194161016

Saikiran (194161016)

20/03/2020

Linear Regression:

Linear regression is a statistical approach to model a relationship between a response variable and one or more predictor(independent) variables.If there is only one predictor variable then it is simple linear regression.If more than one predictor variable is used then it is multiple linear regression.

An expression for a simple linear model is given as

$$\mathbf{Y = \beta_0 + \beta_1 x}$$

where Y is the response variable,x is a predictor variable and β_0, β_1 are intercept and slope estimates respectively.

Assumptions:

A linear model will make several assumptions about the data.

1.The relationship between the predictor and response variable is linear.

2.The residuals(or errors) are assumed to be normally distributed with expected value of zero.

3.The residuals(or errors) are assumed to have constant variance(σ^2).

4.The residuals(or errors) (ϵ_i) are uncorrelated with each other.

In our case the linear equation can be written as

lifeExp = $\beta_0 + \beta_1$***gdpPercap**

Let us try to divide the dataset into train(1952-1992) and test(1993-2007) and fit the model on the train data.

```
library("gapminder")
library("ggplot2")
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("gam")
```

```
## Warning: package 'gam' was built under R version 3.6.3
```

```
## Loading required package: splines
```

```
## Loading required package: foreach
```

```
## Loaded gam 1.16.1
```

```
library("MASS")
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
data.gapminder <- data.frame(gapminder)
train.data <- data.gapminder %>% filter(year>=1952,year<=1992)
test.data <- data.gapminder %>% filter(year>=1993,year<=2007)

lm.fit<-lm(lifeExp~gdpPercap,train.data)
summary.lm<-summary(lm.fit)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = lifeExp ~ gdpPercap, data = train.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -76.914  -8.048   2.143   8.889  18.419
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.284e+01  3.620e-01  145.98   <2e-16 ***
## gdpPercap   7.233e-04  3.301e-05   21.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.66 on 1276 degrees of freedom
## Multiple R-squared:  0.2734, Adjusted R-squared:  0.2728
## F-statistic: 480.1 on 1 and 1276 DF,  p-value: < 2.2e-16
```

Summary of the model:

Formula :

In our case we have used **lifeExp** as a response variable and **gdpPercap** as predictor variables.

Residual Summary:

- One of the assumptions for linear models is that the errors follow a **normal distribution**, and as a consequence the residuals should as well.
- The residuals summary tells us about the **symmetry** of the residual distribution.
- As per the assumptions of the linear model our median should be close to 0 as the expected value of the residuals is 0.In our model the **median is not close to zero**.
- Further, the **first Quartile** and **third Quartile** should be close to each other in **magnitude**, as they would be under a symmetric zero mean distribution.
- The maximum and minimum residual should also have similar magnitude.
- However in our case **first quartile** and **third quartile** have similar magnitude but **minimum** and **maximum** residuals are far apart from each other in magnitude **violating symmetry** assumption.

Coefficients

Estimates:

- The estimate of the **intercept** gives us the expected response variable value when all the features are **zero**. In our case intercept gives us the **life expectancy** when the **gdpPercapita** is **zero**. i.e 5.284e+01 will be the life expectancy when the gdp per Capita is zero.
- If we increase the **gdpPercap** by **one unit**, there will be an increase of `7.233e-04` in response variable(lifeExpectancy).
- **Positive estimate** implies **increase/decrease** in predictor **increases/decreases** the response value and **negative estimate** implies **increase/decrease** in predictor **decreases/increases** the **response value.**
- If the estimated value of any variable is 0, it means that the corresponding predictor variable is not significant(not showing any impact in predicting the response variable value) to the model.

Standard Error:

- The **standard error** tells us the standard error of our estimate, which allows us to **construct marginal confidence intervals** for the estimate of that particular variable.
- If **S.E(β_i)** is the standard error and β_i is the estimated coefficient for a variable i, then we can construct confidence interval by $\beta_i \pm Z_{\alpha/2}$*S.E(β_i).

t-value:

- The coefficient **t-value** tells us about how far our estimated parameter is from a hypothesized 0 value, scaled by the standard deviation of the estimate.
- We want it to be far away from zero as we could reject the null hypothesis and we could declare that a relationship between our predictor and response exists.
- t-values are also used to compute p-values.

Pr(>|t|):

- Under the t distribution with n-p-1 degrees of freedom, this tells us the **probability** of observing a value at least as extreme as our β_i, where n is no. of observations and p is no of predictor variables.
- If this probability is sufficiently low, we can reject the null hypothesis that this coefficient is 0 and declare that there exists a relationship between corresponding variable and the response variable.
- A **p-value of 0.05** or less is a good cut-off point.
- In our case the **intercept, gdpPercap** are having **p value very less than 0.05** and hence considered as significant to our model.
- The 'signif. Codes' associated with each estimate tells us the significance level of the respective variables. Three stars (or asterisks) represent a highly significant p-value and " " blank represents a least significant.

Residual Standard Error:

- It gives us the sum of squares of all the residuals.
- While fitting the model, the linear regression algorithm will try to adjust the beta's(coefficients) such that the RSE is as minimum as possible.

F statistic:

- The model with zero predictor variables is also called "**Intercept Only Model**".
- F – Test for overall significance compares an **intercept only regression model** with our current model.
- It tries to check if the addition of these variables together into the model is significant enough or not .
- We can think of hypothesis test for this as:
- H0: The fit of the **intercept only model** and our **current model** is the same. i.e. Additional variables do not provide value taken together
- Ha : The fit of the **intercept only model** is significantly less compared to our **current model**. i.e. Additional variables do make the **model significantly better**.

