DataBased Statistical Decision Model_Final

Hyeonho Lee

2018년 9월 9일

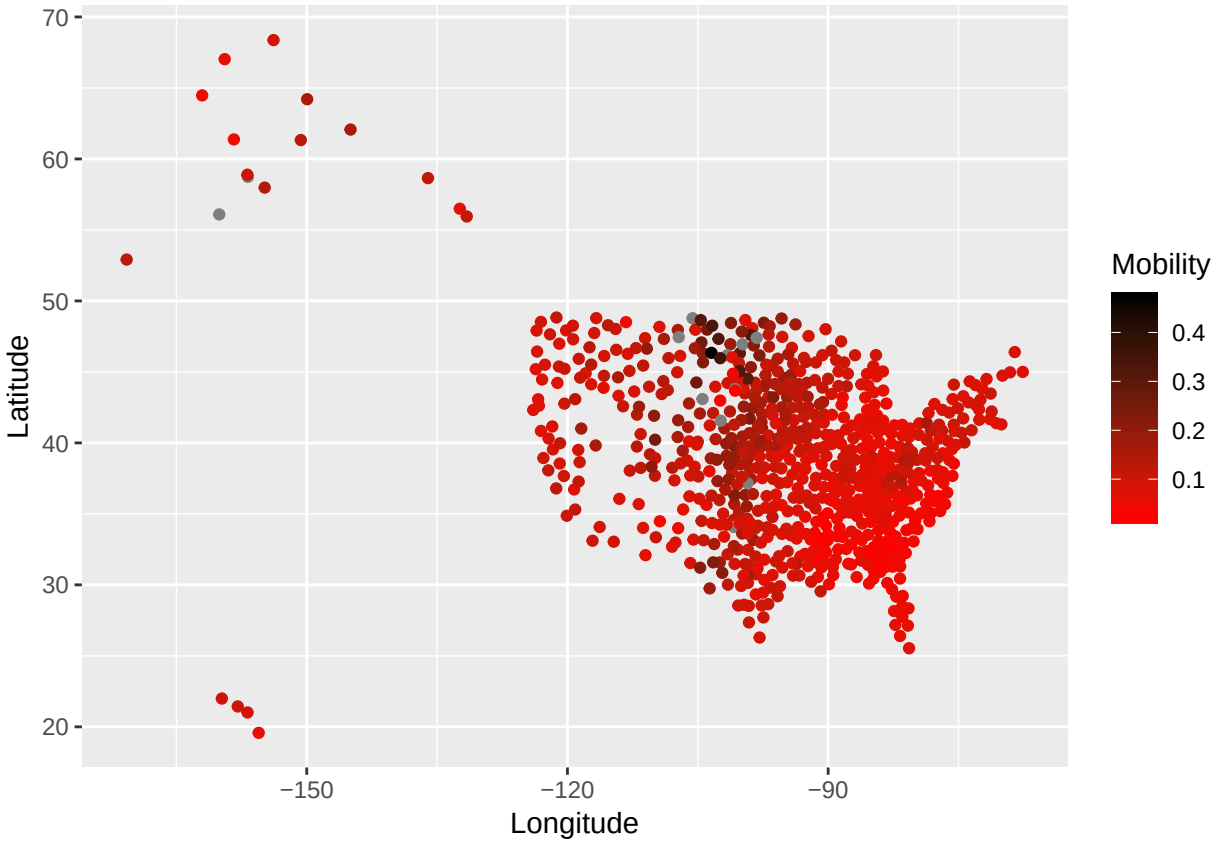
```
library(ggmap)
library(ggplot2)
library(gridExtra)
library(dplyr)
library(car)
```

Question1

1. A map of mobility

a. Make a plot where the x and y coordinates are longitude and latitude, and mobility is indicated by color (possibly grey scale), by a third coordinate, or some other suitable device. Make sure your map is legible. Describe the geographic pattern in words.

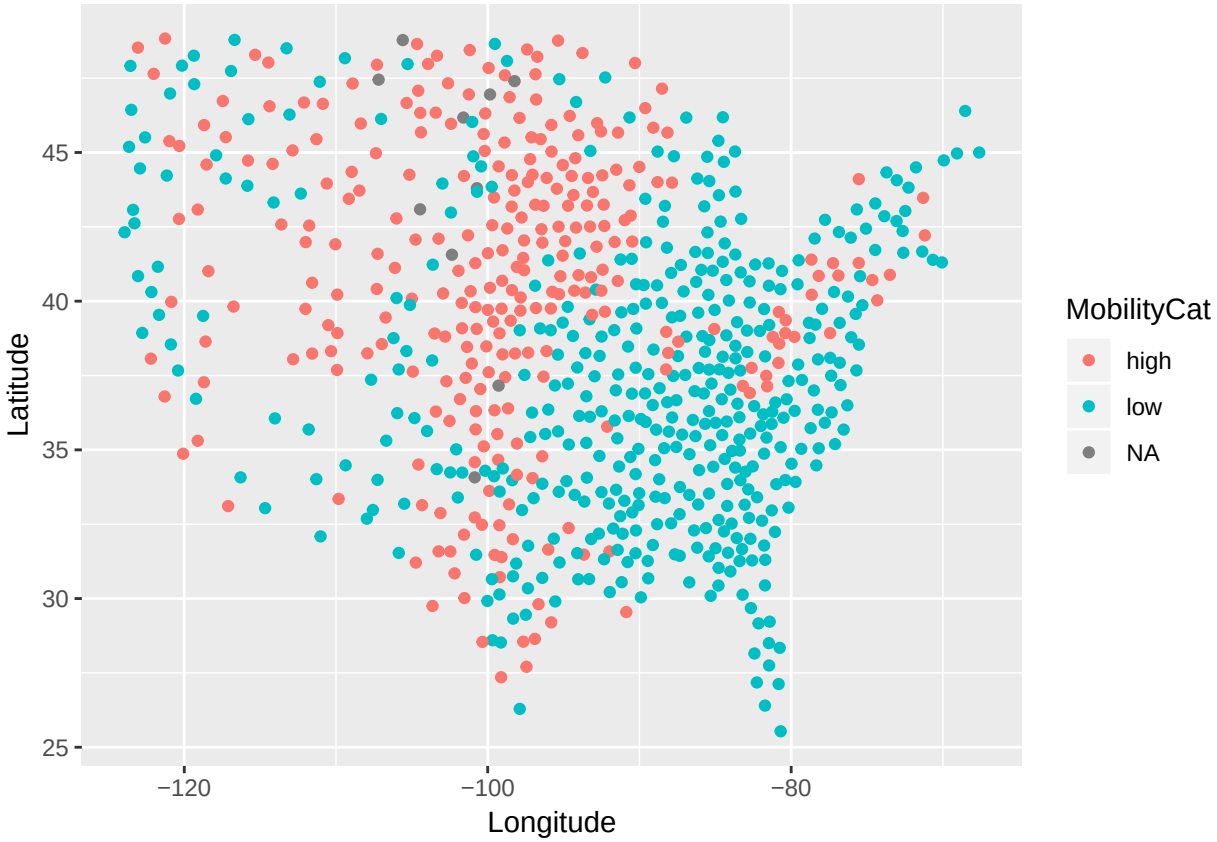
```
ggplot(dat, aes(x = Longitude, y = Latitude)) + geom_point(aes(color = Mobility)) +
  scale_color_gradient(low = "red", high = "black")
```



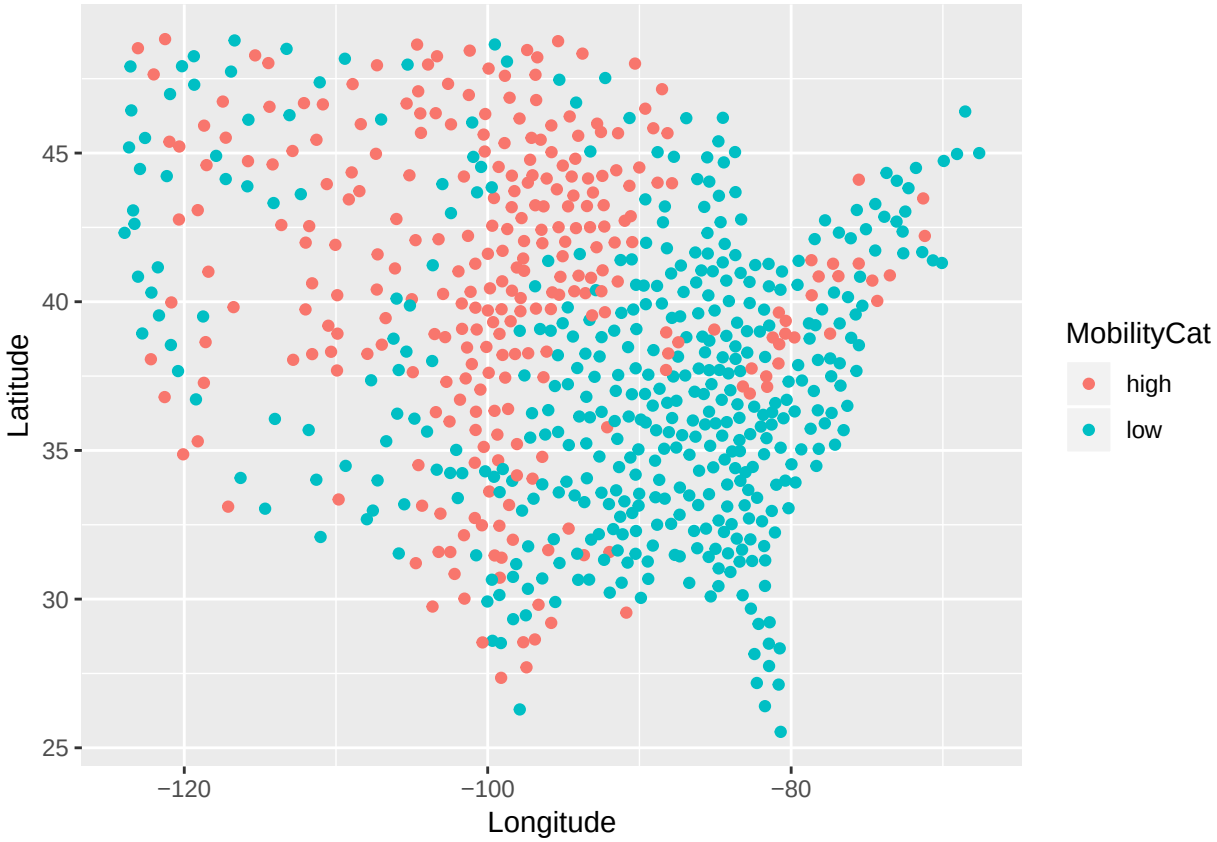
미국 본토의 중북부, 중부, 중남부 지방에서 출생한 아이가 상위 1분위에 속할 확률이 가장 높음이 보인다.

