12/4/2020

DATA MINING & MACHINE LEARNING - ASSIGNMENT

Sean Gavin
A00251388

Table of Contents

Linear Regression

Polynomial Regression

Linear Regression

1.1 Business Understanding

The dataset which I used to explore Linear Regression is an [insurance dataset]() from Kaggle. My goal or primary objective from a business perspective is to help out an insurance company to see if they should charge a customer a premium or not for insurance. My plan is to take the customer details such as their age, bmi, children and their existing medical expense to predict future medical expenses to help medical insurance companies decide on whether or not to charge a premium.

1.2 Data Understanding & Preparation

The dataset insurance.csv includes 1338 rows and seven attributes (columns). Insurance.csv comprises of four numerical characteristics (bmi, age, kids and expenses) and three nominal characteristics (region, smoker, and sex) that have been changed into numerical value factors allocated for each class. The dataset is not missing any fields which means no preparation of the data has to be carried out at the moment.

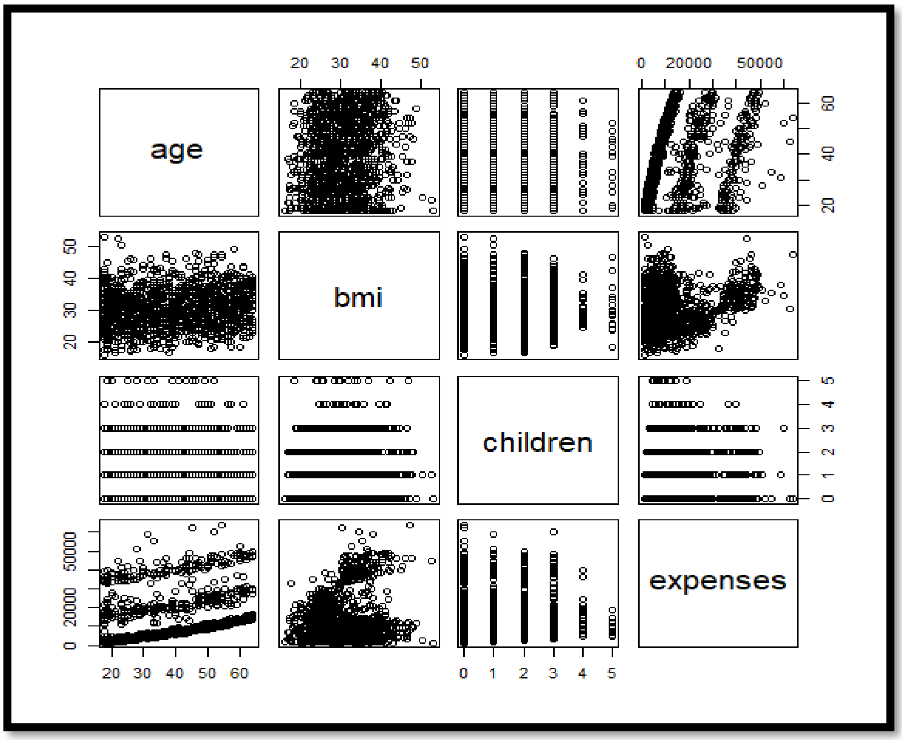
Summary of the dataset below:

```
> dataset <- read.csv("data/insurance.csv", stringsAsFactors = T)
> View(dataset)
> summary(dataset)
      age            sex           bmi           children       smoker          region        expenses    
 Min.   :18.00   female:662   Min.   :16.00   Min.   :0.000   no :1064   northeast:324   Min.   : 1122  
 1st Qu.:27.00   male  :676   1st Qu.:26.30   1st Qu.:0.000   yes: 274   northwest:325   1st Qu.: 4740  
 Median :39.00                Median :30.40   Median :1.000              southeast:364   Median : 9382  
 Mean   :39.21                Mean   :30.67   Mean   :1.095              southwest:325   Mean   :13270  
 3rd Qu.:51.00                3rd Qu.:34.70   3rd Qu.:2.000                              3rd Qu.:16640  
 Max.   :64.00                Max.   :53.10   Max.   :5.000                              Max.   :63770  
> 
```

I produced a correlation matrix to depict the relationship of four features in the dataset:

```
> cor(dataset[c("age", "bmi", "children", "expenses")])
               age        bmi   children   expenses
age      1.0000000 0.10934101 0.04246900 0.29900819
bmi      0.1093410 1.00000000 0.01264471 0.19857626
children 0.0424690 0.01264471 1.00000000 0.06799823
expenses 0.2990082 0.19857626 0.06799823 1.00000000
> 
```

I also produced a scatterplot matrix to further visualise the correlation between the four features in the dataset.



