Homework 4 - Report

3170105743 李政达

2020/7/9

We continue examining the diffusion of tetracycline among doctors in Illinois in the early 1950s, building on our work in lab 6. You will need the data sets `ckm_nodes.csv` and `ckm_network.dat` from the labs.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
ckm_nodes <- read_csv("../data/ckm_nodes.csv")
```

```
## Parsed with column specification:
## cols(
##   city = col_character(),
##   adoption_date = col_double(),
##   medical_school = col_character(),
##   attend_meetings = col_character(),
##   medical_journals = col_double(),
##   free_time_with = col_character(),
##   discuss_medicine_socially = col_character(),
##   club_with_drs = col_character(),
##   drs_among_three_best_friends = col_double(),
##   practicing_here = col_character(),
##   office_visits_per_week = col_character(),
##   proximity_to_other_drs = col_character(),
##   specialty = col_character()
## )
```

```
ckm_network <- read_table("../data/ckm_network.dat", col_names = FALSE)
```

```
## Parsed with column specification:
## cols(
##   .default = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

1. Clean the data to eliminate doctors for whom we have no adoption-date information, as in the labs. Only use this cleaned data in the rest of the assignment.

```
valid_nodes <- !is.na(ckm_nodes$adoption_date)
ckm_network <- ckm_network[valid_nodes, valid_nodes]
ckm_nodes <- ckm_nodes[valid_nodes, ]
```

2. Create a new data frame which records, for every doctor, for every month, whether that doctor began prescribing tetracycline that month, whether they had adopted tetracycline before that month, the number of their contacts who began prescribing strictly *before* that month, and the number of their contacts who began prescribing in that month or earlier. Explain why the dataframe should have 6 columns, and 2125 rows. Try not to use any loops.

```
doc.info <- data.frame("doctor" = rep(seq(1, 125), each = 17),
                       "month" = rep(seq(1, 17), time = 125))
doc.info <- doc.info %>%
  mutate(begin = ckm_nodes$adoption_date[doc.info$doctor] == doc.info$month) %>%
  mutate(before = ckm_nodes$adoption_date[doc.info$doctor] < doc.info$month)
contact <- ckm_network[rep(1:125, each = 17), ]
m1 <- matrix(ckm_nodes$adoption_date[doc.info$doctor] < doc.info$month,
             nrow = 17)
m2 <- matrix(ckm_nodes$adoption_date[doc.info$doctor] <= doc.info$month,
             nrow = 17)
doc.info <- doc.info %>%
  mutate(num_strict_before = rowSums(m1[rep(seq(1,17), time = 125), ] &
                                     contact)) %>%
  mutate(num_begin_before = rowSums(m2[rep(seq(1,17), time = 125), ] &
                                     contact))
```

Ans: In order to distinguish the different doctors and different months, we need to list all the combinations of doctors and months, which need two columns and $125 \times 17 = 2125$ rows. We also need to record the four attributes asked, which need four columns. In conclusion, the dataframe should have 6 columns and 2125 rows.

3. Let

   p_k = Pr(A doctor starts prescribing tetracycline this month | Number of doctor's contacts prescribing before this mon

   and

   q_k = Pr(A doctor starts prescribing tetracycline this month | Number of doctor's contacts prescribing this month = k)

   We suppose that p_k and q_k are the same for all months.

   a. Explain why there should be no more than 21 values of k for which we can estimate p_k and q_k directly from the data.