b. Discretizing the Mobility values may enhance visualizing. Create a new variable, called MobilityCat with values high if Mobility > 0.1, and low otherwise. Make a plot where the x and y coordinates are longitude and latitude, and the categorized mobility (i.e. MobilityCat) is indicated by color. This time, filter your observations so that only the continental part of USA is visible (that is, remove data corresponding to Alaska and Hawaii). Has the geographic pattern become clearer?

```
dat$MobilityCat = dat$Mobility
dat$MobilityCat[dat$MobilityCat>0.1] = 'high'
dat$MobilityCat[dat$MobilityCat<=0.1] = 'low'


# ggmap(get_map(location=c('United States') ,zoom = 4, maptype = 'roadmap',
#                     color = 'bw')) +
#   geom_point(data = dat, aes(x = Longitude, y = Latitude, color = MobilityCat)) +
#   xlab('Longitude') + ylab('Latitude')
#
# ggmap(get_map(location=c('United States') ,zoom = 4, maptype = 'roadmap',
#                     color = 'bw')) +
#   geom_point(data = dat %>% filter(!is.na(MobilityCat)),
#              aes(x = Longitude, y = Latitude, color = MobilityCat)) +
#   xlab('Longitude') + ylab('Latitude')

ggplot(data = dat %>% filter(State != 'AK' & State != 'HI'),
       aes(x = Longitude, y = Latitude, color = MobilityCat)) +
  xlab('Longitude') + ylab('Latitude') + geom_point()
```



```
ggplot(data = dat %>% filter(State != 'AK' & State != 'HI' & !is.na(MobilityCat)),
           aes(x = Longitude, y = Latitude, color = MobilityCat)) +
  xlab('Longitude') + ylab('Latitude') + geom_point()
```



위에서 본것과 살짝 다르게 보이는데, 더 정확하게 볼 수 있다. 예를들어 중부, 서부지역이 0.1이상임을 확실하게 볼 수 있고, 동부지방의 경우 실리콘밸리 지역외에는 0.1이하임을 볼 수 있다.

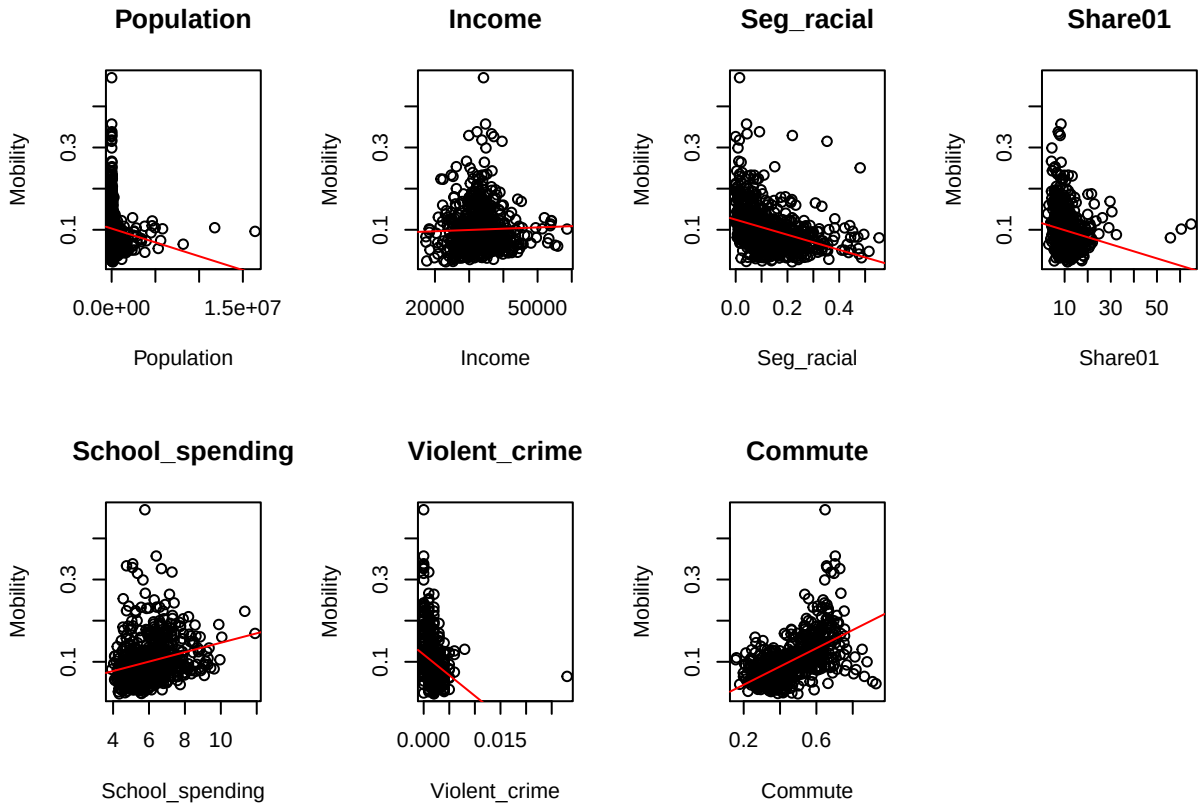
ggmap의 경우 구글서버가 불안정해서 knit가 안됐습니다.

2. A bunch of simple regression models

Make scatter plots of mobility against each of the following variables. Include on each plot a line for the simple or univariate regression, and give a table of the regression coefficients. Carefully explain the interpretation of each coefficient. Do any of the results seem odd?

a. Population
b. Mean household income per capita
c. Racial segregation
d. Income share of the top 1%
e. Mean school expenditures per pupil
f. Violent crime rate
g. Fraction of workers with short commutes.

```r
par(mfrow = c(2, 4))
model1 = lm(Mobility~Population, data = dat)
plot(y = dat$Mobility, x = dat$Population, main = 'Population',
     xlab = 'Population', ylab = 'Mobility')
abline(model1, col = 'red')

model2 = lm(Mobility~Income, data = dat)
plot(y = dat$Mobility, x = dat$Income, main = 'Income',
     xlab = 'Income', ylab = 'Mobility')
abline(model2, col = 'red')

model3 = lm(Mobility~Seg_racial, data = dat)
plot(y = dat$Mobility, x = dat$Seg_racial, main = 'Seg_racial',
     xlab = 'Seg_racial', ylab = 'Mobility')
abline(model3, col = 'red')

model4 = lm(Mobility~Share01, data = dat)
plot(y = dat$Mobility, x = dat$Share01, main = 'Share01',
     xlab = 'Share01', ylab = 'Mobility')
abline(model4, col = 'red')

model5 = lm(Mobility~School_spending, data = dat)
plot(y = dat$Mobility, x = dat$School_spending, main = 'School_spending',
     xlab = 'School_spending', ylab = 'Mobility')
abline(model5, col = 'red')

model6 = lm(Mobility~Violent_crime, data = dat)
plot(y = dat$Mobility, x = dat$Violent_crime, main = 'Violent_crime',
     xlab = 'Violent_crime', ylab = 'Mobility')
abline(model6, col = 'red')

model7 = lm(Mobility~Commute, data = dat)
plot(y = dat$Mobility, x = dat$Commute, main = 'Commute',
     xlab = 'Commute', ylab = 'Mobility')
abline(model7, col = 'red')
```



그래프를 보았을 때, 회귀식이 대부분 잘 적합되어 보이진 않는다. School_spening, Commute, Seg_racial이 그나마 잘 적합된 것 같지만 이도 많이 부족해 보인다. 래버리지도 많이 보이고, 설명력이 많이 떨어져 보인다. 그러나 알 수 있는 부분은 회귀식의 추세를 봄으로써 각각의 변수가 Mobility에 어떤 영향을 미치는 지 알 수 있다. 회귀선이 아래로 내려가는 추세라면, 반비례관를 추정할 있고, 회귀선이 위로 올라가는 추세라면 비례관계를 추정할 수 있다.

```
rbind(summary(model1)$coef, summary(model2)$coef, summary(model3)$coef,
      summary(model4)$coef, summary(model5)$coef, summary(model6)$coef,
      summary(model7)$coef)
```

Table 1: coefficient table

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1030221	0.0020593	50.0282608	0.0000000
Population	0.0000000	0.0000000	-3.6694570	0.0002608
(Intercept)	0.0902463	0.0113347	7.9619628	0.0000000
Income	0.0000003	0.0000003	0.9114705	0.3623496
(Intercept)	0.1246090	0.0030375	41.0230432	0.0000000
Seg_racial	-0.1835019	0.0183933	-9.9765780	0.0000000

수치적으로 coefficient를 보면 Population, Seg_racial, Share01, Violent_crime는 한 단위 증가할 때, Mobility를 감소시키고, Income, School_spending, Commute는 한 단위 증가할 때, Mobility를 증가시키는 점을 알 수 있다. 즉 Mobility를 증가시키기 위해선 Income, School_spending, Commute가 높아야 됨을 알 수 있다.

3. All things considered

Run a linear regression of mobility against all appropriate covariates.

a. Report all regression coefficients and their standard errors; you may use either a table or a figure as you prefer.

```
model = lm(Mobility~., data = dat[,-c(1:2)])
coef_1 = summary(model)$coef
```

Table 2: coef_1 table(9개만)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0743886	0.0896154	0.8300876	0.4070769
StateAR	-0.0080121	0.0091016	-0.8802934	0.3793277
Religious	0.0071309	0.0139700	0.5104437	0.6100748
Violent_crime	-0.2788176	1.6643201	-0.1675264	0.8670563
Single_mothers	-0.2675164	0.0896329	-2.9845793	0.0030472
Divorced	-0.2100496	0.1676063	-1.2532322	0.2109900
Married	-0.0340868	0.0646619	-0.5271543	0.5984333
Longitude	-0.0009199	0.0006184	-1.4875531	0.1378038
Latitude	0.0002442	0.0010311	0.2368251	0.8129364

Id, Name 변수는 Mobility와 인과관계가 전혀 없으므로 제거한 후 모든 변수를 사용하여 모델을 적합하였다. 어떤 변수가 Mobility를 감소시키고, 증가시키는지 확인 할 수 있다.

b. Explain why the ID variable must be excluded.

Mobility와의 인과관계가 전혀 없기 때문이다. Id나 Name에 따라서 Mobility가 변화한다는 것은 전혀 관계가 없다.

c. Explain which other variables, if any, you excluded from the regression, and why. (If you think they can all be used, explain why.) [For this question, do not use any automated variable selection, and try to keep as many variables as possible.]

```
names(summary(model)$coef[,4][summary(model)$coef[,4] < 0.05])
```

Table 3: p value가 0.05이하인 값들의 변수명(6개만)

x
StateNC
Manufacturing
Migration_out
Foreign_born
Social_capital
Single_mothers

```
model = lm(Mobility~., data = dat %>% select(State, Black, Seg_racial,
                                              Commute, Mobility, Middle_class,
                                              Manufacturing, Migration_out, Foreign_born,
                                              Social_capital, Single_mothers))
coef_2 = summary(model)$coef
```

모델링 과정과 p value로 변수 탈락을 계속 실시한 결과, 최종적으로 선택된 변수는 위의 11개이다.