Multiple R-squared, Adjusted R-squared:

- R^2 is a statistical measure that tells how well our model has fitted the data.
- It tells us the proportion of variance that our model was able to explain out of the total variance in the actual data.
- Mathematically, R^2 is given as:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

- It always lies between 0 and 1. A number near to 0 represents that the model did not explain the variance in the response variable well and a number close to 1 represents that the model was able to explain the total variance in the data.
- However, it's hard to define what level of (R^2) is appropriate to claim the model fits well. It varies with the application and the domain studied.For example in an **airline industry** to fit a model for **block time optimization** with an R^2 of 0.4-0.5 is very good.Where as in **rocket/satellite launching systems,** an R^2 of **0.97-0.98** is also not a good fit
- In multiple regression, the (R^2) will increase as more no of variables are included in the model. That's why the adjusted (R^2) is the preferred measure as it adjusts for the number of variables considered.
- Expression for adjusted R^2 is given by :

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2/(n-p)}{\sum_i (y_i - \bar{y})^2)/(n-1)}$$

where n is the number of samples and p is number of the predictor variables

- **We can check the significance of any variable by looking at its p value.**
- P-value basically tells us the probability of the predictor variable coefficient being zero.
- If the p-value is very less(generally checked if it is less than 0.05), we can say that there is very less chance of the coefficient being zero of the concerned variable and hence the variable is said to be significant(having impact in predicting response/output variable).

Linear Model:

In our case if we observe the summary of our linear fit, the p value of gdpPercap is <2e-16. It shows that the chance of coefficient of gdpPercap being zero is almost close to zero.**Hence our hypothesis "lifeExp is dependent on gdpPercap" is correct.**

```
glm.fit <- glm(lifeExp~gdpPercap,data=train.data)
summary.glm <- summary(glm.fit)
summary(glm.fit)
```

```
##
## Call:
## glm(formula = lifeExp ~ gdpPercap, data = train.data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -76.914   -8.048    2.143    8.889   18.419
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.284e+01  3.620e-01  145.98   <2e-16 ***
## gdpPercap   7.233e-04  3.301e-05   21.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 113.7217)
##
##     Null deviance: 199710  on 1277  degrees of freedom
## Residual deviance: 145109  on 1276  degrees of freedom
## AIC: 9680.5
##
## Number of Fisher Scoring iterations: 2
```

Generalised Linear model:

By observing the summary of the generalised linear model fit the p value of gdpPercap is **less than 2e-16** which says that the chance of coefficient of gdpPercap becoming zero is very less.**Hence our hypothesis "lifeExp is dependent on gdpPercap" is correct.**

```
loess.fit <- loess(lifeExp~gdpPercap,data=train.data)
summary.loess <- summary(loess.fit)
summary(loess.fit)
```

```
## Call:
## loess(formula = lifeExp ~ gdpPercap, data = train.data)
##
## Number of Observations: 1278
## Equivalent Number of Parameters: 5.5
## Residual Standard Error: 7.249
## Trace of smoother matrix: 6.03  (exact)
##
## Control settings:
##   span     :  0.75
##   degree   :  2
##   family   :  gaussian
##   surface  :  interpolate      cell = 0.2
##   normalize:  TRUE
##  parametric:  FALSE
## drop.square:  FALSE
```

```
gam.fit <- gam(lifeExp~gdpPercap,data=train.data)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
argument
## ignored
```

```
summary.gam <- summary(gam.fit)
summary(gam.fit)
```

```
##
## Call: gam(formula = lifeExp ~ gdpPercap, data = train.data)
## Deviance Residuals:
##     Min      1Q  Median      3Q     Max
## -76.914  -8.048   2.143   8.889  18.419
##
## (Dispersion Parameter for gaussian family taken to be 113.7217)
##
##     Null Deviance: 199709.7 on 1277 degrees of freedom
## Residual Deviance: 145108.9 on 1276 degrees of freedom
```

```
## AIC: 9680.544
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##             Df Sum Sq Mean Sq F value    Pr(>F)
## gdpPercap    1  54601   54601  480.13 < 2.2e-16 ***
## Residuals 1276 145109     114
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

generalized additive model:

We can observe from the summary of the generalised additive model fit that the p value of gdpPercap is less than 2e-16 which says that the chance of coefficient of gdpPercap becoming zero is very less.**Hence our hypothesis "lifeExp is dependent on gdpPercap" is correct according to generalised additive model.**

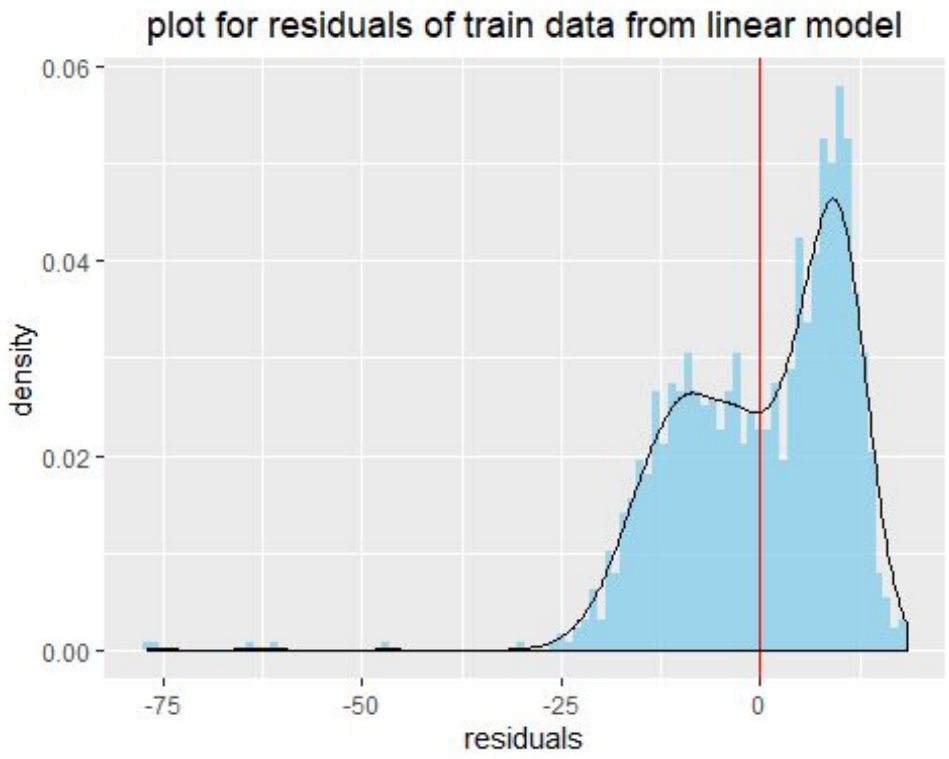
```
rlm.fit <- rlm(lifeExp~gdpPercap,data=train.data)
summary.rlm <- summary(rlm.fit)

summary(rlm.fit)

