MKT 500T

Spring 2018 HW3

Sami Cheong

Feb 12 2018

In Blackboard, I've placed a spreadsheet containing data on the number of prescriptions for a particular drug written by a sample of 1923 doctors in a given month. a. Fit the NBD model to this dataset using MLE, MOM, and "Mean and Zeros." Compare the resulting parameter estimates. b. Using MLE, fit the NBD and NBD with spike at zero. How well does each model fit the data (showing the appropriate histograms and using the chi-square goodness-of-fit test)? c. Using the likelihood ratio test, determine whether we can reject the null hypothesis that p in the NBD with spike at zero model is significantly different from 0. d. Which model would you choose as "best" and why? Using your preferred model, what is the expected distribution for the number of prescriptions over a 12-month period?

Q1. Methods of Moment estimates for NBD

We are given the following parameters for the Negative Binomial Distribution: $\mu = \frac{r}{\alpha}, \sigma^2 = \frac{r}{\alpha}(1+\frac{1}{\alpha})$. With some algebra, we can see that

$$\frac{\sigma^2}{\mu} = 1 + \frac{1}{\alpha} \implies \frac{1}{\alpha} = \frac{\sigma^2}{\mu} - 1 \implies \alpha = (\frac{\sigma^2 - \mu}{\mu})^{-1} = \frac{\mu}{\sigma^2 - \mu},$$

and

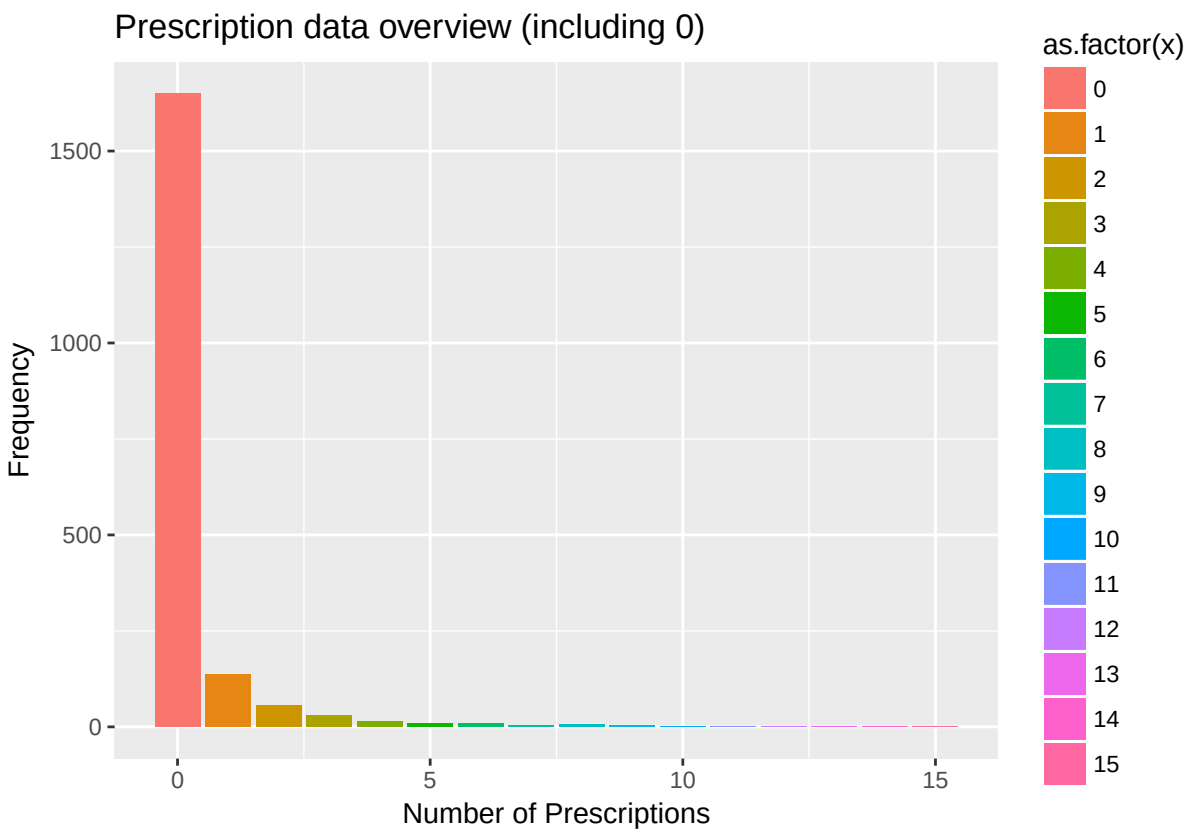
$$r = \frac{\mu}{\alpha} = \mu(\frac{\sigma^2 - \mu}{\mu}) \implies r = \sigma^2 - \mu.$$