Table 4: coef_2 table(5개만)

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0890209	0.0193683	4.596229	0.0000052
StateCA	0.0256309	0.0102093	2.510552	0.0123024
Migration_out	-1.1212002	0.1600811	-7.003951	0.0000000
Foreign_born	0.0982504	0.0282827	3.473865	0.0005480
Social_capital	-0.0068279	0.0015190	-4.495010	0.0000083
Single_mothers	-0.4681448	0.0412987	-11.335582	0.0000000

p value가 0.05이상인 경우 Mobility에 대한 계수의 추정이 유의미하지 않다고 판단하였기 때문에 제거하였다.

d. Compare the coefficients you found in problem 2 to the coefficients for the same variables in this regression. Are they significantly different? Have any changed sign?

```
coef1_Black = coef_1[,1, drop = F][rownames(coef_1)=='Black',]
coef2_Black = coef_2[,1, drop = F][rownames(coef_2)=='Black',]

coef1_Seg_racial = coef_1[,1, drop = F][rownames(coef_1)=='Seg_racial',]
coef2_Seg_racial = coef_2[,1, drop = F][rownames(coef_2)=='Seg_racial',]

coef1_Commute = coef_1[,1, drop = F][rownames(coef_1)=='Commute',]
coef2_Commute = coef_2[,1, drop = F][rownames(coef_2)=='Commute',]

coef1_Middle_class = coef_1[,1, drop = F][rownames(coef_1)=='Middle_class',]
coef2_Middle_class = coef_2[,1, drop = F][rownames(coef_2)=='Middle_class',]

coef1_Single_mothers = coef_1[,1, drop = F][rownames(coef_1)=='Single_mothers',]
coef2_Single_mothers = coef_2[,1, drop = F][rownames(coef_2)=='Single_mothers',]
```

Table 5: coefficient of Black

	Black
coef1_Black	0.0774584
coef2_Black	0.1193369

Table 6: coefficient of Seg_racial

	Seg_racial
coef1_Seg_racial	-0.0635631
coef2_Seg_racial	-0.0530622

Table 7: coefficient of Commute

	Commute
coef1_Commute	0.0591301
coef2_Commute	0.0555756

Table 8: coefficient of Middle_class

	Middle_class
coef1_Middle_class	0.1746245
coef2_Middle_class	0.1713100

Table 9: coefficient of Single_mothers

	Single_mothers
coef1_Single_mothers	-0.2675164
coef2_Single_mothers	-0.4681448

5개만 비교해 보았는데 계수의 값이 바뀌었음을 볼 수 있다. 회귀분석에 사용된 변수들이 바뀌면서, 공분산이 사라지거나 변경되므로 변수의 계수에 영향을 미친다.

e. Take a look at the variation inflation factor for each variable. Report those variables with VIF greater than 10. Do you suspect a (nearly) multicollinearity? If so, give a reason for, and suggest a way to avoid it.

```
model = lm(Mobility~., data = dat %>% select(State, Black, Seg_racial,
                                              Commute, Mobility,Middle_class, Manufacturing,
                                              Migration_out, Foreign_born, Social_capital,
                                              Single_mothers))
vif_model = vif(model)
```

Table 10: vif1

	GVIF	Df	GVIF^(1/(2*Df))
State	174.985813	48	1.055273
Black	6.355854	1	2.521082
Seg_racial	1.758338	1	1.326023
Commute	3.593537	1	1.895663
Middle_class	6.426227	1	2.535000
Manufacturing	2.578988	1	1.605923
Migration_out	1.945796	1	1.394918
Foreign_born	2.733428	1	1.653308
Social_capital	4.765430	1	2.182986
Single_mothers	6.305231	1	2.511022

State가 매우 높은 vif를 보이고 있다. 통상적으로 10이상일 때, 다중공선성이 존재한다고 하는데 State가 너무 높으므로 다중공선성이 의심이 됩니다. multicollinearity를 피하기 위해서 변수를 제거하는 방법을 택합니다. 또다른 방법은 주성분 분석 등을 사용하는 방법이 있지만, 여기선 변수제거를 선택했습니다.

```
model = lm(Mobility~., data = dat %>% select(Black, Seg_racial, Commute,
                                              Mobility, Middle_class, Manufacturing,
                                              Migration_out, Foreign_born, Social_capital,
                                              Single_mothers))
vif_model = vif(model)
```

Table 11: vif2

	x
Black	3.090674
Seg_racial	1.369772
Commute	2.132088
Middle_class	4.337990
Manufacturing	1.485505
Migration_out	1.388117
Foreign_born	1.682466
Social_capital	2.378217
Single_mothers	4.170952

State를 제거한 후의 회귀분석은 vif가 사라진것을 확인 할 수 있습니다.(VIF>10이 다중공선성에 대한 의심 조건)

4. Please in my front yard

a. Inspect the missingness pattern in variables Colleges, Tuition and Graduation. [Note: NA is a missing value.] How many observations have no measurements for these variables?

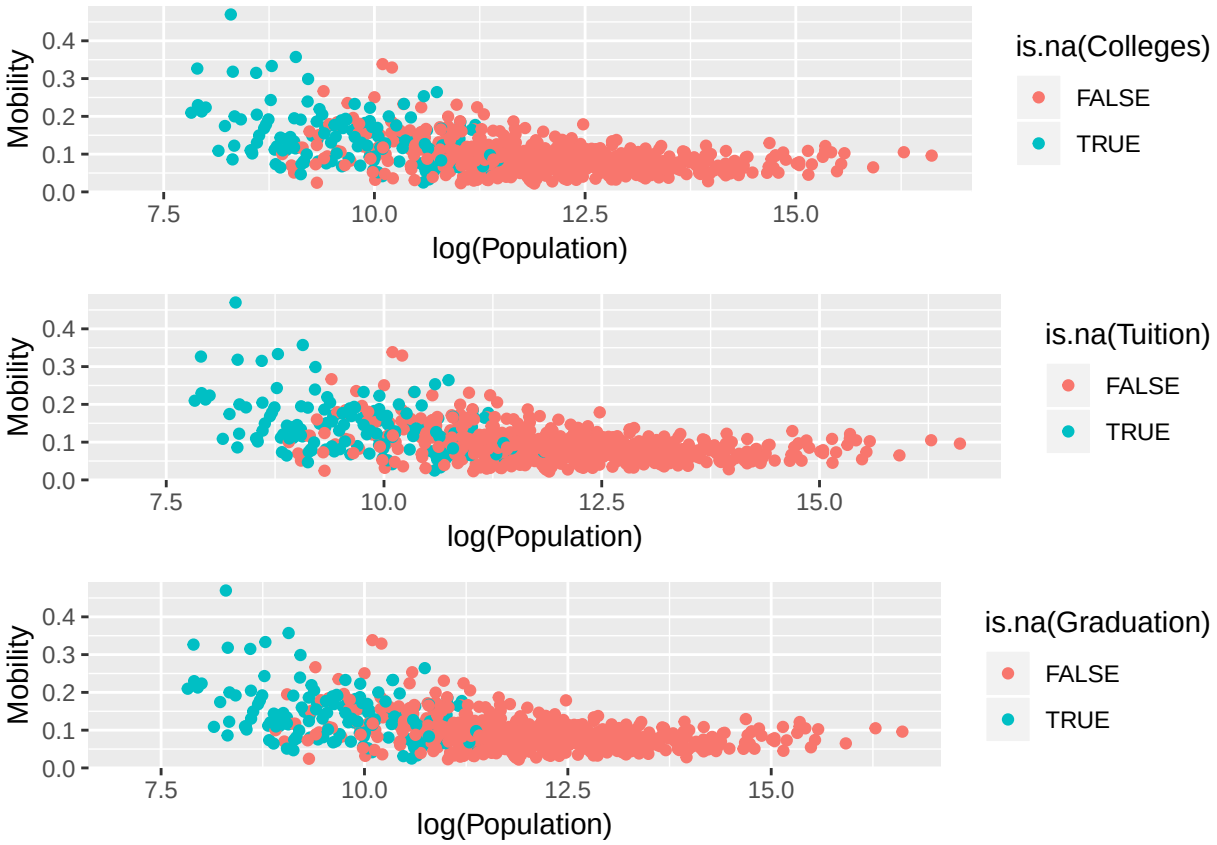
```
Colleges = sum(is.na(dat[,'Colleges']))
Tuition = sum(is.na(dat[,'Tuition']))
Graduation = sum(is.na(dat[,'Graduation']))
```

Table 12: Number of NA

Colleges	Tuition	Graduation
157	161	160

b. Did the missing values happen at random? To answer this, plot a scatter of Mobility and Population (choose a suitable scale for Population), and inspect which data points have missing values in (all of, or some of) variables Colleges, Tuition and Graduation.

```
g1 = ggplot(dat, aes(x = log(Population), y = Mobility)) + geom_point(aes(
  color = is.na(Colleges)))
g2 = ggplot(dat, aes(x = log(Population), y = Mobility)) + geom_point(aes(
  color = is.na(Tuition)))
g3 = ggplot(dat, aes(x = log(Population), y = Mobility)) + geom_point(aes(
  color = is.na(Graduation)))

grid.arrange(g1, g2, g3, ncol=1)
```



누락된 부분이 비슷함을 알 수 있다. 한 변수가 누락됐을 경우 다른 두 변수가 누락될 확률이 높음을 의미한다.

c. Create a new variable, called HE, whose value is TRUE if there is a higher education institution in the community, is FALSE if not. Replace all NA values in variables Colleges, Tuition and Graduation with 0.

```
dat['Colleges'][is.na(dat['Colleges'])] = 0
dat['Tuition'][is.na(dat['Tuition'])] = 0
dat['Graduation'][is.na(dat['Graduation'])] = 0

dat$HE = NA
dat$HE[dat$Colleges == 0 & dat$Tuition == 0 & dat$Graduation == 0] = F
dat$HE[is.na(dat$HE)] = T
```

Table 13: Table of HE

Var1	Freq
FALSE	151
TRUE	590

5. All things considered, again.

Fit a linear regression model, incorporating your findings in problems 3 and 4. If you have removed, created, or modified variables, explain. Report all regression coefficients and their standard errors. Use this model for all problems below.

```
model = lm(Mobility ~ Black + Seg_racial + Seg_poverty + Commute + Middle_class + Manufacturing +
             Migration_out + Foreign_born + Social_capital + Single_mothers + HE +
             Colleges + Tuition + Graduation, data = dat)
coef = summary(model)$coef
```

3번에서 VIF가 높았던 State를 제거하고, 4번에서 언급한 고등교육현황과 새로 만든 컬럼인 HE를 사용한 회귀분석을 실시하였다.

