STAT 206 Lab 5

Xin Feng(Vanessa)

11/1/2019

Due Monday, November 4, 5:00 PM

General instructions for labs: Labs must be completed as a pdf file. Give the commands to answer each question in its own code block, which will also produce plots that will be automatically embedded in the output file. Each answer must be supported by written statements as well as any code used.

Agenda: Fitting models by optimization; transforming data from one representation to another; handling missing data

Many theories of the diffusion of innovations (new technologies, practices, beliefs, etc.) suggest that the fraction of members of a group who have adopted the innovation by time t, $p(t)$, should follow a logistic curve or logistic function,

$$p(t) = \frac{e^{b(t-t_0)}}{1+e^{b(t-t_0)}}.$$

We will look at a classic data set on the diffusion of innovations, which is supposed to show such a curve. It concerns a survey of 246 doctors in four towns in Illinois in the early 1950s, and when they began prescribing (adopted) a then-new antibiotic, tetracycline, and how they became convinced that they should do so (from medical journals, from colleagues, etc.).

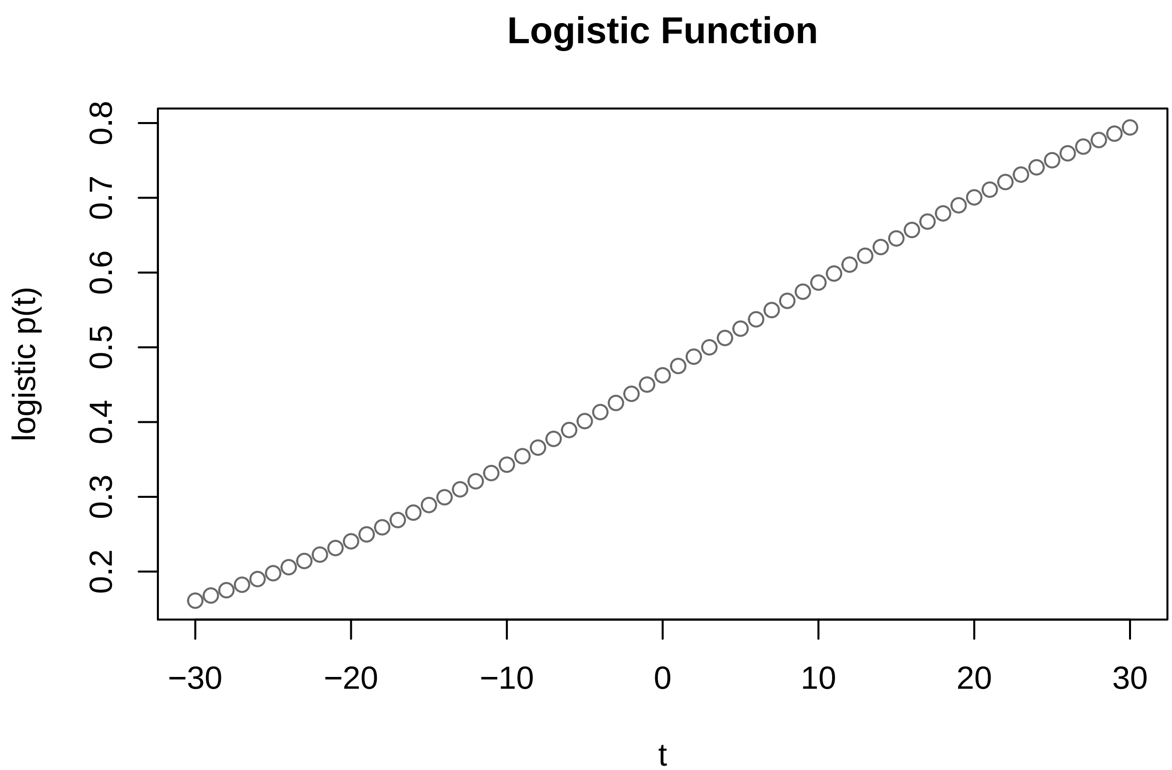
Load the file [http://faculty.ucr.edu/~jflegal/206/ckm_nodes.csv]. Each row is a doctor. The column adoption date shows how many months, after it became available, each doctor began prescribing tetracycline. Doctors who had not done so by the end of the survey, i.e., after month 17, have a value of `Inf` in this column. This information is not available (`NA`) for some doctors. There are twelve other variables which may also be `NA`.

```
ckm_nodes <- read.csv("http://faculty.ucr.edu/~jflegal/206/ckm_nodes.csv")
# View(ckm_nodes)
```

