homework3

zza

2025-06-29

1

```
# 1. 读取数据并预处理
gmp <- read.table("gmp.dat", header = TRUE)
gmp$pop <- round(gmp$gmp / gmp$pcgmp)

a1 <- 0.1
numerator1 <- sum(gmp$pcgmp * (gmp$pop)^a1)
denominator1 <- sum((gmp$pop)^(2 * a1))
y0_1 <- numerator1 / denominator1

a2 <- 0.15
numerator2 <- sum(gmp$pcgmp * (gmp$pop)^a2)
denominator2 <- sum((gmp$pop)^(2 * a2))
y0_2 <- numerator2 / denominator2
# 定义幂律函数
power_law <- function(N, y0, a) {
  y0 * N^a
}

# 绘制散点图（对数坐标）
plot(gmp$pop, gmp$pcgmp,
     log = "xy",
     xlab = "log(Pop)",
     ylab = "l0g(PcGMP)",
     main = "Per Capita GMP vs. Population",
     pch = 20,     # 点样式
     col = "blue")
```

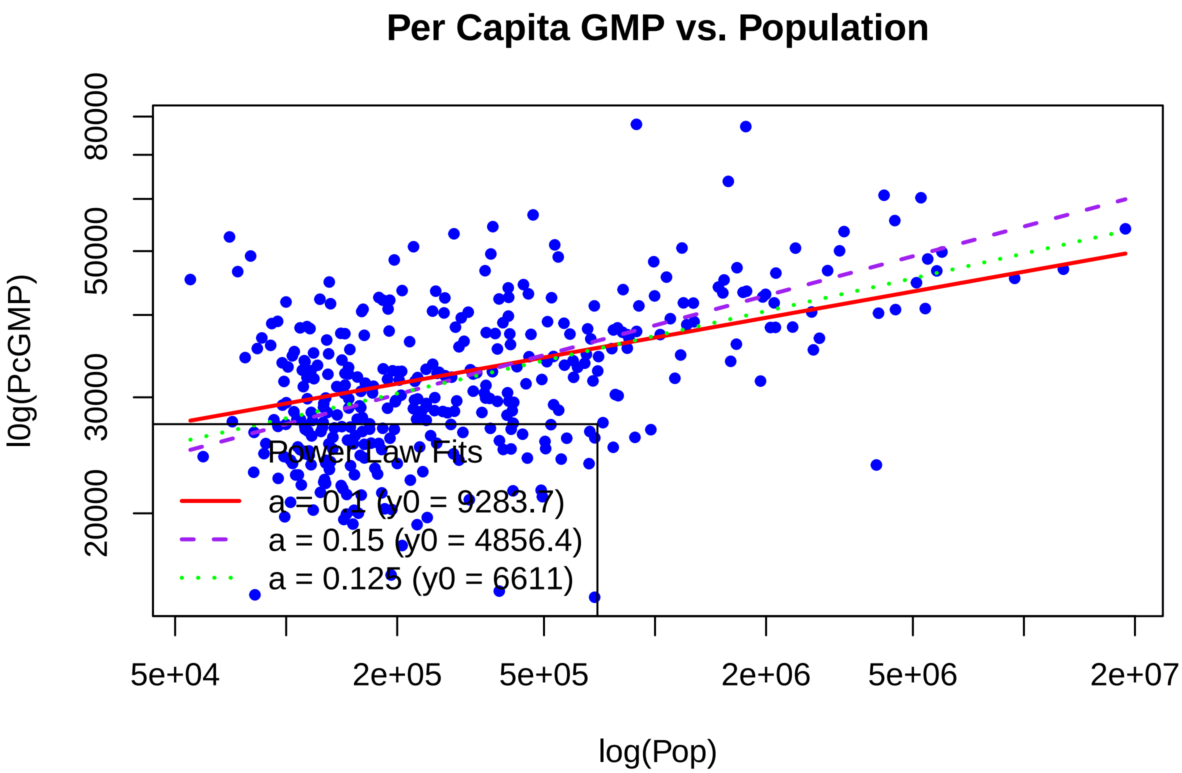
```r
# 添加 3 条曲线
curve(power_law(x, y0 = y0_1, a = a1),
      add = TRUE, col = "red", lwd = 2, lty = 1)



curve(power_law(x, y0 = y0_2, a = a2),  # a = 0.10
      add = TRUE, col = "purple", lwd = 2, lty = 2)

# y0=6611, a=1/8
curve(power_law(x, y0 = 6611, a = 1/8),
      add = TRUE, col = "green", lwd = 2, lty = 3)

# 图例
legend("bottomleft",
       legend = c(
         paste0("a = ", a1, " (y0 = ", round(y0_1, 1), ")"),
         paste0("a = ", a2, " (y0 = ", round(y0_2, 1), ")"),
         "a = 0.125 (y0 = 6611)"
       ),
       col = c("red", "purple", "green"),
       lty = c(1, 2, 3),
       lwd = 2,
       title = "Power Law Fits")
```



