Homework 5 – Due April 7

Name: Navya Mittal netID: navya_0215 Collaborated with: No one

Your homework **must be submitted in Word or PDF format, created by calling "Knit Word" or "Knit PDF" from RStudio on your R Markdown document. Submission in other formats may receive a grade of 0**. Your responses must be supported by both textual explanations and the code you generate to produce your result. Note that all R code used to produce your results must be shown in your knitted file.

We continue examining the diffusion of tetracycline adoption 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`.

1. Clean the data to eliminate doctors for whom we have no adoption-date information, as in Lab 7. Only use this cleaned data for the rest of the assignment.

```
data = read.csv("ckm_nodes.csv")
pos = which(!is.na(data$adoption_date))
cleaned_nodes = data[pos,]
```

2. Create a new data frame which records, for every doctor and every month, whether that doctor began prescribing tetracycline that month (as a boolean variable), whether they had adopted tetracycline strictly before that month (as a boolean variable), 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 data frame should have 6 columns, and 2125 rows. (Try not to use any loops. But points will *not* be deducted if you use loops.)

Specifically, the data frame should contain the following columns:

a. `doctor`: Doctor ID (the row indices of the original data)

b. `month`: Month

c. `begin`: whether that doctor began prescribing tetracycline that month

d. `before`: whether they had adopted tetracycline before that month

e. `contacts_before`: the number of their contacts who began prescribing strictly before that month

f. `contacts_in_or_before`: the number of their contacts who began prescribing in that month or earlier

Display the rows corresponding to the second doctor.

```
#df <- data.frame(df[,-1], row.names = df[,1])
data2 <- read.table("ckm_network.dat")
#data2
as.numeric(row.names(cleaned_nodes))
```

```
##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##  [19]  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##  [37]  37  38  39  58  70  72  73  74  75  76  77  78  79  80  81  82  91  92
##  [55]  93  94  95  96  97  98 105 108 119 121 122 123 124 125 126 127 128 129
##  [73] 130 131 132 133 134 135 136 137 151 152 153 154 155 156 168 169 170 171
##  [91] 172 173 174 175 176 177 178 179 180 181 182 195 196 197 198 199 200 212
## [109] 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 229 241
```

```
#cleaned_nodes]
doctor <- rep(as.numeric(rownames(cleaned_nodes)), each=17)
month <- rep(1:17, 125)
adopt <- rep(cleaned_nodes$adoption_date, each=17)
begin <- month == adopt
before <- month > adopt
inor <-  month >= adopt
inor <- adopt <= month
network = read.table("ckm_network.dat")
dim(network) #therefore square
```

```
## [1] 246 246
```

```
clean_network = network[pos,pos]

numcon <- function(i, x, dat, n_month=17, n_doctor=125){
  contacts <- matrix(x, nr=n_month, nc=n_doctor)
  index <- which(dat[i,]==1)
  if (length(index)>1){
    return(apply(contacts[,index, drop=F],1,sum))
  } else {
    return(rep(0, nrow(contacts)))
  }
}
contacts_before <- as.vector(sapply(1:125, numcon, x=before, dat=clean_network))
contacts_in_or_before <- as.vector(sapply(1:125, numcon, x= inor, dat=clean_network))
output <- data.frame(doctor, month, begin, before, contacts_before, contacts_in_or_before)
output[output$doctor==2,]
```

```
##    doctor month begin before contacts_before contacts_in_or_before
## 18      2     1 FALSE  FALSE               0                     0
## 19      2     2 FALSE  FALSE               0                     0
## 20      2     3 FALSE  FALSE               0                     1
## 21      2     4 FALSE  FALSE               1                     2
## 22      2     5 FALSE  FALSE               2                     2
## 23      2     6 FALSE  FALSE               2                     2
## 24      2     7 FALSE  FALSE               2                     2
## 25      2     8 FALSE  FALSE               2                     2
## 26      2     9 FALSE  FALSE               2                     2
## 27      2    10 FALSE  FALSE               2                     2
## 28      2    11 FALSE  FALSE               2                     2
## 29      2    12  TRUE  FALSE               2                     2
## 30      2    13 FALSE   TRUE               2                     2
## 31      2    14 FALSE   TRUE               2                     2
## 32      2    15 FALSE   TRUE               2                     2
## 33      2    16 FALSE   TRUE               2                     2
## 34      2    17 FALSE   TRUE               2                     2
```

We have 6 columns and 2125 rows as we repeat the experimennt for all 125 doctors for each month which gives us 17*2125 rows and since we have 6 vairables, we end up with 6 columns. 3. Let

p_k = Pr(Doctor starts prescribing tetracycline this month |
Doctor did not previously prescribe, and number of doctor's contacts prescribing before this month is k)

```
**When computing $p_k$ it is important to note that the number of doctors who *could* start prescribing
```

We suppose the p_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 directly from the data.

```
c <- apply(clean_network, MARGIN=2, FUN = sum) #2 for columns
min(c)
```

```
## [1] 0
```

```
max(c)
```

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

The range of values for k are determined by the minimum and maximum number of contacts in `ckm_network`. We see that the minimum number of contacts is 0 and the maximum is 20, therefore we have 21 values of k.

