

# Analysis

Dingyi Li

4/14/2021

## Multiple Linear Regression

### Data preview

#### Read in data

```
dt = read.csv("data&figures/dt.csv")
summary(dt)
```

```
##      State      County      HPI      Personal_Income
## Length:2703    Length:2703    Min.   : 82.32    Min.   : 22440
## Class :character Class :character 1st Qu.: 184.81 1st Qu.: 38374
## Mode  :character Mode  :character Median : 238.80 Median : 43578
##                                     Mean  : 311.45 Mean   : 45972
##                                     3rd Qu.: 367.36 3rd Qu.: 50469
##                                     Max.   :2266.07 Max.   :229825
## Poverty_Percentage Population HighSchoolLess HighSchoolOnly
## Min.   : 2.70    Min.   : 728    Min.   : 1.40    Min.   : 7.80
## 1st Qu.:10.10    1st Qu.: 15785 1st Qu.: 8.20    1st Qu.:29.40
## Median :13.10    Median : 32924 Median :11.40    Median :34.30
## Mean   :13.83    Mean   : 167666 Mean   :12.49    Mean   :33.89
## 3rd Qu.:16.70    3rd Qu.: 90870 3rd Qu.:15.80    3rd Qu.:38.90
## Max.   :41.10    Max.   :10039107 Max.   :46.70    Max.   :54.50
## SomeCollege BachelorAndHigher Unemployment_Rate
## Min.   :11.2    Min.   : 7.20    Min.   : 1.600
## 1st Qu.:27.7    1st Qu.:15.80    1st Qu.: 3.050
## Median :31.0    Median :20.20    Median : 3.700
## Mean   :31.0    Mean   :22.62    Mean   : 3.928
## 3rd Qu.:34.2    3rd Qu.:27.40    3rd Qu.: 4.600
## Max.   :47.3    Max.   :75.30    Max.   :18.300
```

### Correlation Check

```
cor(scale(as.matrix(dt[,c(7,8,9,10)])))
```

### Education parameters

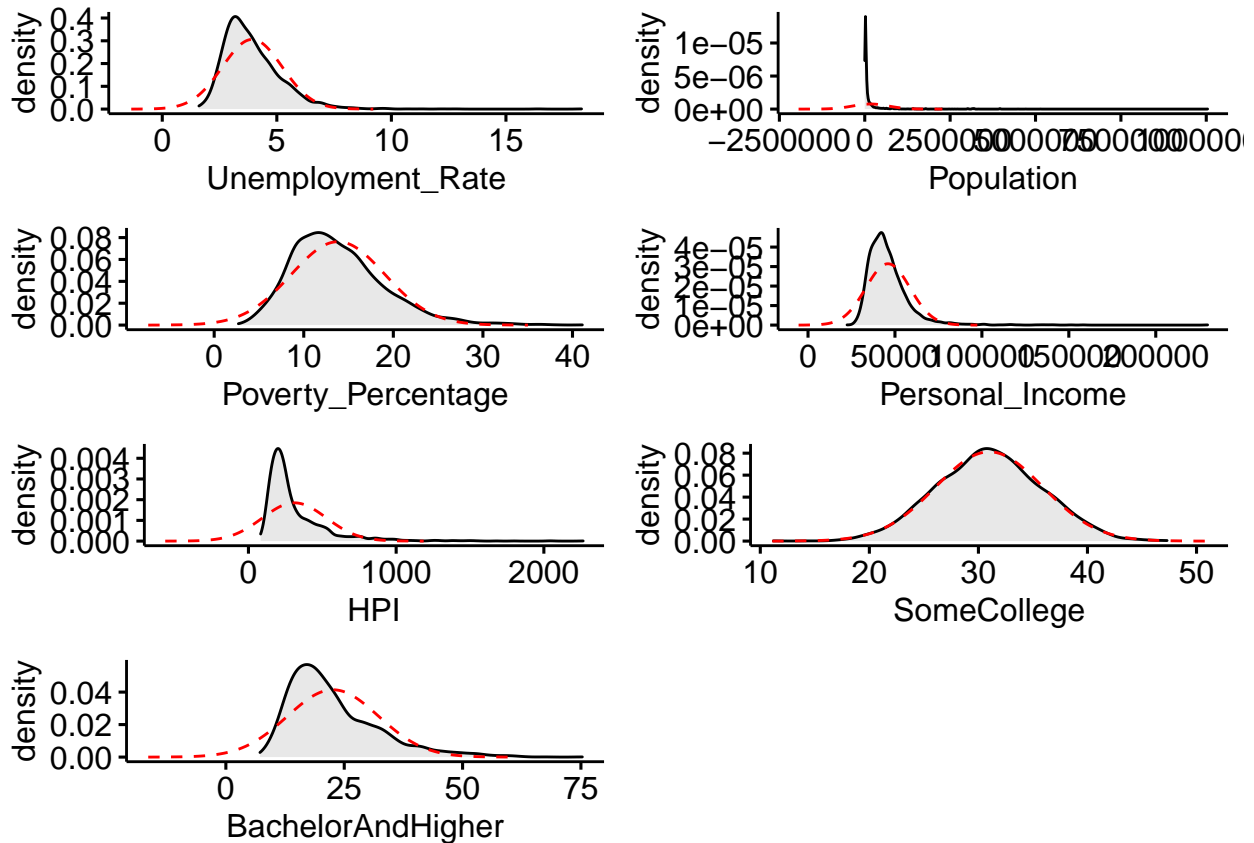
##	HighSchoolLess	HighSchoolOnly	SomeCollege	BachelorAndHigher
## HighSchoolLess	1.0000000	0.3066823	-0.41862584	-0.61009423
## HighSchoolOnly	0.3066823	1.0000000	-0.27129305	-0.80308215
## SomeCollege	-0.4186258	-0.2712931	1.00000000	-0.05682677
## BachelorAndHigher	-0.6100942	-0.8030821	-0.05682677	1.00000000