1.3 Modelling

How I plan to carry out Linear regression on the model is implementing expenses as the dependant variable and all the other values are independent factors withinside the model. I used the fitting linear model function to implement linear regression to the dataset. I assigned the value expenses as y just before the tilde and using "." as a notation to show the other features from the dataset as multiple independent variables and irrespective of numerical or categorical variables.

Original Model:

```
> model <- lm(expenses ~ ., data = dataset)
```

```
Call:
lm(formula = expenses ~ ., data = dataset)

Residuals:
     Min       1Q    Median       3Q      Max
-11302.7  -2850.9   -979.6   1383.9  29981.7

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     -11941.6      987.8 -12.089  < 2e-16 ***
age                256.8       11.9  21.586  < 2e-16 ***
sexmale           -131.3      332.9  -0.395 0.693255
bmi                339.3       28.6  11.864  < 2e-16 ***
children           475.7      137.8   3.452 0.000574 ***
smokeryes        23847.5      413.1  57.723  < 2e-16 ***
regionnorthwest   -352.8      476.3  -0.741 0.458976
regionsoutheast  -1035.6      478.7  -2.163 0.030685 *
regionsouthwest   -959.3      477.9  -2.007 0.044921 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6062 on 1329 degrees of freedom
Multiple R-squared:  0.7509,    Adjusted R-squared:  0.7494
F-statistic: 500.9 on 8 and 1329 DF,  p-value: < 2.2e-16
```

Smoker, bmi, children, age, southwest or southeast are significant and the rest can be removed for the model improvement.

How I improved the original model is by adding few extra features, for an example I made an assumption that expenses and age can't be a linear sequence. Most of the time when people get older their medical fees increase. Hence, why the age squared(age^2) feature was added to the prediction model.

The other assumption I made is that BMI (body-mass-index) will affect medical expenses if a customer reaches a certain value. For an example, if someone is considered obese surely that would increase their medical expenses. A new term is added bmi 30 which will class the numerical element BMI value into two classes; zero for BMI under 30 and one for BMI greater than 30

Prediction Model:

```
> dataset$age2 <- dataset$age^2
> dataset$bmi30 <- ifelse(dataset$bmi >= 30, 1, 0)
> ins_model2 <- lm(expenses ~ age + age2 + children + bmi + sex +
+                      bmi30*smoker + region, data = dataset)
> 
```

```
Call:
lm(formula = expenses ~ age + age2 + children + bmi + sex + bmi30 *
    smoker + region, data = dataset)

Residuals:
     Min       1Q   Median       3Q      Max 
-17297.1  -1656.0  -1262.7   -727.8  24161.6 

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)       139.0053  1363.1359   0.102 0.918792    
age               -32.6181    59.8250  -0.545 0.585690    
age2                3.7307     0.7463   4.999 6.54e-07 ***
children          678.6017   105.8855   6.409 2.03e-10 ***
bmi               119.7715    34.2796   3.494 0.000492 ***
sexmale          -496.7690   244.3713  -2.033 0.042267 *  
bmi30            -997.9355   422.9607  -2.359 0.018449 *  
smokeryes       13404.5952   439.9591  30.468  < 2e-16 ***
regionnorthwest  -279.1661   349.2826  -0.799 0.424285    
regionsoutheast  -828.0345   351.6484  -2.355 0.018682 *  
regionsouthwest -1222.1619   350.5314  -3.487 0.000505 ***
bmi30:smokeryes 19810.1534   604.6769  32.762  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4445 on 1326 degrees of freedom
Multiple R-squared:  0.8664,	Adjusted R-squared:  0.8653 
F-statistic: 781.7 on 11 and 1326 DF,  p-value: < 2.2e-16
```

1.4 Evaluation

The estimation on coefficients column indicates the estimate values of coefficients inside the model for each respective value to the left, Nominal values are given dummy variables depend upon their wide variety of levels using n-1 method. One unit growth of the feature to the left can have an effect on the value of the dependent variable. When the p value is analysed against the alpha level = 0.05 suggests the significance of every coefficient term. A number less than 0.05 in the p value sequence indicated a rejection of the null hypothesis, confirming the alternative hypothesis that the beta-i term is significant at the 0.05 level and the beta term should be kept. R squared = 0.75 indicates that about 75% of the variation in expenses is represented by the original model.

