SAGAR KUMAR

194161013

Importing libraries

```
library(gapminder)
library(magrittr)
library(dplyr)
data("gapminder")
```

Question

Training data will incor-porate data from 1952 to 1992 for each continent and for testing/prediction use data from 1993 to 2007

```
training_data<-filter(gapminder,year==1952 | year==1957 |year==1962
|year==1967 | year==1972 |year==1977|year==1982 |year==1987 |year==1992 )

testing_data<-filter(gapminder,year==1997 | year==2002 |year==2007)

training_data<-data.frame(training_data)
testing_data<-data.frame(testing_data)
```

Q1

Train the linear model, where column gdpPercap is input data and predicted value is Column lifeExp. In R Markdown, print the summary of the learned model. Based on the data, we propose a hypothesis,"lifeExp is dependent on gdpPerCap". Analyze the summary of model and explain, Is Hypothesis correct?

lm

```
lm.fit=lm(lifeExp~gdpPercap,data=training_data)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = lifeExp ~ gdpPercap, data = training_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
```

Analyze the Summary of Model:

##Residuals are essentially the difference between the actual observed response values and the response values that the model predicted.

from summary of the Residuals we can see that the distribution of the residuals is not symmetry.(left skewed) **max=18.419 ,min=-76.914,and mean=2.143**

Linear regression makes several assumptions about the Residuals(Errors), such as :

1.The errors are uncorrelted with each other.Violation of this assumption can lead to very misleading assesments of the strength of the regression.

2.The expected value of errors is zero.

3.The assumption of homoscedasity. (Constant Variance) 4.Distribution of errors (Residulals) follows normal distribution.

Mean and Variance of the residual

```
mean(lm.fit$residuals)
```

```
## [1] -9.310497e-16
```

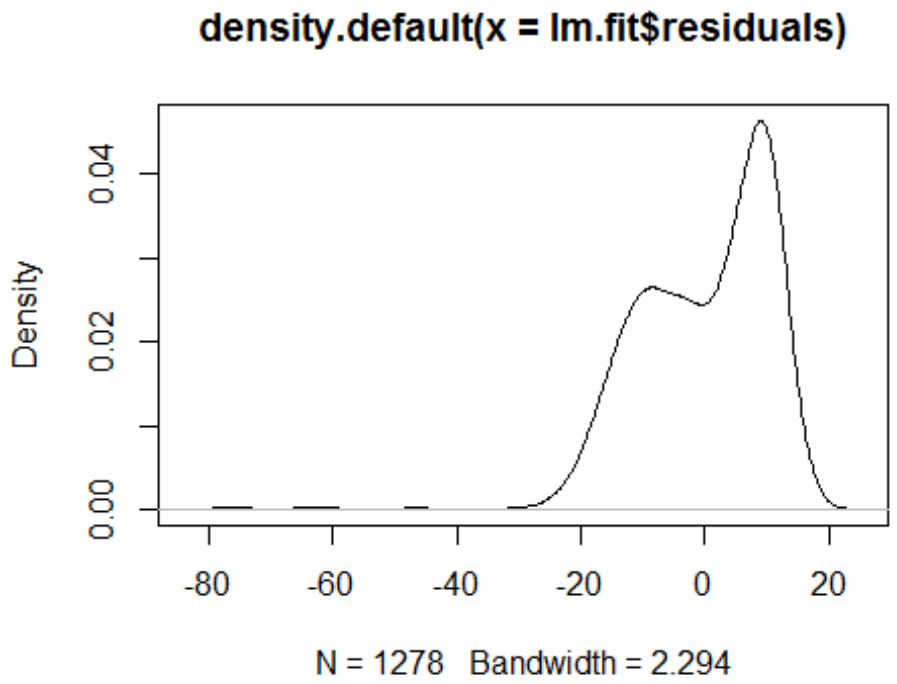
```
var(lm.fit$residuals)
```

```
## [1] 113.6327
```

Expected value(mean) of the residulas is: -9.310497e-16 (it means residuals expected value is almost zero) which satisfy our assumption. Variance of the residulas is: 113.6327

I mentioned that the residuals follow a normal distribution. let's check whether residuals follows normal distribution or not ?

```
plot(density(lm.fit$residuals))
```



We can clearly see that the residuals doesn't follows Normal Distribution , but is a multi modal normal distribution which indicates that our model is not very good. And also we can see that density plot of residual is left skewed
(multimodal: mixture of two or more gaussain functions)

```
coef(lm.fit)
```

```
##  (Intercept)    gdpPercap
## 5.283869e+01 7.232742e-04
```

Coefficient Estimate : This is the expected change in Y per unit change in X. In our model, one unit change in gdpPercap contributes to 7.232742e-04 unit change in lifeExp .

