Homework 4 Solutions

3190300985 LUIS LUZERN YUVEN

1

```r
library(tidyverse)
ckm_nodes <- read_csv('data/ckm_nodes.csv')
noinfor <- which(is.na(ckm_nodes$adoption_date))
ckm_nodes <- ckm_nodes[-noinfor, ]
ckm_network <- read.table('data/ckm_network.dat')
ckm_network <- ckm_network[-noinfor, -noinfor]
```

2

```r
mat <- matrix(rep(1:17, each = 125))
record <- data.frame(mat, rep(1:125, times = 17),
                     rep(ckm_nodes$adoption_date, times = 17))
record[,3] <- ifelse(record[1:2125,1] == record[1:2125,3], TRUE, FALSE)
record <- cbind(record,
                ifelse(rep(ckm_nodes$adoption_date, times = 17) < record[,1],
                       TRUE, FALSE))
record <- cbind(record, vector(length = 2125), vector(length = 2125))
k = 1
for(i in 1 : 17){
  for(j in 1 : 125){
    ctc <- which(ckm_network[j,] == 1)
    if(length(ctc) != 0){
       for(p in 1 : length(ctc)){
        record[k,5] <- record[k,5] +
          ifelse(ckm_nodes$adoption_date[ctc[p]] < i, 1, 0)
        record[k,6] <- record[k,6] +
          ifelse(ckm_nodes$adoption_date[ctc[p]] <= i, 1, 0)
      }
    }
    k = k + 1
  }
}
colnames(record) <- c("Month","Doctor","thisMonth","before",
                      "ctc_strictly_before","ctc_before")
```

The data frame must have 2125 rows, since there are 17 months and 125 doctors. It must also have 6 columns, where the columns represent doctors, months, 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.

3

(a)

