Count Data

DR. SUSAN SIMMONS

MSA 2023

Count

Examples of Count Data

- Number of bicycles rented at a bicycle shop
- Number of highway deaths
- Number of customers visiting a store
- Number of diseased trees
- Number of people with Dengue Fever in Peru
- Number of open data science jobs
- Many, many more….

Poisson Distribution

Most common distribution to model count data is the Poisson distribution

Why not Normal distribution?

- **Mean must be positive**
- **Errors are more appropriate with Poisson regression (when dealing with count data)**

Poisson Distribution

The Poisson distribution:

$$P(y) = \frac{e^{-\lambda}\lambda^{y}}{y!}$$

The ‘λ’ in the distribution is the mean (and variance!!) of this distribution!!

NOTE: Mean is EQUAL to variance

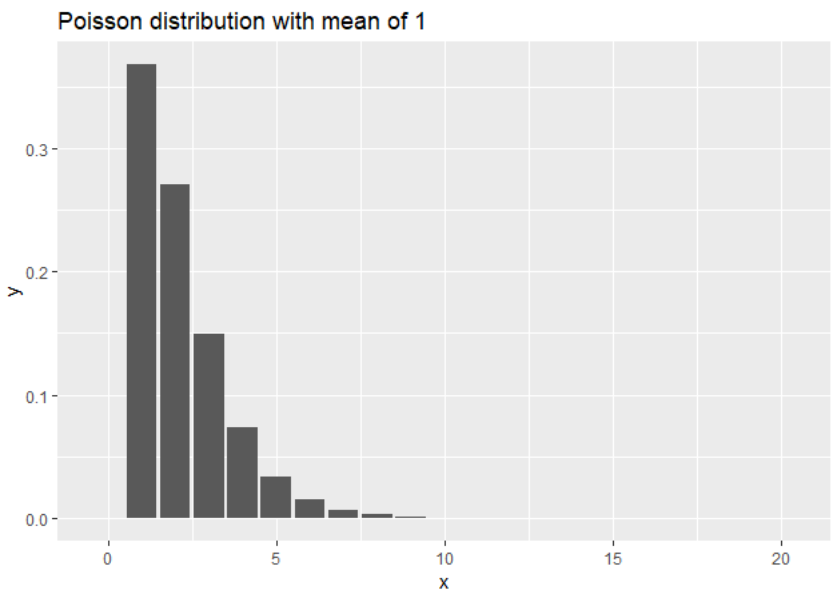
NOTE: Mean is always positive

We will model the mean of this distribution

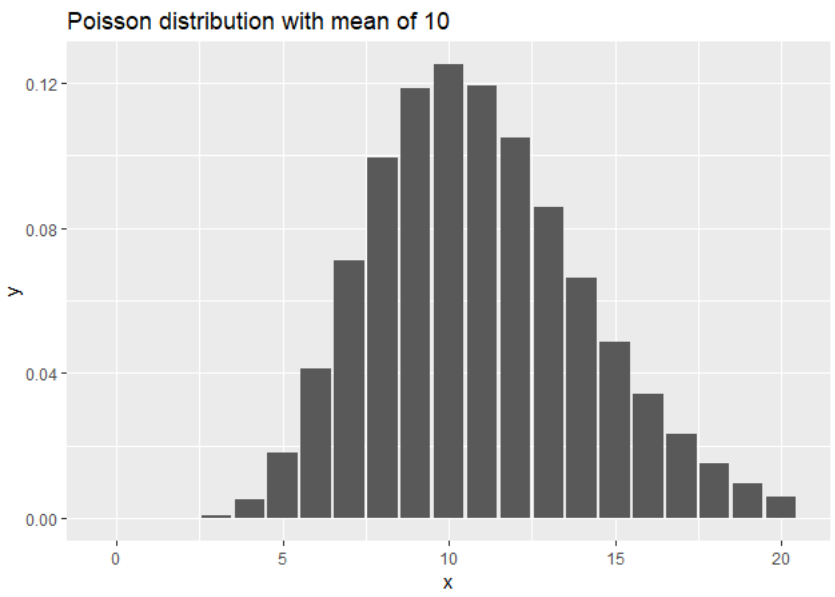
Poisson Regression

Examples of Poisson distribution

λ=1



λ=10



Poisson regression

In Poisson regression, we model the mean (λ_i)

The mean, λ_i, must be positive, however,

$$\lambda_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon_i$$

Can be negative!!!