If the estimate value is 0.0 ,it means it adds no significance to the model and is considered worthless.

Lower the p value allow us to reject null hypothesis.i.e the estimate of any explanatory variable is 0 or there is no relationship between the predictor and the response. Asterisks mark aside p value define significance of value, lower the value ,higher is its significance and higher is the number of asterisks. Hence, if we see a small p-value, then we can infer that there is an association between the predictor and the response. We reject the null hypothesis—that is, we declare a relationship to exist between X and Y —if the p-value is small enough.

Typically, a p-value of 5% or less is a good cut-off point.
A small p-value indicates that there is a relationship between the predictor and response variables

In our dataset gdpPercap is significant vvariales.

Standard error

The coefficient Standard Error measures the average amount that the coefficient estimates vary from the actual.
We expect minimum value of standard error.

Multiple R-squared R-square is a goodness-of-fit measure for linear regression models. It determines the percentage of variation in the response variable that is explained by variation in the explanatory variable. this is use to calculate how well the model is doing to explain the things.R-squared evaluates the scatter of the data points around the fitted regression line. Higher the values of R-square are desirables."How high is High" depends on the context.

It always lies between 0 and 1 (i.e.: a number near 0 represents a regression that does not explain the variance in the response variable well and a number close to 1 does explain the observed variance in the response variable).

In our model the Value of R-square is :0.2734

Adjusted R-squared The adjusted Rsquared increase only if the new terms improves the model more than would be expected by chance.
It decreases when a predictor improves the model by less then expected by chance.The adjusted Rsquared can be negative but it's usually not. It is always lower than The R-squared.

In our Model the value of Adjusted Rsquare is:0.2728

F-Statistic:
F-statistic is a good indicator of whether there is a relationship between our predictor and the response variables. We can also say that if there is greater relationship then our modelis good and not good otherwise. It tells us whether the modelis significant or not. The model is significant if any of the coefficients are non-zero.

glm

```
glm.fit=glm(lifeExp~gdpPercap,data=training_data)
summary(glm.fit)
```

```
##
## Call:
## glm(formula = lifeExp ~ gdpPercap, data = training_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
```

Interpretation

Note: In linear Regression,inference is exact . This is due to the nice properties of the normal, least squares estimation and linearity. As a consequence, the distributions of the coefficients are perfectly known assuming that the assumptions hold. While in logistic regression, the inference is aymptotic. This means that the distributions of the coefficients are unknown except for large sample sizes n, for which we have approximations. The reason is the more complexity of the model in terms of nonlinearity. This is the usual situation for the majority of the regression models. • Estimate: the intercept B0 and the other beta coefficient estimates associated to each predictor variable • Std.Error: the standard deviation of the error of the coefficient estimates. This represents the accuracy of the coefficients. The larger the standard error, the less confident we are about the estimate. • z value: the z-statistic, which is the coefficient estimate (column 2) divided by the standard error of the estimate (column 3) • Pr(>|z|): The p-value corresponding to the z-statistic. The smaller the p-value, the more significant the estimate is.

Deviance • Intuitively, it measures the deviance of the fitted logistic model with respect to a perfect model. • The deviance is basically a measure of how much unexplained variation there is in our logistic regression model – the higher the value the less accurate the model. • As a consequence, the deviance is always larger or equal than zero, being zero only if the fit is perfect. • It compares the difference in probability between the predicted outcome and

the actual outcome for each case and sums these differences together to provide a measure of the total error in the model. • This is similar in purpose to looking at the total of the residuals (the sum of squares) in linear regression analysis in that it provides us with an indication of how good our model is at predicting the outcome. • Don't worry about the technicalities of this – as long as you understand the basic premise you'll be okay! • The deviance has little intuitive meaning because it depends on the sample size and the number of parameters in the model as well as on the goodness of fit. We therefore need a standard to help us evaluate its relative size. One way to interpret the size of the deviance is to compare the value for our model against a 'baseline' model.

Null Deviance When the model includes only intercept term, then the performance of the model is measured by the null deviance.

Residual Deviance When the model has included some variable,then the deviance is residual deviance.

Lower value of residual deviance points out that the model has become better when it has included other variables.