In relation to the prediction model additional terms have been added such as age2, bmi30 which all are all significant by p-values. The prediction model R-square value is 0.87, This is much better than the original model's R Square value, considering the extra independent variable term when analysing the adjusted R Square.

Finally, multiple linear regression was originally constructed using multiple numerical and nominal characteristics such as age, gender, number of children, BMI, and region. The performance of each model can be evaluated by looking at the rest of the range, such as model statistics, which are estimated on regression coefficients, and evaluating the model and feature such as R-square, and p-value.

Polynomial Regression

2.1 Business Understanding

The dataset which I used to explore Polynomial Regression is a [Vertebral Column Data Set] from UCI Machine Learning Repository. My goal or primary objective from a business perspective is to help the HSE identify patient's vertebral data and diagnose them to have normal or abnormal vertebrate. My plan to take the customer Vertebral data such as lumbar lordosis angle and sacral slope and try to predict a pattern or value in order to help HSE diagnose patients.

2.2 Data Understanding & Preparation

The dataset Vertebral.csv includes 310 entries and 7 attributes (columns).The dataset contains six numeric biomechanical features used to classify orthopaedic patients (pelvic_incidence, pelvic_tilt, lumbar_lordosis_angle, sacral_slope, pelvic_radius, degree_spondylolisthesis) and last attribute is classes which is either normal or abnormal. The dataset is not missing any fields which means no preparation of the data has to be carried out at the moment.

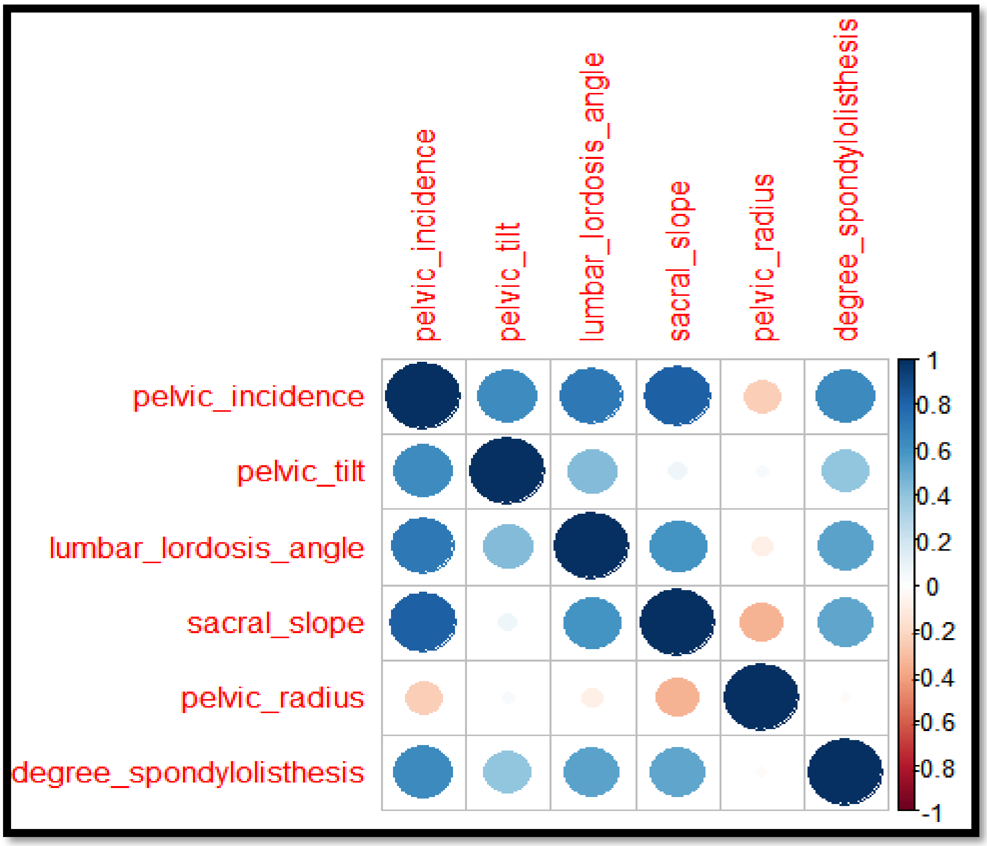
Summary of the dataset below:

```
> summary(backdata)
 pelvic_incidence  pelvic_tilt      lumbar_lordosis_angle
 Min.   : 26.15   Min.   :-6.555   Min.   : 14.00
 1st Qu.: 46.43   1st Qu.:10.667   1st Qu.: 37.00
 Median : 58.69   Median :16.358   Median : 49.56
 Mean   : 60.50   Mean   :17.543   Mean   : 51.93
 3rd Qu.: 72.88   3rd Qu.:22.120   3rd Qu.: 63.00
 Max.   :129.83   Max.   :49.432   Max.   :125.74
  sacral_slope     pelvic_radius
 Min.   : 13.37   Min.   : 70.08
 1st Qu.: 33.35   1st Qu.:110.71
 Median : 42.40   Median :118.27
 Mean   : 42.95   Mean   :117.92
 3rd Qu.: 52.70   3rd Qu.:125.47
 Max.   :121.43   Max.   :163.07
 degree_spondylolisthesis       class
 Min.   :-11.058          Abnormal:210
 1st Qu.:  1.604          Normal  :100
 Median : 11.768
 Mean   : 26.297
 3rd Qu.: 41.287
 Max.   :418.543
```