Q2 Prescription data

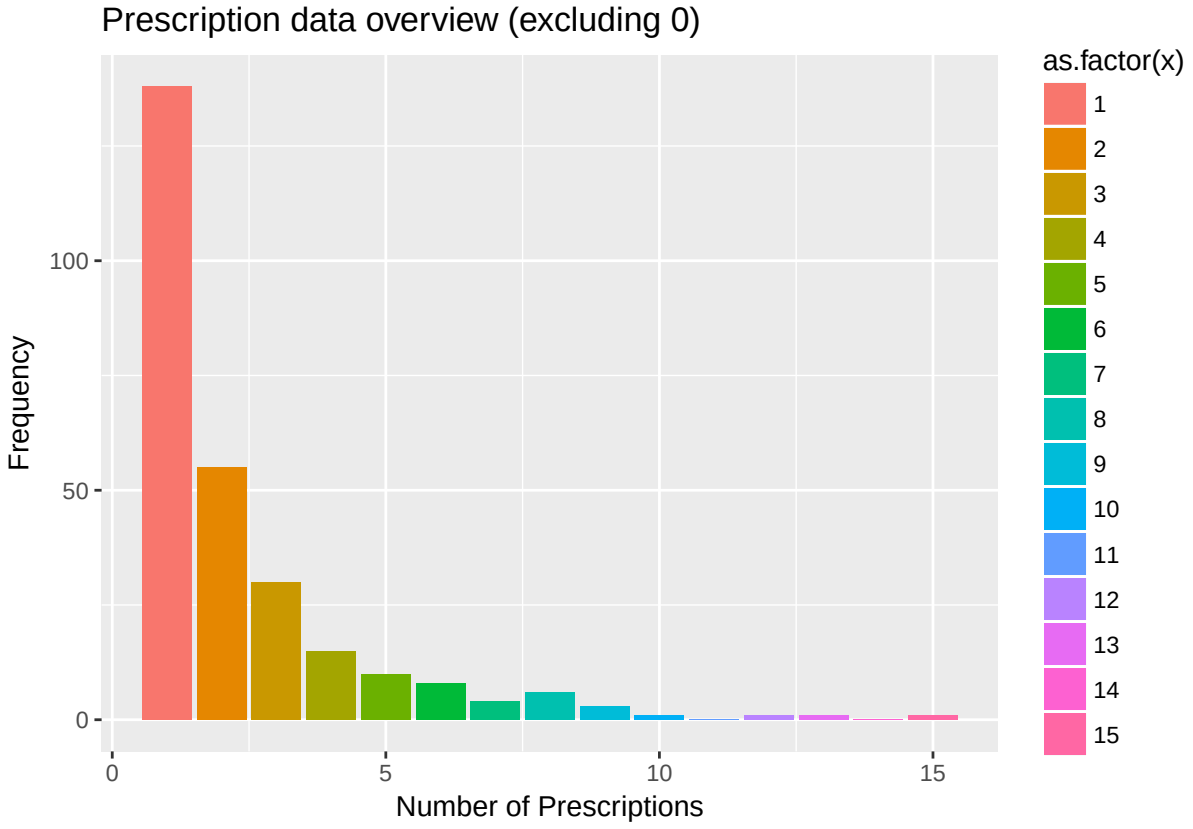
Data Overview

```
library(readxl)
library(knitr)
library(ggplot2)
source('code/util.R')

data<-read_excel('data/HW3 prescription data.xlsx',sheet = 1)
```

```
#kable(data)
ggplot(data)+geom_col(aes(x,n_x,fill=as.factor(x)))+
  labs(x='Number of Prescriptions',y='Frequency')+
  ggtitle('Prescription data overview (including 0)' )
```




```
ggplot(data[which(data$x!=0),])+geom_col(aes(x,n_x,fill=as.factor(x)))+
  labs(x='Number of Prescriptions',y='Frequency')+
  ggtitle('Prescription data overview (excluding 0)' )
```