Poisson regression

In Poisson regression, we model the mean (λ_i)

The mean, λ_i, must be positive, soooo, we force it to be positive…. $\lambda_i = e^{X\beta}$

$$\log(\lambda_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon_i$$

This is called a “link” function…links mean to the linear predictor!

Other link functions….

Identity Link

$$\mu_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon_i$$

Log Link

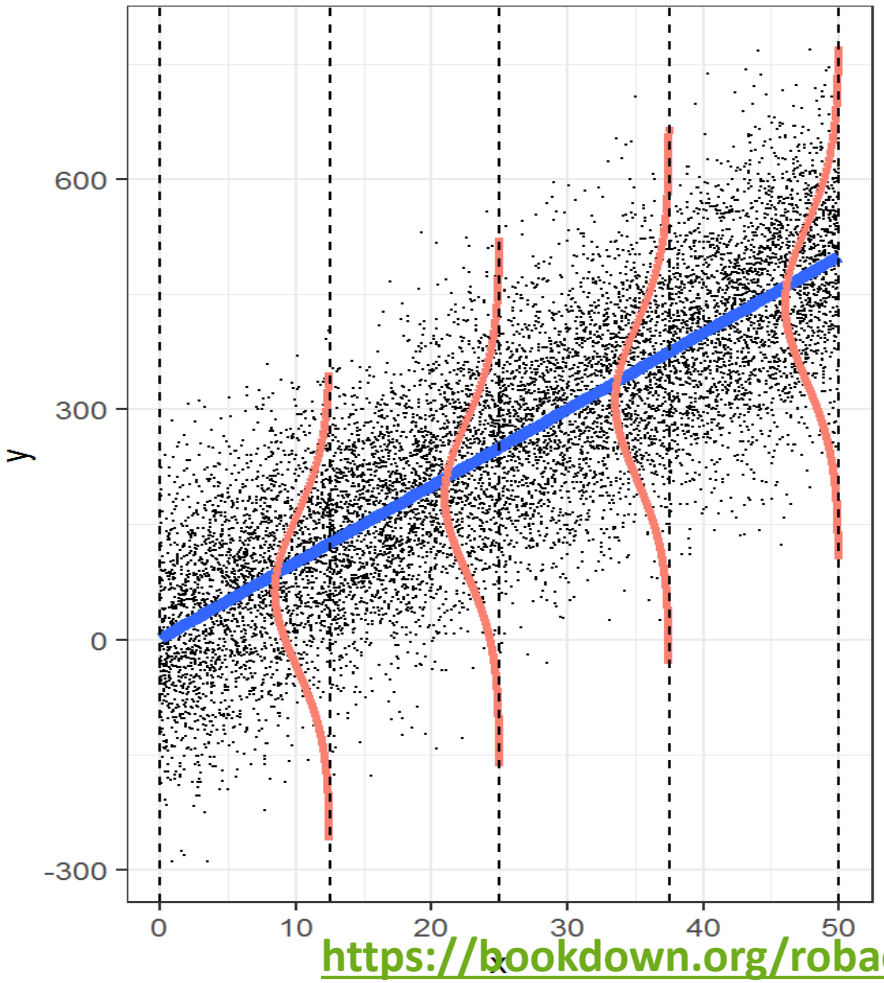
$$\log(\lambda_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon_i$$