AIC

For AIC, a lower number is better, but there's no formal tests for it - you'll still have to make a subjective evaluation of what you think "better" means if you're choosing between two models, but the AIC values should at least provide some guidance

Fischer Scoring It tells how the model was estimated. The algorithm looks around to see if the fit would be improved by using different estimates. If it improves then it moves in that direction and then fits the model again. The algorithm stops when no significant additional improvement can be done. "Number of Fisher Scoring iterations" tells "how many iterations this algorithm run before it stopped". 17

gam

```
library(MASS)

## Warning: package 'MASS' was built under R version 3.6.3

##
## Attaching package: 'MASS'

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

library(mgcv)

## Loading required package: nlme

## Warning: package 'nlme' was built under R version 3.6.3

##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse

## This is mgcv 1.8-31. For overview type 'help("mgcv-package")'.

gam.fit=gam(lifeExp~gdpPercap,family=gaussian(),data=training_data)
summary(gam.fit)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## lifeExp ~ gdpPercap
##
## Parametric 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
##
##
## R-sq.(adj) =  0.273   Deviance explained = 27.3%
## GCV =  113.9  Scale est. = 113.72    n = 1278
```

gam Generalized additive Model In statistics, a generalized additive model (GAM) is a generalized linear model in which the linear predictor depends linearly on unknown

smooth functions of some predictor variables, and interest focuses on inference about these smooth functions

rlm

```
library(MASS)
library(mgcv)
rlm.fit=rlm(lifeExp~gdpPercap,data=training_data)
summary(rlm.fit)
```

```
##
## Call: rlm(formula = lifeExp ~ gdpPercap, data = training_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
```

rlm Robust regression Fit a linear model by robust regression using an M estimator. Robust regression is an iterative procedure that seeks to identify outliers and minimize their impact on the coefficient estimates. The amount of weighting assigned to each observation in robust regression is controlled by a special curve called an influence function.

loess

```
loess.fit=loess(lifeExp~gdpPercap,data=training_data)
summary(loess.fit)
```