a. Fitting NBD using MLE, MOM and 'Mean and Zeros' estimation method.

MLE

To compute the ML estimate, we use `optim` to search for the parameter values α and r that optimizes the log-likelihood function $l(x) = \sum i = 1^n n_x \log(p(x|\alpha, r))$

where $p(\cdot|\alpha, r)$ is the recursive Negative Binomial distribution defined by

$$P(X = x) = \begin{cases} (\frac{\alpha}{\alpha+1})^r & x = 0 \\ \frac{r+x-1}{x(\alpha+1)} P(X = x-1) & x = 1, 2, 3.. \end{cases},$$

and implemented in R as

```
NBD.recur<-function(data,par){
  N<-length(data)
  alpha<-exp(par[1])
  r<-exp(par[2])
  pmf<-vector(mode = 'numeric',N)

  pmf[1]<-(alpha/(1+alpha))^r

  if (N > 1) {
    for (x in 2:N) {
      pmf[x] = pmf[x-1]*(r + x - 1)/(x*(alpha+1))
    }
  }
  return(pmf);
}
```

and the likelihood function is implented in R as :

```
LL.recur<-function(data,pmf=NBD.recur,par){
  pmf = pmf(data,par)
  #print(pmf)
  #print(sum(pmf))
  LLVal = sum(log(pmf)*data)
  return(LLVal)

}
```

```
NBD.par.opt<-optim(par=c(0.01,0.05),fn=LL.recur,pmf=NBD.recur,
                   data =data$n_x, method="L-BFGS-B",control=list(fnscale=-1), lower=1e-10)

print('Optimal paramters for NBD:')
```

```
## [1] "Optimal paramters for NBD:"
```

```
print(exp(NBD.par.opt$par))
```

```
## [1] 2.96302 1.00000
```

```
print('Log-likelihood:')
```

```
## [1] "Log-likelihood:"
```

```
print(NBD.par.opt$value)
```

```
## [1] -1452.88
```

```r
alpha<-exp(NBD.par.opt$par[1])
r<-exp(NBD.par.opt$par[2])
par_MLE<-c(alpha,r)
```

Method of Moments

To solve for α and r using Method of Moments, notice that from Q1 we have a two simple equations

$$\alpha = \frac{\sigma^2}{\mu - \sigma^2}$$

and

$$r = \sigma^2 - \mu$$

The expected value μ can be estimated as $\mu = (\sum_{i=1}^{N} p_i x_i)$ and $\sigma^2 = \sum_{i=1}^{N} p_i (x_i - \mu)^2$

```r
data[,'p_x']<-data$n_x/sum(data$n_x)
N<-nrow(data)
scale<-(N-1)/N
mu = sum(data$p_x * data$x)
sigma2 = ((N-1)/N)*sum(data$p_x * (data$x-mu)^2)


alpha<-sigma2/(sigma2-mu)
r <- sigma2 - mu
par_MOM<-c(alpha,r)
print(par_MOM)
```

```
## [1] 1.3539738 0.9534419
```

Mean and Zeros

Let $p_0 = P(X = 0)$, where X is the number of prescriptions. Form the definition of NBD, we have

$$p_0 = (\frac{\alpha}{\alpha + 1})^r,$$

and as previously defined we have already that $\mu = \frac{r}{\alpha}$. From the data, we can deduce the estimates for p_0 and μ, this allows us to solve for α and r via a system of two non-linear equations:

$$(\frac{\alpha}{\alpha + 1})^r - p_0 = 0, \text{ and}$$
$$\frac{r}{\alpha} - \mu = 0$$