##
## Call: rlm(formula = lifeExp ~ gdpPercap, data = train.data)
## Residuals:
##       Min        1Q    Median        3Q       Max
## -129.2430   -7.1031    0.4585    7.2206   18.6475
##
## Coefficients:
##             Value    Std. Error t value
## (Intercept)  50.4612   0.3171   159.1503
## gdpPercap     0.0012   0.0000    41.6799
##
## Residual standard error: 10.69 on 1276 degrees of freedom

summary.rlm$coefficients[6]

## [1] 41.67986

pValue <- 2*pt(summary.rlm$coefficients[6],summary.rlm$df[2],lower.tail =
FALSE)
cat("pValue of gdpPerCap is :",pValue)

## pValue of gdpPerCap is : 2.399971e-240
```

robust linear model:

We can observe p value of gdpPercap is **2.399971e-240** which says that the chance of coefficient of gdpPercap becoming zero is very less.Hence our hypothesis "lifeExp is dependent on gdpPercap" is correct according to **robust linear model**.

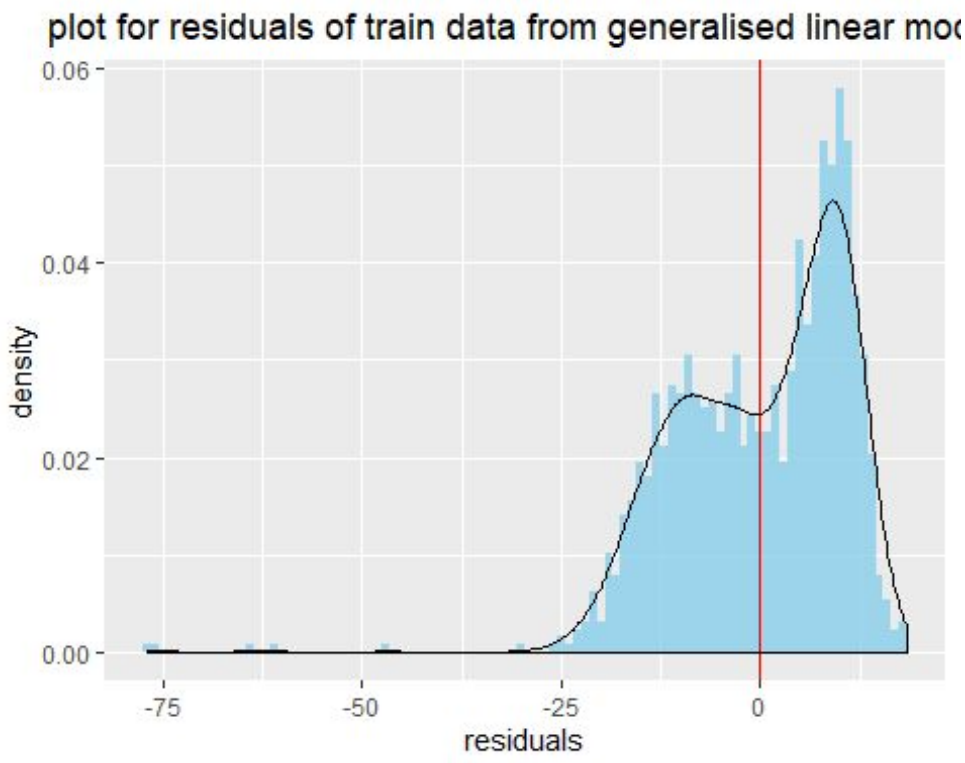
Q2)Use the data from summary and plot a histogram to show how well does your model fits the data.

```
lm.residuals <- data.frame(residuals = residuals(lm.fit))
ggplot(data=lm.residuals,mapping=aes(x=residuals))+
  geom_histogram(binwidth=1,aes(y = ..density..), alpha =
0.8,fill="skyblue")+labs(title="plot for residuals of train data from linear
model")+theme(plot.title =
element_text(hjust=0.5))+geom_vline(data=lm.residuals, aes(xintercept = 0),
colour="red")+geom_density()
```



- Goodness of fit of a model can be checked by plotting histogram plot of residuals.If the residuals are following normal distribution then we can say that our model is fitting the data properly.
- We can observe from the histogram of residuals for linear model that the residuals are not following the normal distribution(we can clearly see its a bimodal distribution).This says that the model is not fitting the data properly.