1. The Model.

   a. Write a function, `logistic`, which calculates the logistic function. It should take two arguments, `t` and `theta`. The `theta` argument should be a vector of length two, the first component being the parameter b and the second component being t_0. Your function may not use any loops. Plot the curve of the logistic function with $b = 0.05$, $t_0 = 3$, from $t = -30$ to $t = 30$.

```
theta_0 <- c(0.05,3)
logistic <- function(t, theta=theta_0){
    p_t <- exp(theta[1]*(t-theta[2]))/(1+exp(theta[1]*(t-theta[2])))
    return(p_t)
}
t_i <- seq(-30, 30, 1)
plot(t_i, logistic(t_i), main="Logistic Function", xlab="t", ylab="logistic p(t)"
     , col="dimgrey")
```




```
b. Explain why $p(t_0)=0.5$, no matter what $b$ is. Use this to check your logistic
function at multiple combinations of $b$ and $t_0$.
```

```r
# When t=t_0, then t-t_0=0
# Logistic function: p(t_0)=e^(0)/(1+e^0)=1/2=0.5
cat("Check t=t_0, p_t=logistic(t_0)=",logistic(theta_0[2], theta_0),"\n")
```

```
## Check t=t_0, p_t=logistic(t_0)= 0.5
```

```r
# Other tests
theta_0 <- c(0.02, 2)
logistic(theta_0[2], theta_0)
```

```
## [1] 0.5
```

```r
theta_0 <- c(0.08, 5)
logistic(theta_0[2], theta_0)
```

```
## [1] 0.5
```

```
c. Explain why the slope of $p(t)$ at $t=t_0$ is $b/4$. (Hint: calculus.) Use this to
check your `logistic` function at multiple combinations of $b$ and $t_0$.
```

```r
# Derivative at the point t=t_0:
# p'(t) = (A'B-AB')/B^2 = b*exp(b*(t-t_0))/(1+exp(b*(t-t_0)))^2
# When t=t_0, then p'(t_0)=b/4
```

```r
Deriv_logis <- function(t, theta=theta_0){
    p_t_deriv <- theta[1]*exp(theta[1]*(t-theta[2]))/
        (1+exp(theta[1]*(t-theta[2])))^2
    return(p_t_deriv)
}
# Checking
theta_0 <- c(0.05, 3)
cat("Slope of p(t_0): ",Deriv_logis(theta_0[2], theta_0),",
    b/4=", theta_0[1]/4, "\n")
```

```
## Slope of p(t_0):  0.0125 ,
##     b/4= 0.0125
```

```r
theta_0 <- c(0.02, 8)
cat("Slope of p(t_0): ",Deriv_logis(theta_0[2], theta_0),",
    and b/4=", theta_0[1]/4, "\n")
```

```
## Slope of p(t_0):  0.005 ,
##     and b/4= 0.005
```

```r
theta_0 <- c(0.08, 5)
cat("Slope of p(t_0): ",Deriv_logis(theta_0[2], theta_0),",
    and b/4=", theta_0[1]/4, "\n")
```

```
## Slope of p(t_0):  0.02 ,
##     and b/4= 0.02
```

2. The Data.

    a. How many doctors in the survey had adopted tetracycline by month 5? Hint: Use `na.omit` carefully.

```r
num_doc <- sum(na.omit(ckm_nodes$adoption_date)<=5)
cat("There are ", num_doc, " doctors had adopted tetracycline by month 5.\n")
```

```
## There are  51  doctors had adopted tetracycline by month 5.
```

```
b. What proportion of doctors, for whom adoption dates are available, had adopted
tetracycline by month 5?
```

```r
# proportaion = num_doctors(adoption_date<=5) / num_doctors(adoption dates available)
prop_doc <- num_doc/(sum(na.omit(ckm_nodes$adoption_date)>0)-
                         length(which(is.infinite(ckm_nodes$adoption_date))))
cat("The proportion in the question is ", prop_doc, "\n")
```

```
## The proportion in the question is  0.4678899
```