## Histogram

```
library(ggpubr)
```

```
## Loading required package: ggplot2
```

```
a<-ggdensity(dt, x = "Unemployment_Rate", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
b<-ggdensity(dt, x = "Population", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
c<-ggdensity(dt, x = "Poverty_Percentage", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
d<-ggdensity(dt, x = "Personal_Income", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
e<-ggdensity(dt, x = "HPI", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
f<-ggdensity(dt, x = "SomeCollege", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
g<-ggdensity(dt, x = "BachelorAndHigher", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
ggarrange(a,b,c,d,e,f,g, ncol = 2, nrow = 4)
```



### Histogram for logtransformation

```
temp=dt
temp$HPI <- log(dt$HPI)
temp$Personal_Income <- log(dt$Personal_Income)
temp$Poverty_Percentage <- log(dt$Poverty_Percentage)
temp$Population <- log(dt$Population)
temp$HighSchoolLess <- log(dt$HighSchoolLess)
temp$BachelorAndHigher <- log(dt$BachelorAndHigher)
temp$Unemployment_Rate <- log(dt$Unemployment_Rate)

library(ggpubr)
a<-ggdensity(temp, x = "Unemployment_Rate", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
b<-ggdensity(temp, x = "Population", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
c<-ggdensity(temp, x = "Poverty_Percentage", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
d<-ggdensity(temp, x = "Personal_Income", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
e<-ggdensity(temp, x = "HPI", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
f<-ggdensity(temp, x = "SomeCollege", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
g<-ggdensity(temp, x = "BachelorAndHigher", fill = "lightgray") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
ggarrange(a,b,c,d,e,f,g, ncol = 2, nrow = 4)
```