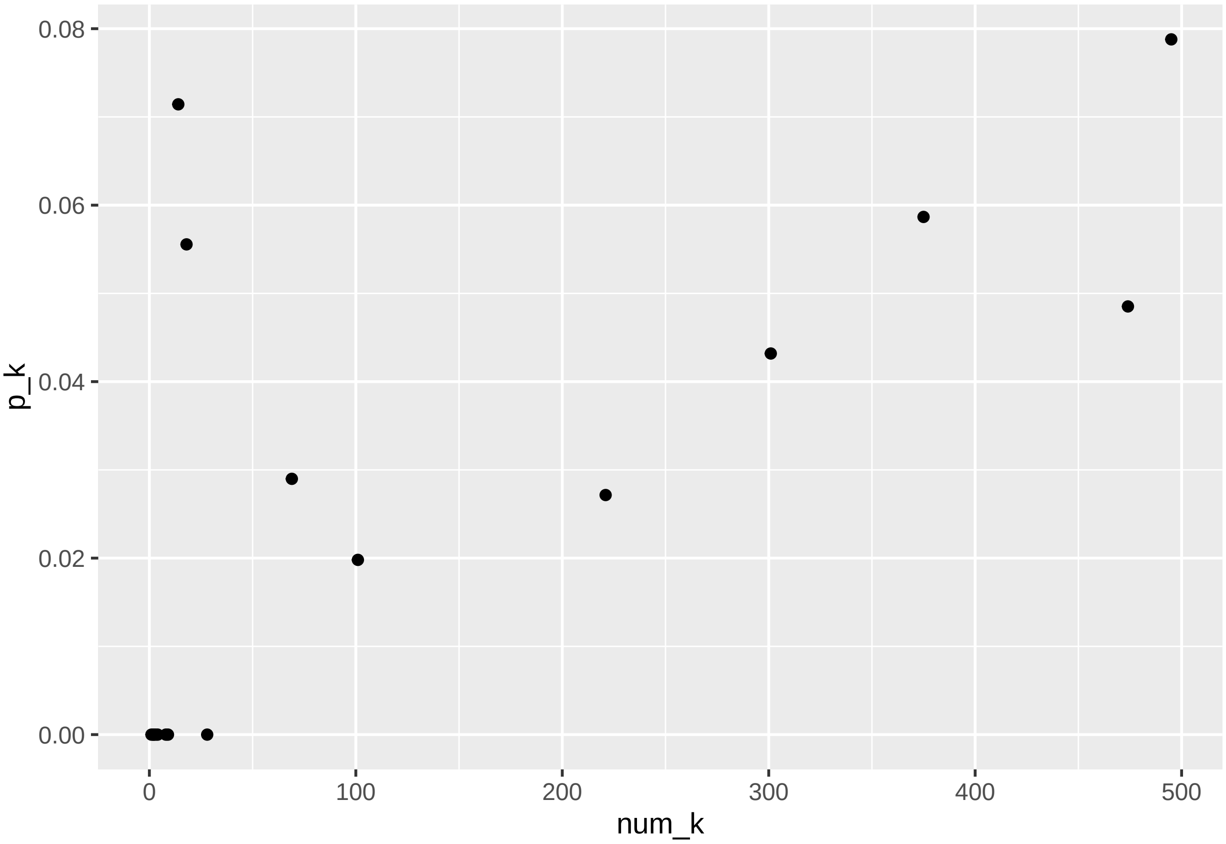
```
max(apply(ckm_network,1,sum))
```

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

The maximum number of contacts from a doctor is 20 contacts, so k can only range from 0 to 20, thus 21 values of k.

(b)

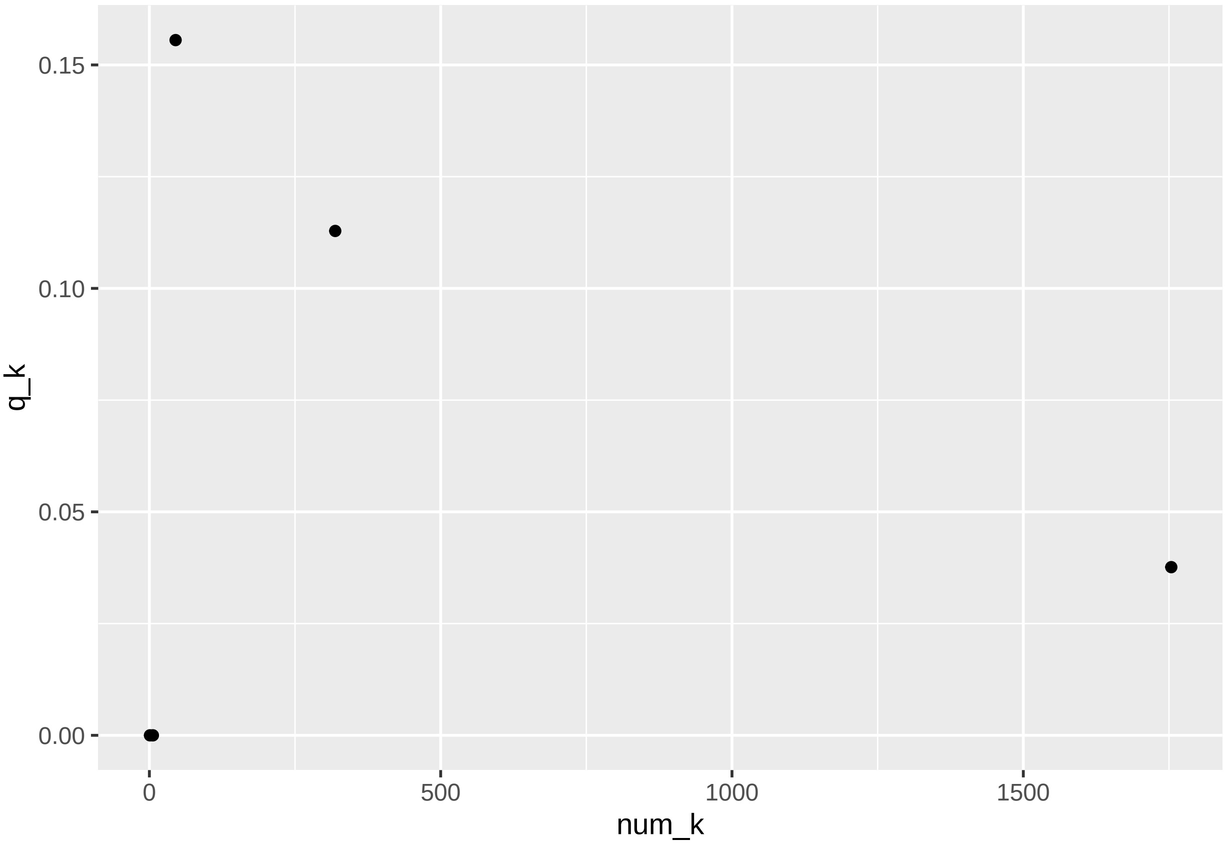
```
p_k <- vector(mode = "numeric", length = 21)
num_k <- p_k
for(k in 0 : 20){
  ind <- record$ctc_strictly_before == k
  num_k[k+1] <- sum(ind)
  if(num_k[k+1] == 0){
    p_k[k+1] <- NA
    next
  }
  record_k <- record[ind,]
  total <- sum(record_k$thisMonth == TRUE)
  p_k[k+1] <- total / num_k[k+1]
}
ggplot() + geom_point(aes(x = num_k, y = p_k)) + labs(x = "num_k", y = "p_k")
```