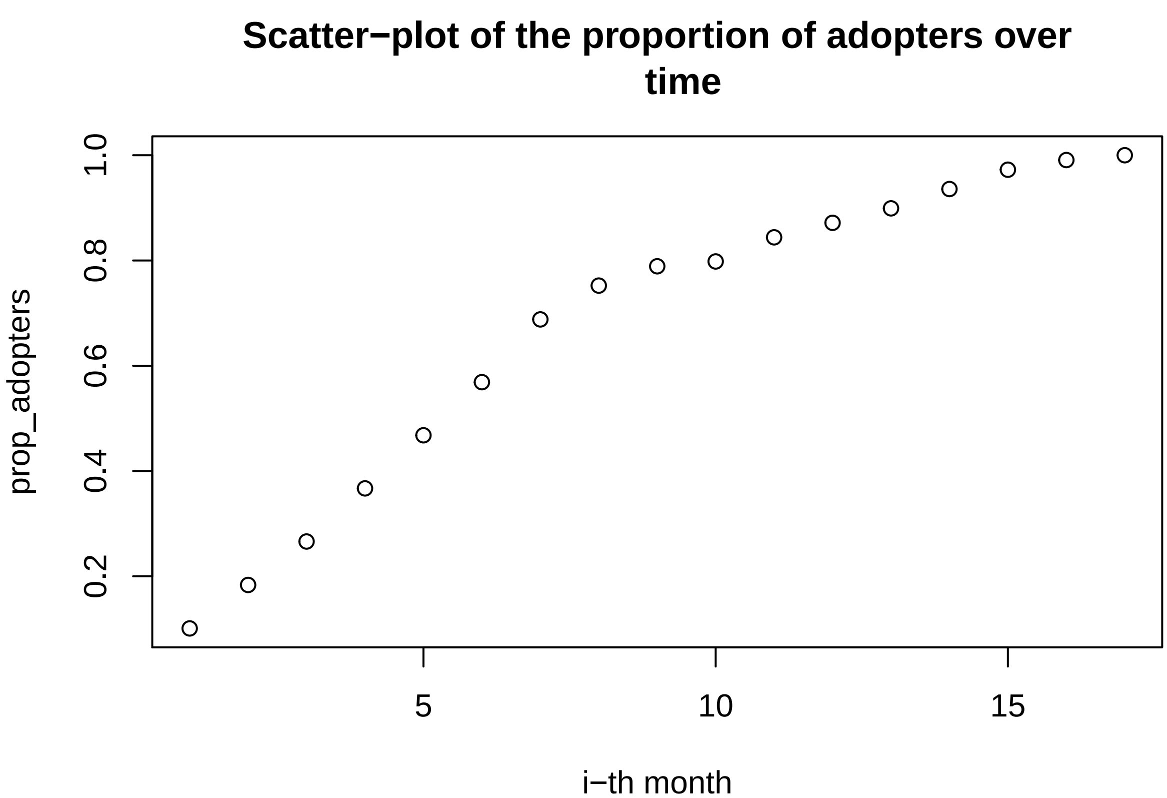
```
c. Create a vector, `prop_adopters`, storing the proportion of doctors who have adopted by
each month. (Be careful about `Inf` and `NA`.)
```

```r
# Number of available dates.
num_sum <- sum(na.omit(ckm_nodes$adoption_date)>0)-
    length(which(is.infinite(ckm_nodes$adoption_date)))
# Except the Inf condition, each month's proportion
max_date <- 17
prop_adopters <- rep(0, max_date)
for(i in 1:max_date){
    prop_adopters[i] <- sum(na.omit(ckm_nodes$adoption_date)==i)/num_sum
}
prop_adopters
```

```
##  [1] 0.100917431 0.082568807 0.082568807 0.100917431 0.100917431
##  [6] 0.100917431 0.119266055 0.064220183 0.036697248 0.009174312
## [11] 0.045871560 0.027522936 0.027522936 0.036697248 0.036697248
## [16] 0.018348624 0.009174312
```