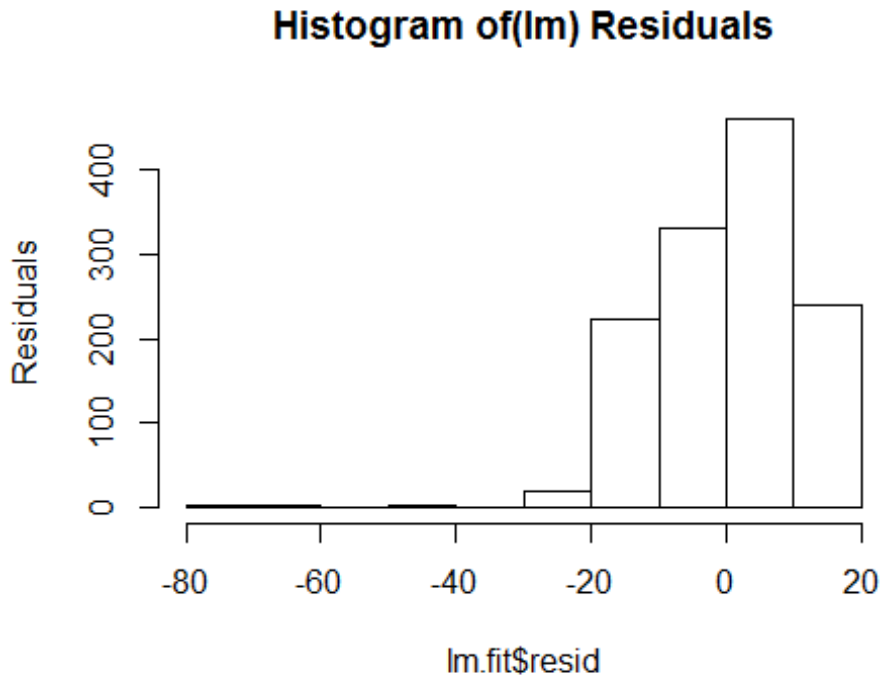
```
## Call:
## loess(formula = lifeExp ~ gdpPercap, data = training_data)
##
## Number of Observations: 1278
## Equivalent Number of Parameters: 5.5
## Residual Standard Error: 7.249
## Trace of smoother matrix: 6.03  (exact)
##
## Control settings:
##   span     :  0.75
##   degree   :  2
##   family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

Loess regression Fit a polynomial surface determined by one or more numerical predictors, using local fitting. Loess regression is a nonparametric technique that uses local weighted regression to fit a smooth curve through points in a scatter plot. Loess curves are can reveal trends and cycles in data