```
glm.residuals <- data.frame(residuals = residuals(glm.fit))
ggplot(data=glm.residuals,mapping=aes(x=residuals))+

geom_histogram(binwidth=1,aes(y=..density..),fill="skyblue",alpha=0.8)+labs(t
itle="plot for residuals of train data from generalised linear
model")+theme(plot.title =
element_text(hjust=0.5))+geom_vline(data=glm.residuals, aes(xintercept = 0),
colour="red")+geom_density()
```

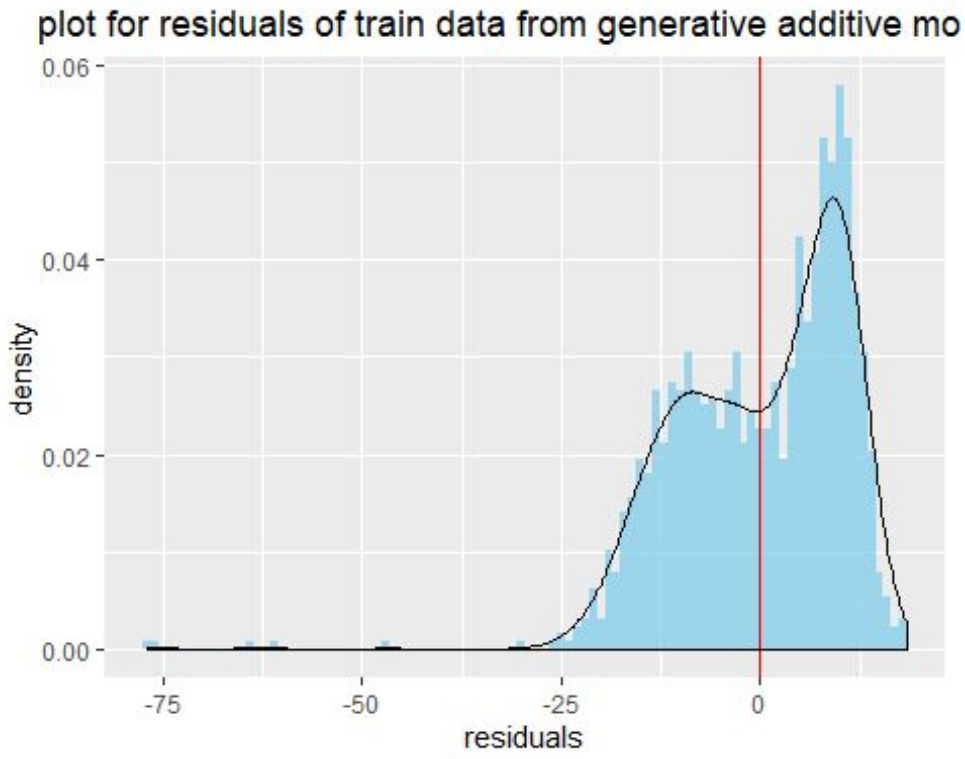


- We can observe from the histogram of residuals for generalised linear model that the residuals are not following the normal distribution.This says that the model is

not fitting the data properly.

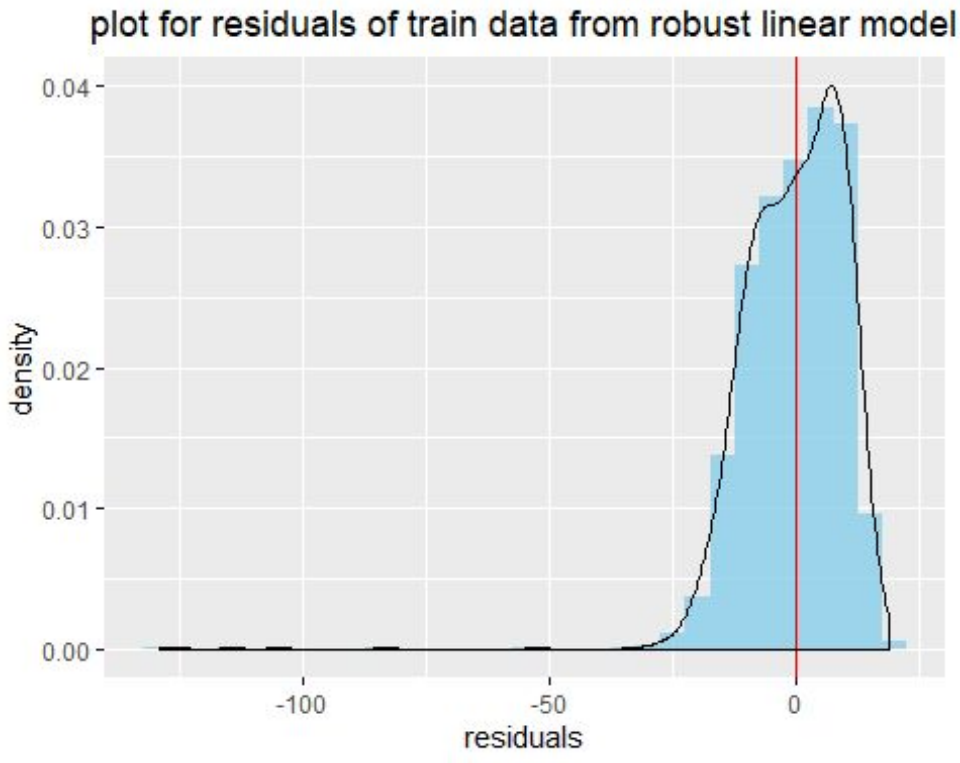
- We can also see that there are some outliers towards the left in the histogram which has very high residual value.