## Model fitting

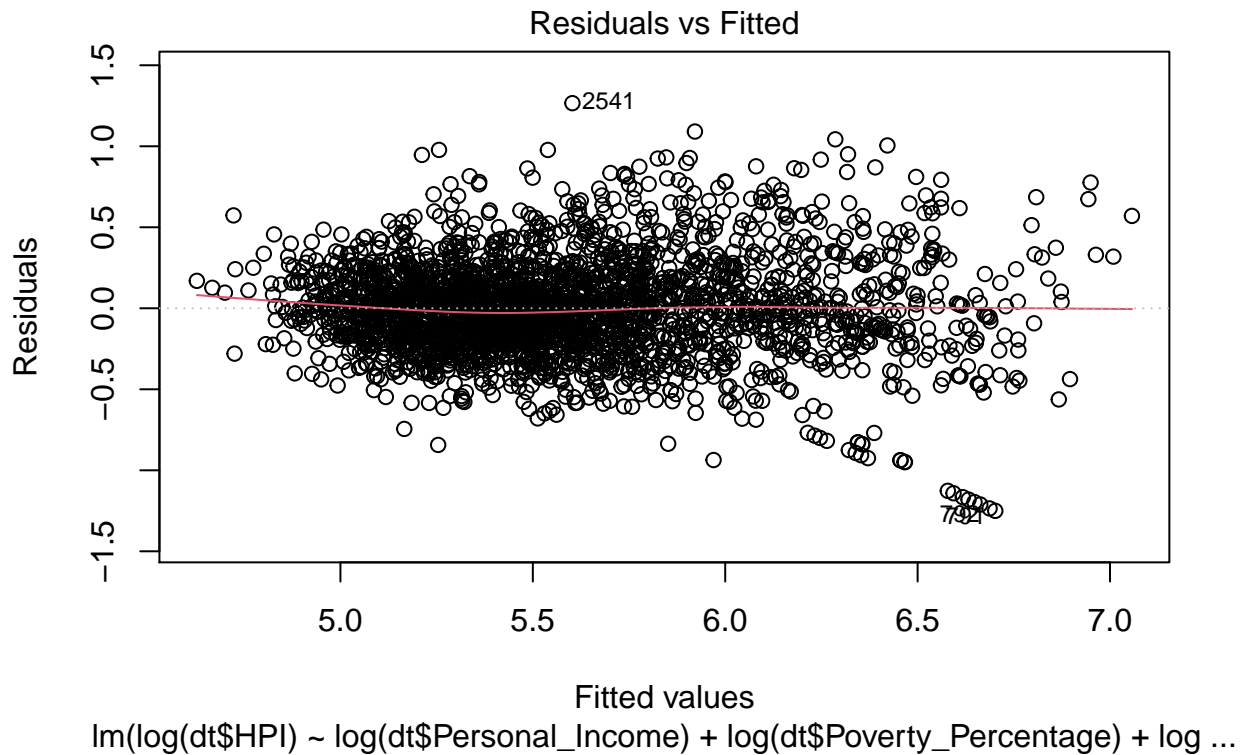
```
m1 = lm(log(dt$HPI)~log(dt$Personal_Income)+log(dt$Poverty_Percentage)+log(dt$Unemployment_Rate)+log(dt$
summary(m1)
```

```
##
## Call:
## lm(formula = log(dt$HPI) ~ log(dt$Personal_Income) + log(dt$Poverty_Percentage) +
##     log(dt$Unemployment_Rate) + log(dt$Population) + dt$SomeCollege +
##     log(dt$BachelorAndHigher))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.25022 -0.17598 -0.01332  0.16097  1.26604
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.236569   0.447373  -7.235 6.06e-13 ***
## log(dt$Personal_Income)  0.526325   0.040401  13.027 < 2e-16 ***
## log(dt$Poverty_Percentage) -0.078788   0.022802  -3.455 0.000558 ***
## log(dt$Unemployment_Rate)  0.072500   0.022477   3.225 0.001273 **
```