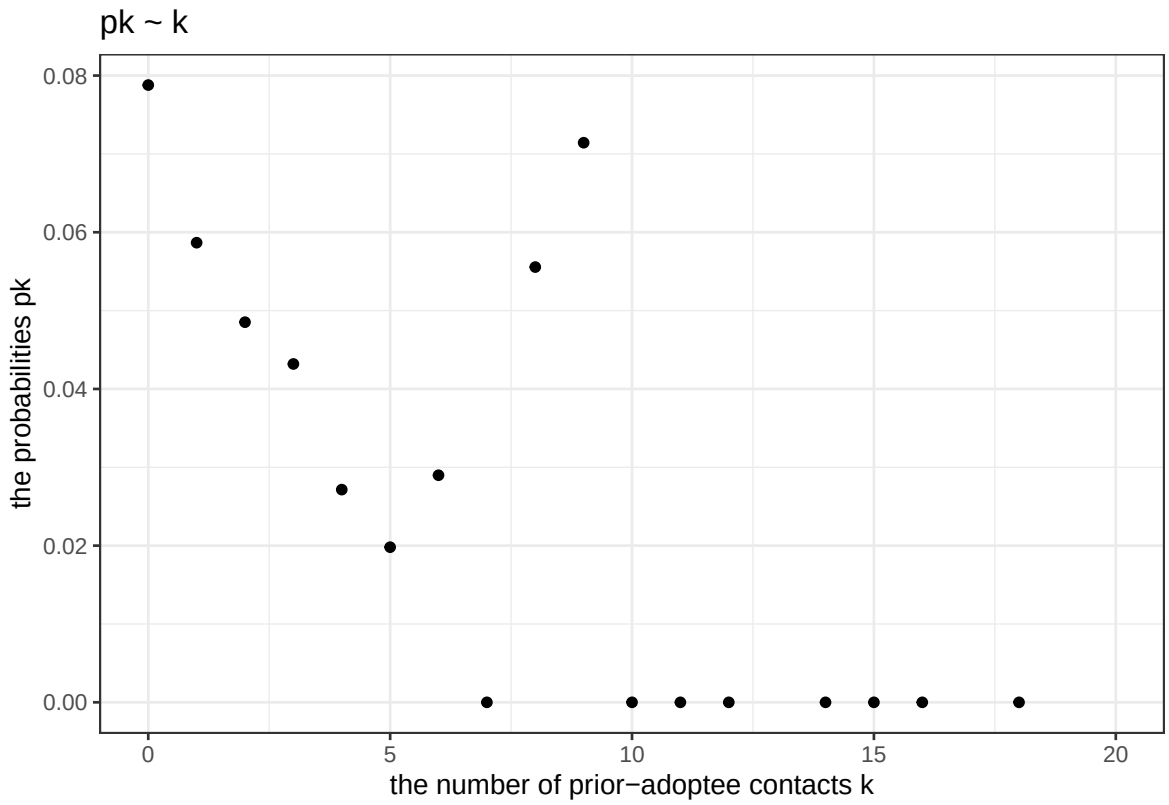
```
max(apply(ckm_network, 1,sum))
```

```
## [1] 20
```

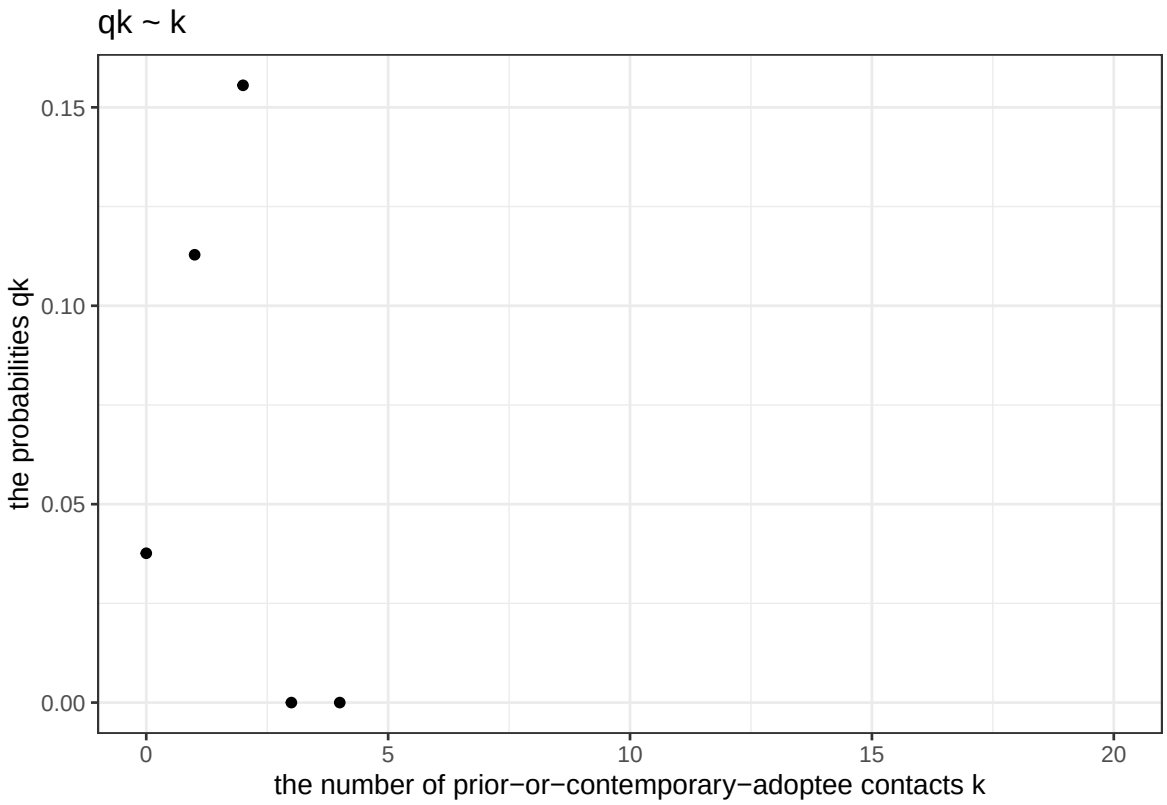
Ans: We can find that the maximum a doctor can contact is 20. So there are at most 21 valid values of k, which is from 0 to 20. In fact, in consideration of that there may exist k so that all the doctors whose number of doctor's contacts prescribing is not equal to k, which means that p_k or q_k is meaningless, the number of real valid k may be less than 21.

