Machine_Learning_PS1

Hao Ran Li, Feiwen Liang, Leila Lan, Susu Zhu, Yitao Hu

04/04/2020

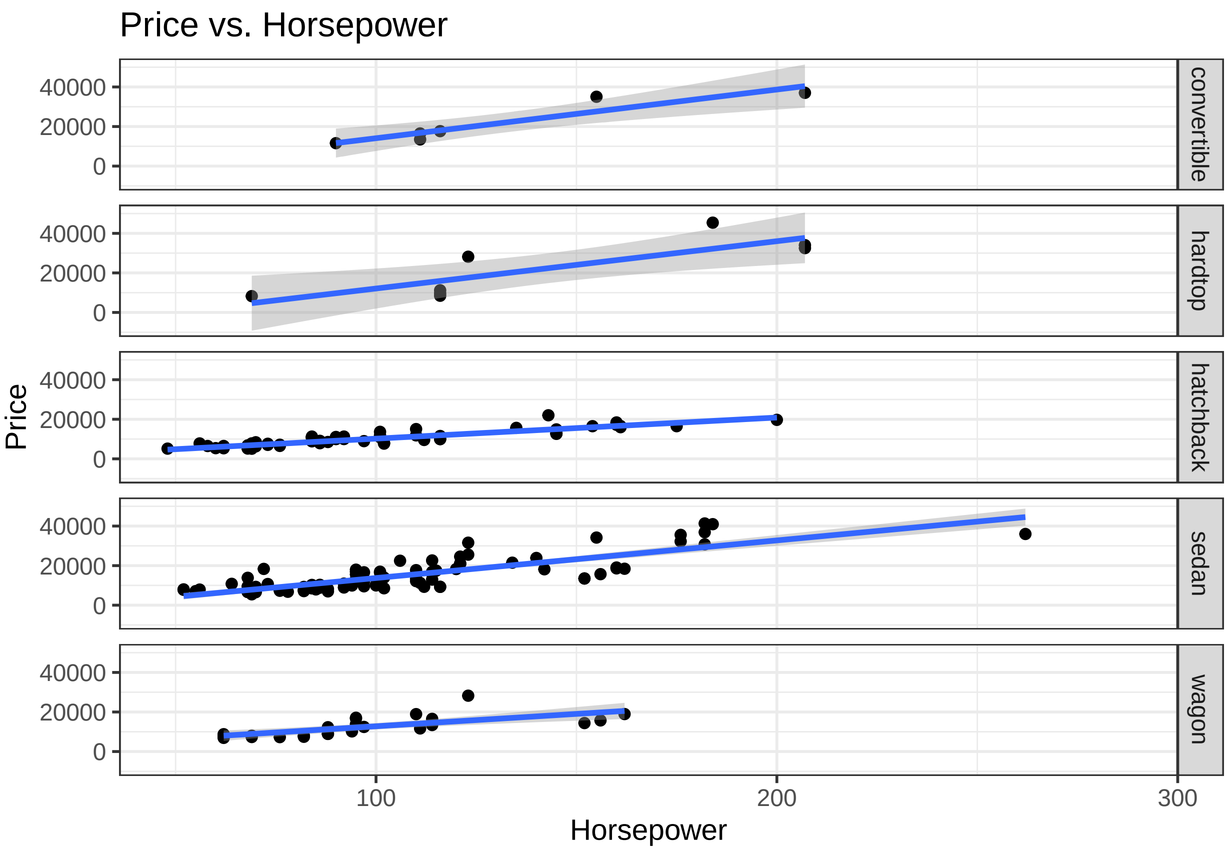
```
#import library and data
 library(ggplot2)
library(readr)
car_data=read_csv("imports-85.csv",
   col_types = cols(horsepower = col_double(),
      price = col_double(),
      `engine-size` = col_double(),
      `city-mpg` = col_double()))
```