I produced a correlation matrix to depict the relationship of four features in the dataset:

```
> cor(backdata[c("pelvic_incidence", "pelvic_tilt", "lumbar_lordosis_angle", "sacral_slope","pelvic_radius","degree_spondylolisthesis")])
                         pelvic_incidence pelvic_tilt lumbar_lordosis_angle sacral_slope pelvic_radius
pelvic_incidence                1.0000000  0.62919878            0.71728237   0.81495999   -0.24746720
pelvic_tilt                     0.6291988  1.00000000            0.43276387   0.06234529    0.03266781
lumbar_lordosis_angle           0.7172824  0.43276387            1.00000000   0.59838689   -0.08034361
sacral_slope                    0.8149600  0.06234529            0.59838689   1.00000000   -0.34212835
pelvic_radius                  -0.2474672  0.03266781           -0.08034361  -0.34212835    1.00000000
degree_spondylolisthesis        0.6387427  0.39786228            0.53366701   0.52355746   -0.02606501
                         degree_spondylolisthesis
pelvic_incidence                       0.63874275
pelvic_tilt                            0.39786228
lumbar_lordosis_angle                  0.53366701
sacral_slope                           0.52355746
pelvic_radius                         -0.02606501
degree_spondylolisthesis               1.00000000
```

I produced a Corrplot is a graphical display of a correlation matrix, this is a good way to understand relations between among the variables. From inspecting the data its clear that a number of variables have good correlations, I will pick lumbar lordosis angle and sacral slope which bout have r value of 0.6 which will make for an interesting model.



2.3 Modelling

Polynomial regression is carried out using lumbar lordosis angle and sacral slope as the dependant variable and the rest of the values are independent factors withinside the model. I used the fitting linear model function which is used for regression and applied ploy() function to the sacral slope along with the polynomial degree. I assigned the lumbar lordosis angle as y before tilde and sacral slope after the tide and the dataset next which contains theses independent variables of numerical values.