```
gam.residuals <- data.frame(residuals = residuals(gam.fit))
ggplot(data=gam.residuals,mapping=aes(x=residuals)) +
  geom_histogram(binwidth=1,aes(y = ..density..), alpha =
0.8,fill="skyblue")+labs(title="plot for residuals of train data from
generative additive model")+theme(plot.title =
element_text(hjust=0.5))+geom_vline(data=gam.residuals, aes(xintercept = 0),
colour="red")+geom_density()
```



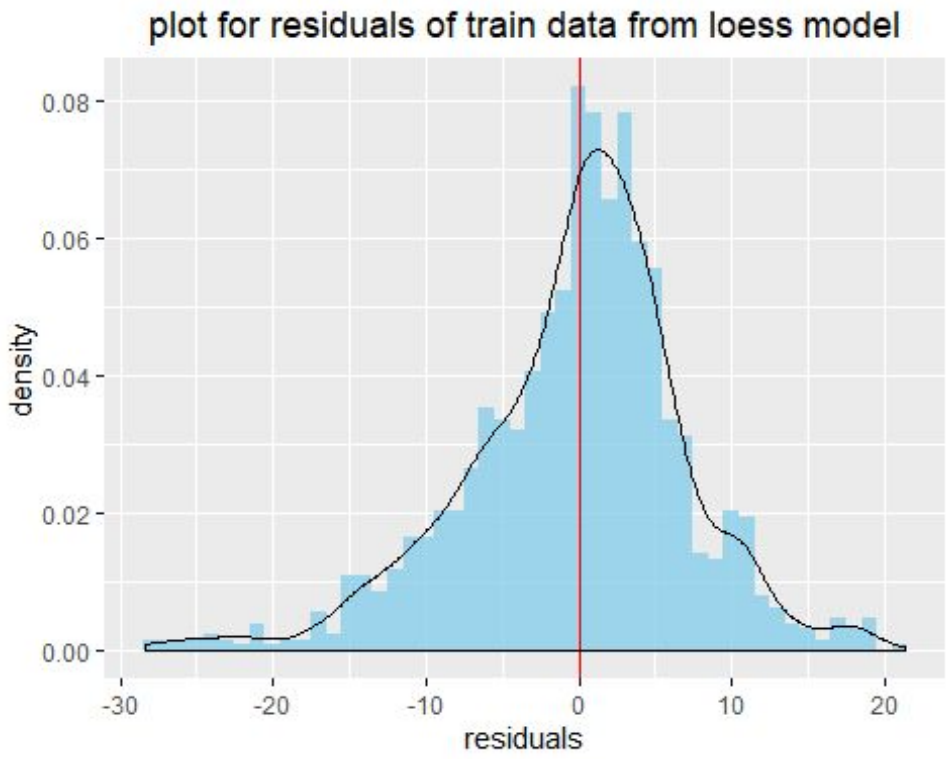
As generative additive model is obtained by combining more than one glm model the residuals of this model is also coming out to be following the similar distribution. We can observe from the histogram of residuals for generalised additive model that the residuals are not following the normal distribution.This says that the model is not fitting the data properly.

