myHomework3

chenpeng

2024-07-04

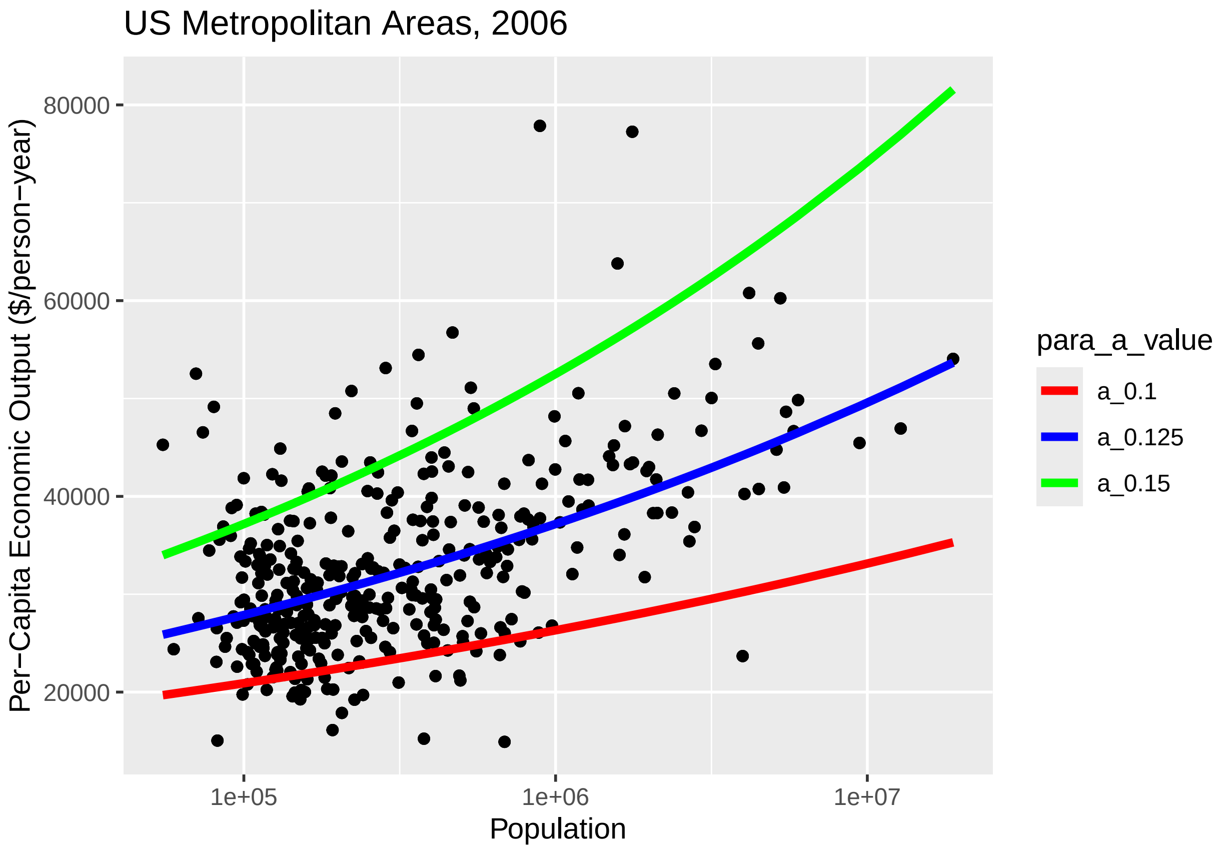
目录

```
gmp <- read.table("../../data/gmp.dat")
gmp$pop <- round(gmp$gmp/gmp$pcgmp)
```

1

```
gmp <- gmp |>
    mutate(a_0.125 = 6611 * (gmp/pcgmp)^(1/8), a_0.1 = 6611 *
        (gmp/pcgmp)^0.1, a_0.15 = 6611 * (gmp/pcgmp)^0.15)
gmp_tidy <- gmp[, 3:7] |>
    gather(para_a_value, nlmfit, -pcgmp, -pop)
```

```r
gmp_tidy |>
    ggplot() + geom_point(aes(x = pop, y = pcgmp)) + labs(x = "Population",
    y = "Per-Capita Economic Output ($/person-year)", title = "US Metropolitan Areas, 2006") +
    geom_line(aes(x = pop, y = nlmfit, color = para_a_value),
        size = 1.5) + scale_x_continuous(trans = "log10") +
    scale_color_manual(values = c("red", "blue", "green"))
```