(c)

```
q_k <- vector(mode = "numeric", length = 21)
num_k <- q_k
for(k in 0 : 20){
  ind <- (record$ctc_before - record$ctc_strictly_before) == k
  num_k[k+1] <- sum(ind)
  if(num_k[k+1] == 0){
    q_k[k+1] <- NA
    next
  }
  record_k <- record[ind,]
  total <- sum(record_k$thisMonth == TRUE)
  q_k[k+1] <- total / num_k[k+1]
}
ggplot() + geom_point(aes(x = num_k, y = q_k)) + labs(x = "num_k", y = "q_k")
```

```
## Warning: Removed 16 rows containing missing values (geom_point).
```



4

(a)

```
record_p <- data.frame(k = 0:20, p = p_k)
model1 <- lm(p~k, data = record_p)
summary(model1)
```

```
##
## Call:
## lm(formula = p ~ k, data = record_p)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.030334 -0.014584 -0.002344  0.005534  0.048694
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0569324  0.0090507   6.290 1.45e-05 ***
## k           -0.0037997  0.0009184  -4.137 0.000877 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02015 on 15 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.533,  Adjusted R-squared:  0.5018
## F-statistic: 17.12 on 1 and 15 DF,  p-value: 0.0008773
```

(b)

This model is a logistic curve. Suppose b > 0. The growth is exponential at the beginning but slowly becomes linear due to saturation and will finally stop at maturity.

```
logistic <- function(k, a, b) return (exp(a+b*k)/(1 + exp(a+b*k)))
model2 <- nls(p ~ logistic(k,a,b), data = record_p, start = list(a=0, b=-0.2))
summary(model2)
```

```
##
## Formula: p ~ logistic(k, a, b)
##
## Parameters:
##   Estimate Std. Error t value Pr(>|t|)
## a -2.56508    0.20610 -12.446 2.62e-09 ***
## b -0.17051    0.05371  -3.174  0.00628 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01957 on 15 degrees of freedom
##
## Number of iterations to convergence: 6
## Achieved convergence tolerance: 1.449e-07
##   (4 observations deleted due to missingness)
```

```
model3 <- lm(p.log ~ k, record_p %>%
               mutate(p.log = ifelse(p==0, log(0.0001/(1-0.0001)), log(p/(1-p)))))
```

