MACHINE LEARNING A-Z

January 9, 2020

Name: Tasnim Ferdous Dima

Regression

0.1 Regression

Regression models (both linear and non-linear) are used for predicting a real value, like salary for example. If your independent variable is time, then you are forecasting future values, otherwise your model is predicting present but unknown values. Regression technique vary from Linear Regression to SVR and Random Forests Regression.
The regression model that I learned are:

- Simple Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Support Vector for Regression (SVR)
- Decision Tree Regression
- Random Forest Regression

0.1.1 Simple Linear Regression

Works on any size of dataset, gives informations about relevance of features. Here we have Years of experiance and salary data of 30 employee. We apply simple linear regression to see the co relation of the data.

Figure 1: (Training Set)

Figure 2: (Testing Set)

0.1.2 Multiple Linear Regression

Works on any size of dataset, gives informations about relevance of features.Works on linear dataset very well. Here we have 50 data of startup companies about their RD spend,administrationnal spend, marketing spend, the staters they are located and we need to find the data how they are related with their profit. Here no graph can be blottesd as they have more then one Independent variable so we can't plot them on XYaxis.

0.1.3 Polynomial Regression

Works on any size of dataset, gives informations about relevance of features.Works on non-linear dataset very well. Here we have 10 data of a position of a person in their company, their level in that company and the salary according to their level and position.Our goal is to predict if the person telling truth about their salary when negotiating while joining. Here we apply both linear regression and polynomial regression on it to see if their is a better prediction on the graph outcome.

Figure 3: (Linear Regression)

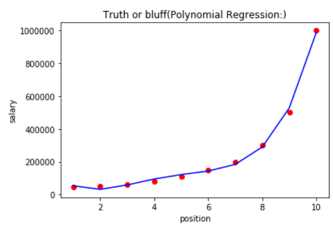

Figure 4: Polynomial in test set)

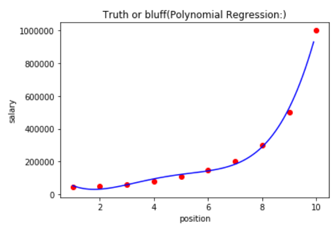

Figure 5: Polynomial in higher resolution

0.1.4 Support Vector for Regression (SVR)

Easily adaptable, works very well on non linear problems, not biased by outliers.Here we apply SVR on the dame data on which we apply the polynomial regression on it to see if their is a better prediction on the graph outcome.

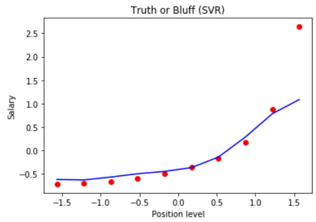

Figure 6: Truth or Bluff (SVR)

Figure 7: Truth or Bluff (SVR high resolution)

0.1.5 Decision Tree Regression

Interpretability, no need for feature scaling, works on both linear / nonlinear problem.Here we apply Decision Tree Regression on the dame data on which we apply the polynomial regression and svr on it to see if their is a better prediction on the graph outcome.

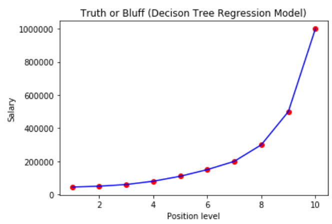

Figure 8: Truth or Bluff (Decison Tree Regression Model)

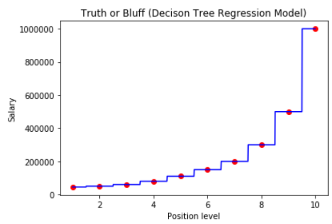

Figure 9: Truth or Bluff (Decison Tree Regression Model in high resolution)

0.1.6 Random Forest Regression

Powerful and accurate, good performance on many problems, including non linear.

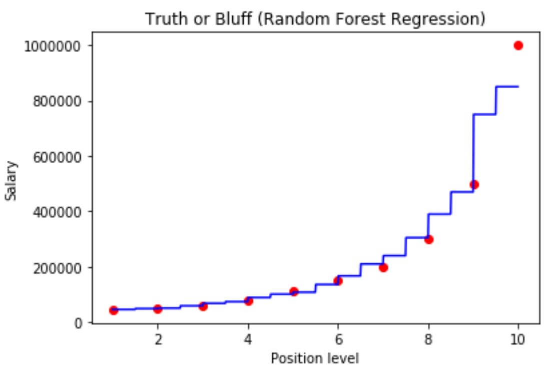

Figure 10: Truth or Bluff (Random Forest Regression Model)

0.2 PART B

Figure 11: change width,hight

Figure 12: change width, height

If we change the canvas width and height then the width and height are gettng bigger or smaller according to the size given.

Figure 13: change canvas size

Figure 14: change canvas size

The viewport takes 4 perameter . The Frist and second one takes the lower lest corner values of 'x' and 'y' axis. If we changes the first 2 perameter then it will shift accordingly the value given. The exis will shift fo the coordinate will be changed and as a result the point that we are drawing wil also change the location according to shifting coordinate system.

Figure 15: viewport on 0,0

Figure 16: viewport on 70,70

0.3 PART C

WE change the attachment serial but the codes still works as it doesn't change the main programme. Before using the program both of the parts are linked.

Figure 17: code works

Figure 18: code works

Figure 19: code doesn't works

Figure 20: empty canva

now we write the 'useprogram()' function before the codes are attached to the shader. Then the code does not work as we try to use the program before attaching the parts. As a result it shows empty canvas.

0.4 PART D

Here we are using the loops to get the points that satisfy the equation and feeding those points to the buffer. WE are wsing a very small incremental value so that it looks link a st line rather then some discreet dot poins. And the count variables are used so that the line goes from one certain position to another certain position. The value of m and c are calculated from the given coordinate value.

Figure 21: use loops to get points

Figure 22: st line