Question1

1

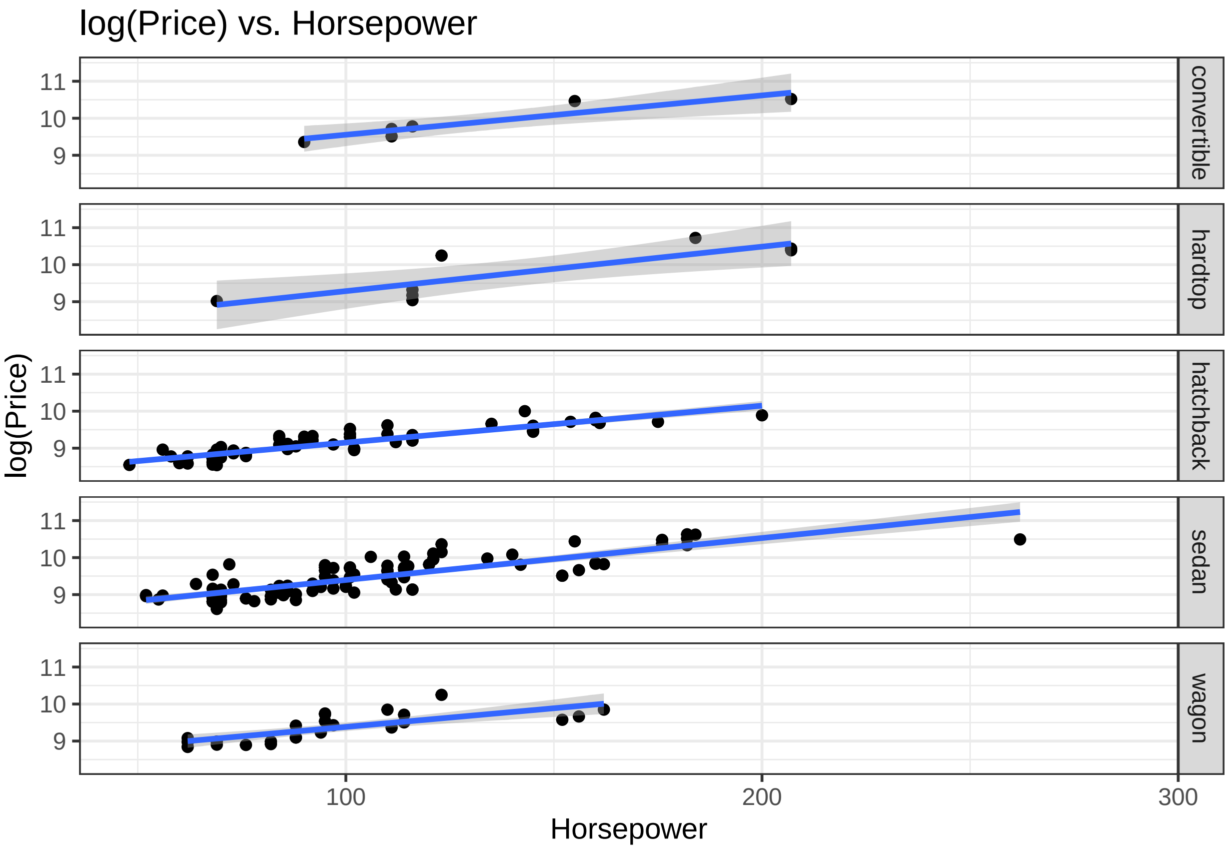
Plot the scatterplot b/w price and horsepower where bodystyle as a discrete variable.

```
qplot(x = horsepower,y = price,facets = `body-style`~.,data = car_data,main = "Price vs. Horsepower",xla
```



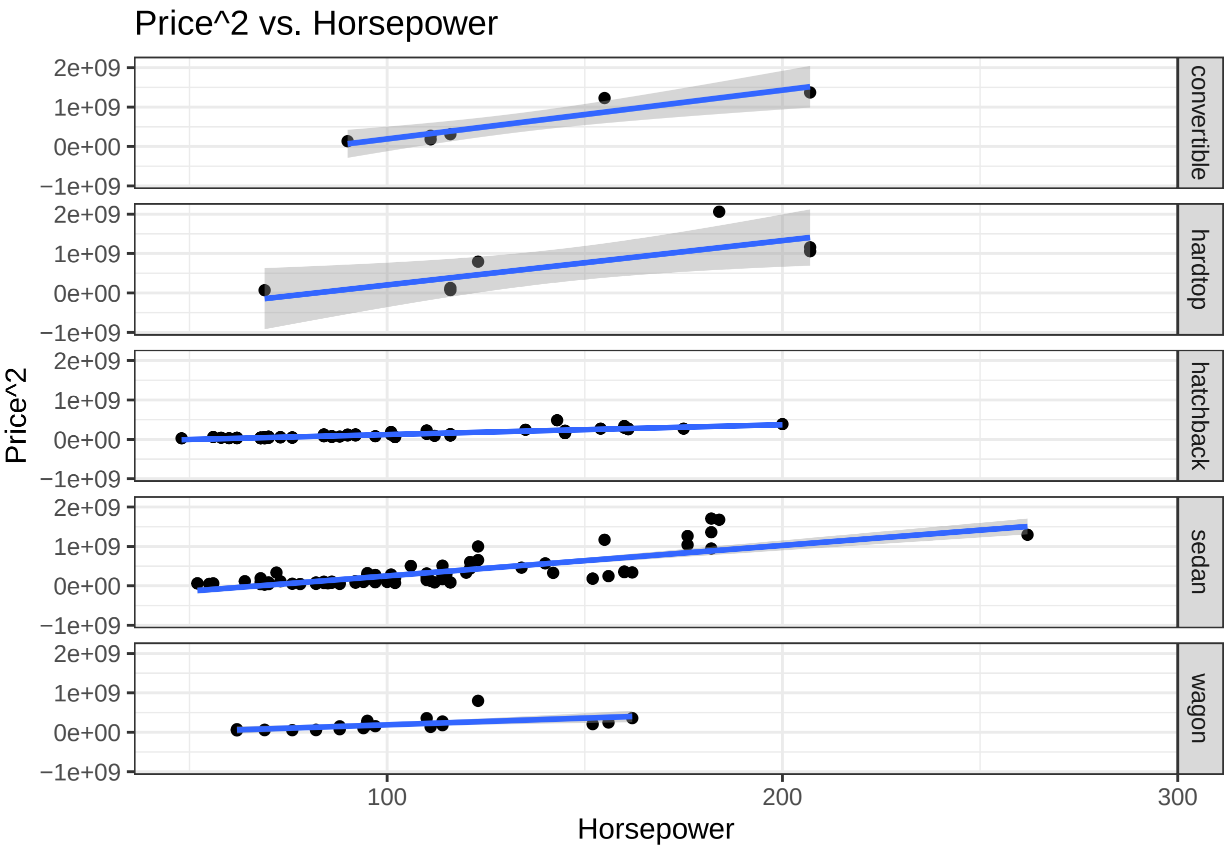
Plot the scatterplot b/w log(price) and horsepower where bodystyle as a discrete variable.

```
qplot(x = horsepower,y = log(price),facets = `body-style`~.,data = car_data,main = "log(Price) vs. Hors
```



Plot the scatterplot b/w price^2 and horsepower where bodystyle as a discrete variable.

```
qplot(x = horsepower,y = price^2,facets = `body-style`~.,data = car_data,main = "Price^2 vs. Horsepower
```



From the graphs above, we cannot see any clear relationship b/w bodystyle and price beyond horsepower.