that might be difficult to model with a parametric curve. Loess regression is one of several algorithms in SAS that can automatically choose a smoothing parameter that best fits the data. LOESS makes less efficient use of data than other least squares methods. It requires fairly large, densely sampled data sets in order to produce good models. This is because LOESS relies on the local data structure when performing the local fitting.

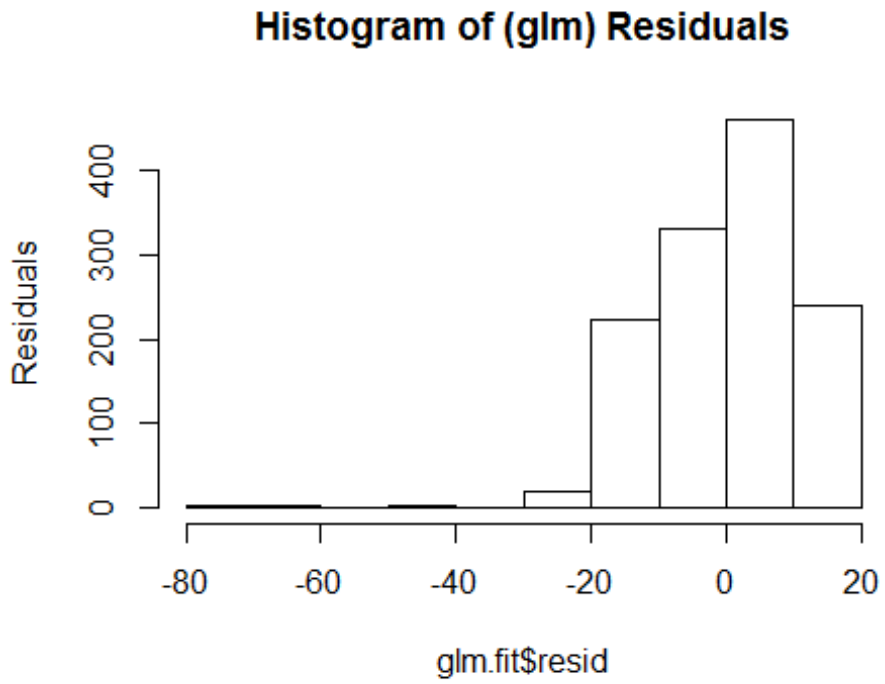
Q2

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

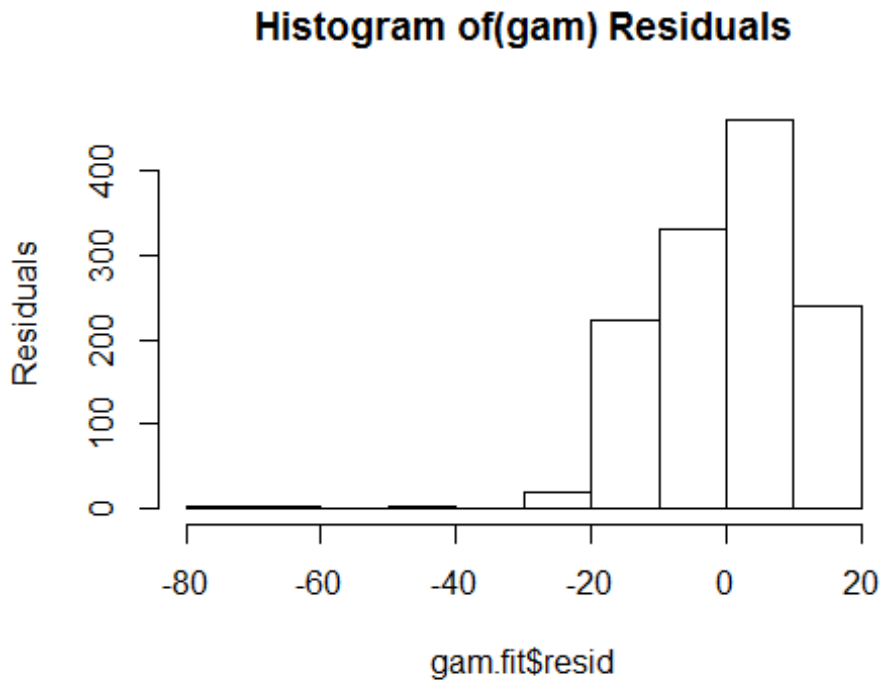
```
hist(lm.fit$resid, main="Histogram of(lm) Residuals",ylab="Residuals")
```



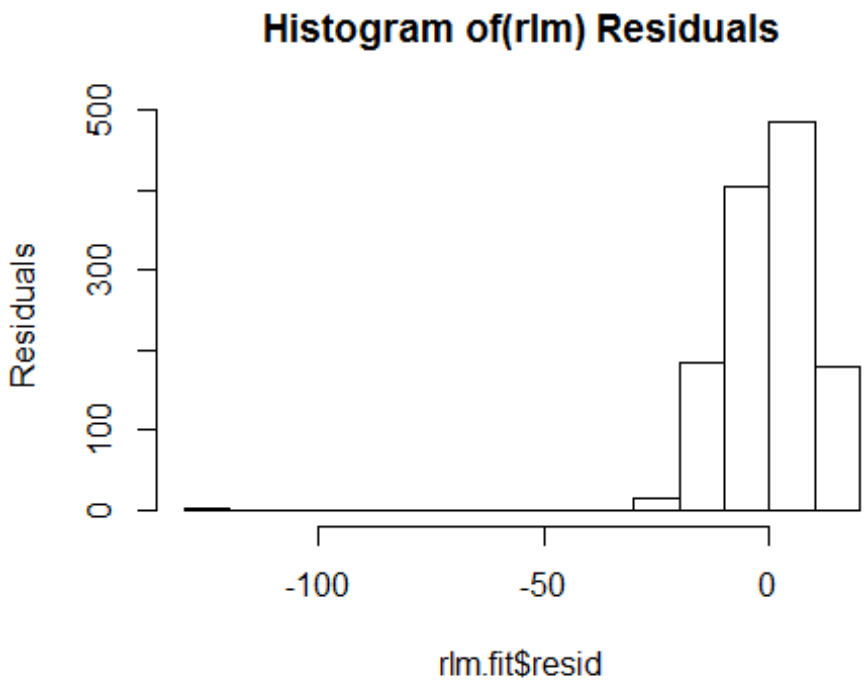