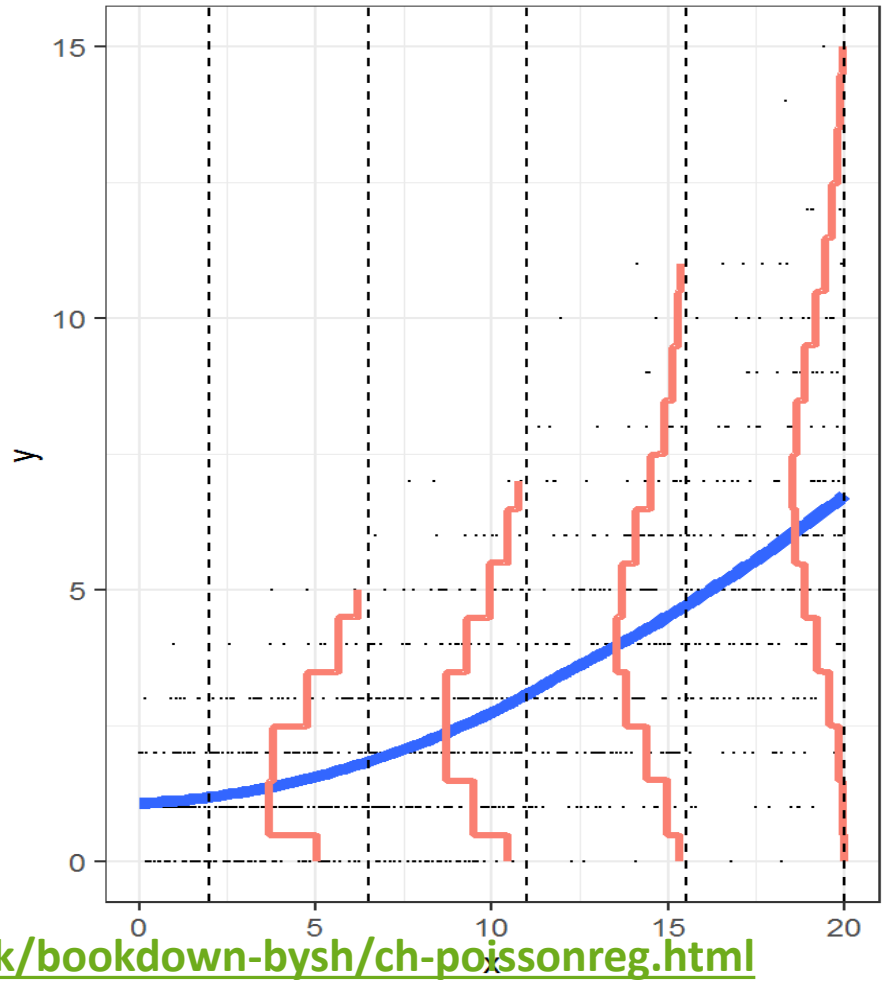
Logit Link

$$log\left(\frac{p_i}{1 - p_i}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon_i$$

Poisson regression

Julie Legler and Paul Roback





https://bookdown.org/roback/bookdown-bysh/ch-poissonreg.html

Poisson regression

We model $\log(\lambda_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_k x_k + \varepsilon_i$

Assumptions

- $E(Y_i|X_i)=V(Y_i|X_i)$ (conditional mean = conditional variance)
- Independent observations
- Linearity in the mean of the response

Some notes:

- In most algorithms, variance of estimators is calculated using the Hessian matrix (inverse of the second derivatives). If you see that that the Hessian is *singular*, you need to respecify model.
- If the algorithm does *not converge*, you need to respecify the model (or try another minimization algorithm…in SAS, default is Newton Raphson (NRA) however, QN (Quasi Newton is an alternative).
- Careful of potential multicollinearity.

Poisson example

Estimating household size in the Philippines from the Family Income and Expenditure Survey (FIES), which is done every three years by the Philippine Statistics Authority (PSA). This data is from the 2015 FIES and is a subset of the 40,000 observations from five regions: Central Luzon, Metro Manila, Ilocos, Davao, and Visayas.