2

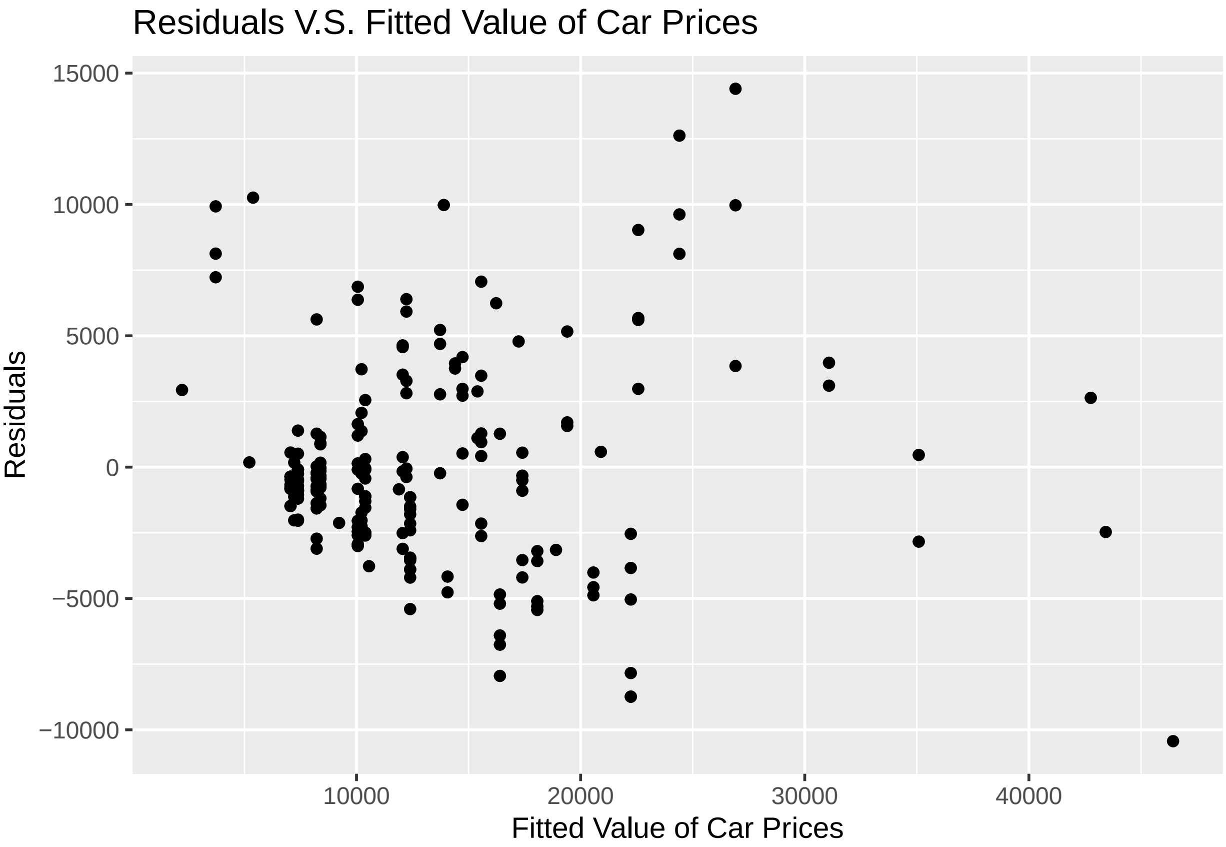
I will regress car prices on engine size

```
PriceOvereEngineSize=lm(formula = car_data$price~car_data$`engine-size`)
summary(PriceOvereEngineSize)
```

```
##
## Call:
## lm(formula = car_data$price ~ car_data$`engine-size`)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10433.0  -2249.4   -469.8   1370.6  14404.6
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -7963.339    884.835   -9.00   <2e-16 ***
## car_data$`engine-size`   166.860      6.629   25.17   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3895 on 199 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.761,  Adjusted R-squared:  0.7598
## F-statistic: 633.5 on 1 and 199 DF,  p-value: < 2.2e-16
```