To implement this in `R`, we define the eqution above and use the package `nlesolve` to solve for the parameters α and r :

```r
library(nleqslv)
MAZ<-function(par,data){
  p0 <-data$p_x[1]
  mu <- sum(data$p_x*data$x)
  alpha <- par[1]
  r <-par[2]

  y<-numeric(2)
  y[1]<-(alpha/(alpha+1))^r - p0
  y[2]<-r/alpha - mu
```

```
  return(y)
}
# solve for alpha and r
par_MAZ_solv<-nleqslv(c(0.2,0.2),data=data,
                  MAZ,
                  control = list(ftol = 1e-10, allowSingular = TRUE),
                  jacobian = TRUE,method = 'Newton')
par_MAZ<-exp(par_MAZ_solv$x)
#print(par_MAZ)
```

Below is the summary after estimating the NBD parameter values using all three methods. A remark of the resulting parameter estimates, the r values for all three estimation method are relatively close, while the $\alpha's$ had a much wider range. This implies that the shaping parameters should be similar under all these parameter estimation scheme.s

```
par_summary<-data.frame(rbind(par_MLE,par_MOM,par_MAZ))
names(par_summary)<-c('alpha','r')

kable(par_summary,align = 'c',col.names = c('alpha','r'))
```

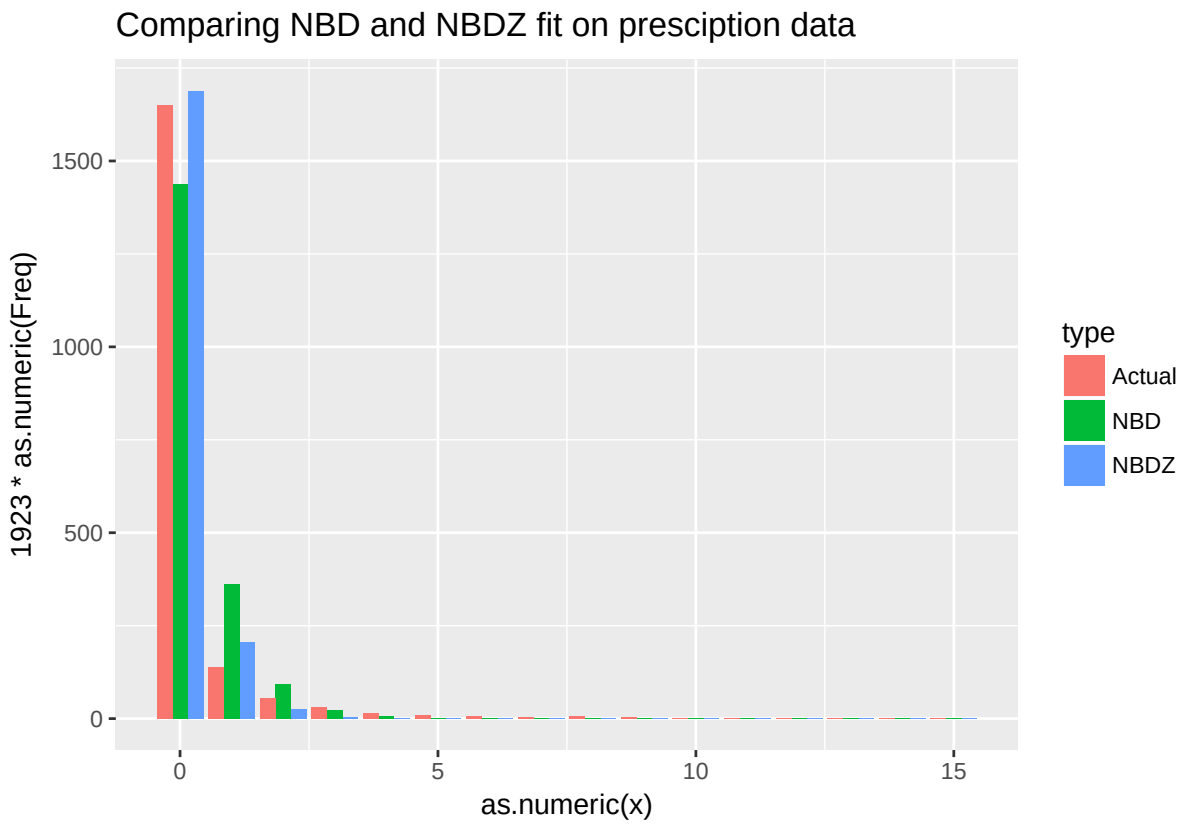
	alpha	r
par_MLE	2.963020	1.0000000
par_MOM	1.353974	0.9534419
par_MAZ	1.377567	1.1141654

```
pmf_compare<-lapply(1:3, function(i){
  pmf=NBD.recur(data$n_x,par=log(par_summary[i,]))
  return(pmf)
})
pmf_compare_summary<-do.call(cbind,pmf_compare)
NBD_compare_summary<-data.frame(cbind(pmf_compare_summary,data$p_x))
names(NBD_compare_summary)<-c('MLE','MOM','MAZ','ACTUAL')
```