Since the probabilities are the same for all months, we can pool the data from individual months together by considering $A = \bigcup_{j=1}^{17} A_j$ where

$A_j = \{$At month j, a doctor did not previously prescribe, and number of doctor's contacts prescribing before this month is

which corresponds to a subset of rows in the dataframe created in Q2. Note that each row corresponds to an observation of a particular doctor in a specific month.

The estimated p_k is calculated as

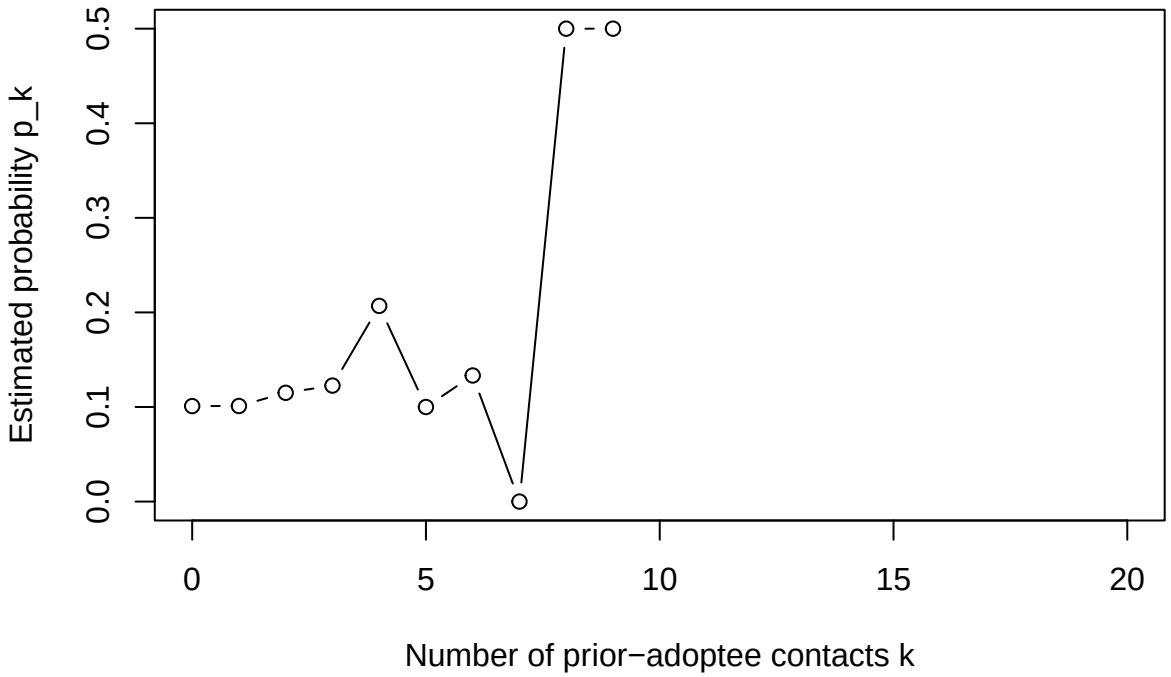
$$\frac{\text{Number of rows that correspond to a doctor that began prescribing that month and are in } A}{\text{Number of rows belonging to } A}$$

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. (A number of estimated p_k are NaN, for large k.)

```
pk_function <- function(i){
  mean(output$begin[(!output$before) & (output$contacts_before == i)])
}

# Create vector of estimated p_k probabilities using sapply
p_k <- sapply(0:20, pk_function)

# Plot probabilities against k
plot(0:20, p_k, type="b", xlab="Number of prior-adoptee contacts k", ylab="Estimated probability p_k")
```



4. Suppose $p_k = a + bk$. This would mean that each contact who adopts the new drug increases the probability of adoption by an equal amount on average. Estimate this model by least squares (i.e., linear regression), using the values you constructed in (3c). Report the estimates of a and b (i.e., the slope and intercept of a linear model with predictor k (from part 3a), and response p_k.)

```
df <- data.frame(k = 0:20, p_k = p_k)
model <- lm(p_k ~ k, data = df)
summary(model)
```

```
##
## Call:
## lm(formula = p_k ~ k, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.27487 -0.08233  0.02430  0.06109  0.19038
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03161    0.08472   0.373   0.7188
## k            0.03475    0.01587   2.190   0.0599 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1441 on 8 degrees of freedom
##   (11 observations deleted due to missingness)
## Multiple R-squared:  0.3748, Adjusted R-squared:  0.2966
## F-statistic: 4.795 on 1 and 8 DF,  p-value: 0.05994
```

```
coef(model)
```

```
## (Intercept)           k
##  0.03160572  0.03475199
```