homework4

Huang Xubin 3180102999

2021/7/15

1

```
# data about the month:
ckm_nodes <- read_csv("data\\ckm_nodes.csv")

# data about the contacts:
ckm_network <- read.table("data\\ckm_network.dat")

# a Boolean vector that shows whether the doctor got the adopted month
# recorded:
dct.bool.got_data <- !is.na(ckm_nodes$adoption_date)

# the vector of indexes of those doctors:
dct.idx.got_data <- which(dct.bool.got_data)

# cut off all useless rows in month data and rows and cols of those
# in contacts data:
ckm_nodes <- ckm_nodes[dct.idx.got_data,]
ckm_network <- ckm_network[dct.idx.got_data, dct.idx.got_data]
dim(ckm_nodes)
```

```
## [1] 125  13
```

```
dim(ckm_network)
```

```
## [1] 125 125
```

2

```
# number of rows after cleaning the data:
dct.n <- nrow(ckm_nodes)

# make a combination of doctor number and month:
dct.tbl.a <- ckm_nodes %>% mutate(begin = 1, doctor = 1 : nrow(ckm_nodes)) %>%
  dplyr::select(doctor, month = adoption_date, begin)

# develop a table of indexes of each obs:
dct.tbl <- data.frame(doctor = rep(1 : dct.n, each = 17),
                      month = rep(1 : 17, times = 125))

# do one left join operation to form a one-hot vector by hand:
```

```
dct.tbl <- dplyr::left_join(dct.tbl, dct.tbl.a, by = c("doctor", "month"))
dct.tbl$begin[is.na(dct.tbl$begin)] <- 0

# whether that doctor had begun before that month:
dct.tbl <- dct.tbl %>% group_by(doctor) %>%
  mutate(begin_before = (cumsum(begin) - begin)) %>%
  ungroup()

# table of the number of contacts of each doctor in each month that begins adopting:
invisible(dct.tbl.contacts_each_month <- data.frame(
  doctor = rep(1 : dct.n, times = rowSums(ckm_network)),
  month = ckm_nodes$adoption_date[
    unlist(apply(as.matrix(ckm_network), 1,
                 function(e){return(which(as.logical(e)))}))]) %>%
  group_by(doctor, month) %>% summarise(contacts.begin = n()))
```

```
## `summarise()` has grouped output by 'doctor'. You can override using the `.groups` argument.
```

```
# left join into the ultra table:
invisible(dct.tbl <- dplyr::left_join(dct.tbl, dct.tbl.contacts_each_month))
```

```
## Joining, by = c("doctor", "month")
```

```
dct.tbl$contacts.begin[is.na(dct.tbl$contacts.begin)] <- 0

# adding the two last columns into the ultra table:
dct.tbl <- dct.tbl %>% group_by(doctor) %>%
  mutate(contacts.begin_before = cumsum(contacts.begin) - contacts.begin,
         contacts.begin_in_or_before = cumsum(contacts.begin)) %>%
  ungroup() %>% dplyr::select(-contacts.begin)
head(dct.tbl)
```

```
## # A tibble: 6 x 6
##   doctor month begin begin_before contacts.begin_before contacts.begin_in_or_be~
##    <int> <dbl> <dbl>        <dbl>                 <dbl>                    <dbl>
## 1      1     1     1            0                     0                        1
## 2      1     2     0            1                     1                        1
## 3      1     3     0            1                     1                        2
## 4      1     4     0            1                     2                        3
## 5      1     5     0            1                     3                        3
## 6      1     6     0            1                     3                        3
```

```
tail(dct.tbl)
```

```
## # A tibble: 6 x 6
##   doctor month begin begin_before contacts.begin_before contacts.begin_in_or_be~
##    <int> <dbl> <dbl>        <dbl>                 <dbl>                    <dbl>
## 1    125    12     0            0                     0                        0
## 2    125    13     0            0                     0                        0
## 3    125    14     0            0                     0                        0
## 4    125    15     0            0                     0                        0
## 5    125    16     1            0                     0                        0
## 6    125    17     0            1                     0                        0
```

where 6 columns of which 2 columns are for **identifying each observation** and 4 columns left are just for the data required, and 2125 rows because **125** (number of doctor) **plus 17** (max adopted time) **equals 2125**.