b. By fitting a NBDZ model, we get the following parameters: $alpha = 7.1289$, $r = 1$ and $p_0 = 0.01$. However, the resulting fit is still extremely close to the regular NBD, and neitherof the models provide a great fit on the prescription data, as we can see for example at x=1. Also, both models have a chi-squared with a p-value close to 0, further suggesting that the model result is not satisfactory.

```
NBDZ.recur<-function(data,par,pos=T){
  N<-length(data)
  if(pos){
  alpha<-exp(par[1])
  r<-exp(par[2])
  p0<-(par[3])^2/(1+par[3])^2

  }
  else{
  alpha<-par[1]
  r<-par[2]
  p0<-par[3]/(1+par[3])

  }
  #print(alpha)
  #print(r)
```

```
  #print(p0)

  pmf<-vector(mode = 'numeric',N)

  pmf[1]<-p0 + (1-p0)*(alpha/(1+alpha))^r

  if (N > 1) {
    for (x in 2:N) {
      pmf[x] = (1-p0)*pmf[x-1]*(r + x - 1)/(x*(alpha+1))
    }
  }
  return(pmf);

}

NBD.par.opt<-optim(par=c(1,0.2),fn=LL.recur,pmf=NBD.recur,
                   data =data$n_x, method="L-BFGS-B",control=list(fnscale=-1), lower=1e-10)


NBDZ.par.opt<-optim(par=c(0.1,0.1,0.1),fn=LL.recur,pmf=NBDZ.recur,
                   data =data$n_x, method="L-BFGS-B",control=list(fnscale=-1), lower=1e-10)

par_NBDZ<-c((NBDZ.par.opt$par[1]),
            (NBDZ.par.opt$par[2]),
            (NBDZ.par.opt$par[3]))


par_NBDZ_new<-c(exp(NBDZ.par.opt$par[1]),
            exp(NBDZ.par.opt$par[2]),
            (NBDZ.par.opt$par[3])^2/(1+NBDZ.par.opt$par[3]))^2

NBD_pmf<-data.frame(cbind(data$x,NBD.recur(data$n_x,NBD.par.opt$par)))

NBDZ_pmf<-data.frame(cbind(data$x,NBDZ.recur(data$n_x,par_NBDZ_new,pos = F)))
```




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

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

c. Using the liklihood ratio test, we found that the p-value associated with $-2(LL_{NBD} - LL_{NBDZ})$ is close to 1, this means that we cannot reject the hypothesis that p is significantly different than 0 in the NBDZ model.

```
LL_nbd<-NBD.par.opt$value
LL_nbdz<-NBDZ.par.opt$value
chi2<- -2*(LL_nbd-LL_nbdz)
print(pchisq(chi2,1,lower.tail = F))
```

```
## [1] 0.999968
```