Plot the residuals over fitted

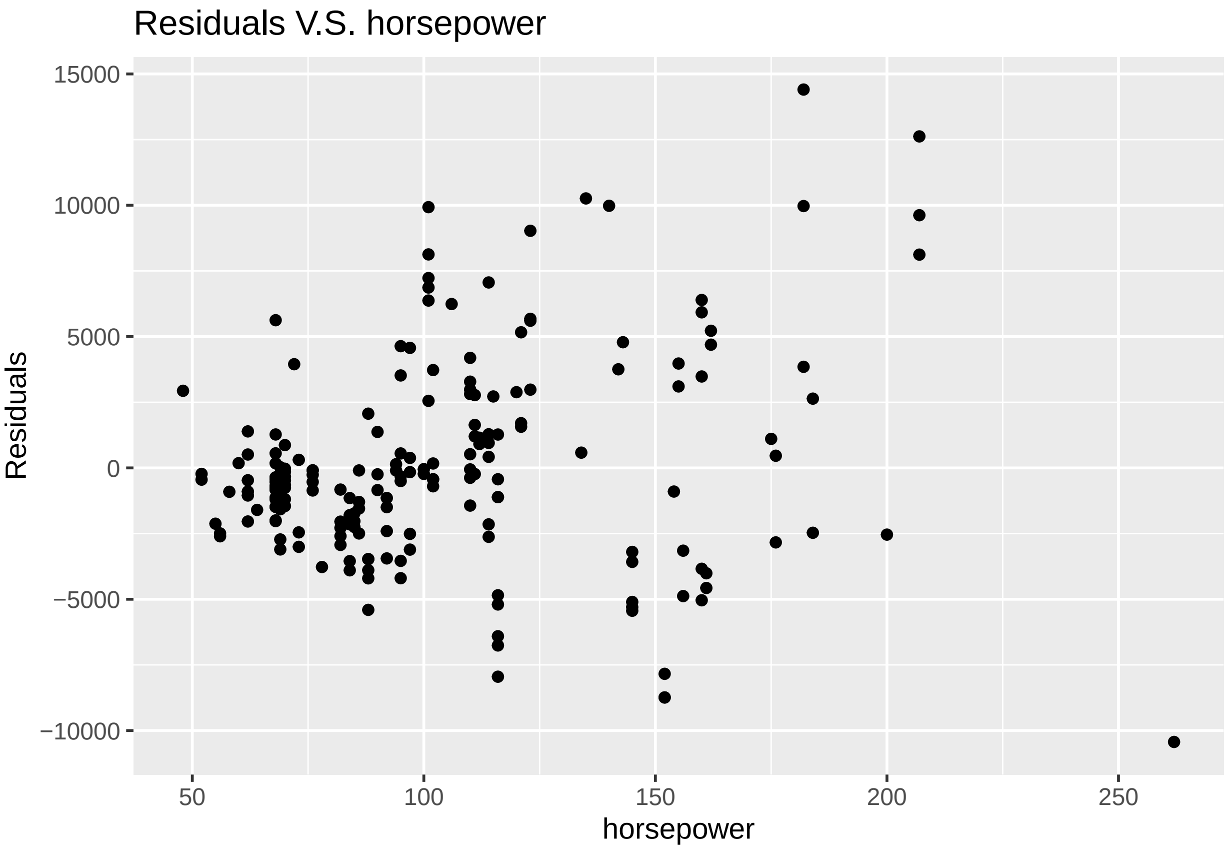
```
qplot(x = PriceOvereEngineSize$fitted.values,y = PriceOvereEngineSize$residuals,main = 'Residuals V.S. F
```



From the graph above, we observe that the residuals tend to randomly distributed given any level of Fitted value, and that no non-linear relationship exist b/w the residuals and the fitted values. Therefore, we can conclude that our linear model is appropriate in this case.

Plot the residuals over horsepower

```
qplot(x = car_data$horsepower[which(!is.na(car_data$price))],y = PriceOvereEngineSize$residuals,main = 
```

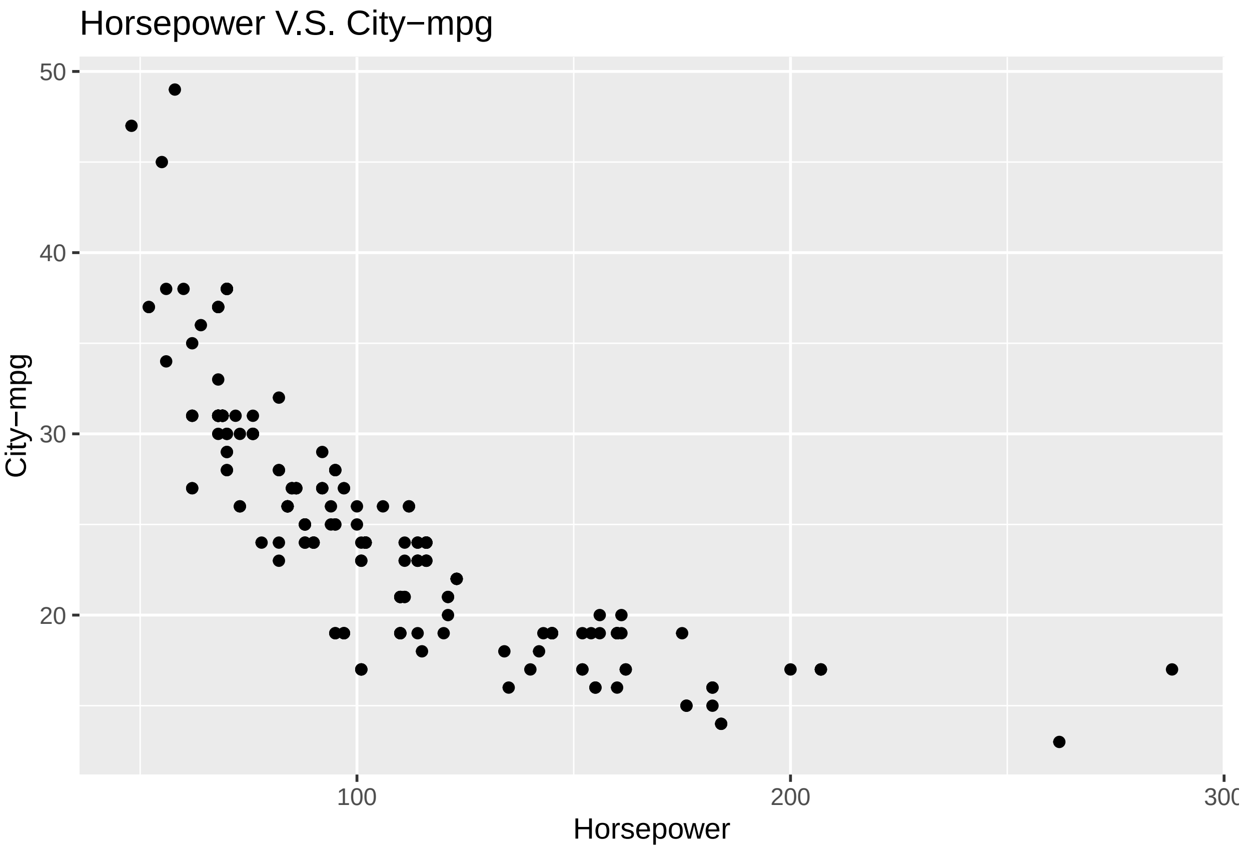


From the graph above, we can observe that the variance of residuals tends to increases with horsepower. Therefore, we can conclude that the horsepower can be an omitted variable to our orginial linear model.

