

# Negbin\_fit2.R

SIU850486795

2023-11-10

```
# Negbin_fit2.R
# Fitting the negative binomial to frequency data

# Load necessary libraries
library(ggplot2)
library(MASS)

# Read in data set
nbdata <- read.table(header=T,colClasses=c("numeric","numeric"),text="
y obsfreq
0 24
1 16
2 16
3 18
4 15
5 9
6 6
7 5
8 3
9 4
10 3
11 0
12 1
")

# Generate offset y values for plot
nbdata <- transform(nbdata,yexp=y-0.1)
nbdata <- transform(nbdata,yobs=y+0.1)

# Print data
nbdata
```

| ##    | y | obsfreq | yexp | yobs |
|-------|---|---------|------|------|
| ## 1  | 0 | 24      | -0.1 | 0.1  |
| ## 2  | 1 | 16      | 0.9  | 1.1  |
| ## 3  | 2 | 16      | 1.9  | 2.1  |
| ## 4  | 3 | 18      | 2.9  | 3.1  |
| ## 5  | 4 | 15      | 3.9  | 4.1  |
| ## 6  | 5 | 9       | 4.9  | 5.1  |
| ## 7  | 6 | 6       | 5.9  | 6.1  |
| ## 8  | 7 | 5       | 6.9  | 7.1  |
| ## 9  | 8 | 3       | 7.9  | 8.1  |
| ## 10 | 9 | 4       | 8.9  | 9.1  |

```
## 11 10      3  9.9 10.1
## 12 11      0 10.9 11.1
## 13 12      1 11.9 12.1

# Convert tabulated data to raw form for R
ydata <- rep(npdata[, "y"], npdata$obsfreq)

# Calculate mean and variance
mean(ydata)

## [1] 3.166667

var(ydata)

## [1] 7.770308

# Fit negative binomial distribution
# Note that k = size, m = mu
fitout <- fitdistr(ydata, "negative binomial")
fitout

##      size      mu
## 1.7604320 3.1666614
## (0.4023404) (0.2717661)

# Calculate expected frequencies
npdata <- transform(npdata, nbprob = dnbinom(npdata$y, size = fitout$estimate[1], mu = fitout$estimate[2]))
npdata <- transform(npdata, expfreq = fitout$n * nbprob)

# Print revised data
npdata

##      y obsfreq yexp yobs      nbprob      expfreq
## 1  0      24 -0.1  0.1 0.163356715 19.6028058
## 2  1      16  0.9  1.1 0.184827709 22.1793250
## 3  2      16  1.9  2.1 0.163955116 19.6746139
## 4  3      18  2.9  3.1 0.132084632 15.8501559
## 5  4      15  3.9  4.1 0.101029802 12.1235762
## 6  5       9  4.9  5.1 0.074807542  8.9769051
## 7  6       6  5.9  6.1 0.054172568  6.5007081
## 8  7       5  6.9  7.1 0.038599183  4.6319020
## 9  8       3  7.9  8.1 0.027165924  3.2599109
## 10 9       4  8.9  9.1 0.018934849  2.2721819
## 11 10      3  9.9 10.1 0.013094906  1.5713888
## 12 11      0 10.9 11.1 0.008997955  1.0797546
## 13 12      1 11.9 12.1 0.006149486  0.7379383

# Generate plot showing frequencies
ggplot(npdata, aes(yobs, obsfreq)) +
  geom_bar(stat = "identity", width = 0.05, fill = "blue") +
  geom_point(color = "blue", size = 3) +
  geom_bar(aes(yexp, expfreq), stat = "identity", width = 0.05, fill = "red") +
  geom_point(aes(yexp, expfreq), color = "red", size = 3) +
  ggtitle("Fitting the negative binomial to frequency data")
```