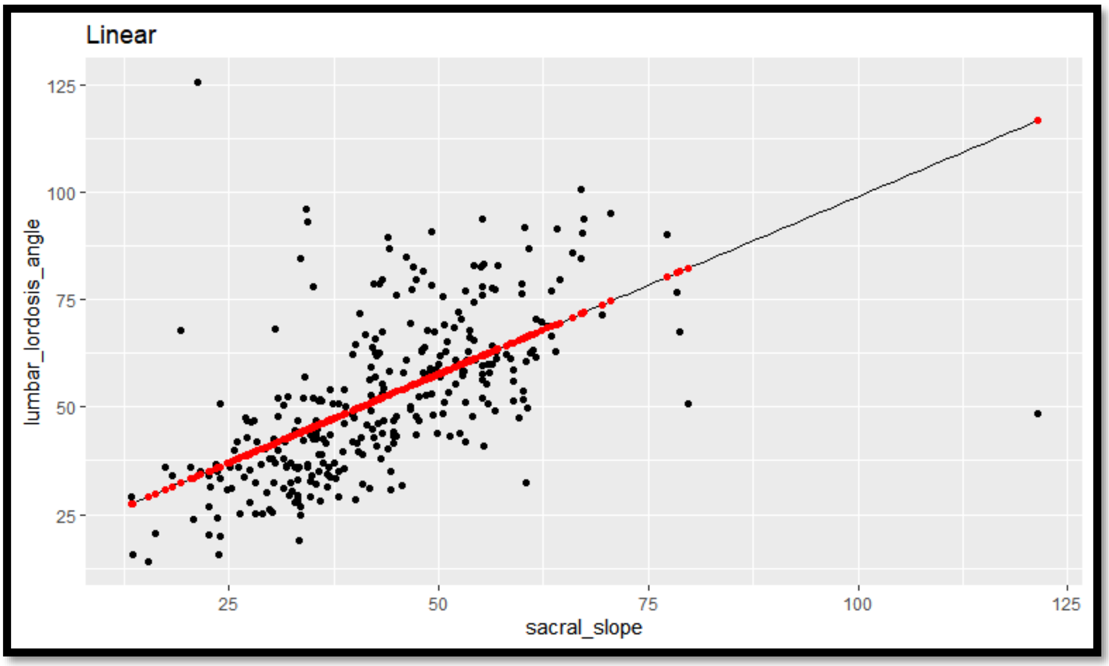
```
> lindata<-lm(lumbar_lordosis_angle~sacral_slope,data=backdata)
> lindata

Call:
lm(formula = lumbar_lordosis_angle ~ sacral_slope, data = backdata)

Coefficients:
 (Intercept)  sacral_slope
     16.4030        0.8271
```

Linear Regression:

```
> linear<-ggplot(data=backdata,aes(x=sacral_slope,y=lumbar_lordosis_angle))+geom_point()+
+   stat_function(aes(sacral_slope),fun=linfun)+ggtitle("Linear")+ geom_point(data=meanslinear,aes(x=x,y=y),color="red")
```

```
Call:
lm(formula = lumbar_lordosis_angle ~ sacral_slope, data = backdata)

Residuals:
    Min      1Q  Median      3Q     Max 
-68.456  -9.342  -2.217   7.051  91.763 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)   16.4030     2.8394   5.777 1.86e-08 ***
sacral_slope   0.8271     0.0631  13.107  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.89 on 308 degrees of freedom
Multiple R-squared:  0.3581,	Adjusted R-squared:  0.356 
F-statistic: 171.8 on 1 and 308 DF,  p-value: < 2.2e-16
```