3

Scatterplot between city-mpg and horse power

```
qplot(x = car_data$horsepower,y = car_data$`city-mpg`,main = 'Horsepower V.S. City-mpg',xlab = 'Horsepo
```



Regress City-mpg on horsepower

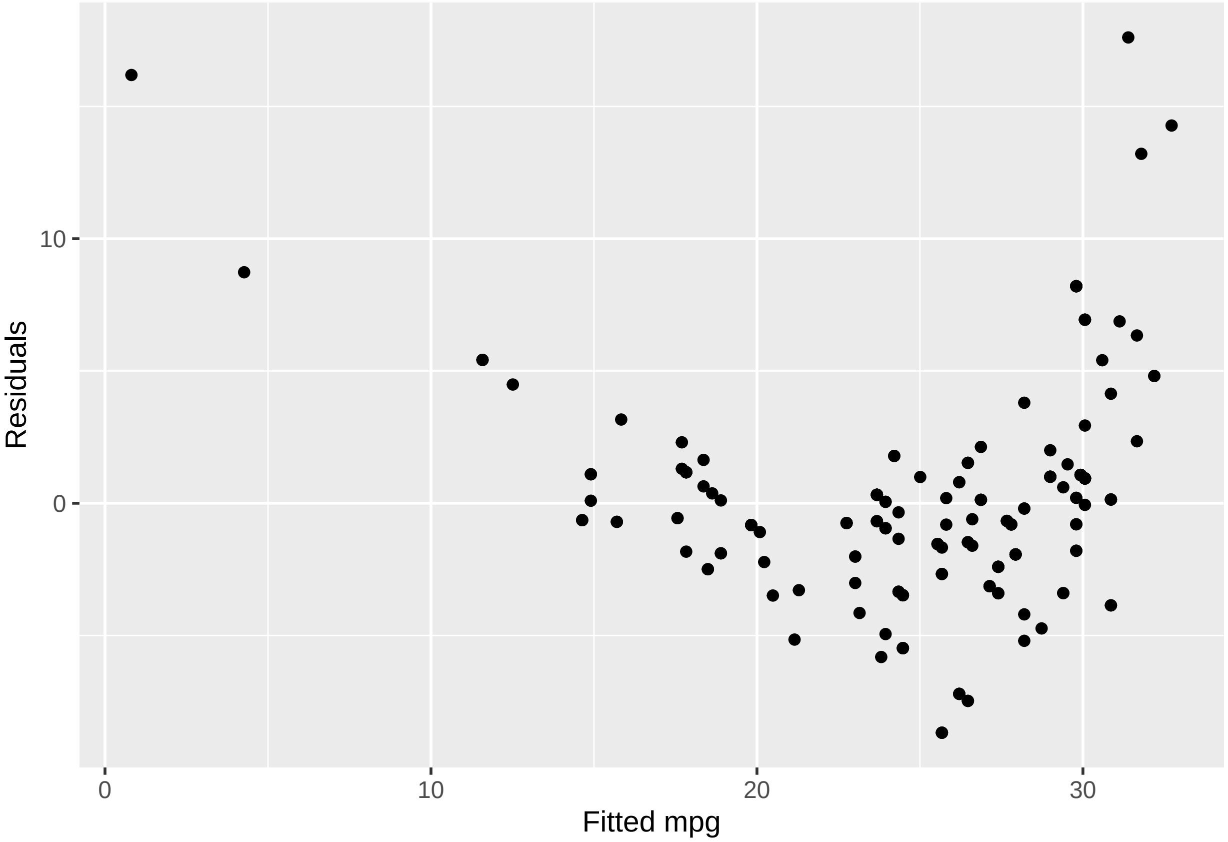
```
MPGonHorsepower=lm(formula = car_data$`city-mpg`~car_data$horsepower)
summary(MPGonHorsepower)
```

```
##
## Call:
## lm(formula = car_data$`city-mpg` ~ car_data$horsepower)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6743 -1.9346 -0.3447  1.0710 17.6085
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         39.103083   0.774598   50.48   <2e-16 ***
## car_data$horsepower -0.132958   0.006945  -19.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.92 on 201 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.6458, Adjusted R-squared:  0.644
## F-statistic: 366.5 on 1 and 201 DF,  p-value: < 2.2e-16
```

The regression is inconsistent with the scatterplot. From the scatterplot, we can observe that city-mpg is negatively correlated with horsepower, but the relationship is non-linear.