3

a

The reason why there should be no more than 21 entries of k is just there is no one who got more than 20 contacts, and by counting up all cases plus the 0 case gives the number of 21.

```
# there is no one who got more than 20 contacts:
max(rowSums(ckm_network))
```

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

b

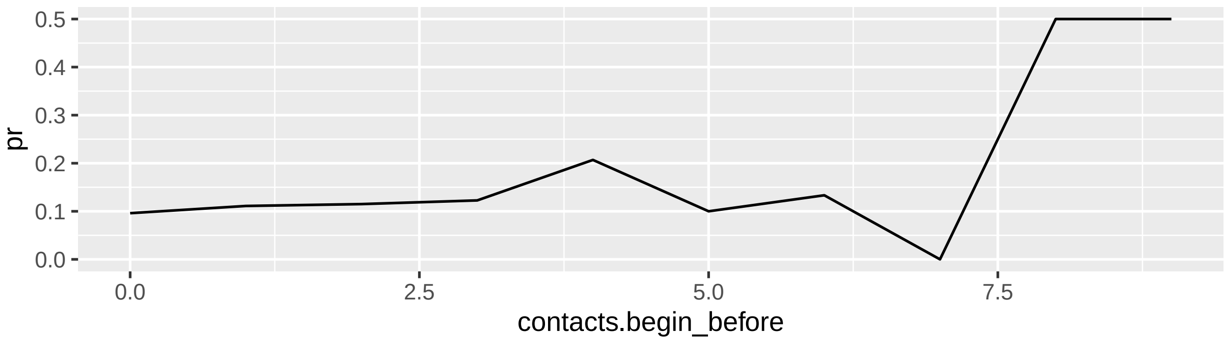
```
# generating p_k:
dct.p.dmnt <- dct.tbl %>% filter(begin_before == 0) %>%
  group_by(contacts.begin_before) %>% summarise(dominator = n())
dct.p.nmrt <- dct.tbl %>% filter(begin == 1) %>%
  group_by(contacts.begin_before) %>% summarise(numerator = n())
invisible(dct.p <- dplyr::full_join(dct.p.dmnt, dct.p.nmrt) %>%
  mutate(pr = numerator / dominator))
```

```
## Joining, by = "contacts.begin_before"
```

```
dct.p[is.na(dct.p)] <- 0
dct.p
```

```
## # A tibble: 10 x 4
##    contacts.begin_before dominator numerator     pr
##                    <dbl>     <int>     <int>  <dbl>
##  1                     0       406        39 0.0961
##  2                     1       198        22 0.111
##  3                     2       200        23 0.115
##  4                     3       106        13 0.123
##  5                     4        29         6 0.207
##  6                     5        20         2 0.1
##  7                     6        15         2 0.133
##  8                     7         3         0 0
##  9                     8         2         1 0.5
## 10                     9         2         1 0.5
```

```
# plot:
dct.p %>% ggplot() + geom_line(aes(x = contacts.begin_before, y = pr))
```