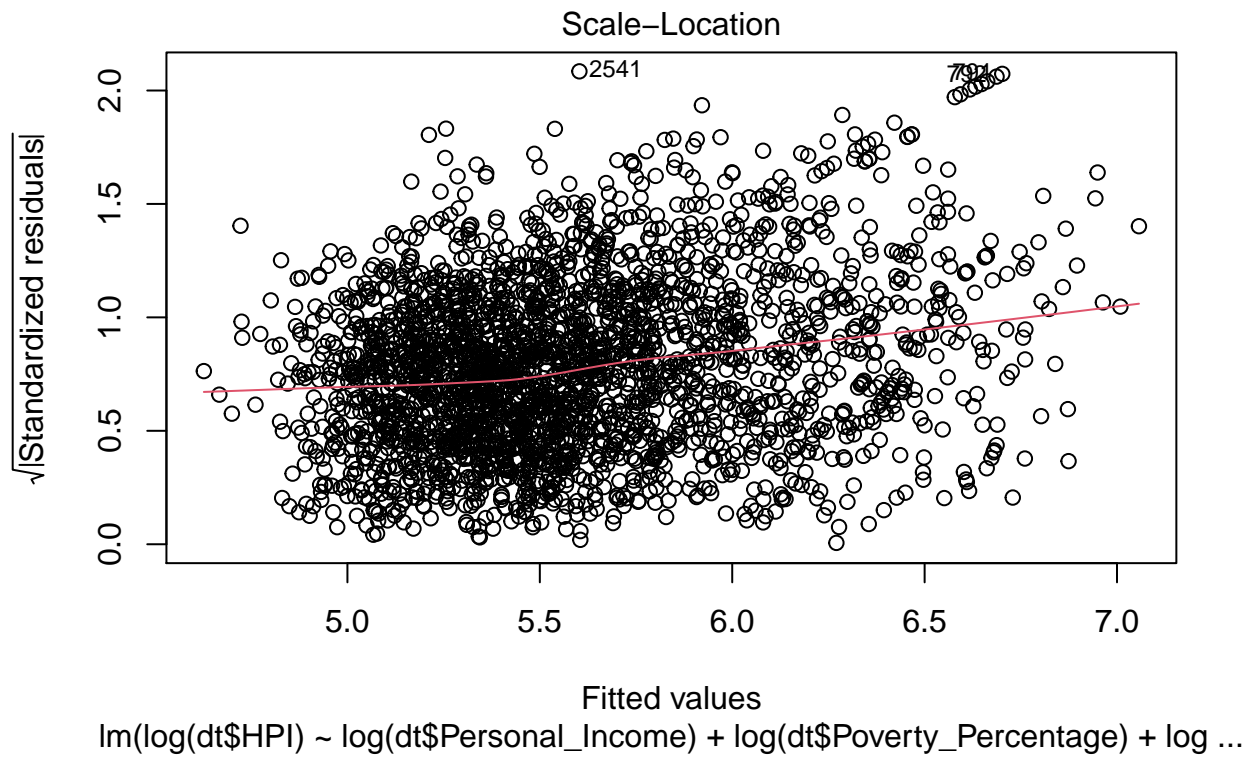
```
## log(dt$Population)          0.199449    0.004604  43.321 < 2e-16 ***
## dt$SomeCollege              0.009598    0.001212   7.918 3.49e-15 ***
## log(dt$BachelorAndHigher)  0.286792    0.023175  12.375 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2917 on 2696 degrees of freedom
## Multiple R-squared:  0.684, Adjusted R-squared:  0.6832
## F-statistic: 972.4 on 6 and 2696 DF, p-value: < 2.2e-16
```

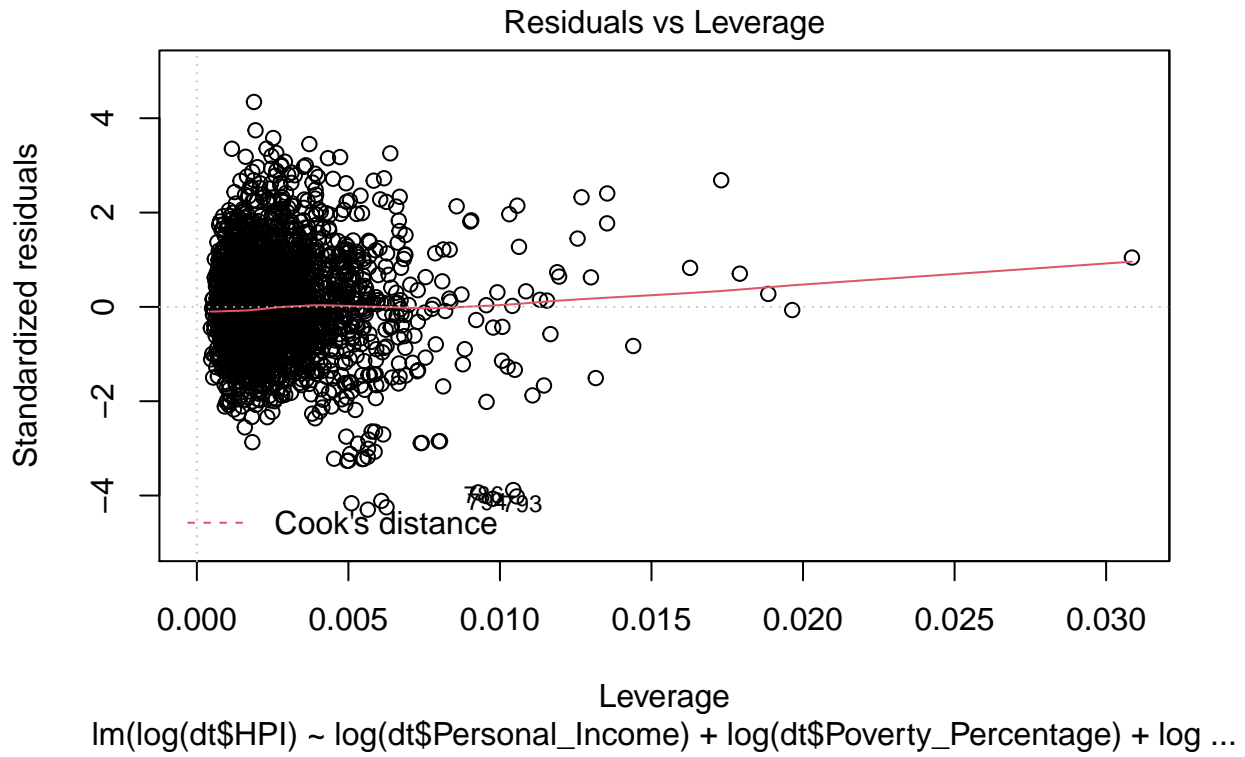
## Diagnostic Plots

```
plot(m1)
```









```
car::vif(m1)
```

```
##      log(dt$Personal_Income) log(dt$Poverty_Percentage)
##                2.623520                2.307242
##      log(dt$Unemployment_Rate)      log(dt$Population)
##                1.410572                1.388636
##                dt$SomeCollege      log(dt$BachelorAndHigher)
##                1.091027                2.559650
```