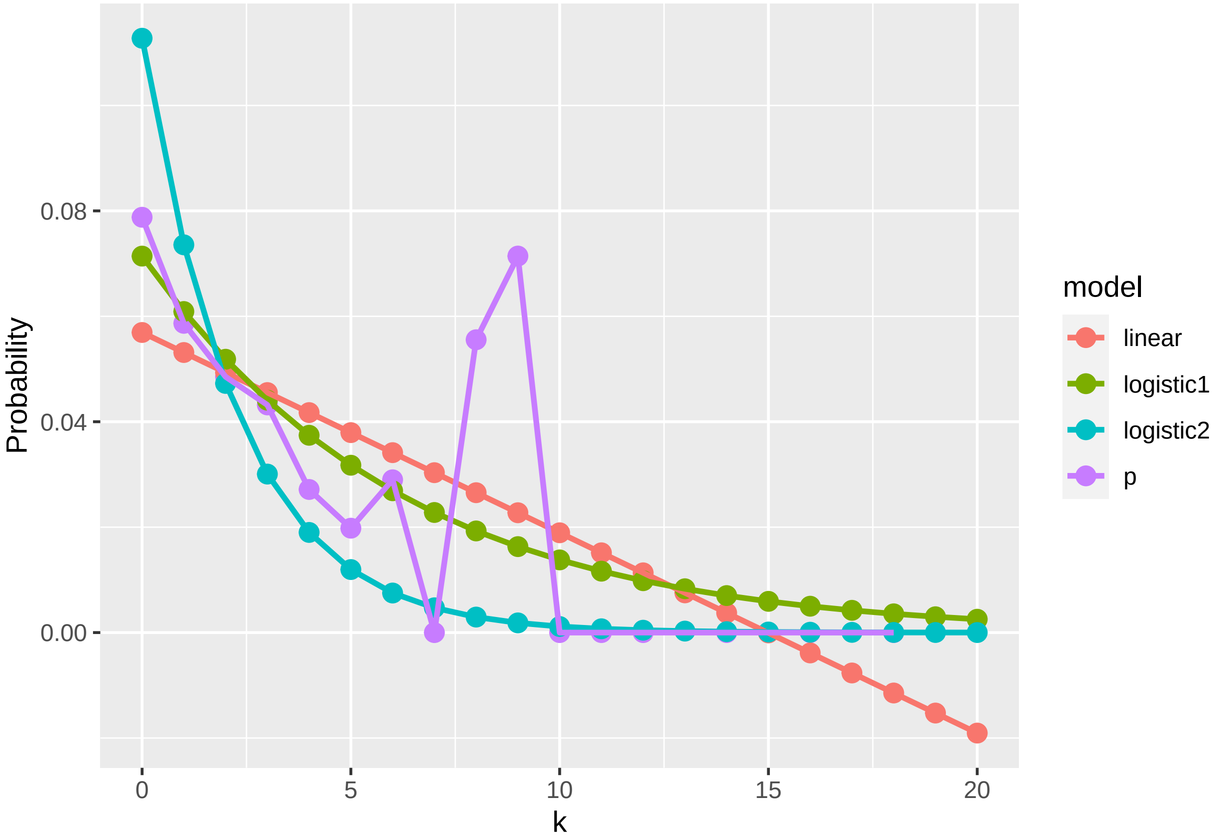
(c)

```
m1 <- predict(model1, newdata = data.frame(k = c(0:20)))
m2 <- predict(model2, newdata = data.frame(k = c(0:20)))
y <- predict(model3, newdata = data.frame(k = c(0:20)))
m3 <- exp(y) / (1 + exp(y))
record_p <- record_p %>% mutate(linear = m1, logistic1 = m2, logistic2 = m3)
```

```
k <- c(0:20)
tidy_record <- record_p %>% gather(model,res,-k) %>% na.omit()
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```
tidy_record %>% ggplot() + geom_point(aes(x = k, y = res, color = model),
                                      size = 3) +
  geom_line(aes(x = k, y = res, color = model), size = 1) +
  labs(y = "Probability")
```



From the result, we can conclude that the linear model is unsuitable to predict the probabilities. The model `logistic1` is probably better than `logistic2`, since the latter is prone to overfitting at the tail, and the former gives better prediction at the head.