Table 14: coefficient of model

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1121455	0.0205391	5.4600882	0.0000001
Black	0.0991310	0.0153759	6.4471806	0.0000000
Seg_racial	-0.0189824	0.0139081	-1.3648400	0.1727574
Seg_poverty	-0.0327832	0.0635116	-0.5161763	0.6059001
Commute	0.0938872	0.0150680	6.2309086	0.0000000
Middle_class	0.1334213	0.0283244	4.7104723	0.0000030
Manufacturing	-0.1531639	0.0170082	-9.0053056	0.0000000
Migration_out	-0.5865270	0.1792084	-3.2728758	0.0011188
Foreign_born	0.1159446	0.0283644	4.0876842	0.0000488
Social_capital	0.0007741	0.0014151	0.5470328	0.5845367
Single_mothers	-0.5126024	0.0436207	-11.7513693	0.0000000
HETRUE	-0.0046801	0.0041225	-1.1352657	0.2566664
Colleges	0.0555591	0.0698965	0.7948762	0.4269645
Tuition	0.0000003	0.0000004	0.8350311	0.4039952
Graduation	-0.0210348	0.0094977	-2.2147278	0.0271115

위에서 다중공선성으로 인해 state를 제거하였더니 p value가 낮은 변수들이 다시 나타났다. 이번에도 p value가 0.05가 되지 않으면 제거한다.

```
model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing +
             Migration_out + Foreign_born + Single_mothers + Graduation, data = dat)
coef = summary(model)$coef
```

Table 15: coefficient of model

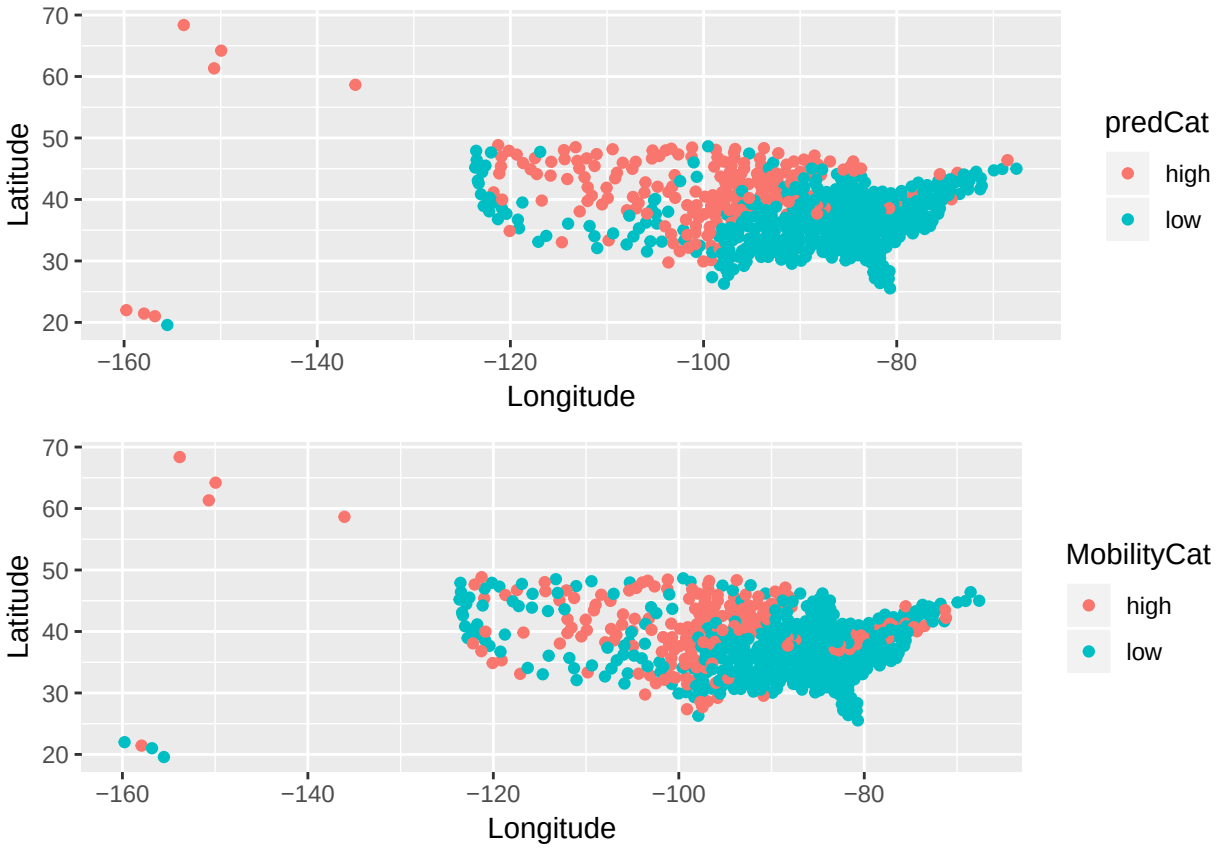
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.1164823	0.0193941	6.006061	0.0000000
Black	0.1037717	0.0151926	6.830412	0.0000000
Commute	0.0913177	0.0103715	8.804706	0.0000000
Middle_class	0.1362925	0.0249950	5.452789	0.0000001
Manufacturing	-0.1506582	0.0159435	-9.449475	0.0000000
Migration_out	-0.6903865	0.1593224	-4.333268	0.0000168
Foreign_born	0.0978698	0.0269655	3.629449	0.0003048
Single_mothers	-0.5567208	0.0402058	-13.846787	0.0000000
Graduation	-0.0189382	0.0089478	-2.116517	0.0346542

현 보고서에서 변수선택의 기준이 되는 p value가 모두 0.05이므로 더이상 제거하지 않는다.

6. Make a map of predicted mobility.

Make a map of the model's predicted mobility. How does it compare, qualitatively, to the map of actual mobility?

```
dat = dat %>% select(Black, Commute, Middle_class, Manufacturing, Migration_out,
                                  Foreign_born, Single_mothers, Graduation, Mobility,
                      Longitude, Latitude)
dat = na.omit(dat)

model = lm(Mobility ~ ., data = dat)

pred = predict(model, newdata = dat)
pred = cbind(pred, dat)

pred$predCat = pred$pred
pred$predCat[pred$predCat>0.1] = 'high'
pred$predCat[pred$predCat!='high'] = 'low'
g1 = pred %>%
  ggplot(aes(x = Longitude, y = Latitude)) + geom_point(aes(color = predCat))

dat$MobilityCat = dat$Mobility
dat$MobilityCat[dat$MobilityCat>0.1] = 'high'
dat$MobilityCat[dat$MobilityCat!='high'] = 'low'

g2 = dat %>%
  ggplot(aes(x = Longitude, y = Latitude)) + geom_point(aes(color = MobilityCat))

grid.arrange(g1, g2, ncol=1)
```

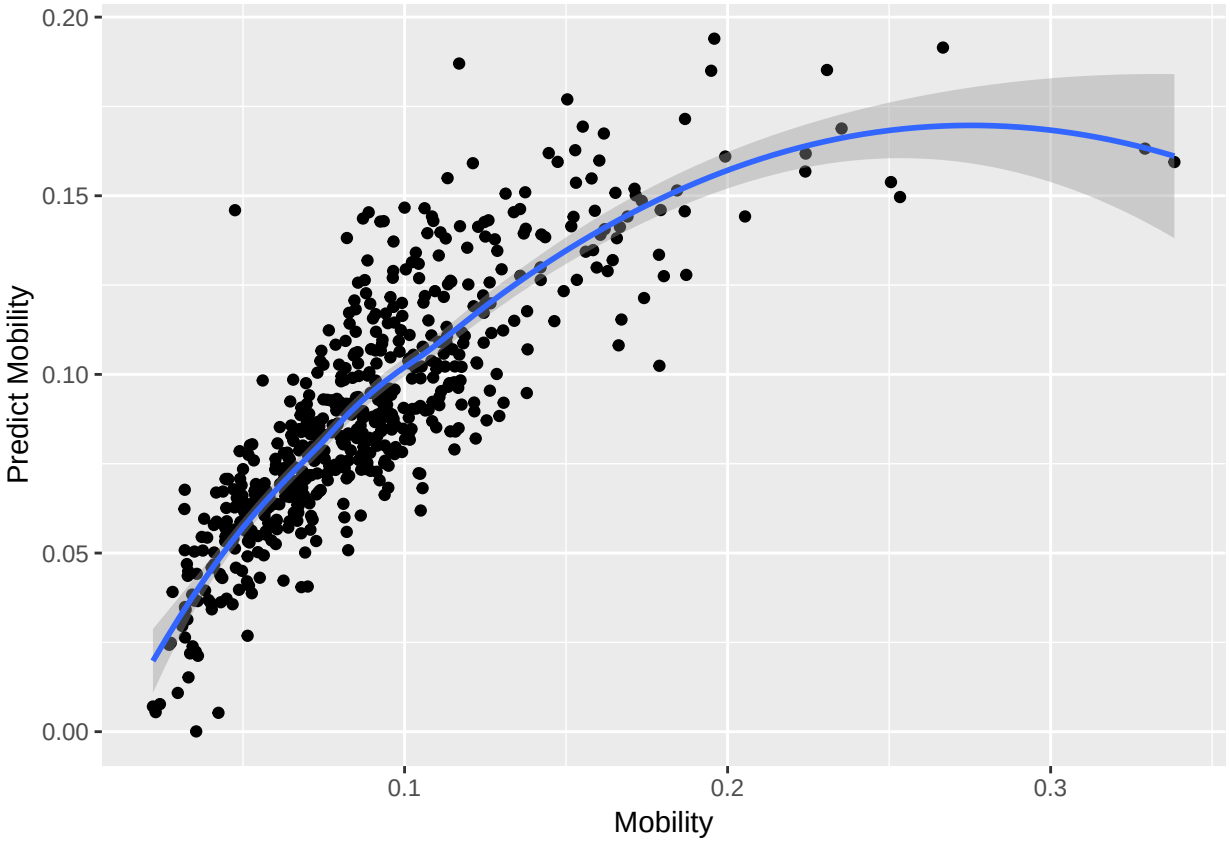


NA값을 제거하고 본 결과이다. 미국본토의 서부지역을 제외한 나머지 지역은 굉장히 비슷함을 볼 수 있고, 예측을 잘 했다고 생각된다.