## correlation plot

```
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
```

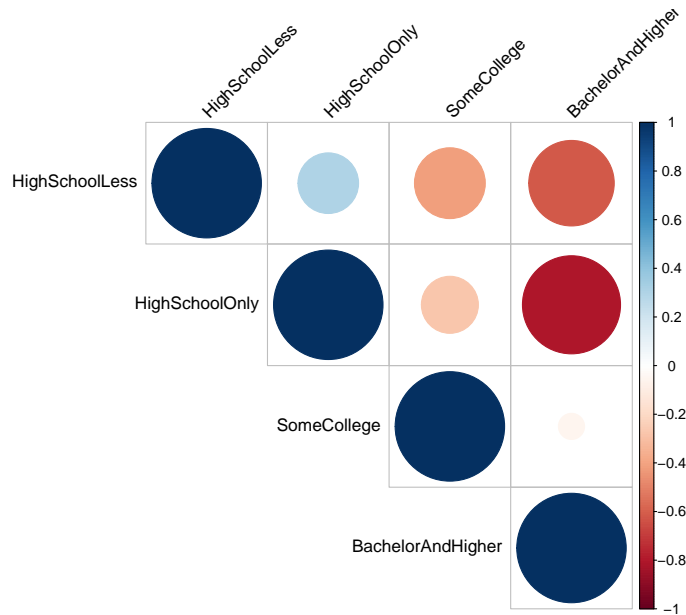


```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
df <- dt %>% dplyr::select(HighSchoolLess:BachelorAndHigher)
res <- cor(df)
```

```
corrplot(res, type = "upper", order = "hclust",
          tl.col = "black", tl.srt = 45)
```



## old pca

```
library("tidymodels")
```

```
## -- Attaching packages ----- tidymodels 0.1.3 --
```

```
## v broom      0.7.6      v rsample      0.0.9
## v dials      0.0.9      v tibble      3.1.0
## v infer      0.5.4      v tidyr       1.1.3
## v modeldata  0.1.0      v tune        0.1.5
## v parsnip    0.1.5      v workflows   0.2.2
## v purrr      0.3.4      v workflowsets 0.0.2
## v recipes    0.1.16     v yardstick   0.0.8
```