We can also check the appropriateness of linear model by ploting the residuals over the fitted values.

```
qplot(x = MPGonHorsepower$fitted.values,y = MPGonHorsepower$residuals,
      xlab = 'Fitted mpg',ylab = 'Residuals')
```



Question2

```
#import data and libraries
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 3.5.2
```

```
library(foreign)
```

```
## Warning: package 'foreign' was built under R version 3.5.2
```

```
StockRetAcct_DT= as.data.table(read.dta("StockRetAcct_insample.dta"))
#set the key as firm ID and year
setkey(x = StockRetAcct_DT,FirmID,year)
#creat returns
StockRetAcct_DT[,AnnRet:=exp(lnAnnRet)-1]
```

a

First we need to define the issuance ranking variable. To make the strategy tradable, we need to do it in a loop. Then, we value-weighted for each portfolio for each year, and then average across years

```
# loop through the years in the data base
for (i in 1981:2014)
```

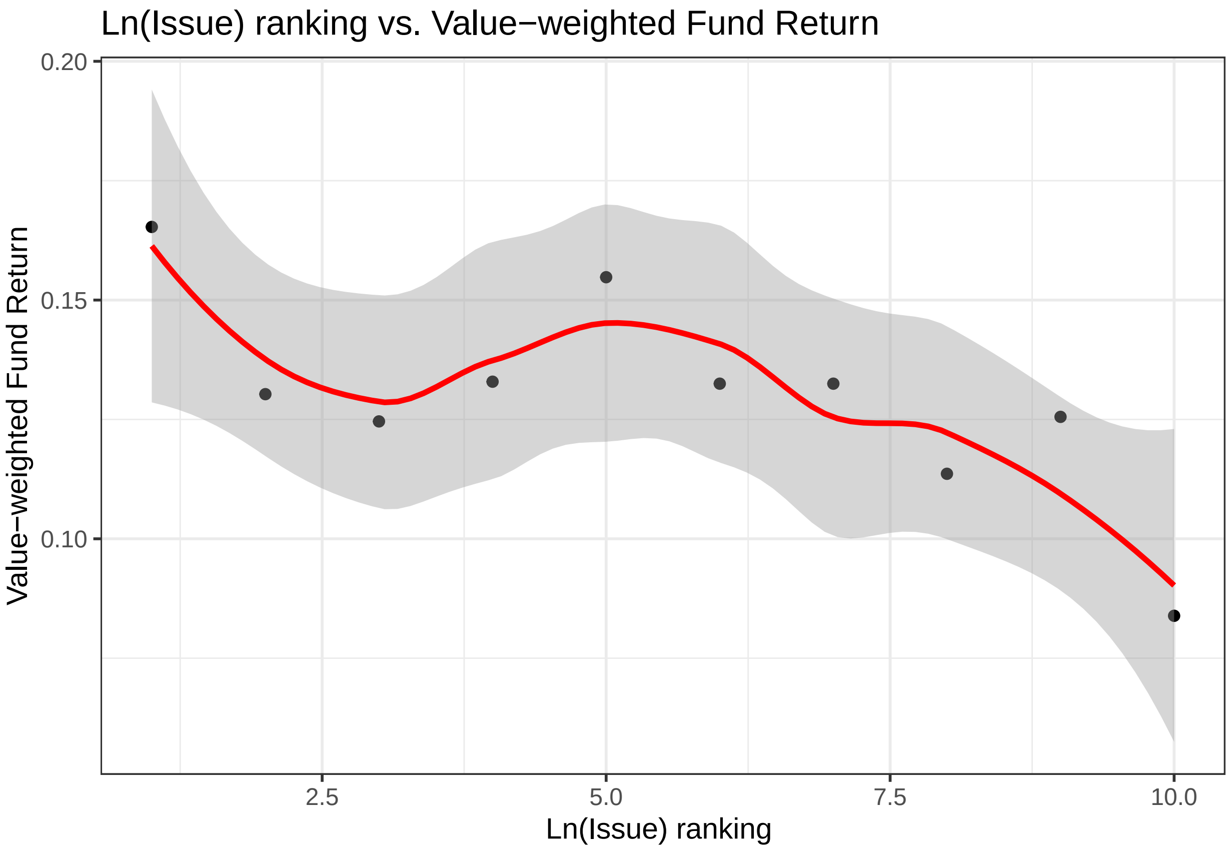
```r
{
 StockRetAcct_DT[year == i,Issue_vingtile_yr:=cut(StockRetAcct_DT[year == i,]$lnIssue,
breaks=quantile(StockRetAcct_DT[year == i,]$lnIssue,probs=c(0:10)/10,na.rm=TRUE), include.lowest=TRUE, l

# first, we need to value-weight stocks within each portfolio for each year
VW_lnIssue_Funds_yr = StockRetAcct_DT[,list(MeanAnnRet = weighted.mean(x = AnnRet,w = MEwt)), by = list

# then we average across years
VW_lnIssue_Funds_yr = VW_lnIssue_Funds_yr[,list(MeanAnnRet = mean(MeanAnnRet)), by = Issue_vingtile_yr]
#drop the nas
VW_lnIssue_Funds_yr =na.omit(VW_lnIssue_Funds_yr)
VW_lnIssue_Funds_yr[order(Issue_vingtile_yr)]
```

```
##     Issue_vingtile_yr MeanAnnRet
##  1:                 1 0.16530884
##  2:                 2 0.13029273
##  3:                 3 0.12458327
##  4:                 4 0.13290052
##  5:                 5 0.15478433
##  6:                 6 0.13248696
##  7:                 7 0.13248788
##  8:                 8 0.11361812
##  9:                 9 0.12553570
## 10:                10 0.08387553
```

b

Plot the relationship

```r
qplot(x =VW_lnIssue_Funds_yr$Issue_vingtile_yr, y = VW_lnIssue_Funds_yr$MeanAnnRet,na.rm = TRUE,
main = "Ln(Issue) ranking vs. Value-weighted Fund Return",
xlab = 'Ln(Issue) ranking ',
ylab = 'Value-weighted Fund Return'
) + geom_smooth(col=I("red")) + theme_bw()
```



From the graph, we can conclude that the relationship between Issue ranking and sorted Fund returns are not linear.