Linear Regression plot



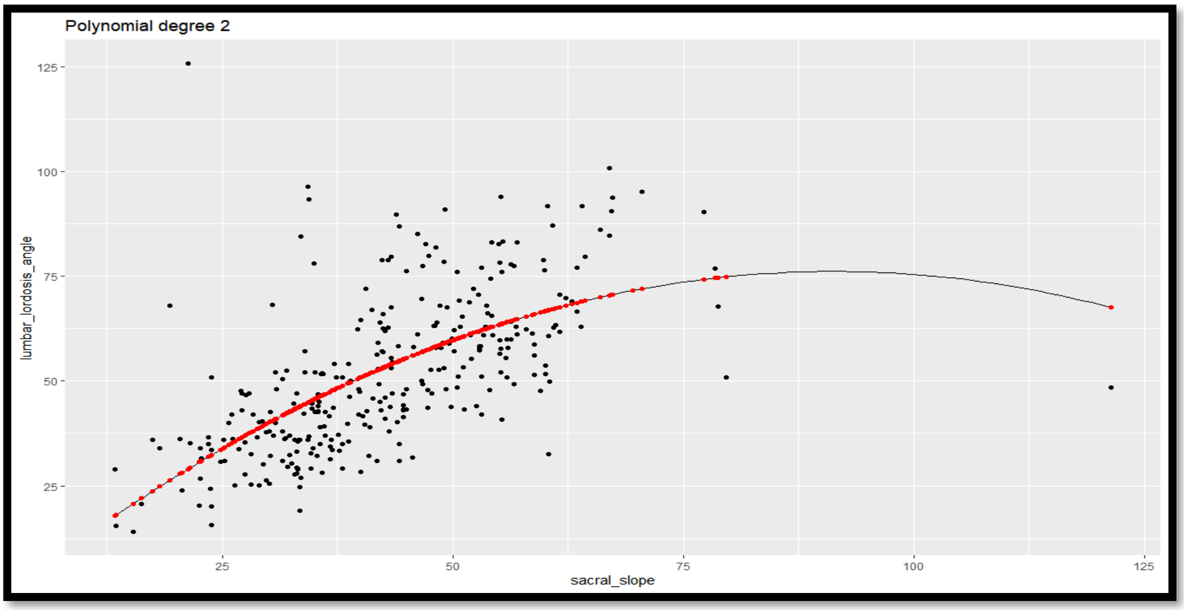
Second degree polynomial

```
> poly2<-lm(lumbar_lordosis_angle~poly(sacral_slope,2,raw=T),data=backdata)
> poly2

Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 2, raw = T), 
    data = backdata)

Coefficients:
                    (Intercept)  poly(sacral_slope, 2, raw = T)1  poly(sacral_slope, 2, raw = T)2  
                       -3.90134                          1.75137                         -0.00958  
```

```
> summary(poly2)

Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 2, raw = T), 
    data = backdata)

Residuals:
    Min      1Q  Median      3Q     Max 
-34.430  -9.861  -2.223   6.946  96.752 

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)                     -3.901336   5.423937  -0.719    0.473    
poly(sacral_slope, 2, raw = T)1  1.751371   0.221210   7.917 4.47e-14 ***
poly(sacral_slope, 2, raw = T)2 -0.009580   0.002203  -4.349 1.87e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.47 on 307 degrees of freedom
Multiple R-squared:  0.3953,	Adjusted R-squared:  0.3914 
F-statistic: 100.4 on 2 and 307 DF,  p-value: < 2.2e-16
```

```
> poly2<-ggplot(data=backdata,aes(x=sacral_slope,y=lumbar_lordosis_angle))+geom_point()+
+   stat_function(aes(sacral_slope),fun=funpoly2)+ggtitle("Polynomial degree 2")+geom_poin
t(data=meanspoly2,aes(x=xpoly2,y=ypoly2),color="red")
```

Second degree polynomial plot code:

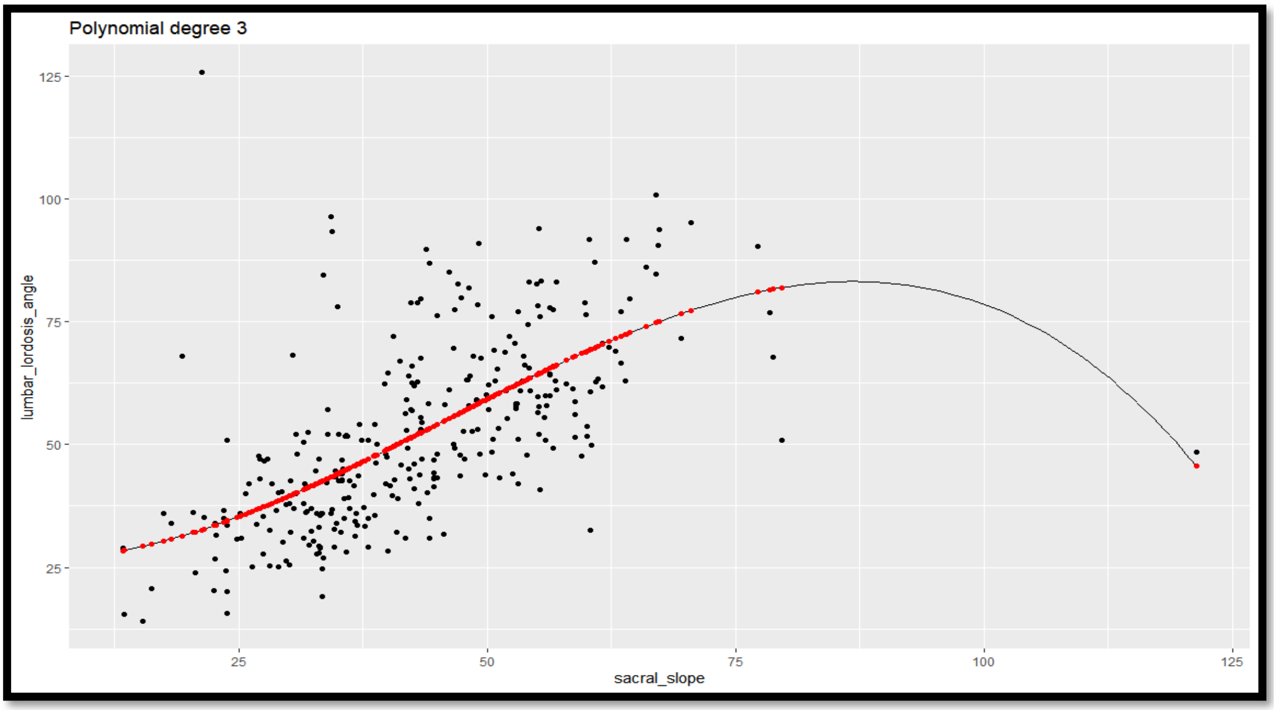


Polynomial degree 3

```
> poly3<-lm(lumbar_lordosis_angle~poly(sacral_slope,3,raw=T),data=backdata)
> poly3

Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 3, raw = T), 
    data = backdata)

Coefficients:
                   (Intercept)  poly(sacral_slope, 3, raw = T)1  
                    27.0618805                       -0.2326161  
poly(sacral_slope, 3, raw = T)2  poly(sacral_slope, 3, raw = T)3  
                     0.0275658                       -0.0002009  
```

```
Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 3, raw = T), 
    data = backdata)

Residuals:
    Min      1Q  Median      3Q     Max 
-36.810  -9.362  -1.966   6.800  93.103 

Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                      2.706e+01  1.130e+01   2.395  0.01724
poly(sacral_slope, 3, raw = T)1 -2.326e-01  6.742e-01  -0.345  0.73030
poly(sacral_slope, 3, raw = T)2  2.757e-02  1.214e-02   2.271  0.02385
poly(sacral_slope, 3, raw = T)3 -2.009e-04  6.458e-05  -3.110  0.00205
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.27 on 306 degrees of freedom
Multiple R-squared:  0.4138,	Adjusted R-squared:  0.4081 
F-statistic: 72.02 on 3 and 306 DF,  p-value: < 2.2e-16
```

```
> poly3<-ggplot(data=backdata,aes(x=sacral_slope,y=lumbar_lordosis_angle))+geom_point()+
+   stat_function(aes(sacral_slope),fun=funpoly3)+ggtitle("Polynomial degree 3")+geom_poin
t(data=meanspoly3,aes(x=xpoly3,y=ypoly3),color="red")
> poly3
```