2.

```r
mse <- function(para, N = gmp$pop, Y = gmp$pcgmp) {
    return(mean((Y - para[1] * N^para[2])^2))
}

mse(c(6611, 0.15))
```

```
## [1] 207057513
```

```r
mse(c(5000, 0.1))
```

```
## [1] 298459914
```

3.

```r
nlm(mse, c(y0 = 6611, a = 1/8))
```

```
## $minimum
## [1] 61857060
##
## $estimate
## [1] 6611.0000000    0.1263177
##
## $gradient
## [1] 50.048639 -9.983778
##
## $code
## [1] 2
##
## $iterations
## [1] 3
```

minimum represents the the value of the estimated minimum of f. estimate represents the point at which the minimum value of f is obtained.

4.

```r
plm <- function(para, N = gmp$pop, Y = gmp$pcgmp) {
    t <- nlm(mse, c(para[1], para[2]), N, Y)
    return(list(parameters = c(t$estimate[1], t$estimate[2]),
        MSE = t$minimum))
}
plm(c(6611, 0.15))
```

```
## $parameters
## [1] 6610.9999997    0.1263182
##
## $MSE
## [1] 61857060
```

```
plm(c(5000, 0.1))
```

```
## $parameters
## [1] 5000.0000008    0.1475913
##
## $MSE
## [1] 62521484
```

They differ because their y0 are different. The starting value, y0 = 6611 and a = 0.15, has the lower MSE.

5.

a.

```
mean(gmp$pcgmp)
```

```
## [1] 32922.53
```

```
sem1 <- sd(gmp$pcgmp)/sqrt(nrow(gmp))
sem1
```

```
## [1] 481.9195
```

formula

$$SEM = \sqrt{\frac{\Sigma_{i=1}^{n}(x_i - \bar{x})^2}{n(n-1)}}$$

b.

```
mean.jackknife <- function(i, pc = gmp$pcgmp) {
    return(mean(pc[-i]))
}
```

c.

```
n = nrow(gmp)
jackknifed.means <- sapply(1:n, mean.jackknife)
```

d.

```
sem2 <- sqrt((n - 1)^2/n * var(jackknifed.means))
sem2
```

```
## [1] 481.9195
```

```
abs(sem2 - sem1)/sem1
```

```
## [1] 4.010371e-15
```

非常符合

6.

```
plm.jackknife <- function(para, N = gmp$pop, Y = gmp$pcgmp) {
    # para = c(6611, 0.125)
    jackknife <- function(i) {
        return(plm(para, N[-i], Y[-i])$parameters)
    }
    n = length(N)
    jackknifed.para <- sapply(1:n, jackknife)
    y.sd <- sqrt((n - 1)^2/n * var(jackknifed.para[1, ]))
    a.sd <- sqrt((n - 1)^2/n * var(jackknifed.para[2, ]))
    sd <- c(y.sd, a.sd)
    names(sd) = c("y.sd", "a.sd")
    return(sd)
}
```

```
plm.jackknife(c(6611, 0.125))
```

```
##         y.sd         a.sd
## 1.136653e-08 9.901003e-04
```

7.

```
gmp_2013 <- read.csv("../../data/gmp-2013.dat", header = T,
    sep = " ") |>
    mutate(pop = round(gmp/pcgmp))
plm(c(6611, 1/8), N = gmp_2013$pop, Y = gmp_2013$pcgmp)
```

```
## $parameters
## [1] 6611.0000002    0.1433688
##
## $MSE
## [1] 135210524
```

```
plm.jackknife(c(6611, 0.125), N = gmp_2013$pop, Y = gmp_2013$pcgmp)
```

```
##         y.sd         a.sd
## 2.692652e-08 1.098548e-03
```

It haven't changed significantly.