Multilevel Pspline from Kakao Search Data

Choi TaeYoung

2020-10-14

Contents

1 필요한 패키지 **1**

2 데이터 **2**

3 Multilevel 모델에 적용 **3**
3.1 Naive's GCV vector 찾기 4

4 그래프 **4**

1 필요한 패키지

2 데이터

- Y data : Y데이터의 경우 139주(2018년 1월 ~ 2020년 8월)동안의 카카오로 “홈트레이닝”을 검색한 횟수를 지역별로 나타냄
- X data : X데이터의 경우 17개의 지역별 인구수

3 Multilevel 모델에 적용

- 논문의 방법인 EM알고리즘을 통해 multilevel spline 방법으로 최적의 μ 벡터를 찾았다.

```
#multilevel
  #beta_hat_vector 구하기
  grain_out <- NULL
  J=139
  beta_hat <- NULL
  for(m in 1:139){
    result2_out <- NULL
    results2 <- glm(unlist(y_list[m]) ~ unlist(x_list), maxit=2000)
      kth_beta_hat <- coef(results2)[2]
      kth_var <- diag(vcov(results2))[2]
      grain_out <- list(kth_beta_hat, kth_var)
      grain_out
    beta_hat <- rbind(beta_hat,grain_out)
  }
```

- p-spline 기법을 활용하여 새롭게 짠 코드로 리얼데이터에 적용

```
EM_out <- MultiEM_ps(x=z_month,
                     beta_hat_vec=unlist(beta_hat[,1]),
                     V=diag(unlist(beta_hat[,2])),
                     lambda=1)
tail(EM_out$mu)
```

```
##                 [,1]
## [134,] 1.927178e-06
## [135,] 2.008419e-06
## [136,] 2.238056e-06
## [137,] 2.616470e-06
## [138,] 3.134398e-06
## [139,] 4.102928e-06
```

3.1 Naive's GCV vector 찾기

- Multilevel과 성능을 비교하기위해서 Naive한 방법으로 구해보자.
- Naive기법 역시 P-spline으로 코드를 짠 후 실행했다.

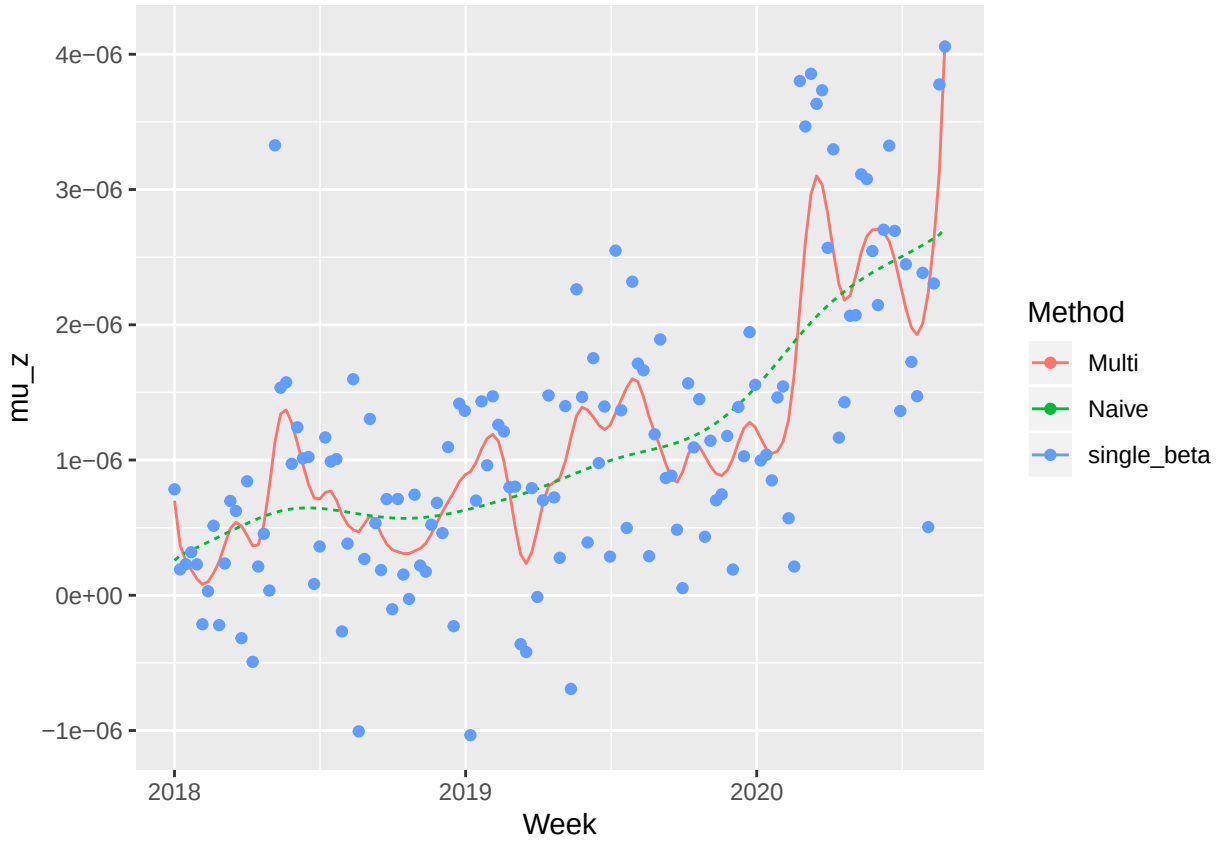
```
naive_out <- naive_ps(x=z_month,
                      beta_hat_vec=unlist(beta_hat[,1]),
                      lambda = 10000)
tail(naive_out$mu)
```

```
##                [,1]
## [134,] 2.565626e-06
## [135,] 2.589098e-06
## [136,] 2.613291e-06
## [137,] 2.638159e-06
## [138,] 2.665680e-06
## [139,] 2.715045e-06
```

4 그래프

```
# hat_all
single_beta <- unlist(beta_hat[,1]) %>% as.vector()
mu_z_multi <- EM_out$mu %>% as.vector()

#mu_z_naive <- naive_out$mu %>% as.vector()
test_mon <- fread("HT_kakao_search.csv")
test_mon <- test_mon[-1,1]

mu_z <- cbind(obs_y[-1,2],mu_z_multi) %>% as.data.frame
mu_z <- rename(mu_z, Week = V2)

# naive

df2 <- cbind(test_mon,single_beta)
df2 <- rename(df2, Week =V1)

df2_naive <- naive_out$betaEst
df2 <- cbind(df2,df2_naive)
df2 <- rename(df2, Naive =df2_naive)

df2 <- cbind(df2,mu_z$mu_z_multi)
df2 <- rename(df2, Multi =V2)

# gather함수 사용
df2 <- gather(df2[, c("Week",  "single_beta", "Naive", "Multi")],
              key = "Method", value = "mu_z", -Week)
df2$Week <- parse_date_time(df2$Week, "ymd")
df2$Week <- as.Date(df2$Week, format="%Y-%m-%d")

g <- ggplot(df2, aes(x=Week, y=mu_z, group=Method)) +
  geom_line(data= df2 %>% dplyr::filter(Method != "single_beta"),aes(x = Week, y = mu_z, color = Method
  geom_point(data=df2 %>% dplyr::filter(Method == "single_beta"), aes(x = Week, y = single_beta, color =
  guides(linetype = "none") +
  scale_color_discrete(name = "Method")
```

```
g
```


4e−06
3e−06
2e−06
1e−06
0e+00
−1e−06
mu_z
2018
2019
2020
Week
Method
Multi
Naive
single_beta