c

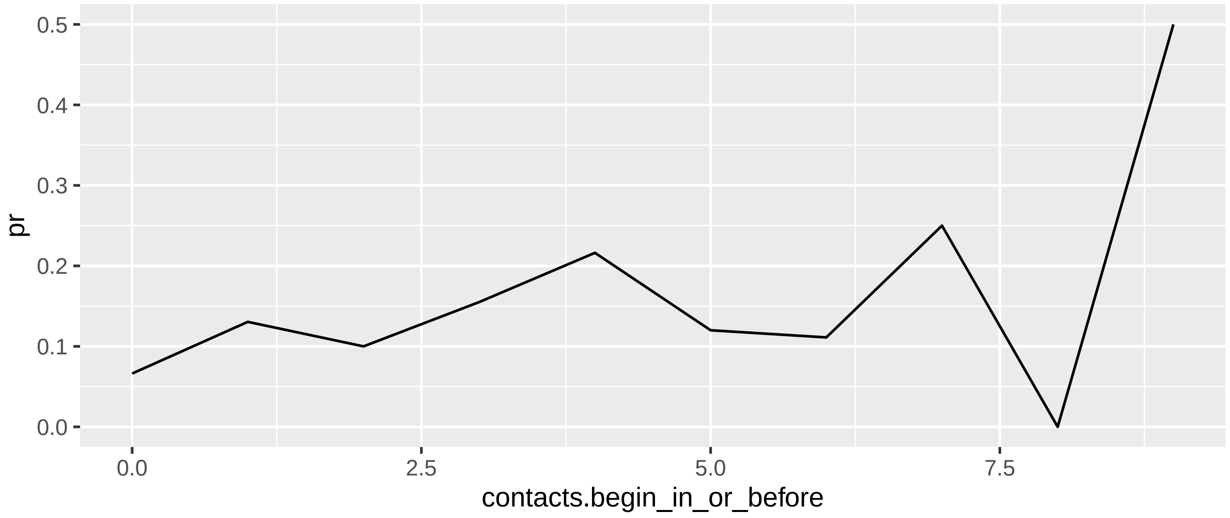
```
# generating q_k:
dct.q.dmnt <- dct.tbl %>% filter(begin_before == 0) %>%
  group_by(contacts.begin_in_or_before) %>% summarise(dominator = n())
dct.q.nmrt <- dct.tbl %>% filter(begin == 1) %>%
  group_by(contacts.begin_in_or_before) %>% summarise(numerator = n())
invisible(dct.q <- dplyr::full_join(dct.q.dmnt, dct.q.nmrt) %>%
  mutate(pr = numerator / dominator))
```

```
## Joining, by = "contacts.begin_in_or_before"
```

```
dct.q[is.na(dct.q)] <- 0
dct.q
```

```
## # A tibble: 10 x 4
##    contacts.begin_in_or_before dominator numerator     pr
##                          <dbl>     <int>     <int>  <dbl>
##  1                           0       302        20 0.0662
##  2                           1       230        30 0.130
##  3                           2       230        23 0.1
##  4                           3       129        20 0.155
##  5                           4        37         8 0.216
##  6                           5        25         3 0.12
##  7                           6        18         2 0.111
##  8                           7         4         1 0.25
##  9                           8         2         0 0
## 10                           9         4         2 0.5
```

```
# plot:
dct.q %>% ggplot() + geom_line(aes(x = contacts.begin_in_or_before, y = pr))
```



4

a

```
# estimation via least squares:
f1 <- function(a, b, X = dct.p$contacts.begin_before){
  return(a + b * X)
}
f2 <- function(a, b, X = dct.p$contacts.begin_before){
  return(exp(a + b * X) / (1 + exp(a + b * X)))
}
ls <- function(params, f, Y = dct.p$pr){
  return(mean((Y - f(params[1], params[2]))^2))
}
p1 <- c(0.03284105 , 0.03459169)
result_1 <- nlm(ls, p1, f1)$estimate
result_1
```

```
## [1] 0.03284105 0.03459169
```

where the parameters estimated are $(a,\ b) = (0.03284105,\ 0.03459169)$.

b

Suppose $b > 0$, then according to the derivative of the model w.r.t. k, the image of this model is at first a steep curve right up to the top (probability of 1) and soon becomes a straight (horizontal) line heading to the right end, which indicates the impact of adding 1 contacts is at first a factor to consider and soon being nothing to consider about.

```
p2 <- c(-3.6872181, 0.3834268)
result_2 <- nlm(ls, p2, f2)$estimate
result_2
```