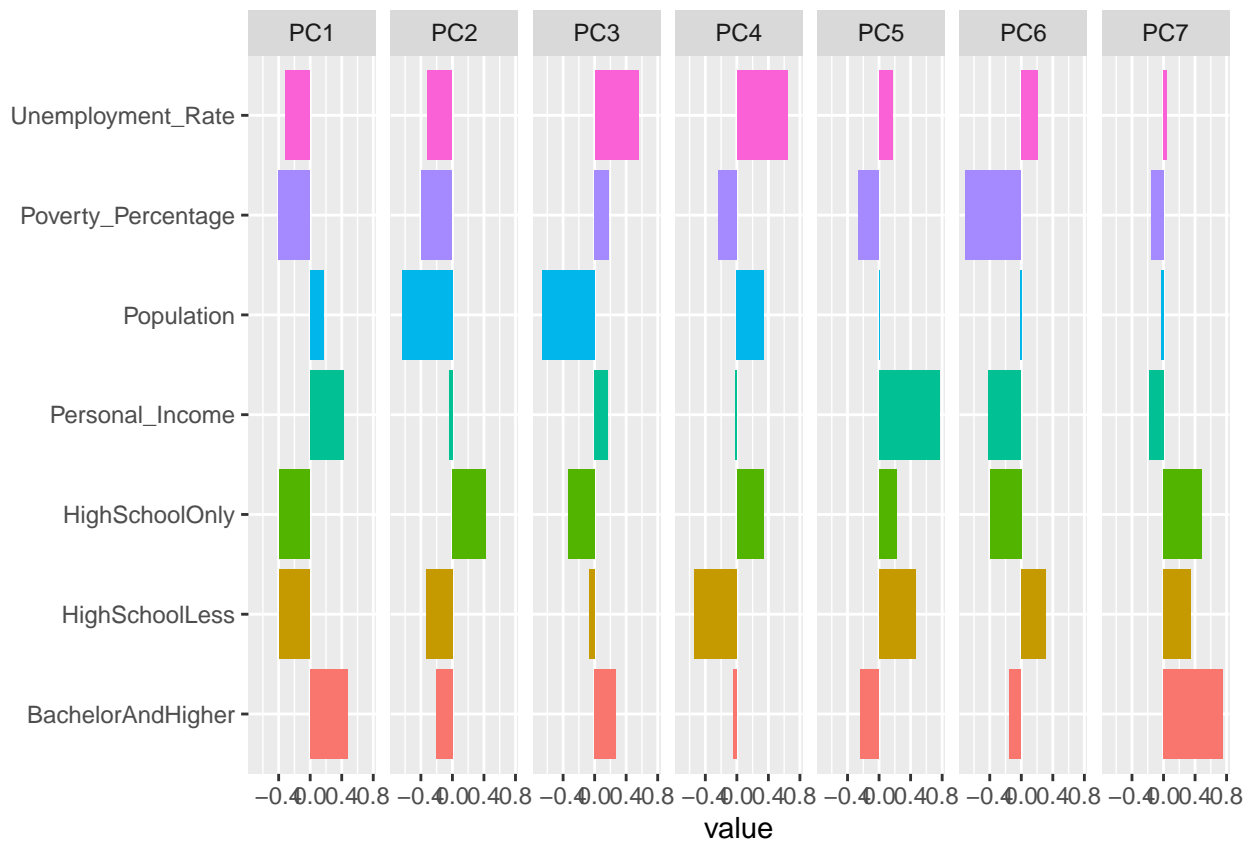
```
## -- Conflicts ----- tidymodels_conflicts() --
```

```
## x purrr::discard() masks scales::discard()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## x recipes::step()  masks stats::step()
## * Use tidymodels_prefer() to resolve common conflicts.
```