   b. Create a vector of estimated p_k probabilities, using the data frame from (2). Plot the probabilities against the number of prior-adoptee contacts k.
   c. Create a vector of estimated q_k probabilities, using the data frame from (2). Plot the probabilities against the number of prior-or-contemporary-adoptee contacts k.

```r
# create pk and qk
pk <- qk <- c()
for (k in 0:20) {
  obs <- doc.info %>% filter(num_strict_before == k)
  pk[k+1] <- sum(obs$begin) / dim(obs)[1]
  obs <- doc.info %>%
    filter(num_begin_before - num_strict_before == k)
  qk[k+1] <- sum(obs$begin) / dim(obs)[1]
}
# plot
prop <- data.frame(k = 0:20, pk, qk)
prop %>%
  ggplot(aes(x = k, y = pk)) +
  geom_point() +
  labs(x = "the number of prior-adoptee contacts k",
       y = "the probabilities pk",
       title = "pk ~ k") +
  theme_bw()
```




```r
prop %>%
  ggplot(aes(x = k, y = qk)) +
  geom_point() +
  labs(x = "the number of prior-or-contemporary-adoptee contacts k",
       y = "the probabilities qk",
       title = "qk ~ k") +
  theme_bw()
```



4. Because it only conditions on information from the previous month, p_k is a little easier to interpret than q_k. It is the probability per month that a doctor adopts tetracycline, if they have exactly k contacts who had already adopted tetracycline.

```
# clean the data
prop <- prop %>%
  filter(!is.na(pk)) %>%
  select(-qk)
```

   a. Suppose $p_k = a + bk$. This would mean that each friend who adopts the new drug increases the probability of adoption by an equal amount. Estimate this model by least squares, using the values you constructed in (3b). Report the parameter estimates.

```
mse1 <- function(parameters, x = prop$k, y = prop$pk) {
  a <- parameters[1]
  b <- parameters[2]
  y.estimate = a + b * x
  return(sum((y - y.estimate) ^ 2) / length(x))
}
par1 <- nlm(mse1, c(0, 0))
par1$estimate[1]
```

```
## [1] 0.05693243
```

```
par1$estimate[2]
```

```
## [1] -0.003799739
```

   b. Suppose $p_k = e^{a+bk}/(1 + e^{a+bk})$. Explain, in words, what this model would imply about the impact of adding one more adoptee friend on a given doctor's probability of adoption. (You can

suppose that $b > 0$, if that makes it easier.) Estimate the model by least squares, using the values you constructed in (3b).

```
mse2 <- function(parameters, x = prop$k, y = prop$pk) {
  a <- parameters[1]
  b <- parameters[2]
  y.estimate = exp(a + b * x) / (1 + exp(a + b * x))
  return(sum((y - y.estimate) ^ 2) / length(x))
}
par2 <- nlm(mse2, c(0, 0))
par2$estimate[1]
```