C

First,we need to construct the transformed Issue-related feature.

```
StockRetAcct_DT[Issue_vingtile_yr>0,Issue_trans:=0]
StockRetAcct_DT[Issue_vingtile_yr==1,Issue_trans:=-1]
StockRetAcct_DT[Issue_vingtile_yr==10,Issue_trans:=1]
```

Then, we can perform Fama-MacBeth Regression.

Recall the weight-matrix at each timestep t-1 to construct Fama-MacBeth factors are

$$W_t = (X_t^T X_t)^{-1} X_t^T$$

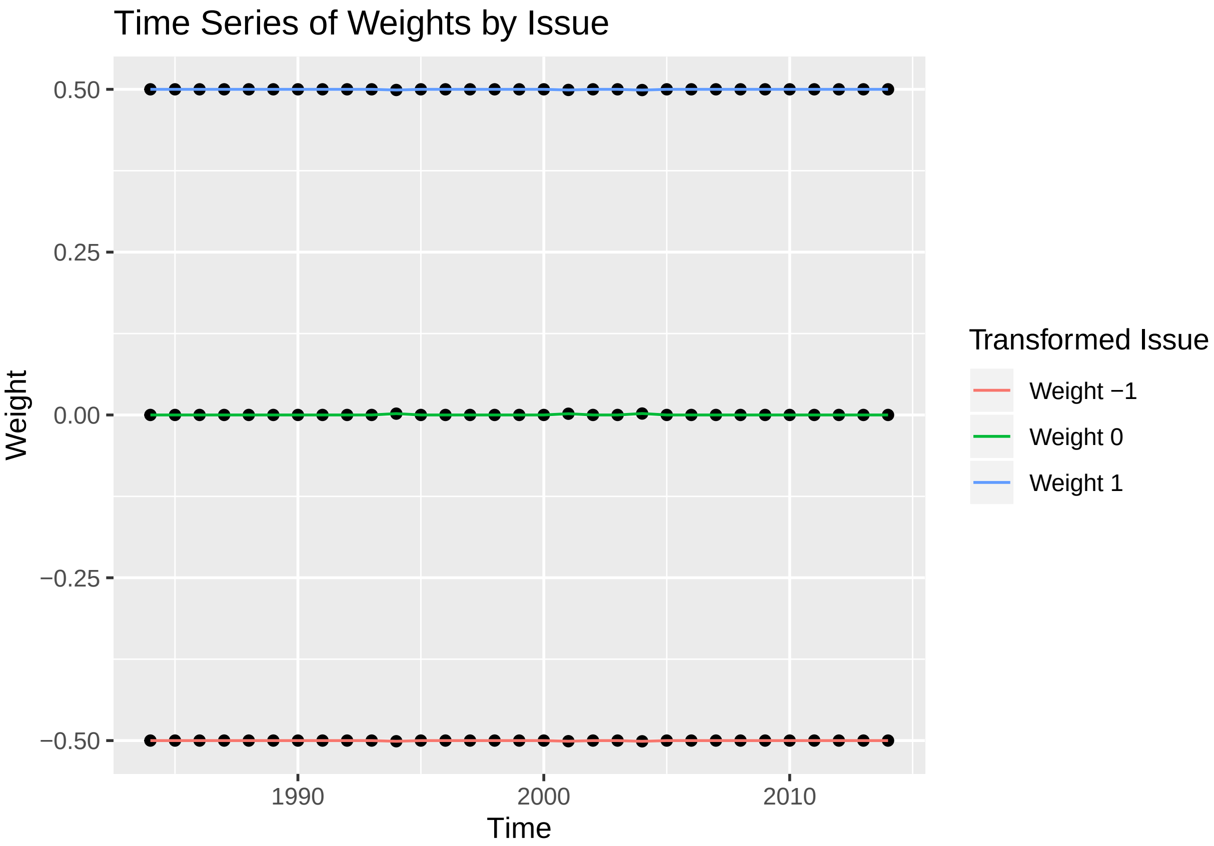
where X is the design matrix at time t.

The 2nd row of W would be the weights vector to construct the transformed Issue-related factor portfolio.

```
#initialize the weights matrix
Weight_mat=matrix(data = 0,nrow = length(1984:2014),ncol = 3)
#loop across time
for (t in 1984:2014){
  #build the desigh matrix
  Xt=cbind(1,StockRetAcct_DT[year==t,]$Issue_trans)
  #drop all the nas
  Xt=na.omit(Xt)
  #compute the Weight matrix
  Wt=solve(t(Xt)%*%Xt)%*%t(Xt)
  #get the weights for the issue-related portfolio
  weights_t=Wt[2,]
  #sum the weights by transformed Issue variable
  grouped_Weights_t=data.table(cbind(Xt[,2],weights_t))[,list(Weights = sum(weights_t)), by = V1]
  grouped_Weights_t=grouped_Weights_t[order(V1),]
  Weight_mat[(t-1983),]=t(grouped_Weights_t[,2])
}
```

Then we can plot the summed weights for stocks grouped by transfromed issue variable.

```
Weight_sr=data.table(cbind(1984:2014,Weight_mat))
colnames(Weight_sr)=c('time','Weight -1','Weight 0','Weight 1')
#melt the data
Weight_sr=melt(Weight_sr, id.vars <- 'time', variable.name='Transformed_issue')
Weiht_Plt=qplot(x = Weight_sr$time,y = Weight_sr$value,main = 'Time Series of Weights by Issue',xlab =
Weiht_Plt+labs(colour = "Transformed Issue")
```



From the graph above, we can conclude that, the implied positions of Fama-MacBeth regression is long the first 10 percentile portfolio and short the last 10 percentile portfolio with the same weights of 0.5.