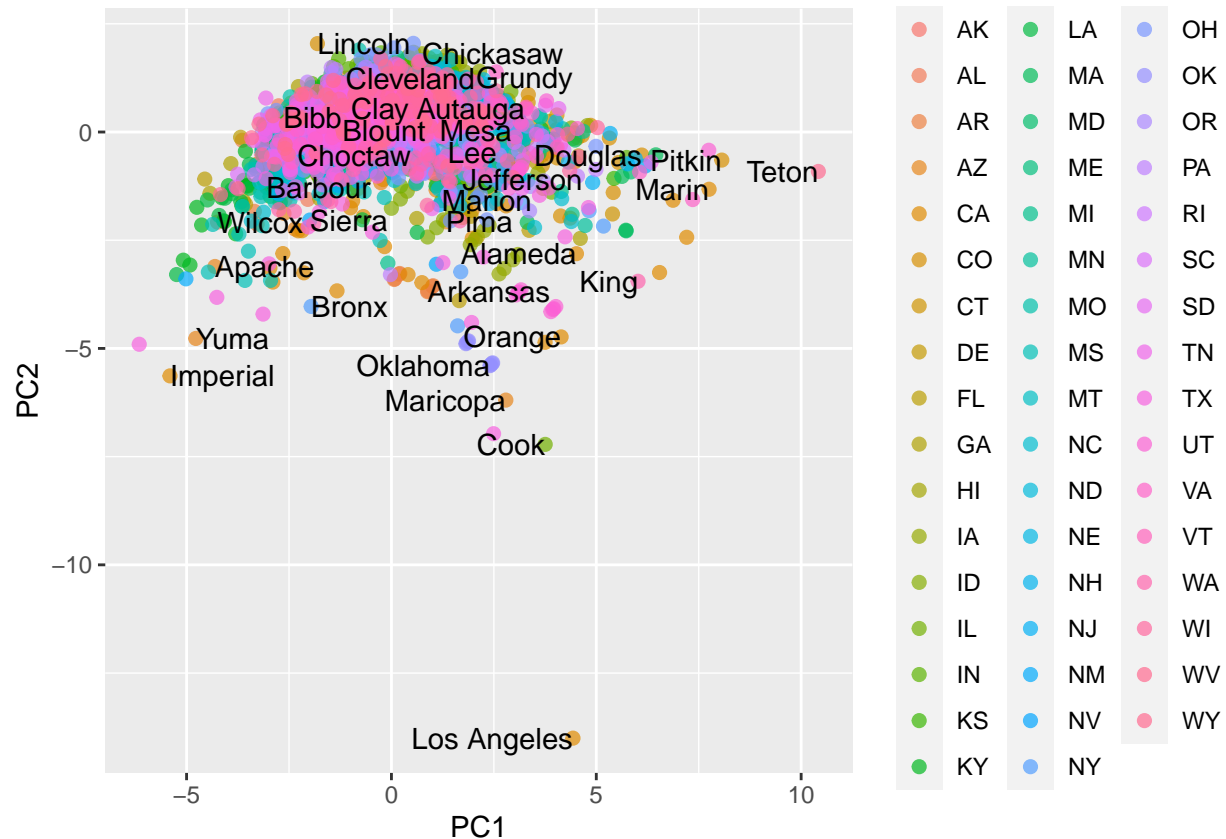
```
library(forcats)
pca_rec <- recipe(HPI ~., data = dt) %>%
  update_role(State, County, SomeCollege, new_role = "id") %>%
  step_normalize(all_predictors()) %>%
  step_pca(all_predictors())

pca_prep <- prep(pca_rec)
tidied_pca <- tidy(pca_prep, 2)

tidied_pca %>%
  filter(component %in% paste0("PC", 1:7)) %>%
  mutate(component = fct_inorder(component)) %>%
  ggplot(aes(value, terms, fill = terms)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~component, nrow = 1) +
  labs(y = NULL)
```



```
juice(pca_prep) %>%
  ggplot(aes(PC1, PC2, label = County)) +
  geom_point(aes(color = State), alpha = 0.7, size = 2) +
  geom_text(check_overlap = TRUE, hjust = "inward") +
  labs(color = NULL)
```



## stepwise regression

```
library(MASS)
```

```
##
```

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

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

```
##
```

```
## select
```

```
# Fit the full model
```

```
full.model <- lm(HPI ~ Personal_Income + Poverty_Percentage + Population + HighSchoolLess + HighSchoolO
```

```
# Stepwise regression model
```

```
step.model <- stepAIC(full.model, direction = "both",  
                      trace = FALSE)
```

```
summary(step.model)
```

```
##
```

```
## Call:
```

```
## lm(formula = HPI ~ Personal_Income + Poverty_Percentage + Population +
```

```
## HighSchoolLess + SomeCollege + BachelorAndHigher + Unemployment_Rate,
```

```

##      data = dt)
##
## Residuals:
##      Min        1Q      Median        3Q        Max
## -543.63   -72.53   -13.70    49.97   1194.45
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.192e+02  3.937e+01 -13.188 < 2e-16 ***
## Personal_Income    3.105e-03  3.320e-04   9.352 < 2e-16 ***
## Poverty_Percentage -5.949e+00  8.334e-01  -7.138 1.21e-12 ***
## Population        9.522e-05  5.779e-06  16.478 < 2e-16 ***
## HighSchoolLess    1.312e+01  8.559e-01  15.334 < 2e-16 ***
## SomeCollege       6.434e+00  7.132e-01   9.022 < 2e-16 ***
## BachelorAndHigher  1.414e+01  5.075e-01  27.863 < 2e-16 ***
## Unemployment_Rate  1.806e+01  2.570e+00   7.027 2.67e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 146 on 2695 degrees of freedom
## Multiple R-squared:  0.5291, Adjusted R-squared:  0.5279
## F-statistic: 432.6 on 7 and 2695 DF,  p-value: < 2.2e-16

```