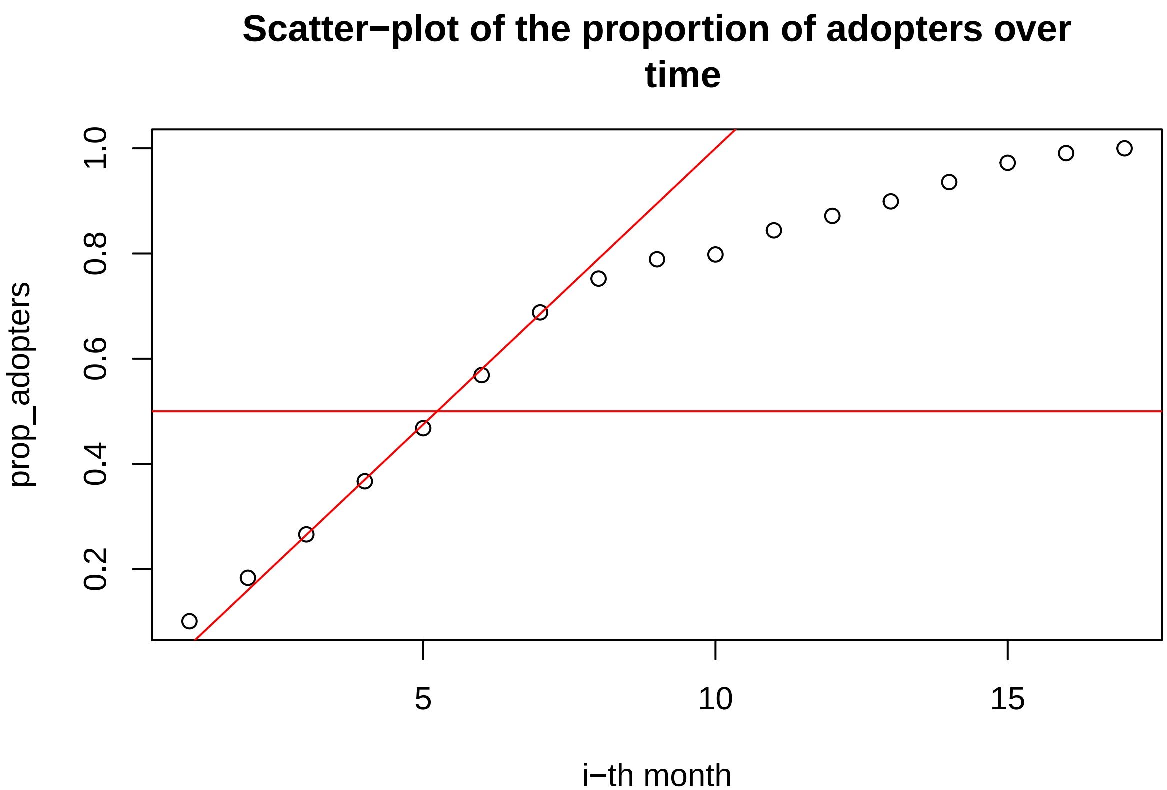
d. Make a scatter-plot of the proportion of adopters over time.

```r
prop_adopter_accu <- rep(0, max_date)
for(i in 1:max_date){
    prop_adopter_accu[i] <- sum(na.omit(ckm_nodes$adoption_date)<=i)/num_sum
}
plot(prop_adopter_accu, main="Scatter-plot of the proportion of adopters over
     time", xlab="i-th month", ylab="prop_adopters")
```



e. Make rough guesses about t_0 and b from the plot, and from your answers in problem 1.

```r
plot(prop_adopter_accu, main="Scatter-plot of the proportion of adopters over
     time", xlab="i-th month", ylab="prop_adopters")
abline(h=0.5, col=2)
abline(a=-0.05, b=0.105, col=2)
```




```r
# p(t_0)=0.5, t_0~=5.5
# Slope of p(t_0)=b/4, slope(t_0)~=0.105, b~=0.42
# Rough guesses: t_0=5.5, b=0.42
```

3. The Fit.

   a. Write a function, `logistic_mse`, which calculates the mean squared error of the logistic model on this data set. It should take a single vector, `theta`, and return a single number. This function cannot contain any loops, and must use your `logistic` function.

```r
theta_0 <- c(0.42, 5.5)
logistic_mse <- function(theta){
    m <- c(1:17)
    logi_mse <- mean((prop_adopter_accu-logistic(m, theta))^2)
    return(logi_mse)
}
```

```
b. Use `optim` to minimize `logistic_mse`, starting from your rough guess in problem 2e.
Report the location and value of the optimum to reasonable precision. (By default,
R prints to very unreasonable precision.)
```

```r
theta_0<-c(0.42, 5.5)
fit_1 <- optim(theta_0,logistic_mse,method="BFGS")
fit_1
```

```
## $par
## [1] 0.3738471 5.5196363
##
## $value
## [1] 0.001037297
##
## $counts
## function gradient
##       12       11
##
## $convergence
## [1] 0
##
## $message
## NULL
```

c. Add a curve of the fitted logistic function to your scatterplot from Problem 2d. Does it seem like a reasonable match?

```r
plot(prop_adopter_accu, main="Scatter-plot of the proportion of adopters over
     time", xlab="i-th month", ylab="prop_adopters")
curve(logistic(x, c(fit_1$par[1], fit_1$par[2])), add=TRUE, col=2)
```