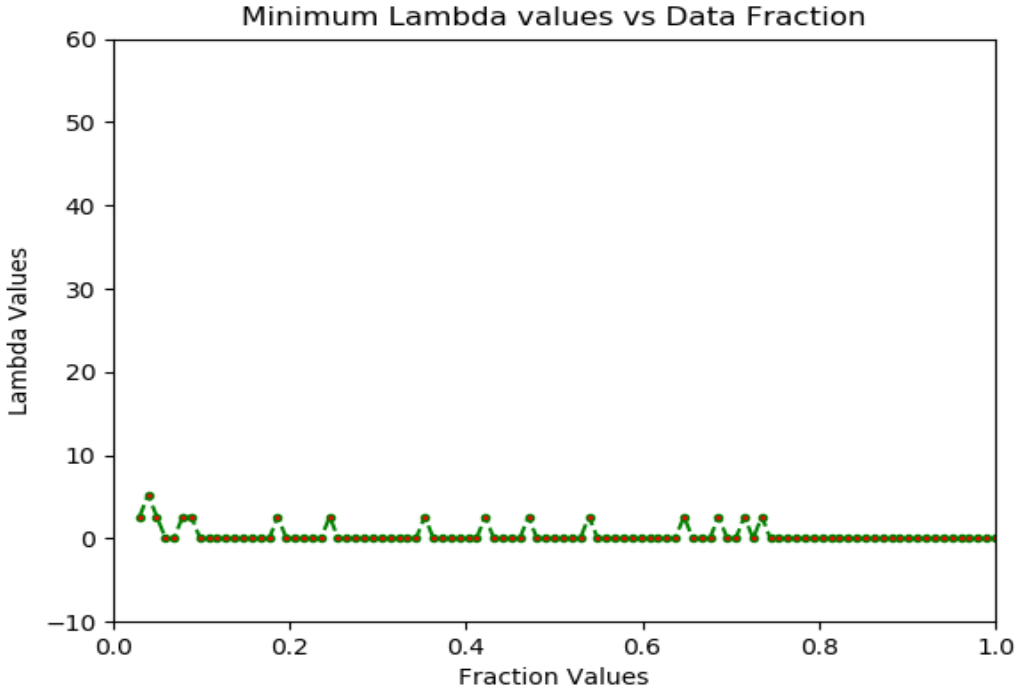
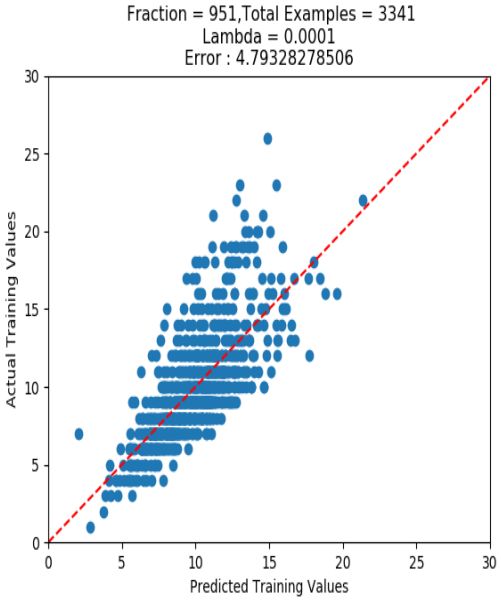
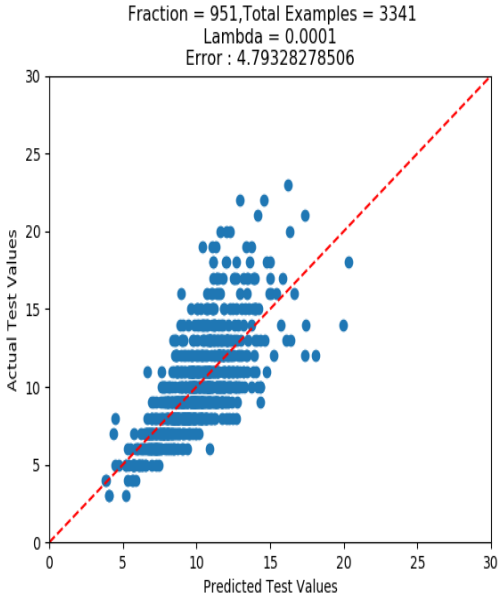