```
rlm.residuals <- data.frame(residuals = residuals(rlm.fit))
ggplot(data=rlm.residuals,mapping=aes(x=residuals)) +
  geom_histogram(binwidth=5,aes(y = ..density..), alpha =
0.8,fill="skyblue")+labs(title="plot for residuals of train data from robust
linear model")+theme(plot.title =
element_text(hjust=0.5))+geom_vline(data=rlm.residuals, aes(xintercept = 0),
colour="red")+geom_density()
```



We can observe that the residuals of robust linear are following normal distribution with some outliers lying towards the left in the histogram plot.
Hence we can say that robust linear model is properly fitting our data.

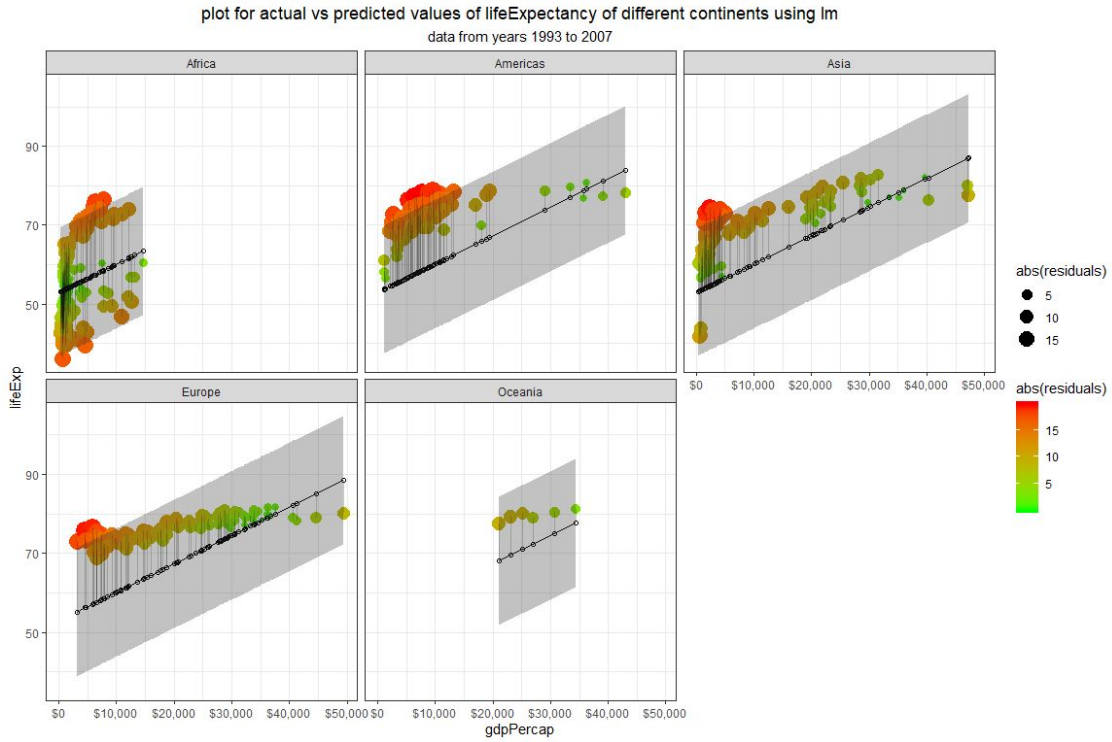
```
loess.residuals <- data.frame(residuals = residuals(loess.fit))
ggplot(data=loess.residuals,mapping=aes(x=residuals)) +
  geom_histogram(binwidth=1,aes(y = ..density..), alpha =
0.8,fill="skyblue")+labs(title="plot for residuals of train data from loess
model")+theme(plot.title =
element_text(hjust=0.5))+geom_vline(data=loess.residuals, aes(xintercept =
0), colour="red")+geom_density()
```



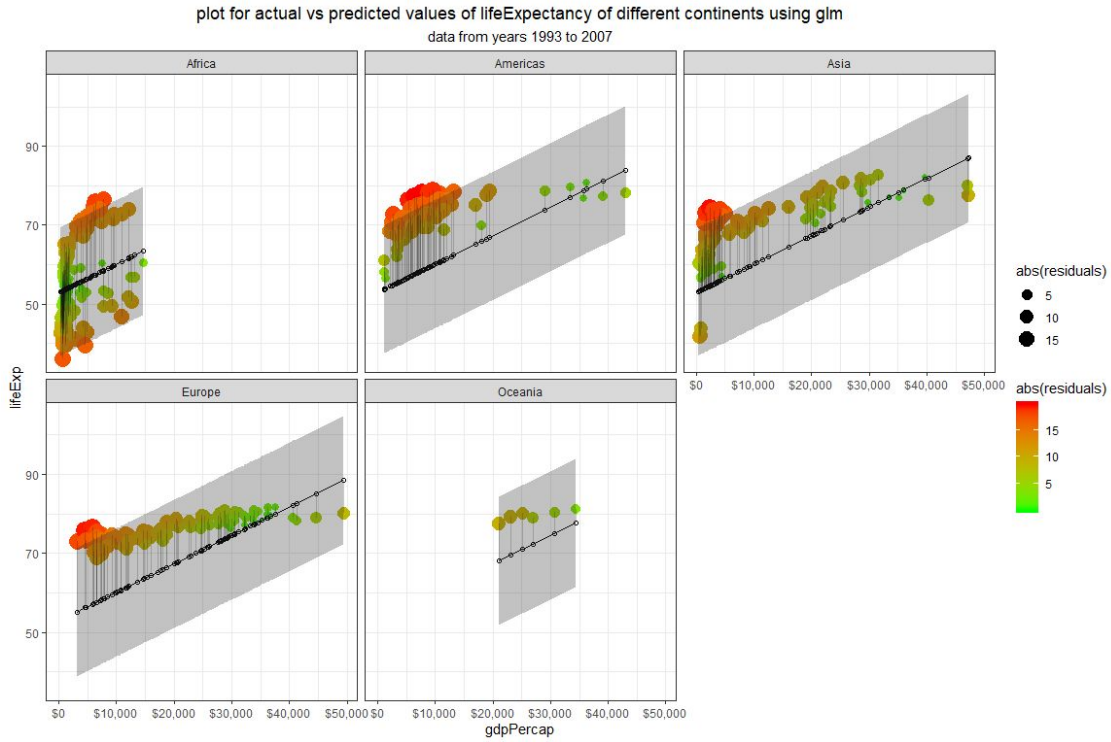
- We can observe that the residuals are lying symmetric the mean 0 which is expected and also We can observe that the residuals of loess model are following normal distribution.
- Hence we can say that loess is perfectly fitting our data.
- **Out of all the model loess is fitting the data perfectly but loess is not preferred for the datasets greater than 1000 samples as it is computationally not storable.It requires O(n^2) in memory.**

Q3)Plot a graph showing the prediction on test data points, linear regression line fitting the data. Also draw the area covered by prediction intervals. Facet the data based on continents.

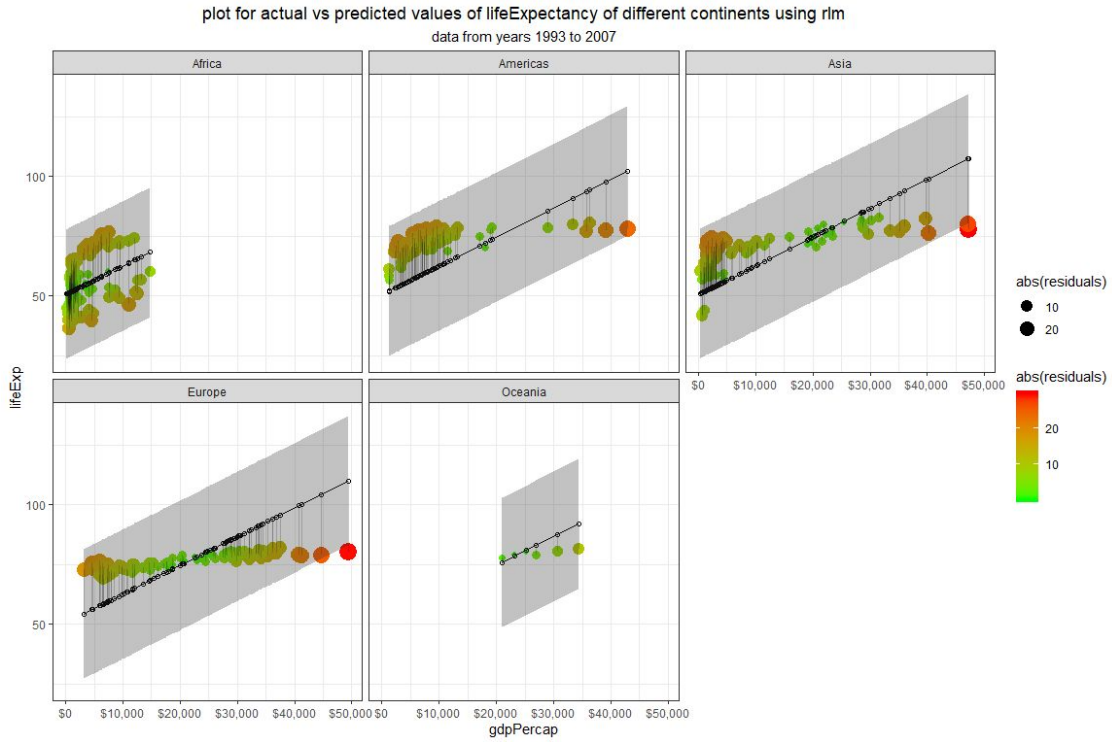
```
temp <- predict(lm.fit,test.data,interval = "predict")
test.data$predicted <- temp[,1]
test.data$residuals <- test.data$lifeExp-test.data$predicted
sd1 <- sd(test.data$predicted)