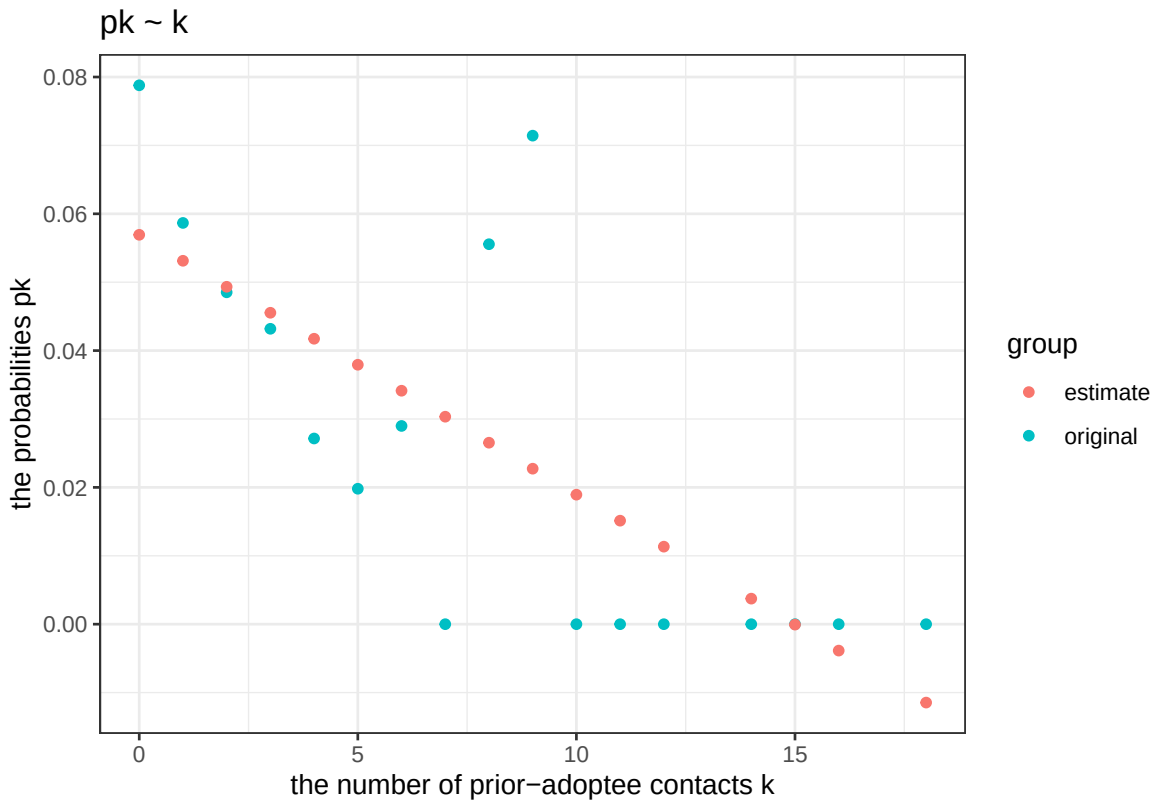
```
## [1] -2.565049
```

```
par2$estimate[2]
```