Figure 2: Part 8B : Best Lambda vs Fraction Values



(a) Part 9A : Training Predicted vs Actual



(b) Part 9B : Test Predicted vs Actual

2 Logistic Regression Model and Analysis

This is Analysis of Report of Logistic Regression model trained through Gradient Descent and Newton Raphson method.In this Assignment we have to classify two-dimensional data that based on feature to predict whether to give credit or not.

For Observation of Data I have plotted data to see what basis must be chosen to classify data.Then created a generic function for creating polynomial features and trained model for that degree.

2.1 Basic Analysis of Logistic Regression

This Analysis is based on Convergence of Cost Function by Gradient Descent and Newton Raphson.Cost Function log likleyhood.

Let Cost function = J,Feature Vector = X,Data Labelled = Y
Predicted Hypothesis = f and λ = Regularization Parameter.

$$f = \sigma(XW) = \frac{1}{1 + e^{-XW}}$$

$$J = -\sum_{i=1}^{N}(y_i \log(f_i) + (1 - y_i)\log(1 - f_i)) + \lambda||W||^2$$

For Gradient Descent :-

$$W_{new} = W_{prev} - \alpha(X^T(f - Y) + \lambda W_{prev})$$

For Newton Raphson :-

$$W_{new} = W_{prev} - (X^T R X + \lambda I_D)^{-1}(X^T(f - Y) + \lambda W_{prev})$$

R is Diagonal Matrix with :-

$$R_{ii} = f_i(1 - fi)$$

2.2 Approach Applied and Graphs Plotted

- Plot the Dataset for Representation of Data.
- Train model with Gradient Descent for various Iterations and compare with Newton Raphson.
- Plot the simple Decision Boundary
- Plot Polynomial Decision Boundary and Perform Overfitting and UnderFitting of Data

2.2.1 Plot the DataSet :-

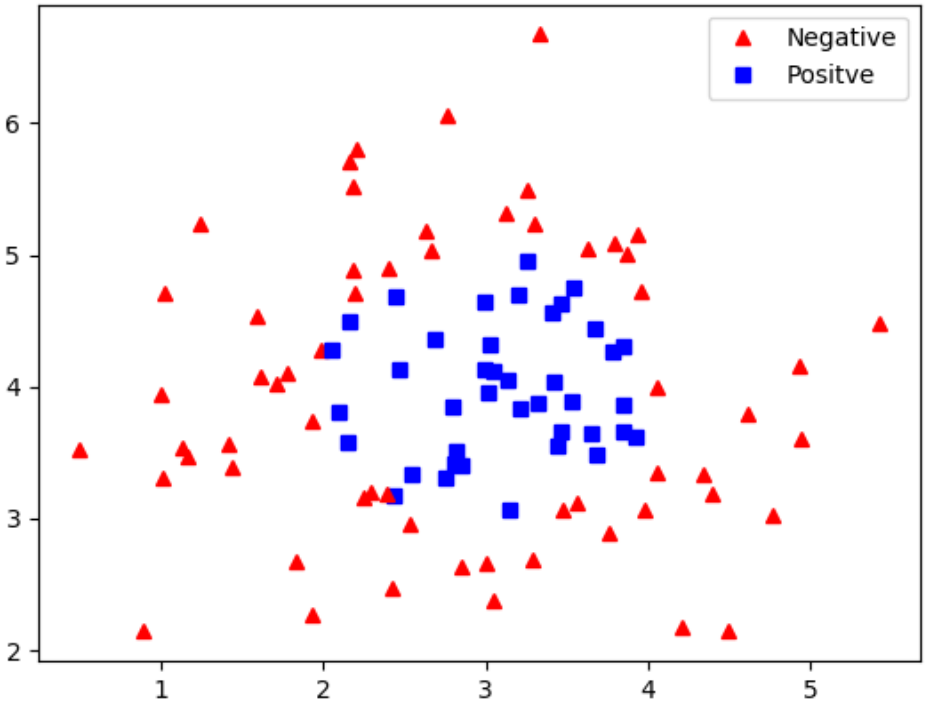
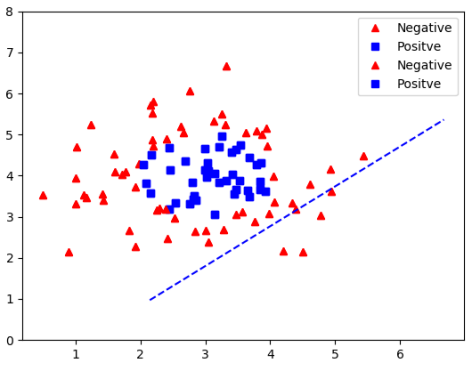