```
hist(glm.fit$resid, main="Histogram of (glm) Residuals",ylab="Residuals")
```



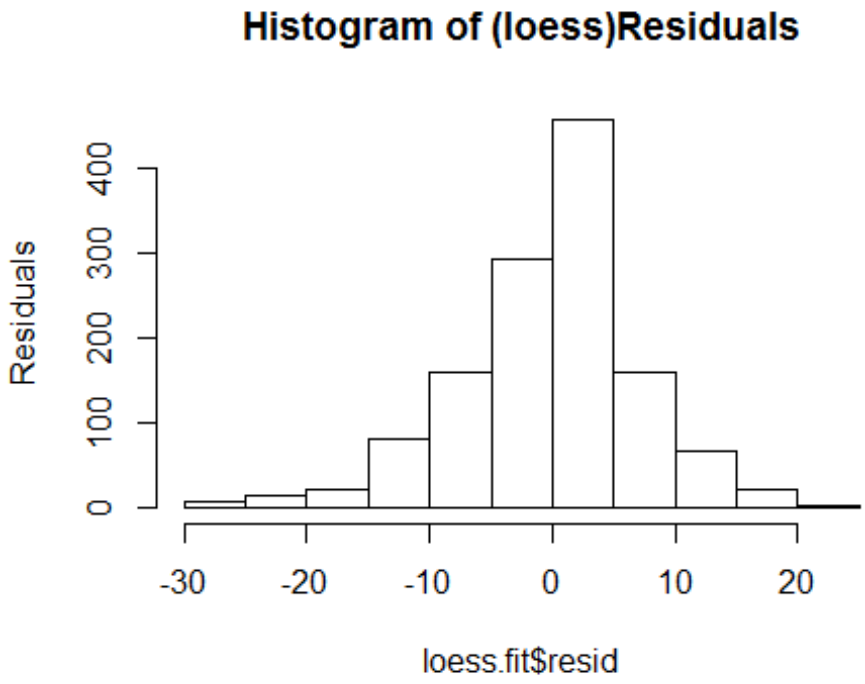

```r
hist(gam.fit$resid, main="Histogram of(gam) Residuals",ylab="Residuals")
```



```r
hist(rlm.fit$resid, main="Histogram of(rlm) Residuals",
 ylab="Residuals")
```




```r
hist(loess.fit$resid, main="Histogram of (loess)Residuals",
 ylab="Residuals")
```



loess method best fit the residuals and this is symmetric arount 0. and all other method is left skewed.

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.

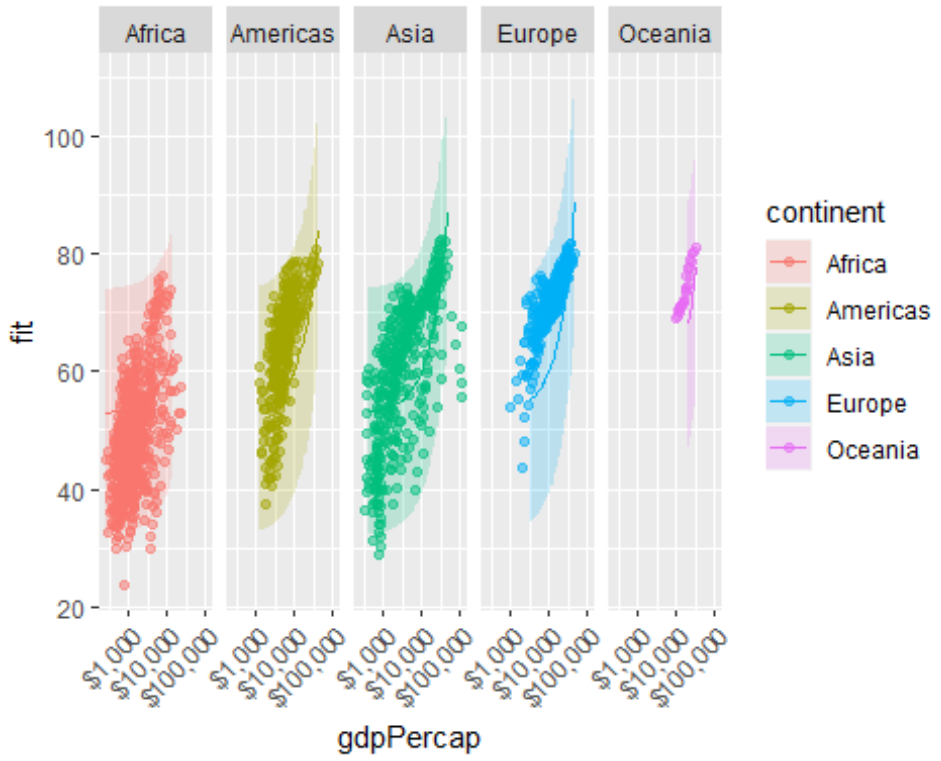
lm