7. Just because I was there

Find Pittsburgh in the data set. For this question, assume that the model (you have fitted just before) is well-fitted. a. What its actual mobility? What is its predicted mobility, according to the model?

```
ggplot(dat, aes(pred$Mobility, pred$pred, xlab = 'predict', ylab = 'Mobility')) +
  geom_point() + geom_smooth() + xlab('Mobility') + ylab('Predict Mobility')
```

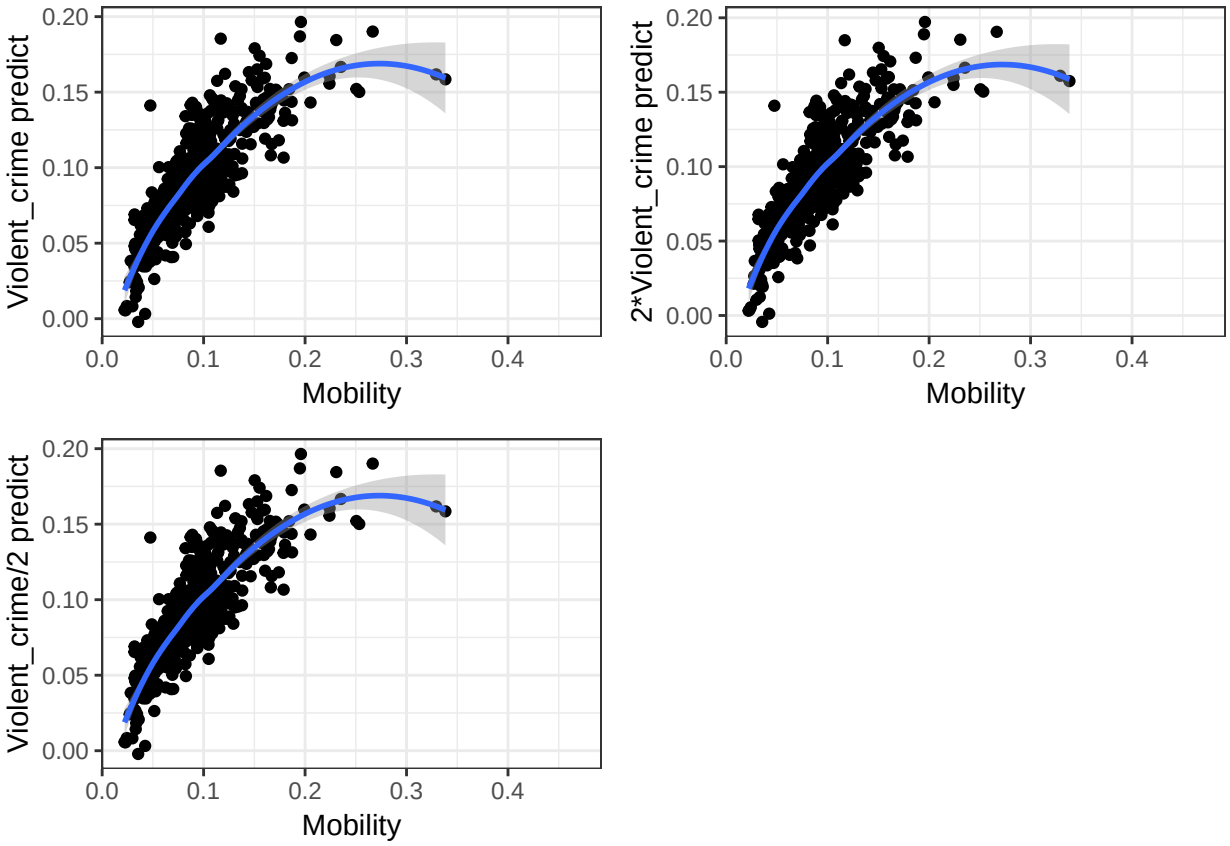


실제 Mobility가 증가함에 따라서 예측된 Mobility도 증가함을 볼 수 있다. 모델의 경우 동일한 증가치를 가지진 않지만 추세를 잘 따라가고 있는것으로 보인다.

b. Holding all else fixed, what is the predicted mobility if the violent crime rate is doubled? If it is halved?

```
model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing + Migration_out +
                                Foreign_born + Single_mothers + Graduation + Violent_crime,
           data = dat)
pred1 = predict(model, newdata = dat)

model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing + Migration_out +
                                Foreign_born + Single_mothers + Graduation + I(Violent_crime^2),
           data = dat)
pred2 = predict(model, newdata = dat)

model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing + Migration_out +
                                Foreign_born + Single_mothers + Graduation + I(Violent_crime/2),
           data = dat)
pred3 = predict(model, newdata = dat)

pred = data.frame(cbind(dat$Mobility, pred1, pred2, pred3))
colnames(pred) = c('a', 'b', 'c', 'd')

require('gridExtra')

theme_set(theme_bw())
p1 = pred %>% ggplot(aes(pred$a, pred$b)) + geom_point() + geom_smooth() +
  xlab('Mobility') + ylab('Violent_crime predict')
```

```
p2 = pred %>% ggplot(aes(pred$a, pred$c)) + geom_point() + geom_smooth() +
  xlab('Mobility') + ylab('2*Violent_crime predict')
p3 = pred %>% ggplot(aes(pred$a, pred$d)) + geom_point() + geom_smooth() +
  xlab('Mobility') + ylab('Violent_crime/2 predict')

grid.arrange(p1, p2, p3, ncol=2)
```



한 변수의 값을 2배 혹은 1/2하여도 회귀식의 영향을 주지 않는다. 분산에 따라 계수가 바뀌기 때문에 배수를 하면 영향을 주지 않는다. 그러나 제곱 혹은 로그 등 분산에 영향을 주는 변수변환을 한다면 계수가 달라져, 회귀식도 변환될 것이다.

c. Provide a 95% confidence interval for the expected mobility at Pittsburgh.

```
model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing + Migration_out +
                                Foreign_born + Single_mothers + Graduation, data = dat)

pred = predict(model, newdata = dat, interval='confidence')

pred = cbind(pred, dat)
Pittsburgh_conf = pred[pred$Name == 'Pittsburgh',][1:3]
```

Table 16: confidence of Pittsburgh

	fit	lwr	upr
229	0.0774405	0.0727929	0.0820881

d. Provide a 95% prediction interval for the expected mobility at Pittsburgh. Explain the difference.

```
model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing + Migration_out +
                                Foreign_born + Single_mothers + Graduation, data = dat)

pred = predict(model, newdata = dat, interval='prediction')

pred = cbind(pred, dat)
Pittsburgh_pred = pred[pred$Name == 'Pittsburgh',][1:3]
```

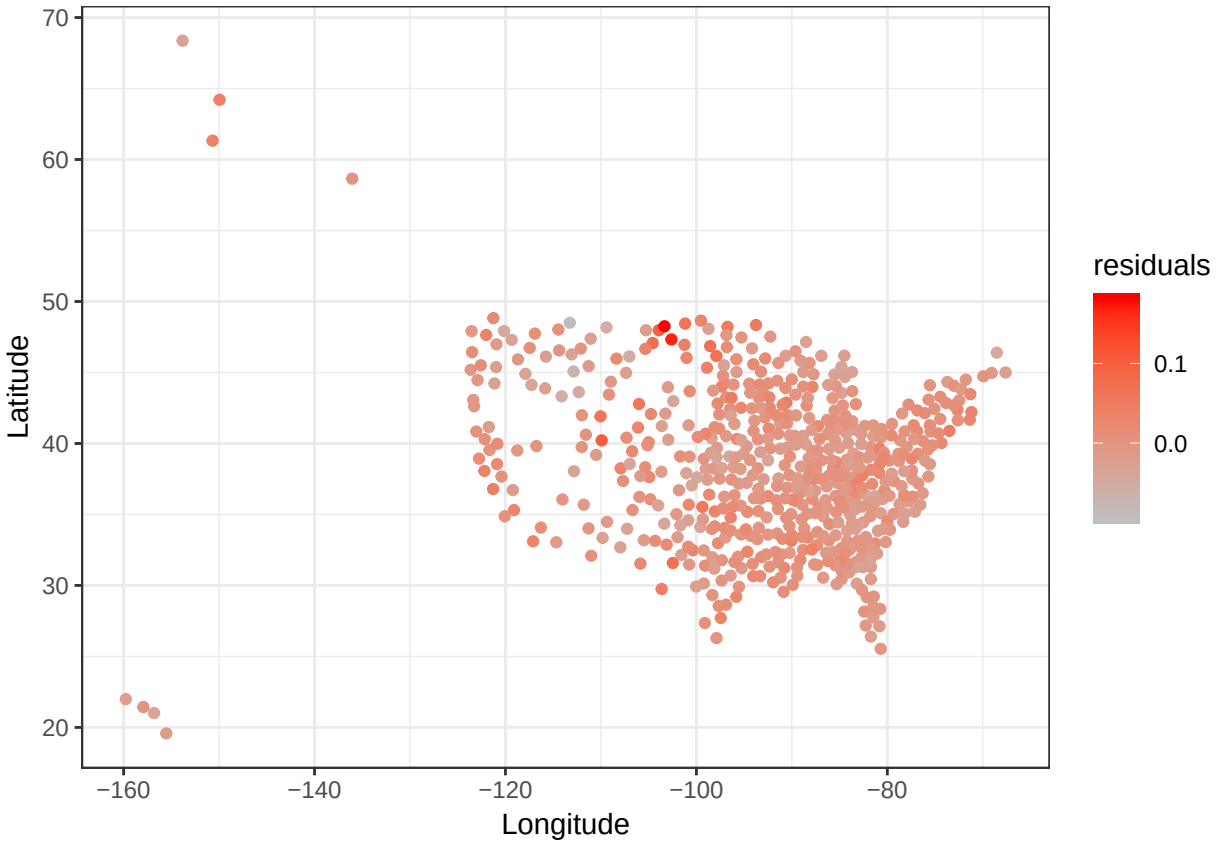
Table 17: prediction of Pittsburgh

	fit	lwr	upr
229	0.0774405	0.0302827	0.1245983

8. After making proper allowances

a. Make a map of the model's residuals.

```
model = lm(Mobility ~ Black + Commute + Middle_class + Manufacturing +
             Migration_out + Foreign_born + Single_mothers + Graduation, data = dat)

pred = predict(model, newdata = dat)

dat_resi = mutate(dat, dat$Mobility - pred)

colnames(dat_resi)[14] = 'residuals'

dat_resi %>%
  ggplot(aes(x = Longitude, y = Latitude)) + geom_point(aes(color = residuals)) +
  scale_color_gradient(low = "gray", high = "red")
```



회귀식에 사용하는 변수만 선택한 후 NA값은 모두 제거 하였다.

b. What are the five communities with the largest positive residuals? The five with the most negative residuals? Provide the names of the communies. (Can you mark these on the map?)

```
dat_resi_head = dat_resi %>% arrange(desc(residuals)) %>% head(5)
dat_resi_tail = dat_resi %>% arrange(desc(residuals)) %>% tail(5)

dat_resi$head = NA
dat_resi$head[dat_resi$ID %in% dat_resi_head$ID] = 'positive top5'
dat_resi$head[dat_resi$ID %in% dat_resi_tail$ID] = 'negative top5'
dat_resi$head[is.na(dat_resi$head) == T] = 'not top5'

dat_resi %>% ggplot(aes(x = Longitude, y = Latitude)) + geom_point(aes(col = head))
```