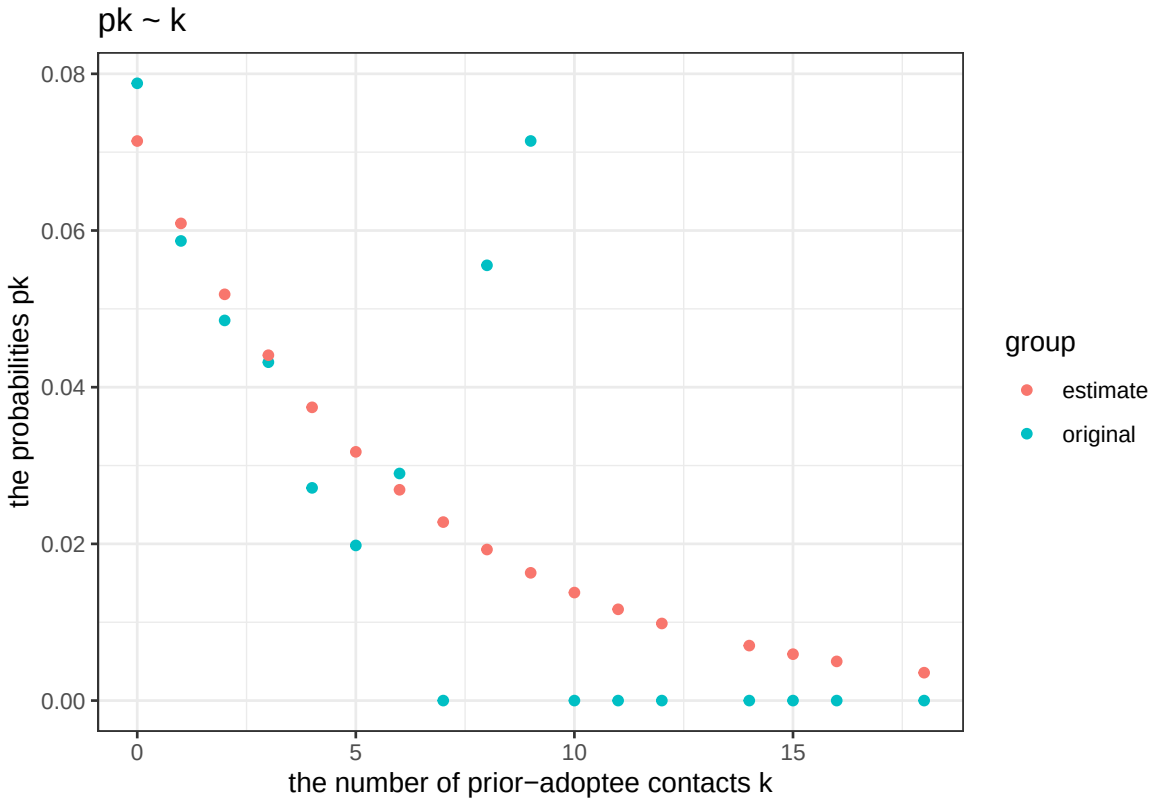
```
## [1] -0.1705144
```

c. Plot the values from (3b) along with the estimated curves from (4a) and (4b). (You should have one plot, with k on the horizontal axis, and probabilities on the vertical axis .) Which model do you prefer, and why?

```
# 4a
a <- par1$estimate[1]
b <- par1$estimate[2]
prop1 <- data.frame(k = rep(prop$k, time = 2),
                    pk = c(prop$pk, a+b*prop$k),
                    group = rep(c("original", "estimate"),
                                each=length(prop$k)))
prop1 %>%
  ggplot(aes(x = k, y = pk)) +
  geom_point(aes(colour = group)) +
  labs(x = "the number of prior-adoptee contacts k",
       y = "the probabilities pk",
       title = "pk ~ k") +
  theme_bw()
```




```
# 4b
a <- par2$estimate[1]
b <- par2$estimate[2]
prop2 <- data.frame(k = rep(prop$k, time = 2),
                    pk = c(prop$pk, exp(a+b*prop$k)/(1+exp(a+b*prop$k))),
                    group = rep(c("original", "estimate"),
                                each=length(prop$k)))
prop2 %>%
  ggplot(aes(x = k, y = pk)) +
  geom_point(aes(colour = group)) +
  labs(x = "the number of prior-adoptee contacts k",
       y = "the probabilities pk",
       title = "pk ~ k") +
  theme_bw()
```



We can choose the model with the less minimum square error.

```
par1$minimum
```

```
## [1] 0.0003583686
```

```
par2$minimum
```

```
## [1] 0.0003377583
```

So I prefer the model in 4b.

For quibblers, pedants, and idle hands itching for work to do: The p_k values from problem 3 aren't all equally precise, because they come from different numbers of observations. Also, if each doctor with k adoptee contacts is independently deciding whether or not to adopt with probability p_k, then the variance in the number of adoptees will depend on p_k. Say that the actual proportion who decide to adopt is $\hat{p}_k$. A little probability (exercise!) shows that in this situation, $\mathbb{E}[\hat{p}_k] = p_k$, but that $\mathrm{Var}[\hat{p}_k] = p_k(1-p_k)/n_k$, where n_k is the number of doctors in that situation. (We estimate probabilities more precisely when they're really extreme [close to 0 or 1], and/or we have lots of observations.) We can estimate that variance as $\hat{V}_k = \hat{p}_k(1-\hat{p}_k)/n_k$. Find the $\hat{V}_k$, and then re-do the estimation in (4a) and (4b) where the squared error for p_k is divided by $\hat{V}_k$. How much do the parameter estimates change? How much do the plotted curves in (4c) change?