```
pred_out <- predict(object = lm.fit,newdata =testing_data,interval =
"predict")
pred_df<-cbind(testing_data,pred_out)

library(ggplot2)
p <- ggplot(data = subset(pred_df, continent %in% c("Europe",
"Africa","Asia","Americas","Oceania")),
            aes(x = gdpPercap,
                y = fit, ymin = lwr, ymax = upr,
                color = continent,
                fill = continent,
                group = continent))



p + geom_point(data = subset(gapminder,
                             continent %in% c("Europe",
"Africa","Asia","Americas","Oceania")),
               aes(x = gdpPercap, y = lifeExp,
                   color = continent),
               alpha = 0.5,
               inherit.aes = FALSE) +
    geom_line() +
    geom_ribbon(alpha = 0.2, color = FALSE) +
    scale_x_log10(labels = scales::dollar)+facet_grid(~continent)+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

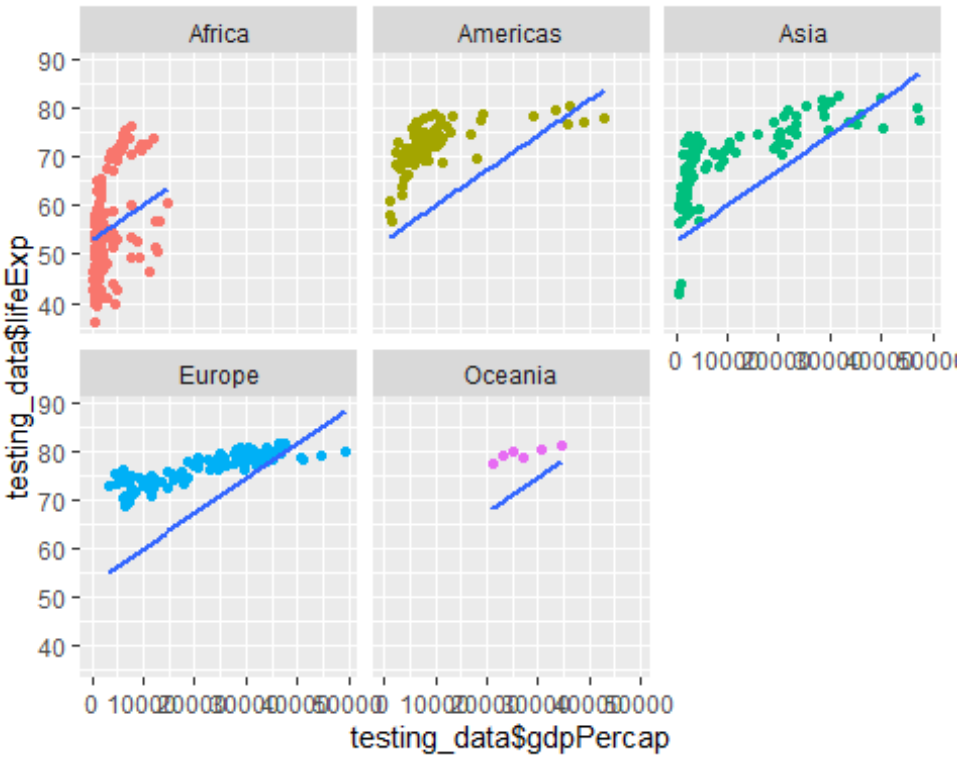


glm

```
predictedLifeExp<-predict(glm.fit,newdata = testing_data)
dframe <-data.frame(testing_data$gdpPercap,predictedLifeExp)

ggplot(data=dframe,mapping = aes(testing_data$gdpPercap))+
  geom_point(mapping = aes(testing_data$gdpPercap,testing_data$lifeExp,col =
testing_data$continent),show.legend = FALSE)+
  geom_smooth(mapping = aes(testing_data$gdpPercap,predictedLifeExp))+
  facet_wrap(~testing_data$continent)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

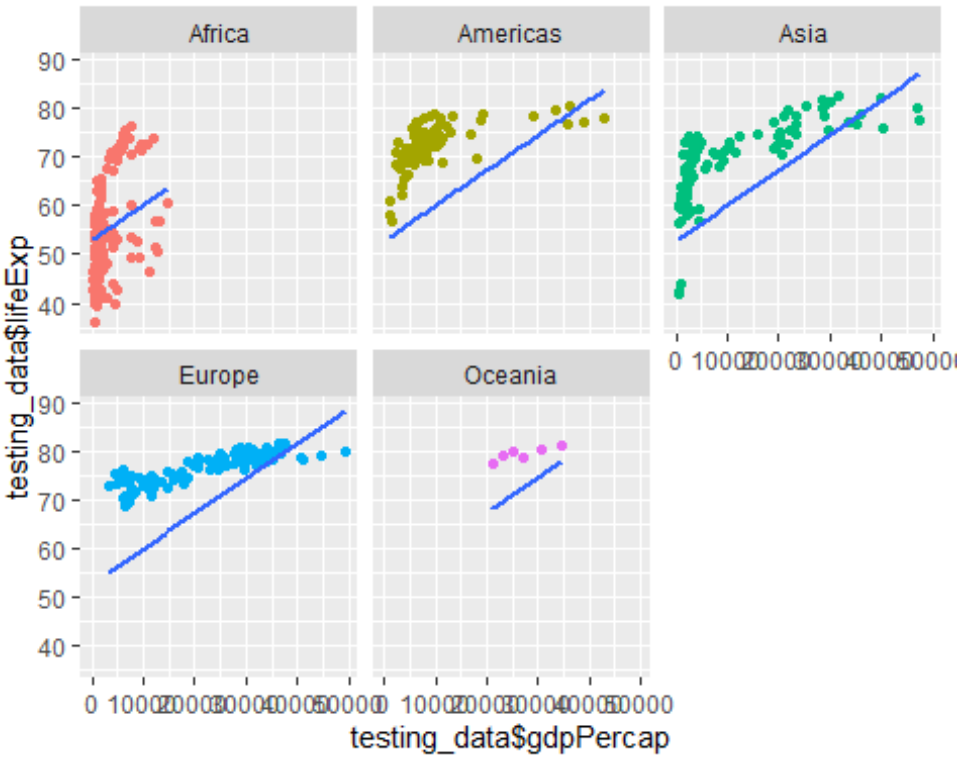


gam

```
predictedLifeExp<-predict(gam.fit,newdata = testing_data)
dframe <-data.frame(testing_data$gdpPercap,predictedLifeExp)

ggplot(data=dframe,mapping = aes(testing_data$gdpPercap))+
  geom_point(mapping = aes(testing_data$gdpPercap,testing_data$lifeExp,col =
testing_data$continent),show.legend = FALSE)+
  geom_smooth(mapping = aes(testing_data$gdpPercap,predictedLifeExp))+
  facet_wrap(~testing_data$continent)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