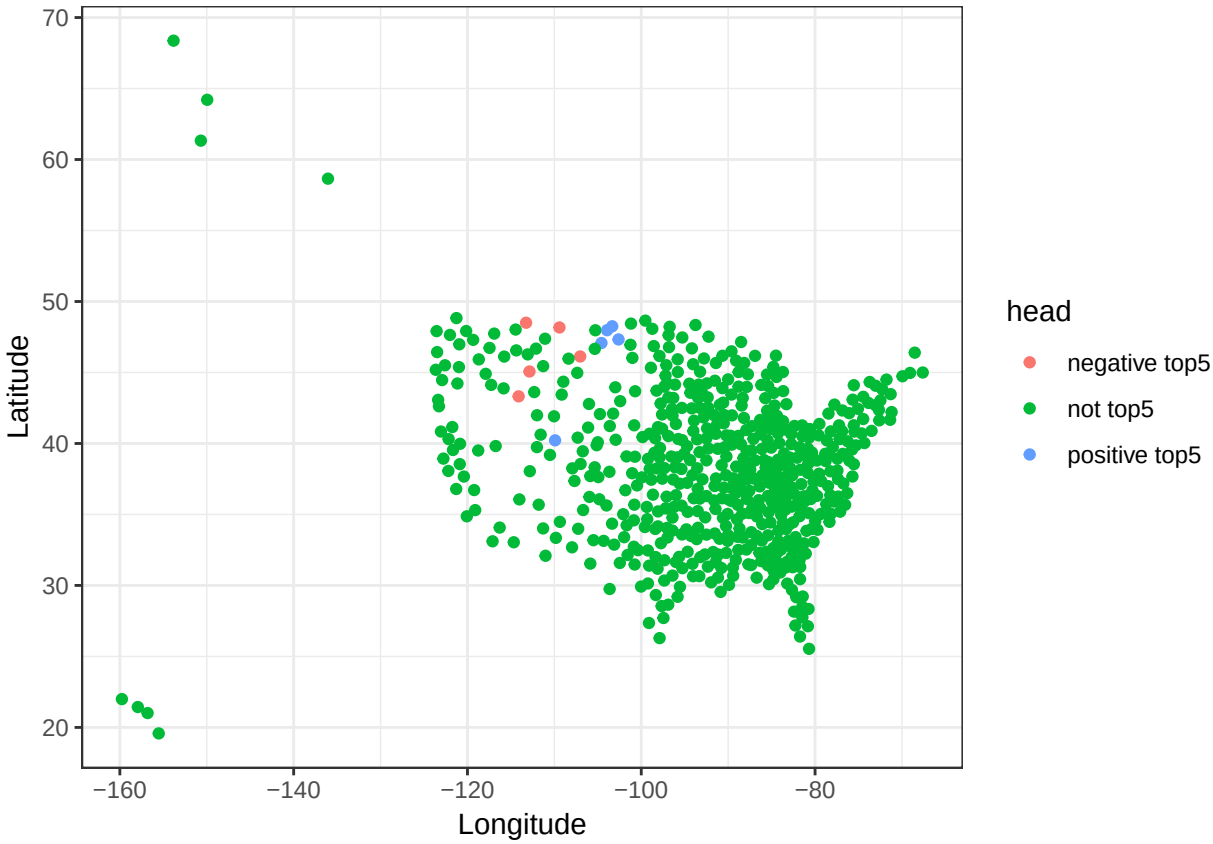


Table 18: Positive top5

x
Williston
Dickinson
Vernal
Sidney
Glendive

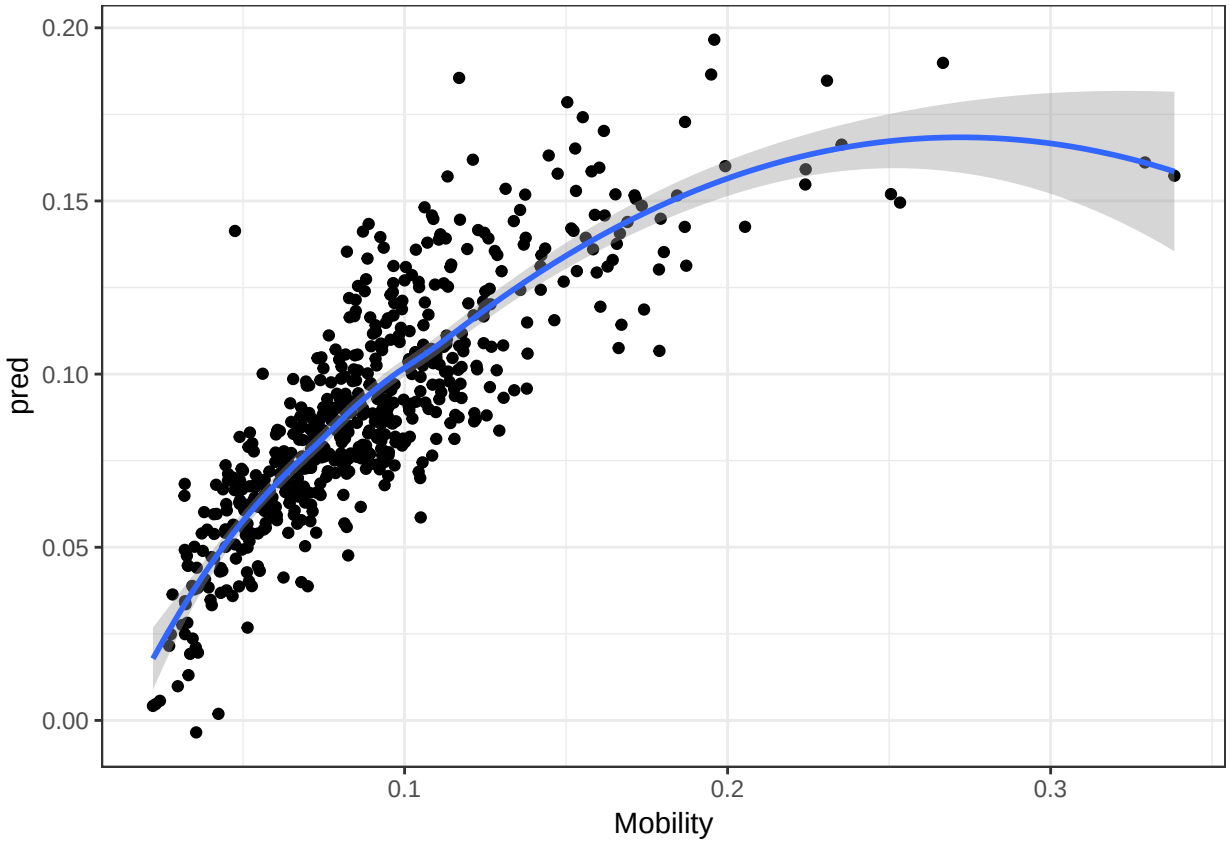
Table 19: negative top5

x
Havre
Twin Falls
Colstrip
Dillon
Shelby

9. Expectations and reality

a. Make a scatterplot of actual mobility against predicted mobility. Is the relationship linear? Should it be, is the model right? Is the relationship flat? Should it be, is the model right?

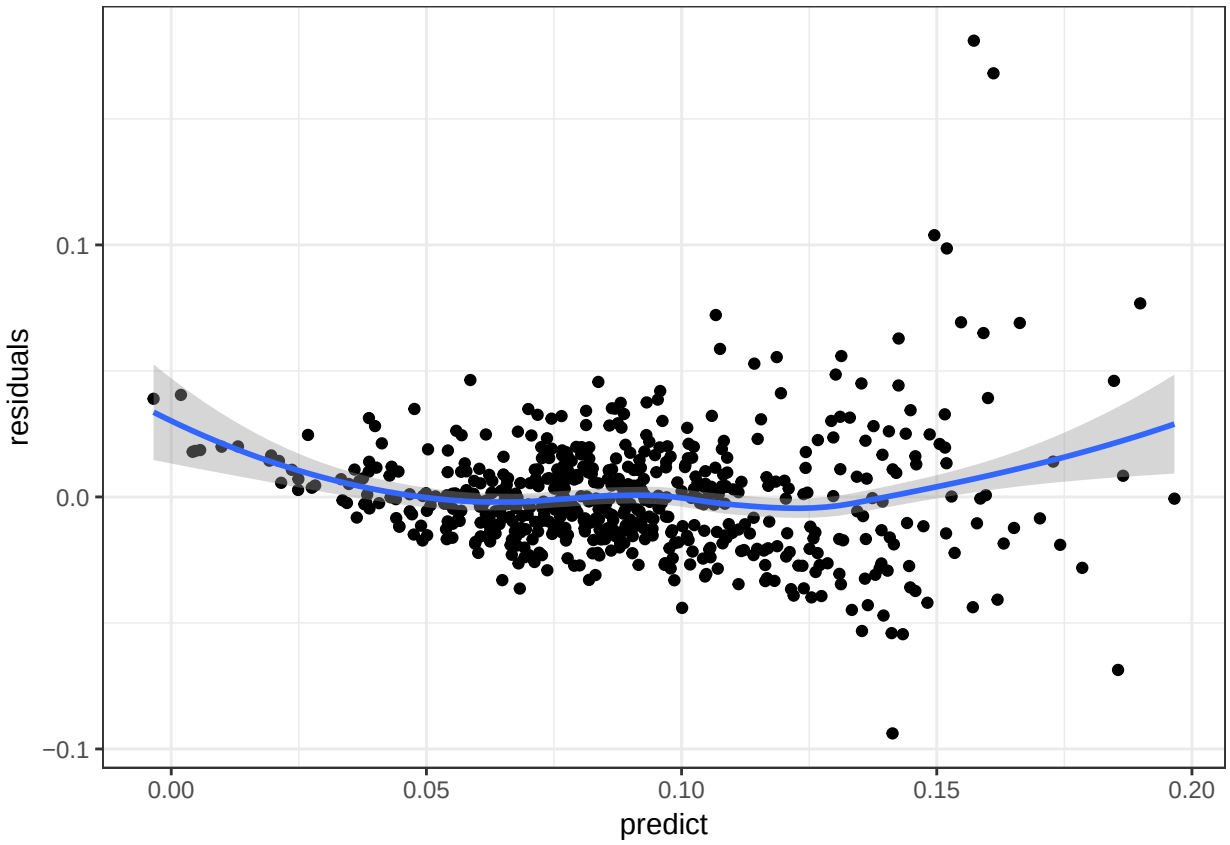
```
model = lm(Mobility ~ Black +  Commute + Middle_class + Manufacturing +
             Migration_out + Foreign_born + Single_mothers + Graduation, data = dat)
pred = predict(model, newdata = dat)


dat_pred = cbind(dat, pred)
dat_pred %>% ggplot(aes(Mobility, pred)) + geom_point() + geom_smooth()
```