ggplot(test.data,mapping=aes(x=gdpPercap,y=lifeExp))+

geom_point(test.data,mapping=aes(x=gdpPercap,y=lifeExp,color=abs(residuals),s
ize=abs(residuals)))+

geom_point(mapping=aes(y=predicted),shape=1)+geom_line(mapping=aes(y=predicte
d))+
  scale_color_continuous(low="green",high="red")+
  geom_segment(aes(xend=gdpPercap,yend=predicted),alpha=.2)+

theme_bw()+facet_wrap(continent~.)+geom_ribbon(mapping=aes(ymin=test.data$pre
dicted-1.96*sd1,ymax=test.data$predicted+1.96*sd1), alpha = 0.3)+
  labs(title="plot for actual vs predicted values of lifeExpectancy of
different continents using lm",subtitle = "data from years 1993 to 2007")+
  theme(plot.title = element_text(hjust=0.5),plot.subtitle =
element_text(hjust=0.5))+
  scale_x_continuous(labels = scales::dollar)
```



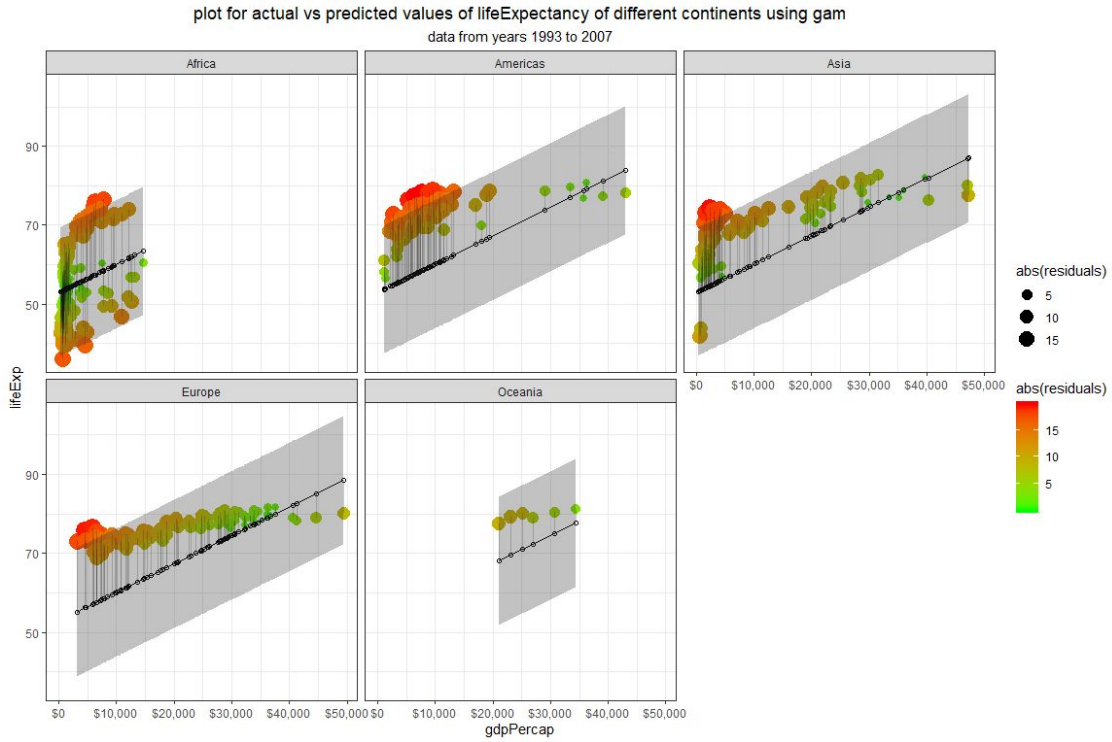

```
test.data$predicted <- predict(glm.fit,test.data)
test.data$residuals <- test.data$lifeExp-test.data$predicted
sd2 <- sd(test.data$predicted)
ggplot(test.data,mapping=aes(x=gdpPercap,y=lifeExp))+

geom_point(test.data,mapping=aes(x=gdpPercap,y=lifeExp,color=abs(residuals),s
ize=abs(residuals)))+

geom_point(mapping=aes(y=predicted),shape=1)+geom_line(mapping=aes(y=predicte
d))+
  scale_color_continuous(low="green",high="red")+
  geom_segment(aes(xend=gdpPercap,yend=predicted),alpha=.2)+