Question 3

a

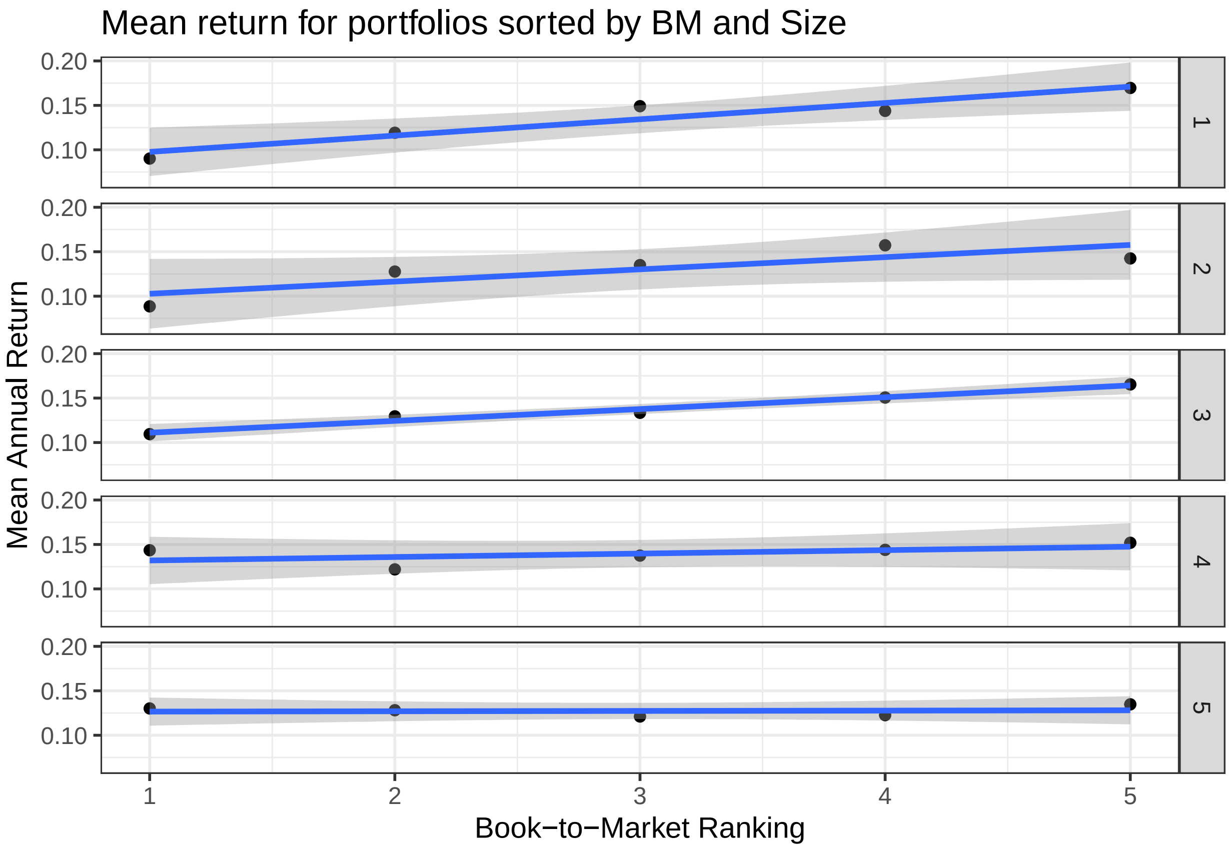
```
#define ranking variable based on lnBM and lnME
for (i in 1981:2014)
{
 StockRetAcct_DT[year == i,BM_vingtile_yr:=cut(StockRetAcct_DT[year == i,]$lnBM,
breaks=quantile(StockRetAcct_DT[year == i,]$lnBM,probs=c(0:5)/5,na.rm=TRUE), include.lowest=TRUE, label
  StockRetAcct_DT[year == i,Size_vingtile_yr:=cut(StockRetAcct_DT[year == i,]$lnME,
breaks=quantile(StockRetAcct_DT[year == i,]$lnME,probs=c(0:5)/5,na.rm=TRUE), include.lowest=TRUE, label
```

b

```
# first, we need to value-weight stocks within each portfolio for each year
VW_BM_ME_Funds_yr = StockRetAcct_DT[,list(MeanAnnRet = weighted.mean(x = AnnRet,w = MEwt)), by = list(B

# then we average across years
VW_BM_ME_Funds_yr = VW_BM_ME_Funds_yr[,list(MeanAnnRet = mean(MeanAnnRet)), by = list(BM_vingtile_yr, S
#drop nas
VW_BM_ME_Funds_yr=na.omit(VW_BM_ME_Funds_yr)
#sorting by ascending order
VW_BM_ME_Funds_yr=VW_BM_ME_Funds_yr[order(BM_vingtile_yr,Size_vingtile_yr)]
```

plot the relationship.

```
BM_Size_Ranking_plt_obj=qplot(data = VW_BM_ME_Funds_yr,x = BM_vingtile_yr,y = MeanAnnRet,facets = Size_
BM_Size_Ranking_plt_obj+geom_smooth(method = 'lm')+labs(facets='Size Ranking')+theme_bw()
```



The graph is consistent with the hypothesis that, holding size constant, there is a linear relationship between value-weighted mean return and the Value Ranking.

We can also observe that, the slope of the line, the value spread, is larger for small stock portfolios.