관계는 선형보다 비선형에 가깝게 보인다. 또한 비례관계에 있음을 볼 수 있다. 그러나 모델이 관계를 잘 표현해서 적합은 잘 되었다고 판단한다.

b. Make a scatterplot of the model's residuals against predicted mobility. Is the relationship linear? Should it be, is the model right? Is the relationship flat? Should it be, is the model right?

```
dat_resi = mutate(dat, dat$Mobility - pred)
dat_resi = cbind(dat_resi, pred)
colnames(dat_resi)[14:15] = c('residuals', 'predict')


dat_resi %>% ggplot(aes(predict, residuals)) + geom_point() + geom_smooth()
```



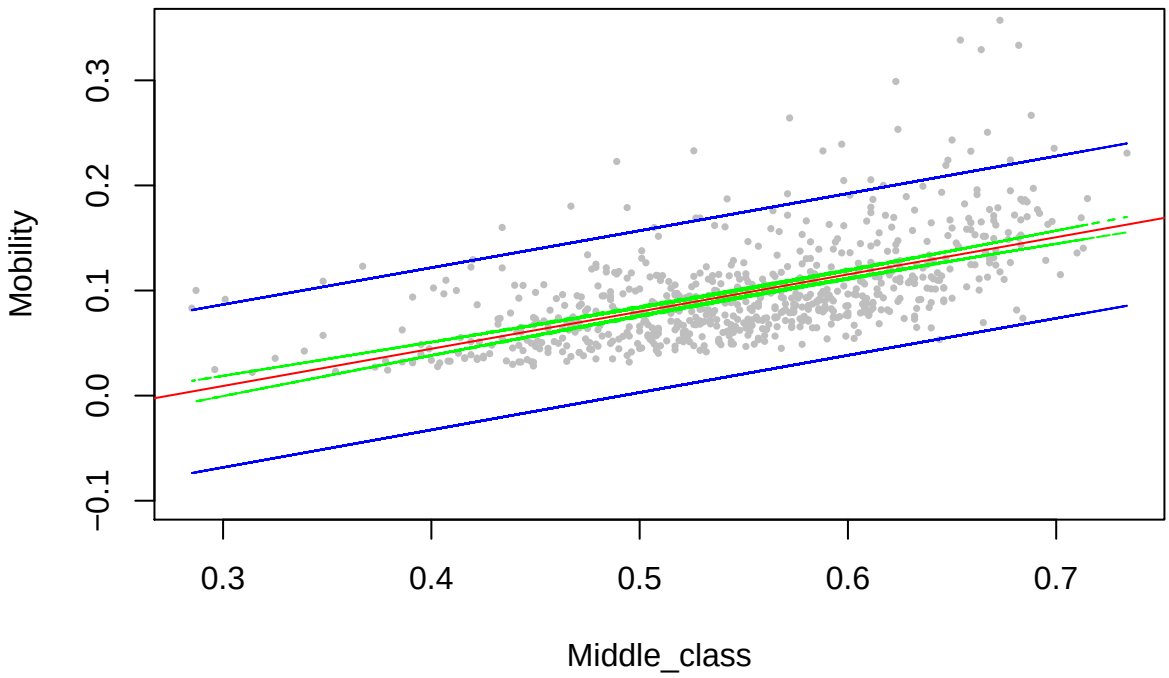
전체적으로 선형관계는 보이지 않는다. 잔차도가 선형성을 띄지 않으므로 좋은 모형으로 볼 수 있다. 그러나 predict가 0.1 이후로 분산이 커지는 경향을 보인다. 그러므로 등분산성을 위배한다고 판단하였고, 선형회귀의 가정이 깨지는 것을 볼 수 있다.

10. Cross-validation, bootstrap and smoothing

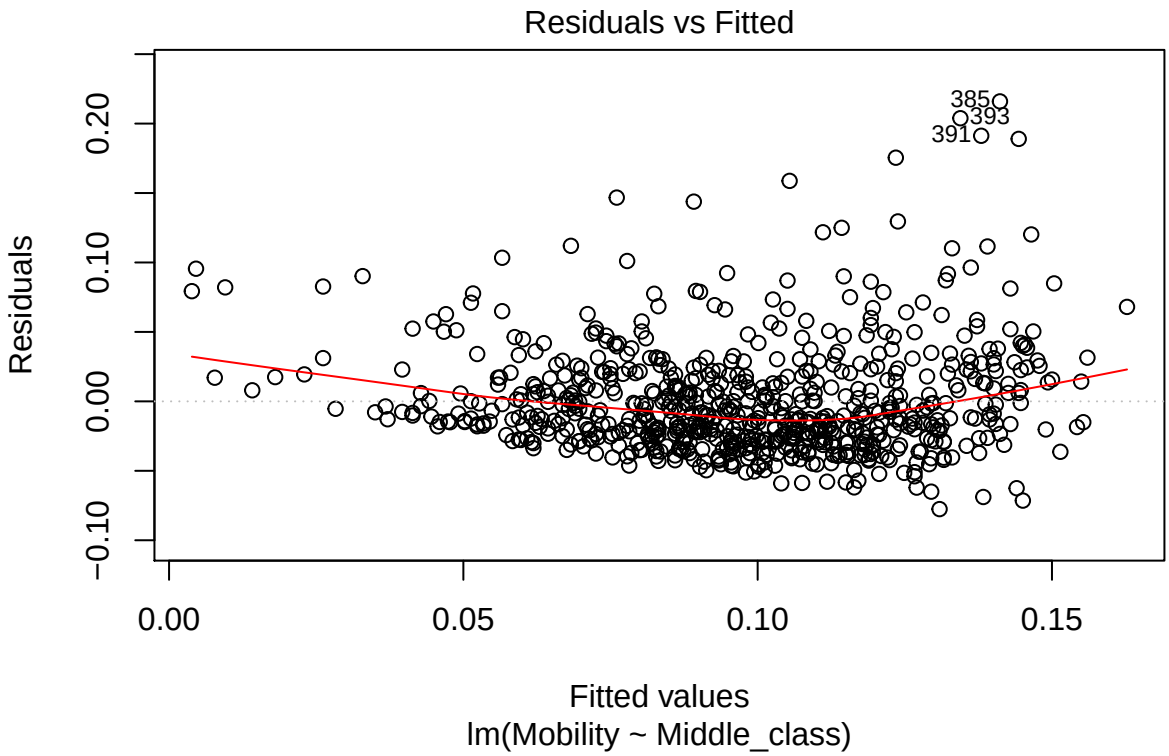
For this question, focus on predicting mobility by the fraction of middle class in the community.

a. Create a plot showing the data points, the fitted regression line, and 95% confidence and prediction intervals.

```
model = lm(Mobility ~ Middle_class, data = dat)
pred_con = predict(model, newdata = data.frame(Middle_class = dat$Middle_class), interval = 'confidence
pred_pre = predict(model, newdata = data.frame(Middle_class = dat$Middle_class), interval = 'prediction

plot(Mobility ~ Middle_class, data = dat, pch = 16, cex = 0.5, col = 'gray', ylim = c(-0.1, 0.35))
abline(model, col ='red')
lines(dat$Middle_class, pred_con[,"lwr"], lty="dashed", col="green")
lines(dat$Middle_class, pred_con[,"upr"], lty="dashed", col="green")
lines(dat$Middle_class, pred_pre[,"lwr"], col="blue")
lines(dat$Middle_class, pred_pre[,"upr"], col="blue")
```

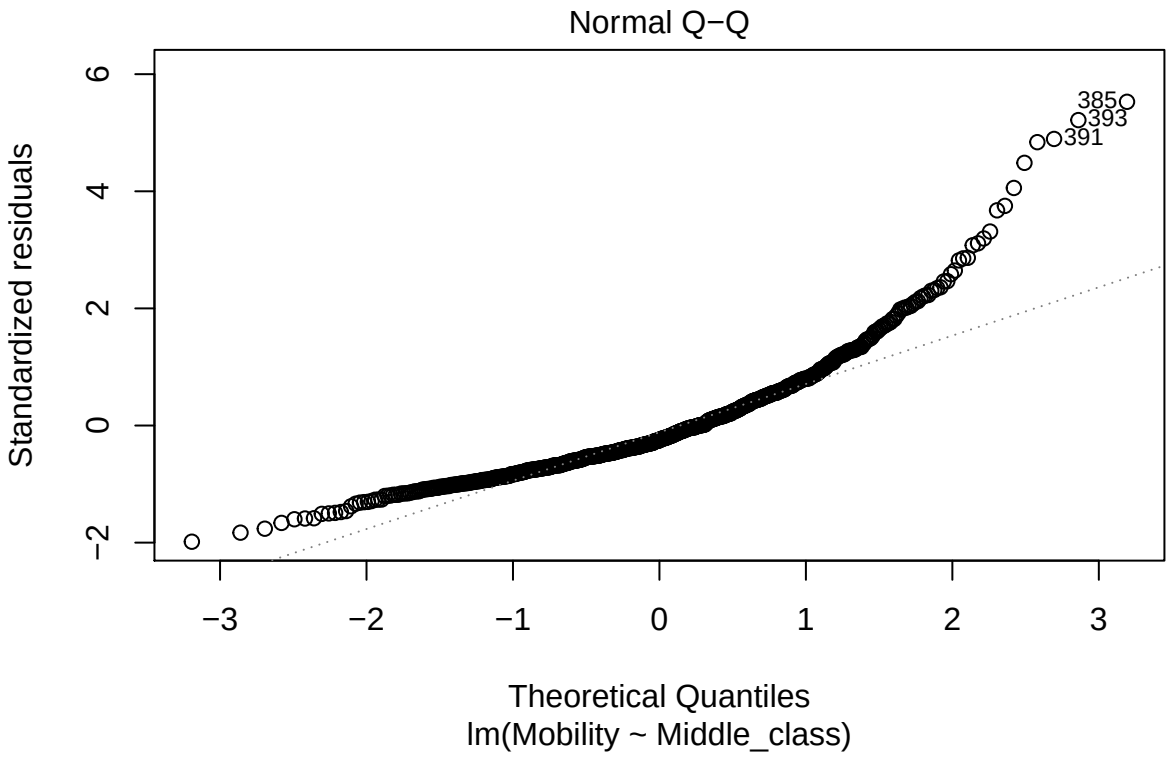



```
plot(model, which = 1)
```



잔차도에서 등분산성을 위배하므로 위에서 가정한 σ^2이 맞지 않는다고 할 수 있다.

```
plot(model, which = 2)
```




```
shapiro.test(scale(residuals(model)))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  scale(residuals(model))
## W = 0.87933, p-value < 2.2e-16
```