Figure 3: Part 1 : Representation of Classification Data

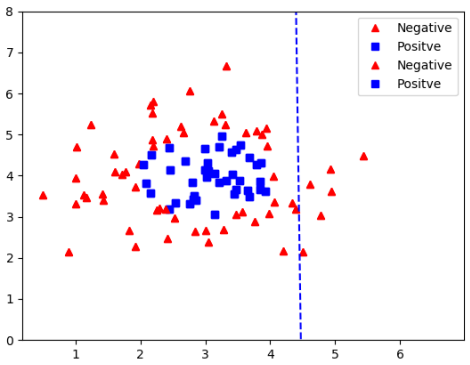
2.2.2 Training the Linear model and Compare Methods :-

For Gradient Descent :-

- Model 1 :- Error = 0.52 ,Iterations = 1000,α = 0.02 4(a)
- Model 2 :- Error = 0.50 ,Iterations = 10000,α 0.02 4(b)
- Model 3 :- Error = 0.45 ,Iterations = 100000,α = 0.02 4



(a) Iterations = 1000



(b) Iterations = 10000

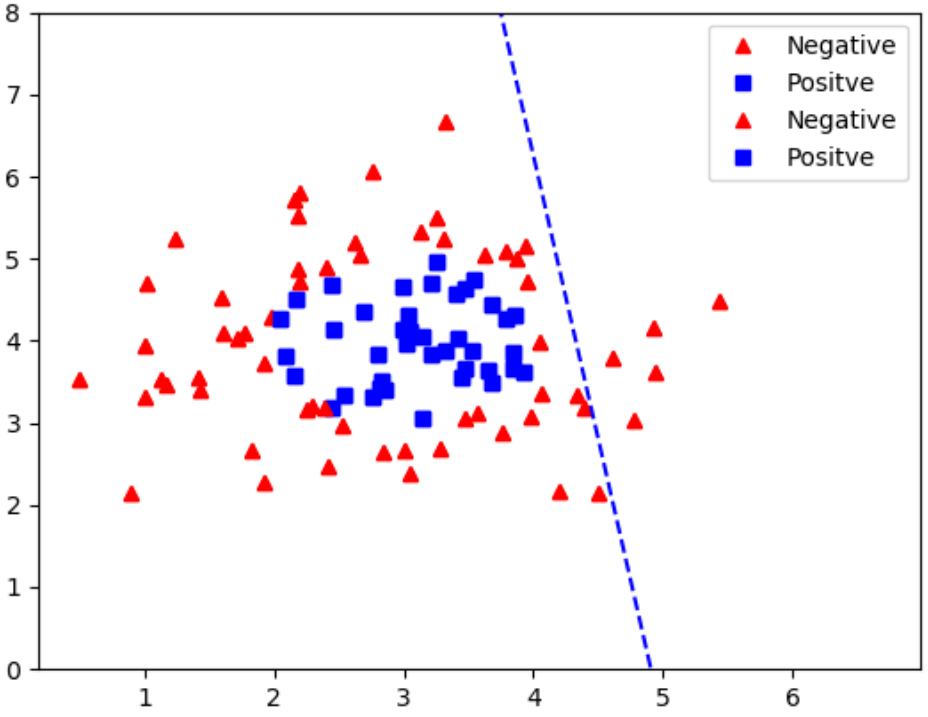


Figure 4: Iterations = 1000000

For Newton Raphson :-

Model :- Error = 0.45,Iterations = 10 5

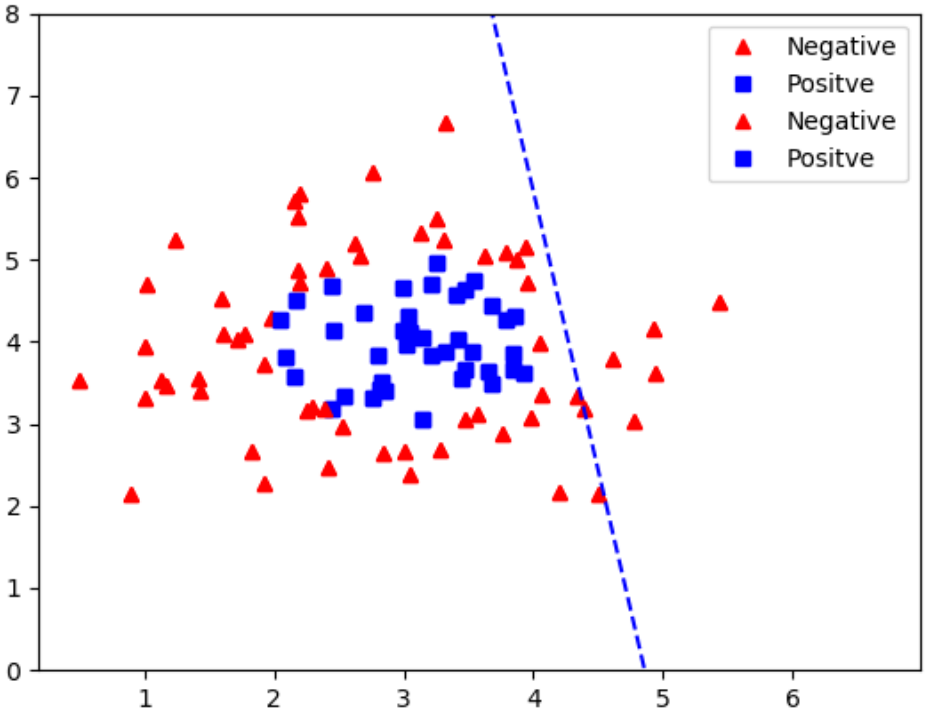


Figure 5: Newton Raphson Linear

Comparisons :-

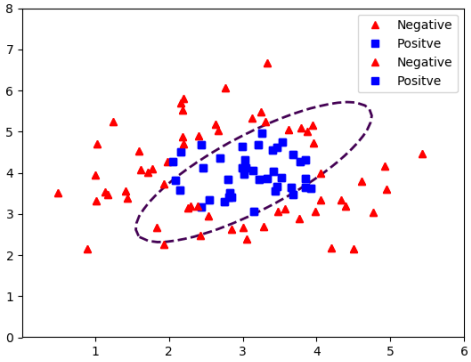
- We could clearly see Newton Raphson is much more faster than Gradient Descent and dependent on less factors.
- Newton Raphson is more dependent on Starting Parameters or it fails to converge to minima .
- Newton Raphson is more complicated requires more computation to calculate second derivative.
- So Gradient Descent is more Used in real life ML problems.

2.3 Training Model for Non-Linear Decision Boundary and Underfitting and Overfitting By change in Regularization Parameter :