2

```r
mse <- function(params, N = gmp$pop, Y = gmp$pcgmp) {
  y0 <- params[1]
  a <- params[2]
  if (a <= 0 || any(N <= 0)) return(1e10)
  predictions <- y0 * (N ^ a)
  mean((Y - predictions)^2)
}



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

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

```r
mse(c(5000, 0.10))
```

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

3

```r
# 使用 nlm() 优化 mse()
result1 <- nlm(mse, c(y0 = 4800, a = 0.1))
result2 <-nlm(mse, c(y0 = 6611, a = 1/8))
result3 <-nlm(mse, c(y0 = 9000, a = 0.05))

# 打印结果
print(result1)
```

```
## $minimum
## [1] 62747987
##
## $estimate
## [1] 4800.000001    0.150686
##
## $gradient
## [1] -1244.134735    -2.503395
##
## $code
## [1] 2
##
## $iterations
## [1] 5
```

```r
print(result2)
```

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

```
## $iterations
## [1] 3
```

```r
print(result3)
```

```
## $minimum
## [1] 62861726
##
## $estimate
## [1] 9000.0000004    0.1026275
##
## $gradient
## [1] 678.014375   1.594424
##
## $code
## [1] 2
##
## $iterations
## [1] 6
```

结果解释：- minimum：最小均方误差（MSE）。- estimate：最优参数（y_0 和 a）。- 初始值不同可能导致不同的收敛速度和结果。

4

```r
plm <- function(y0_init, a_init, N = gmp$pop, Y = gmp$pcgmp) {
  result <- nlm(mse, c(y0_init, a_init), N = N, Y = Y)

  list(
    y0 = result$estimate[1],
    a = result$estimate[2],
    mse = result$minimum
  )
}

result1 <- plm(y0_init = 6611, a_init = 0.15)
print(result1)
```

```
## $y0
```

```
## [1] 6611
##
## $a
## [1] 0.1263177
##
## $mse
## [1] 61857060
```

```
result2 <- plm(y0_init = 5000, a_init = 0.10)
print(result2)
```

```
## $y0
## [1] 5000
##
## $a
## [1] 0.1475913
##
## $mse
## [1] 62521484
```

原因是非线性优化可能收敛到不同的局部最小值, 初始值不同可能导致优化路径不同;

其中 y 从 5000，a0 从 0.10 开始的更小

5

a

```
## 均值: 32922.53
```

```
## 经典标准误: 481.9195
```

b

```
jackknife_mean <- function(i, data = gmp$pcgmp) {
  mean(data[-i])  # 排除第 i 个观测值
}
```

c

```
jackknifed.means <- sapply(1:nrow(gmp), jackknife_mean)
```

d

```
# 计算刀切方差和标准误
n <- nrow(gmp)
jackknife_var <- ((n-1)^2/n) * var(jackknifed.means)
jackknife_sem <- sqrt(jackknife_var)
```

```
## 刀切标准误: 481.9195
```

```
## 与传统标准误的比值: 1
```

6

```
plm.jackknife <- function(y0_init, a_init, N = gmp$pop, Y = gmp$pcgmp) {
  n <- length(N)

  jack_y0 <- numeric(n)
  jack_a <- numeric(n)

  for (i in 1:n) {

    N_sub <- N[-i]
    Y_sub <- Y[-i]


    fit <- plm(y0_init, a_init, N_sub, Y_sub)


    jack_y0[i] <- fit$y0
    jack_a[i] <- fit$a
  }

  se_y0 <- sqrt(((n-1)^2/n) * var(jack_y0))
```

```r
  se_a <- sqrt(((n-1)^2/n) * var(jack_a))


  list(se_y0 = se_y0, se_a = se_a)
}

# 使用与问题 4 相同的初始值
jack_result1 <- plm.jackknife(y0_init = 6611, a_init = 0.15)
jack_result2 <- plm.jackknife(y0_init = 5000, a_init = 0.10)

print(jack_result1)
```

```
## $se_y0
## [1] 1.175826e-08
##
## $se_a
## [1] 0.0009898304
```

```r
print(jack_result2)
```

```
## $se_y0
## [1] 2.033898e-08
##
## $se_a
## [1] 0.0009979824
```

7

```r
gmp2013 <- read.table("gmp-2013.dat", header = TRUE)
gmp2013$pop <- round(gmp2013$gmp / gmp2013$pcgmp)



# 使用与 2006 年相同的初始值
fit2013 <- plm(y0_init = 5000, a_init = 0.10,
               N = gmp2013$pop, Y = gmp2013$pcgmp)

# 计算标准误
```

```r
se2013 <- plm.jackknife(y0_init = 5000, a_init = 0.10,
                        N = gmp2013$pop, Y = gmp2013$pcgmp)


print(fit2013)
```

```
## $y0
## [1] 5000
##
## $a
## [1] 0.164427
##
## $mse
## [1] 139208731
```

```r
print(se2013)
```

```
## $se_y0
## [1] 6.248473e-08
##
## $se_a
## [1] 0.001122513
```