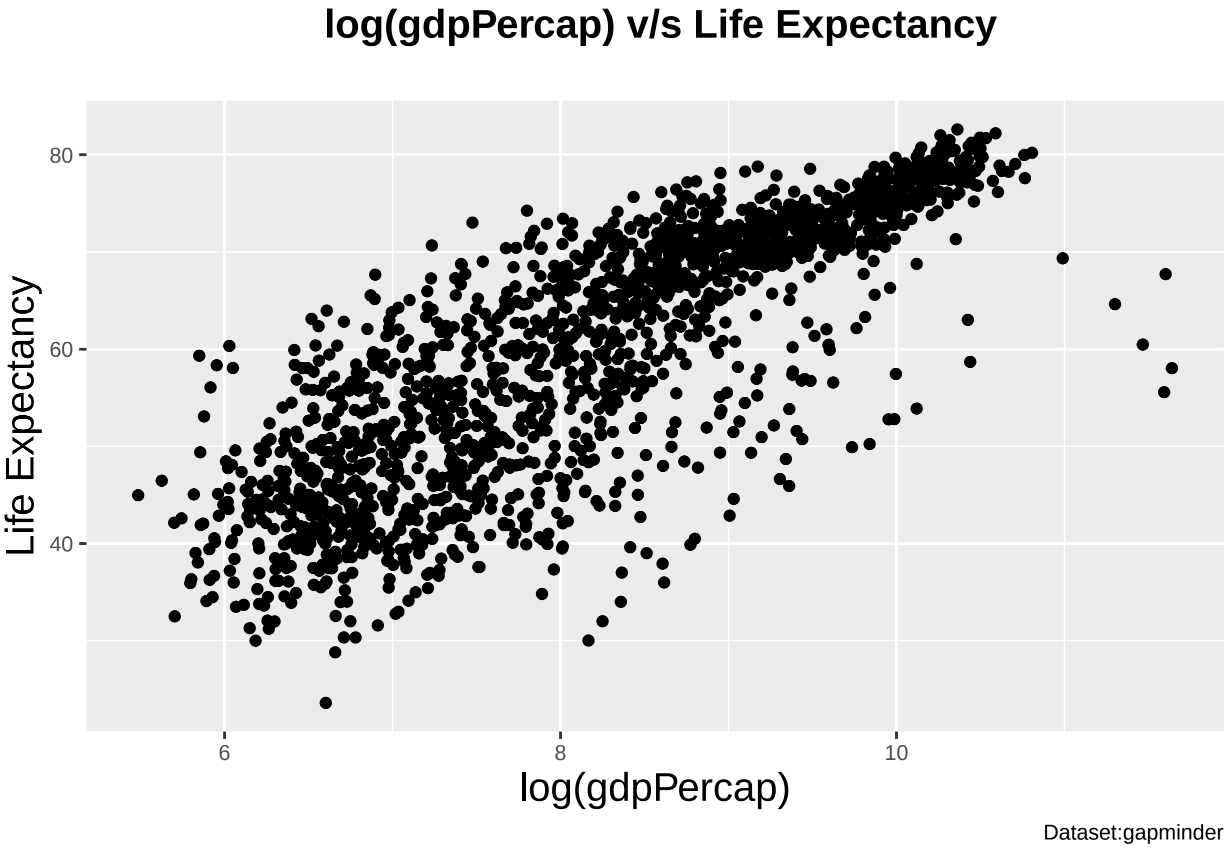
Assignment 3 : Data Visualization

Sagar Kumar | 194161013

Question1:

Using the gapminder dataset, plot a scatter plot with log(gdpPercap) on x-axis and lifeExp on y-axis.