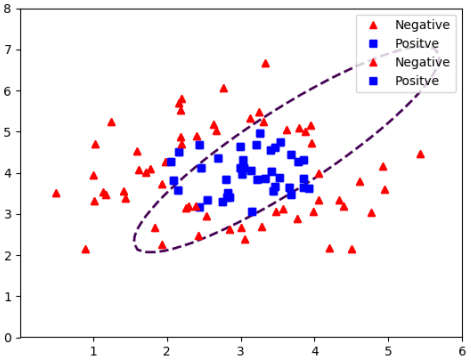
For Gradient Descent :-

- Degree - 2
  - Overfitt Model : Error : 0.17 , Iterations : 100000,$\alpha = 0.001$ $\lambda = 0$ 6(a)
  - Underfitt Model : Error : 0.22,Iterations : 100000,$\alpha = 0.001$ $\lambda = 1$ 6(b)
- Degree - 3
  - Overfitt Model : Error : 0.14 , Iterations : 100000,$\alpha = 0.001$ $\lambda = 0$ 6(c)
  - Underfitt Model : Error : 0.16,Iterations : 100000,$\alpha = 0.001$ $\lambda = 1$ 6(d)
- Degree - 4
  - Overfitt Model : Error : 0.1 , Iterations : 100000,$\alpha = 0.00006$ $\lambda = 0$ 6(e)
  - Underfitt Model : Error : 0.11,Iterations : 100000,$\alpha = 0.00006$ $\lambda = 1$ 6(f)