theme_bw()+facet_wrap(continent~.)+geom_ribbon(mapping=aes(ymin=test.data$pre
dicted-1.96*sd2,ymax=test.data$predicted+1.96*sd2), alpha = 0.3)+
  labs(title="plot for actual vs predicted values of lifeExpectancy of
different continents using glm",subtitle = "data from years 1993 to 2007")+
  theme(plot.title = element_text(hjust=0.5),plot.subtitle =
element_text(hjust=0.5))+
  scale_x_continuous(labels = scales::dollar)
```



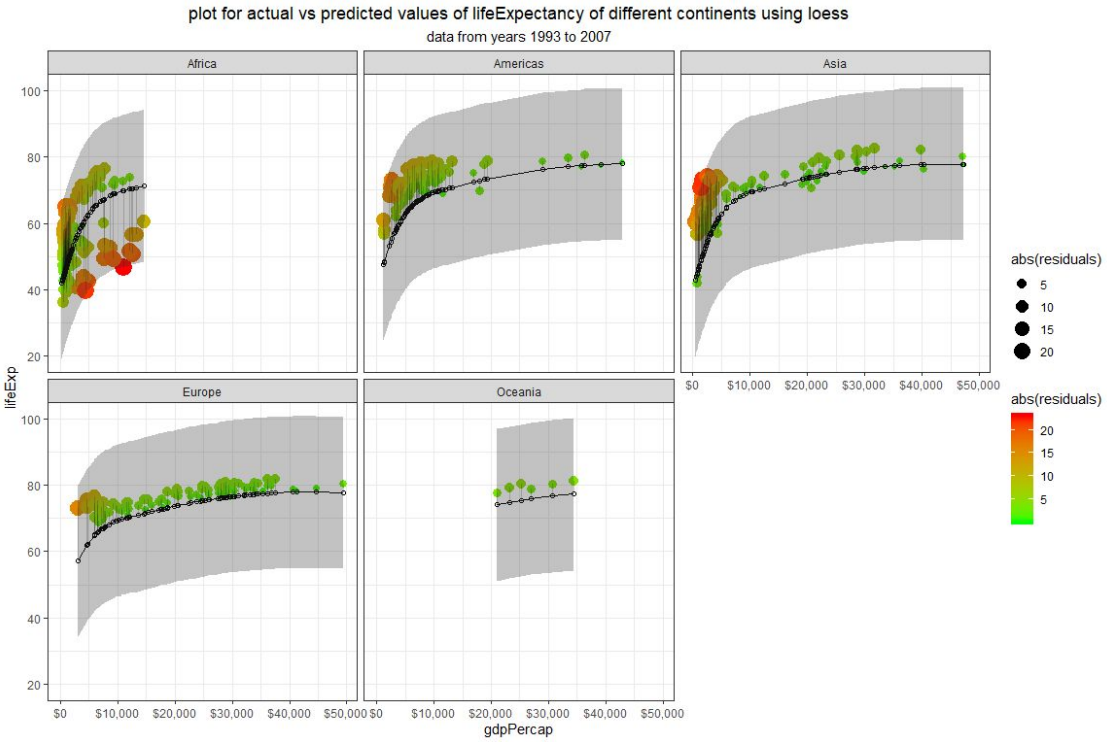

```
test.data$predicted <- predict(rlm.fit,test.data)
test.data$residuals <- test.data$lifeExp-test.data$predicted
sd3 <- sd(test.data$predicted)

ggplot(test.data,mapping=aes(x=gdpPercap,y=lifeExp))+

geom_point(test.data,mapping=aes(x=gdpPercap,y=lifeExp,color=abs(residuals),s
ize=abs(residuals)))+

geom_point(mapping=aes(y=predicted),shape=1)+geom_line(mapping=aes(y=predicte
d))+
  scale_color_continuous(low="green",high="red")+
  geom_segment(aes(xend=gdpPercap,yend=predicted),alpha=.2)+

theme_bw()+facet_wrap(continent~.)+geom_ribbon(mapping=aes(ymin=test.data$pre
dicted-1.96*sd3,ymax=test.data$predicted+1.96*sd3), alpha = 0.3)+
  labs(title="plot for actual vs predicted values of lifeExpectancy of
different continents using rlm",subtitle = "data from years 1993 to 2007")+
  theme(plot.title = element_text(hjust=0.5),plot.subtitle =
element_text(hjust=0.5))+
  scale_x_continuous(labels = scales::dollar)
```




```
test.data$predicted <- predict(gam.fit,test.data)
test.data$residuals <- test.data$lifeExp-test.data$predicted
sd4<-sd(test.data$predicted)

ggplot(test.data,mapping=aes(x=gdpPercap,y=lifeExp))+

geom_point(test.data,mapping=aes(x=gdpPercap,y=lifeExp,color=abs(residuals),s
ize=abs(residuals)))+

geom_point(mapping=aes(y=predicted),shape=1)+geom_line(mapping=aes(y=predicte
d))+
  scale_color_continuous(low="green",high="red")+

geom_segment(aes(xend=gdpPercap,yend=predicted),alpha=.2)+geom_ribbon(mapping
=aes(ymin=test.data$predicted-1.96*sd4,ymax=test.data$predicted+1.96*sd4),
alpha = 0.3)+
  theme_bw()+facet_wrap(continent~.)+
  labs(title="plot for actual vs predicted values of lifeExpectancy of
different continents using gam",subtitle = "data from years 1993 to 2007")+
  theme(plot.title = element_text(hjust=0.5),plot.subtitle =
element_text(hjust=0.5))+
  scale_x_continuous(labels = scales::dollar)
```




```
test.data$predicted <- predict(loess.fit,test.data)
test.data$residuals <- test.data$lifeExp-test.data$predicted
test.data <- test.data[complete.cases(test.data),]
sd5<-sd(test.data$predicted)

ggplot(test.data,mapping=aes(x=gdpPercap,y=lifeExp))+

geom_point(test.data,mapping=aes(x=gdpPercap,y=lifeExp,color=abs(residuals),s
ize=abs(residuals)))+

geom_point(mapping=aes(y=predicted),shape=1)+geom_line(mapping=aes(y=predicte
d))+
  scale_color_continuous(low="green",high="red")+

geom_segment(aes(xend=gdpPercap,yend=predicted),alpha=.2)+geom_ribbon(mapping
=aes(ymin=test.data$predicted-1.96*sd5,ymax=test.data$predicted+1.96*sd5),alp
ha=0.3)+
  theme_bw()+facet_wrap(continent~.)+
  labs(title="plot for actual vs predicted values of lifeExpectancy of
different continents using loess",subtitle = "data from years 1993 to 2007")+
  theme(plot.title = element_text(hjust=0.5),plot.subtitle =
element_text(hjust=0.5))+
  scale_x_continuous(labels = scales::dollar)
```


plot for actual vs predicted values of lifeExpectancy of different continents using loess
data from years 1993 to 2007
Africa
Americas
Asia
Europe
Oceania
lifeExp
gdpPercap
abs(residuals)
5
10
15
20
abs(residuals)
20
15
10
5