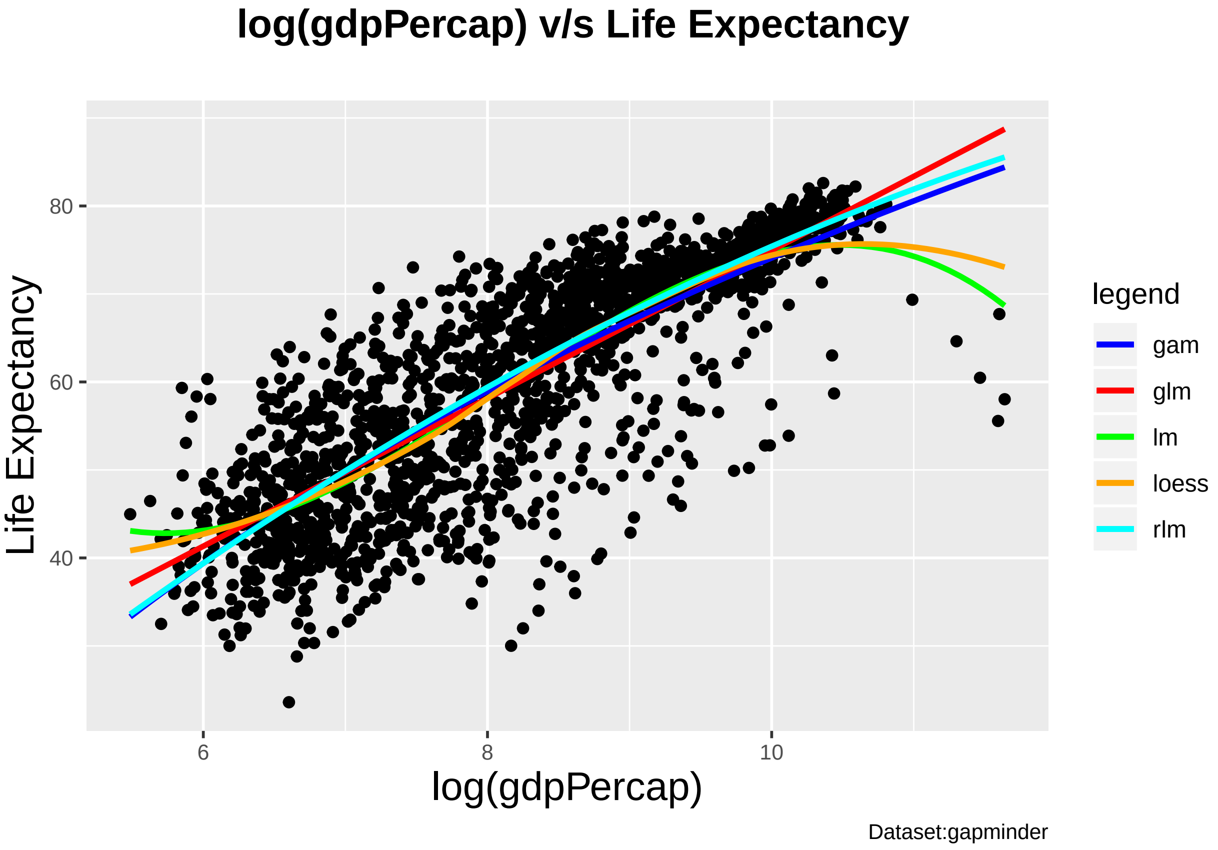
```r
ggplot(data=gapminder)+
  geom_point(aes(x=log(gdpPercap),y=lifeExp))+
   labs(x="log(gdpPercap)", y="Life Expectancy",
       title=" log(gdpPercap) v/s Life Expectancy",
       subtitle=" ",
       caption="Dataset:gapminder")+
  theme
```



Clearly,we can see that Data is monotonically increasing.

Generate a graph with following different smoothing methods (Add legend for clear distinction of each smooth curve).

```
ggplot(data=gapminder,mapping=aes(x=log(gdpPercap),y=lifeExp))+
  geom_point()+
  geom_smooth(method="lm",formula = y~poly(x,3),show.legend = TRUE,aes(colour="lm"),se=FALSE)+
  geom_smooth(method="glm",show.legend = TRUE,aes(colour="glm"),se=FALSE)+
  geom_smooth(method="gam",show.legend = TRUE,formula=y~log(x),aes(colour="gam"),se=FALSE)+
  geom_smooth(method="loess",show.legend = TRUE,aes(colour="loess"),se=FALSE)+
  geom_smooth(method="rlm",show.legend=TRUE,aes(colour="rlm"),formula = y~poly(x,2),se=FALSE)+
  scale_colour_manual(name="legend",values = c("blue","red","green","orange","cyan"))+
  labs(x="log(gdpPercap)", y="Life Expectancy",
       title=" log(gdpPercap) v/s Life Expectancy",
       subtitle=" ",
       caption="Dataset:gapminder")+
  theme
```



legend:
gam<-BLUE
glm<-RED
lm<-GREEN
loess<-orange
rlm<-cyan
In this we have five different smoothing line.
All smootheing line shows the trend between lifeExp and log(gdpPercap).

Clearly,we can see that that there is linear relationship between lifeExp and log(gdpPercap).When we use three degree polynomial curve for linear model,In gam we use log(y-axis) and In rlm two degree polynomial.

Explain the following

Smoothing

Smoothing is a very powerful technique used all across data analysis. It is designed to estimate f(x) when the shape is unknown, but assumed to be smooth. The general idea is to group data points that are expected to have similar expectations and compute the average, or fit a simple parametric model.

Smoothing method (function) to use, accepts either a character vector, e.g. "auto", "lm", "glm", "gam", "loess" or a function, e.g. MASS::rlm or mgcv::gam, stats::lm, or stats::loess

lm
linear Model
lm is used to fit linear models.
It can be used to carry out regression, single stratum analysis of variance and
analysis of covariance

glm
Generalized linear Model

The generalized linear model is a flexible generalization of ordinary linear regression that allows for response variables that have error distribution models other than a normal distribution.

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
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 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.

Which of these methods cannot be used for a large dataset and why?
Ans: **loess**

The memory usage of this implementation of loess is roughly quadratic in the number of points, with 1000 points taking about 10Mb.

Smoothing method (function) to use, accepts either a character vector. e.g. "auto", "lm", "glm", "gam", "loess" or a function, e.g. MASS::rlm or mgcv::gam, stats::lm, or stats::loess.

For method = "auto" the smoothing method is chosen based on the size of the largest group (across all panels). stats::loess() is used for less than 1,000 observations. otherwise mgcv::gam() is used with formula = y ~ s(x, bs = "cs"). Somewhat anecdotally, **loess** gives a better appearance, but is O(Nˆ2) in memory, so **does not work for larger datasets.**

If you have fewer than 1,000 observations but want to use the same gam() model that method = "auto" would use, then set method = "gam", formula = y ~ s(x, bs = "cs").