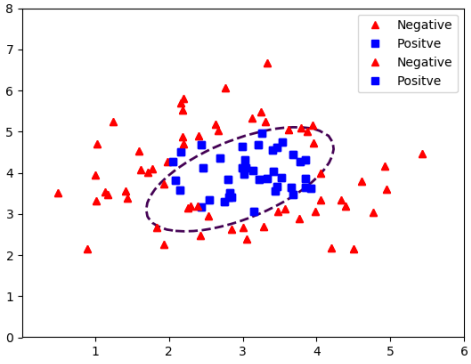
Plot of Decision Boundary are :-



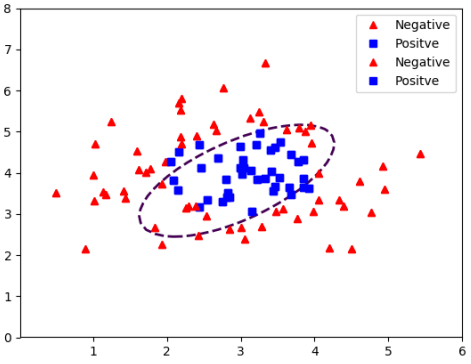
(a) Gradient Degree 2-Overfitt



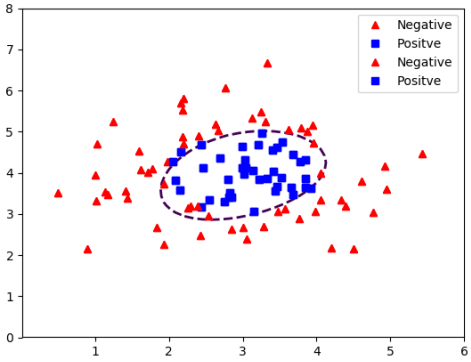
(b) Gradient Degree 2 - Underfitt



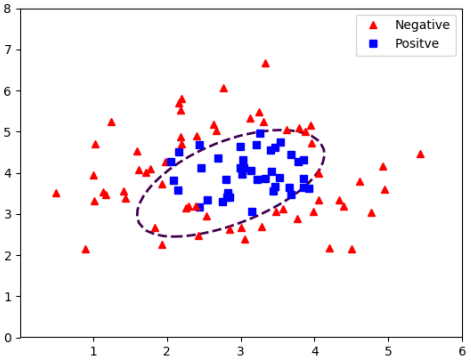
(c) Gradient Degree 3 - Overfitt



(d) Gradient Degree 3 - Undefitt



(e) Gradient Degree 4 - Overfitt

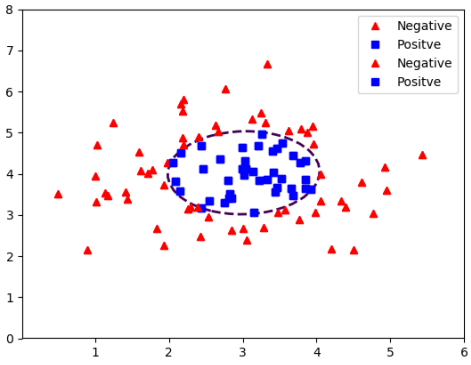


(f) Gradient Degree 4 -Underfitt

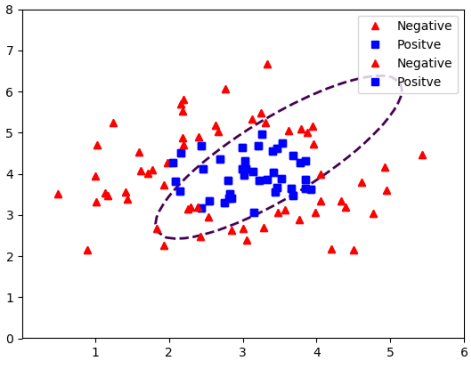
For Newton Raphson :-

- Degree - 2
  - Overfitt Model : Error : 0.0 , Iterations : 100,$\lambda = 0$ 6(g)
  - Underfitt Model : Error : 0.19,Iterations : 100 $\lambda = 1$ 6(h)
- Degree - 3