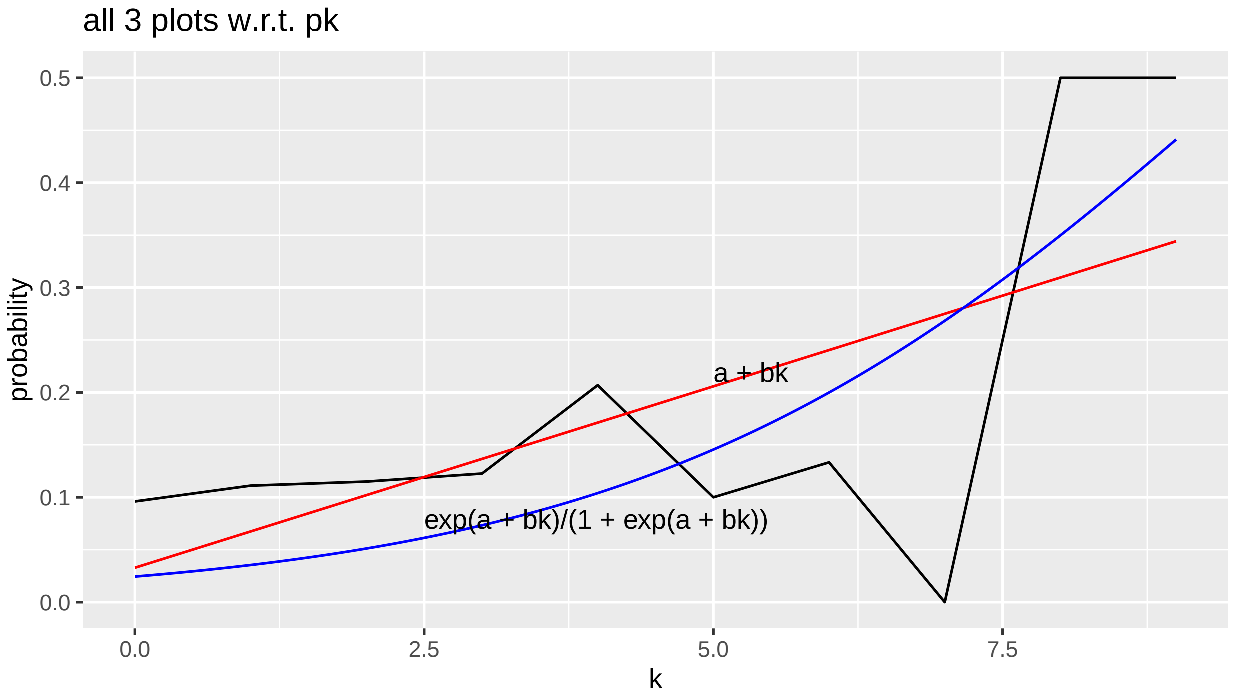
```
## [1] -3.6872181  0.3834268
```

where the parameters estimated are $(a,\ b) = (-3.6872181,\ 0.3834268)$.

c

```
interval.left <- min(dct.p$contacts.begin_before)
interval.right <- max(dct.p$contacts.begin_before)
X <- seq(interval.left, interval.right, (interval.right - interval.left) / 10000)

# plot all the 3 plots:
ggplot() + geom_line(aes(x = dct.p$contacts.begin_before, y = dct.p$pr)) +
  geom_line(aes(x = X, y = f1(result_1[1], result_1[2], X)), colour = 'red') +
  geom_line(aes(x = X, y = f2(result_2[1], result_2[2], X)), colour = 'blue') +
  labs(x = "k", y = "probability", title = "all 3 plots w.r.t. pk") +
  geom_text(data = data.frame(x = 5, y = 0.21),
            aes(x, y, label = "a + bk"), hjust = 0, vjust = 0) +
  geom_text(data = data.frame(x = 2.5, y = 0.07),
            aes(x, y, label = "exp(a + bk)/(1 + exp(a + bk))"), hjust = 0, vjust = 0)
```



Clearly the exponential model fits more of the origin data according to the image above.