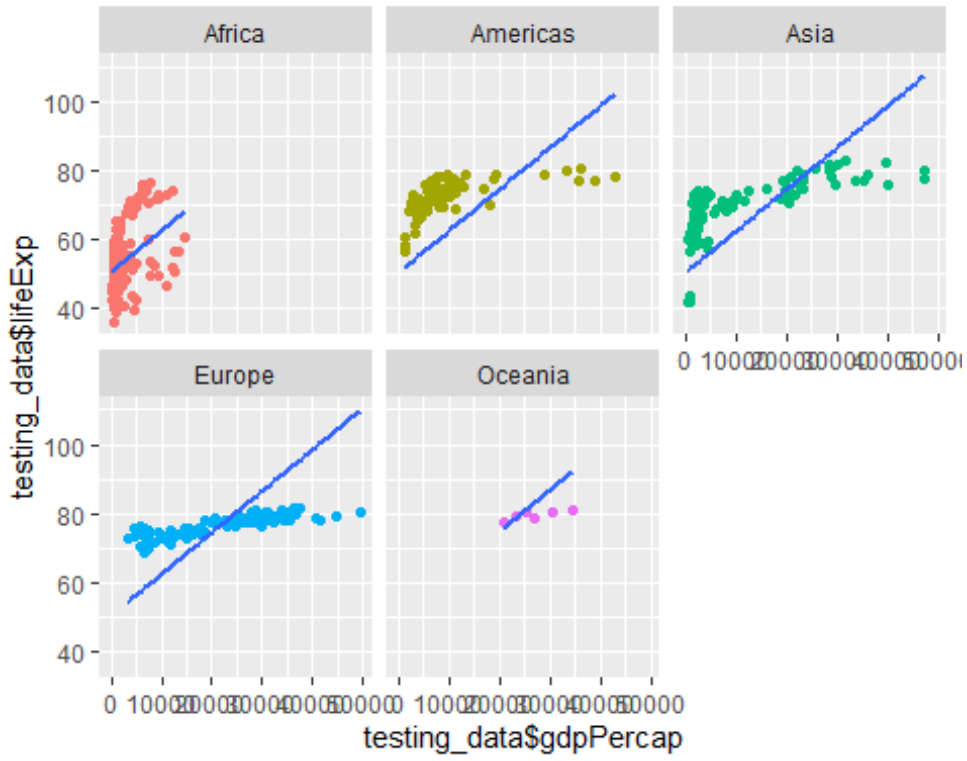


rlm

```
predictedLifeExp<-predict(rlm.fit,newdata = testing_data)
dframe <-data.frame(testing_data$gdpPercap,predictedLifeExp)

ggplot(data=dframe,mapping = aes(testing_data$gdpPercap))+
  geom_point(mapping = aes(testing_data$gdpPercap,testing_data$lifeExp,col =
testing_data$continent),show.legend = FALSE)+
  geom_smooth(mapping = aes(testing_data$gdpPercap,predictedLifeExp))+
  facet_wrap(~testing_data$continent)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

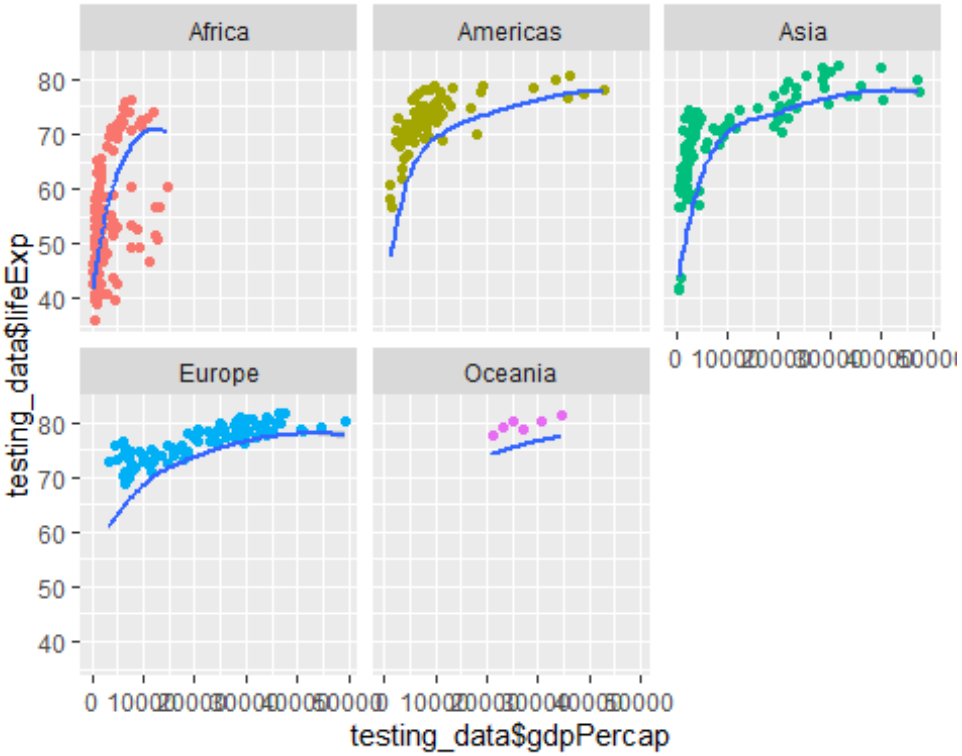


*loess**

```
predictedLifeExp<-predict(loess.fit,newdata = testing_data)
dframe <-data.frame(testing_data$gdpPercap,predictedLifeExp)

ggplot(data=dframe,mapping = aes(testing_data$gdpPercap))+
  geom_point(mapping = aes(testing_data$gdpPercap,testing_data$lifeExp,col =
testing_data$continent),show.legend = FALSE)+
  geom_smooth(mapping = aes(testing_data$gdpPercap,predictedLifeExp))+
  facet_wrap(~testing_data$continent)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```


Africa
Americas
Asia
Europe
Oceania
testing_data$lifeExp
testing_data$gdpPercap
80
70
60
50
40
0
10000
20000
30000
40000
50000