  - Overfitt Model : Error : 0.01 , Iterations : 100,$\lambda = 0$ 6(i)
  - Underfitt Model : Error : 0.05,Iterations : 100, $\lambda = 1$ 6(j)
- Degree - 4
  - Overfitt Model : Error : 0.0 , Iterations : 100,$\lambda = 0$ 6(k)
  - Underfitt Model : Error : 0.01,Iterations : 100,$\lambda = 1$ 6(l)

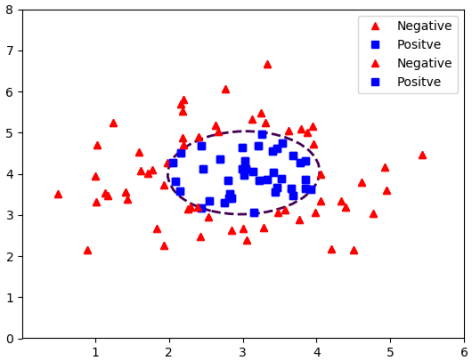
Plot of Decision Boundary are :-



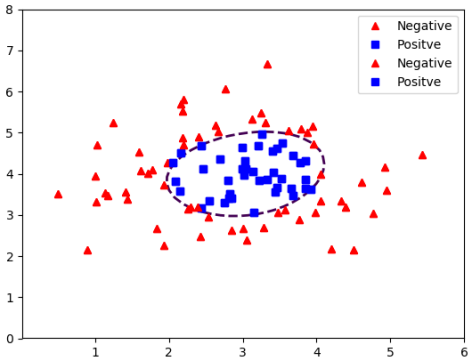
(g) Raphson Degree 2-Overfitt



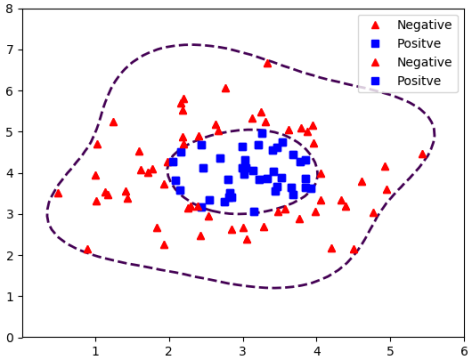
(h) Raphson Degree 2 - Underfitt



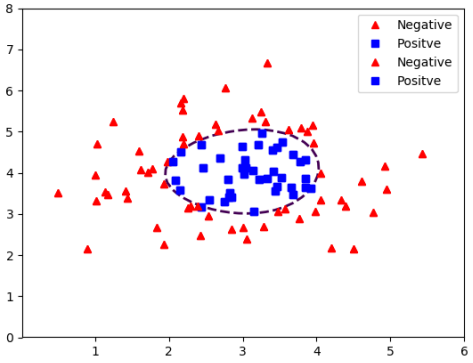
(i) Raphson Degree 3-Overfitt



(j) Raphson Degree 3 - Underfitt



(k) Raphson Degree 4-Overfitt



(l) Raphson Degree 4 - Underfitt