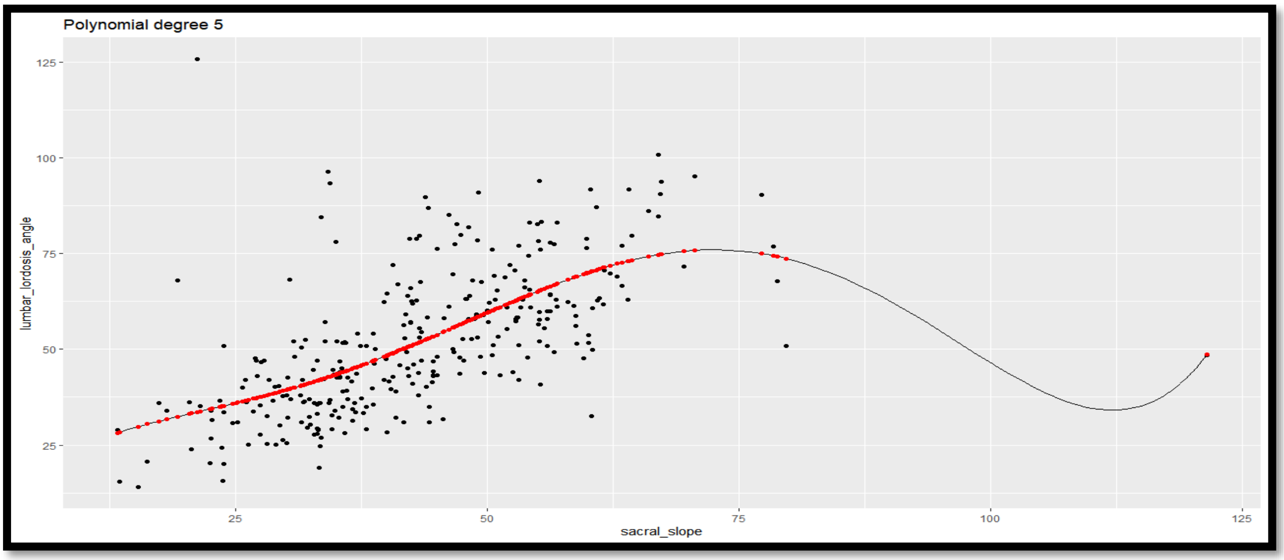


Polynomial degree 5

```
> poly5

Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 5, raw = T),
    data = backdata)

Coefficients:
                   (Intercept)  poly(sacral_slope, 5, raw = T)1
                     4.845e+00                        3.126e+00
poly(sacral_slope, 5, raw = T)2  poly(sacral_slope, 5, raw = T)3
                    -1.466e-01                        3.764e-03
poly(sacral_slope, 5, raw = T)4  poly(sacral_slope, 5, raw = T)5
                    -4.011e-05                        1.442e-07
```

```
> poly5<-ggplot(data=backdata,aes(x=sacral_slope,y=lumbar_lordosis_angle))+geom_po
int()+
+   stat_function(aes(sacral_slope),fun=funpoly5)+ggtitle("Polynomial degree 5")+g
eom_point(data=meanspoly5,aes(x=xpoly5,y=ypoly5),color="red")
> poly5
```

```
> poly5<-lm(lumbar_lordosis_angle~poly(sacral_slope,5,raw=T),data=backdata)
> summary(poly5)

Call:
lm(formula = lumbar_lordosis_angle ~ poly(sacral_slope, 5, raw = T),
    data = backdata)

Residuals:
    Min      1Q  Median      3Q     Max
-37.902  -9.294  -2.094   6.875  92.083

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                     4.845e+00  5.139e+01   0.094    0.925
poly(sacral_slope, 5, raw = T)1  3.126e+00  6.022e+00   0.519    0.604
poly(sacral_slope, 5, raw = T)2 -1.466e-01  2.604e-01  -0.563    0.574
poly(sacral_slope, 5, raw = T)3  3.764e-03  5.187e-03   0.726    0.469
poly(sacral_slope, 5, raw = T)4 -4.011e-05  4.757e-05  -0.843    0.400
poly(sacral_slope, 5, raw = T)5  1.442e-07  1.598e-07   0.902    0.368

Residual standard error: 14.28 on 304 degrees of freedom
Multiple R-squared:  0.417,     Adjusted R-squared:  0.4074
F-statistic: 43.49 on 5 and 304 DF,  p-value: < 2.2e-16
```



2.4 Evaluation

From analysing the graph, you can tell that the data fits much better using quadratic curve rather than a linear line. Majority from the plots have a high bias meaning that the model is unable to fit the data. This results in under fitting as the plot curve line dose not capture the pattern in the data. I feel the best fit for the Polynomial regression data is Polynomial degree 3. I think next time I should carry out Polynomial regression using a training set so I can train the data and Validation Set so I can determine the hyperparameters e.g. the maximum polynomial degree.

I struggled finding the hyperparameters using the vertebrate dataset as I tried number of different Polynomial degree. I feel I should have used better attributes from the dataset as lumbar lordosis angle and sacral slope as the dependant variable are to unique and don't really follow a pattern. The performance of each model can be evaluated by looking at the rest of the range, such as model statistics, which are estimated on regression coefficients, and evaluating the model and feature such as R-square, and p-value.