qq plot이 애매해서 잔차에 대한 정규성 검정을 실시한 결과, 유의수준 5%에서 귀무가설인 정규분포를 따른다를 위반하였으므로 정규성을 띄지 않는다고 판단한다.

```
dwtest(model)
```

```
##
##  Durbin-Watson test
##
## data:  model
## DW = 0.98291, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

유의수준 5%하에서 독립적이라는 가정을 기각하므로 독립적이지 않다는 대립가설이 채택된다. 즉 독립성가정은 위배된다.

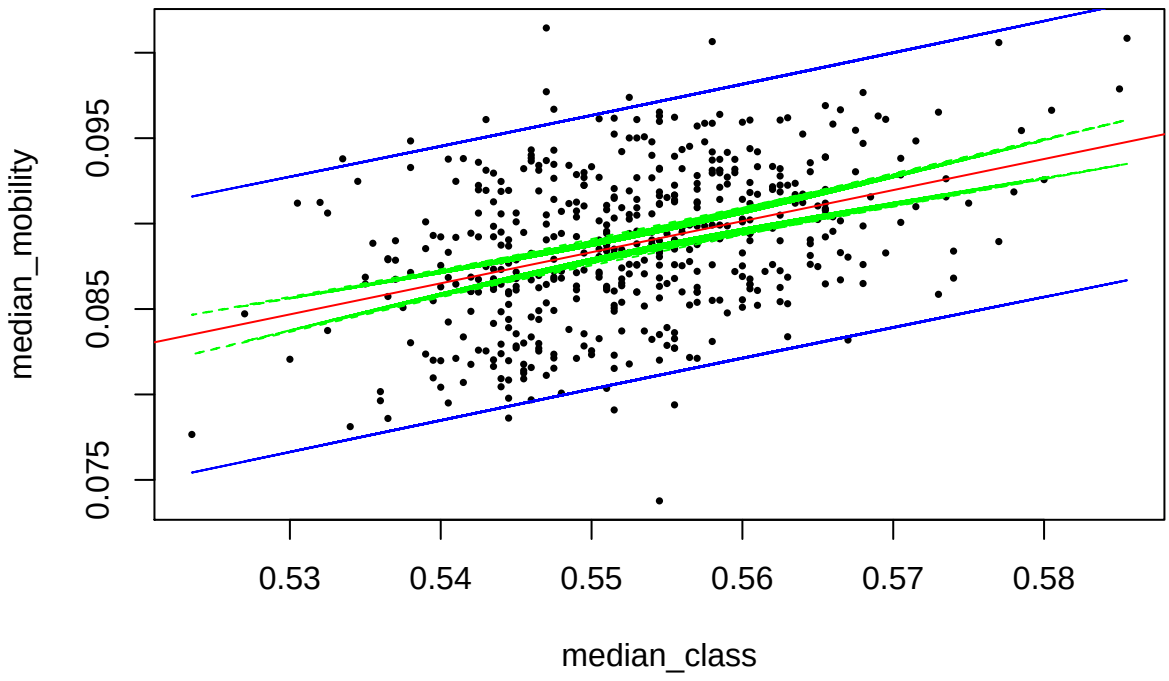
b. Use a resampling method to obtain 95% confidence interval. Can you build a 95% prediction interval?

```
dat_boot <- list()

for(i in 1:500){
  dat_boot[[i]] <- dat %>%
    sample_n(size = 100) %>%
```

```
    summarise(median_mobility = median(Mobility, na.rm = T), median_class =
                median(Middle_class, na.rm = T))
}
```

붓스트랩을 실시하였다.

```
dat_boot = bind_rows(dat_boot)
model = lm(median_mobility ~ median_class, data = dat_boot)
pred_con = predict(model, newdata = data.frame(median_class = dat_boot$median_class),
                   interval = 'confidence')
pred_pre = predict(model, newdata = data.frame(median_class = dat_boot$median_class),
                   interval = 'prediction')

plot(median_mobility ~ median_class, data = dat_boot, pch = 16, cex = 0.5)
abline(model, col ='red')
lines(dat_boot$median_class, pred_con[,"lwr"], lty="dashed", col="green")
lines(dat_boot$median_class, pred_con[,"upr"], lty="dashed", col="green")
lines(dat_boot$median_class, pred_pre[,"lwr"], col="blue")
lines(dat_boot$median_class, pred_pre[,"upr"], col="blue")
```



붓스트랩을 실시한한 후 95% 기준 각각의 신뢰구간과 예측구간을 만들었다.

c. Use a smoothing spline to do a nonparametric regression of Mobility on Middle_class. Use cross-validation to choose the degree of flexibility. Then use a resampling method to obtain 95% confidence interval. Create a plot showing the data points, the spline and the confidence interval.

```
model <- smooth.spline(x = dat$Middle_class, y = dat$Mobility, cv = TRUE)
model$df
```

```
## [1] 5.056275
```

```
resampling <- function(data) {
  n <- nrow(data)
  rs_rows <- sample(1:n, size = n, replace = TRUE)
  return(data[rs_rows,])
}
```

smooth.spline함수를 사용하여 비모수회귀를 실행하였는데, cv를 통해 최적의 df를 찾아내었다. resampling함수는 붓스트랩을 하기위한 함수로써 생성하였다.

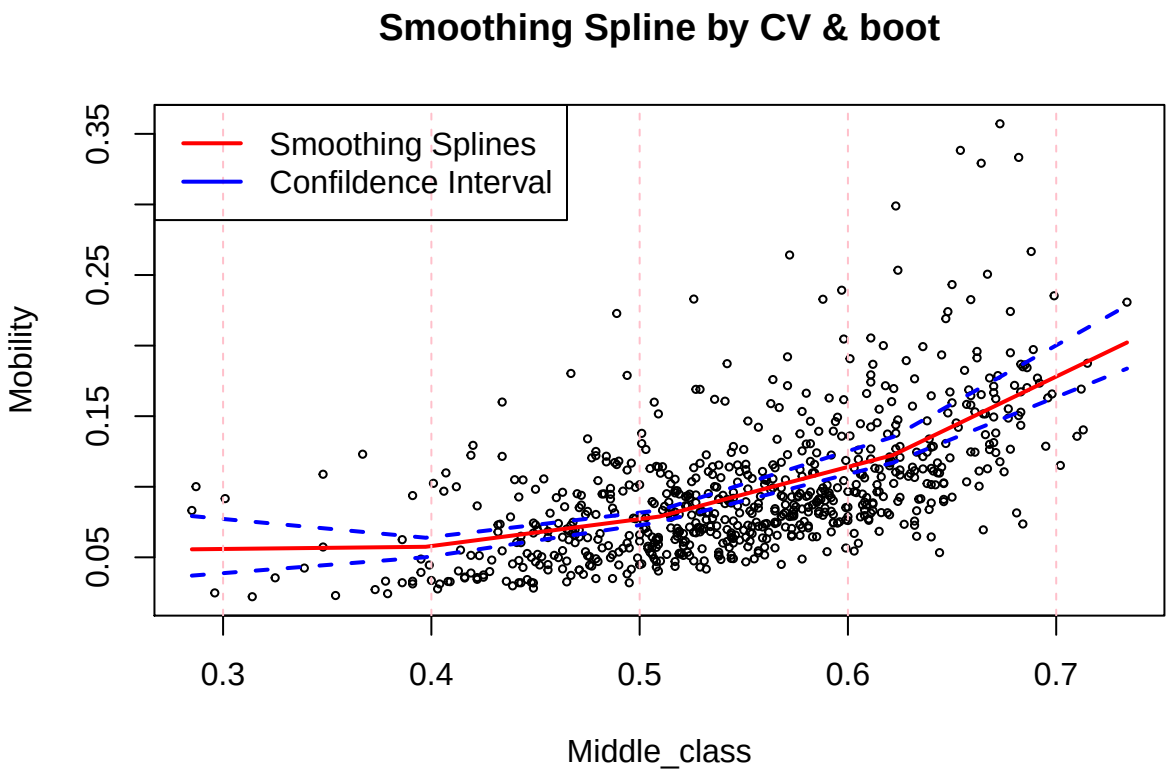
```
spline.estimator <- function(data, m = 300) {
  model <- smooth.spline(x=data[,1], y = data[,2], df = 5.056275)
  eval.grid <- seq(from = min(data[,1]), to = max(data[,1]), length.out = m)
  return(predict(model, x = eval.grid)$y)
}
```

최적의 df로 설정하고, 스플라인 추정치들을 구하는 함수입니다. 데이터의 첫번째 컬럼의 min값과 max값이 매듭의 시작과 끝이 되고 predict값을 반환합니다. 매듭의 조건과 예측값을 반환하는 스플라인의 계수를 생성합니다.

```
spline.cis <- function(data, B, alpha = 0.05, m = 300) {
  spline.main <- spline.estimator(data, m = m)
  spline.boots <- replicate(B, spline.estimator(resampling(data), m = m))
  cis.lower <- 2*spline.main - apply(spline.boots, 1, quantile, probs = 1-alpha/2)
  cis.upper <- 2*spline.main - apply(spline.boots, 1, quantile, probs = alpha/2)
  return(list(main.curve = spline.main, lower.ci = cis.lower, upper.ci = cis.upper,
              x = seq(from = min(data[,1]), to = max(data[,1]), length.out = m)))
}
```

붓스트랩을 통하여 스플라인의 신뢰구간을 구하는 함수이다.

```
sp.cis <- spline.cis(dat, B = 1000, alpha = 0.05, m = 5)

plot(dat$Middle_class, dat$Mobility, cex = 0.5,main = "Smoothing Spline by CV & boot",
     xlab = 'Middle_class', ylab = 'Mobility')
lines(x=sp.cis$x,y=sp.cis$main.curve, col = "red", lwd=2)
lines(x=sp.cis$x,y=sp.cis$lower.ci, lty=2,col = "blue", lwd = 2)
lines(x=sp.cis$x,y=sp.cis$upper.ci, lty=2,col = "blue", lwd = 2)
abline(v=c(0.3,0.4,0.5,0.6,0.7),lty=2,col="pink")
legend('topleft',c('Smoothing Splines', 'Confildence Interval'), col=c("red",'blue'),lwd=2)
```



최종적인 붓스트랩을 통한 스플라인함수의 신뢰구간입니다.

d. Test whether smoothing is needed here.

a번에서 언급한 선형회귀의 가정 독립성가정과 정규성가정, 등분산성가징이 모두 위배 되었으므로 별다른 가정이 필요없는 비모수적인 spline smoothing이 필요하다.

또한 b번과 c번을 통해서 실질적으로 비모수가정도 모집단의 분포를 잘 설명하고, 신뢰구간과 예측구간도 안정적이기 때문에 spline smoothing을 사용하는 것이 좋다고 판단한다.