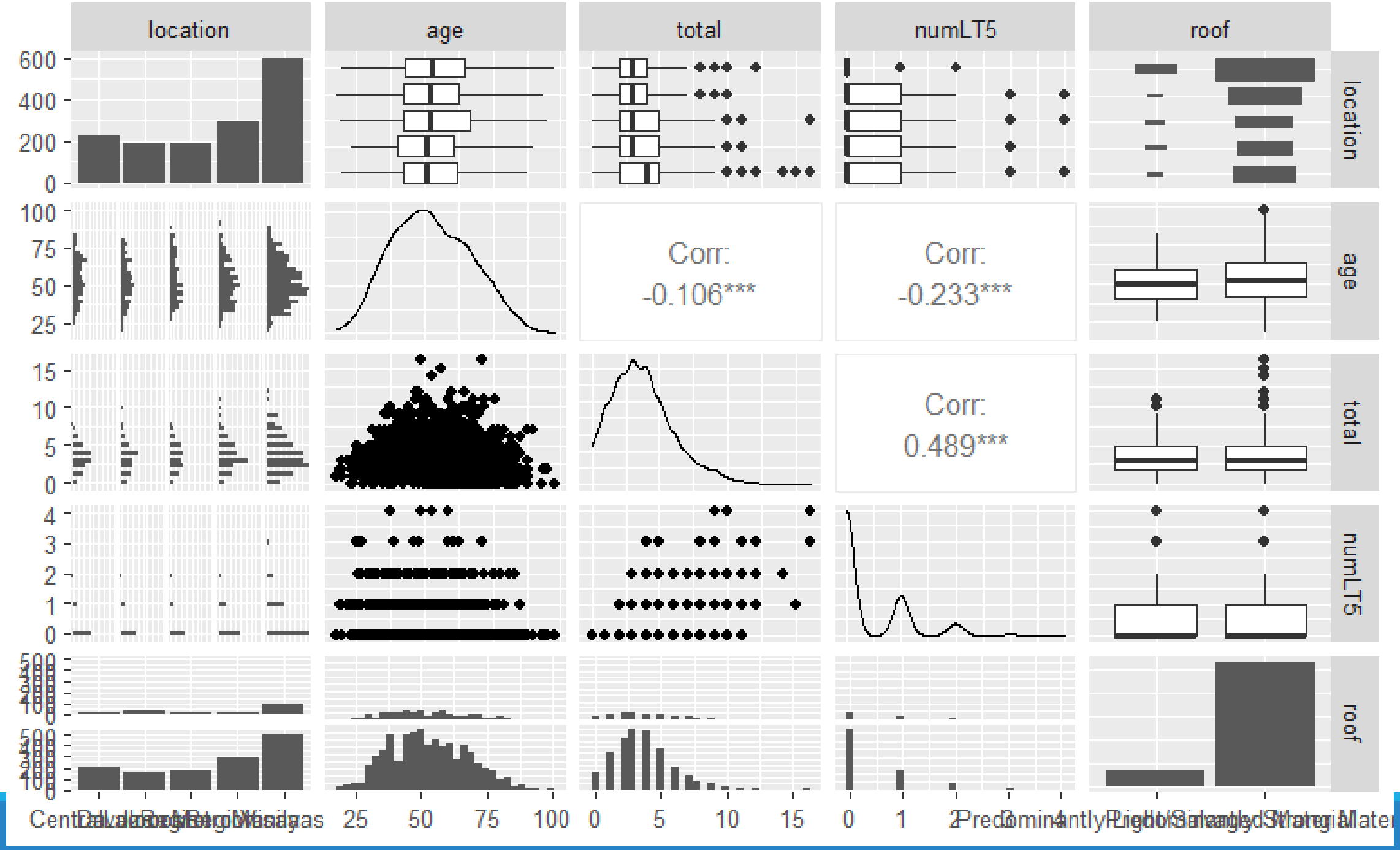
age = the age of the head of household

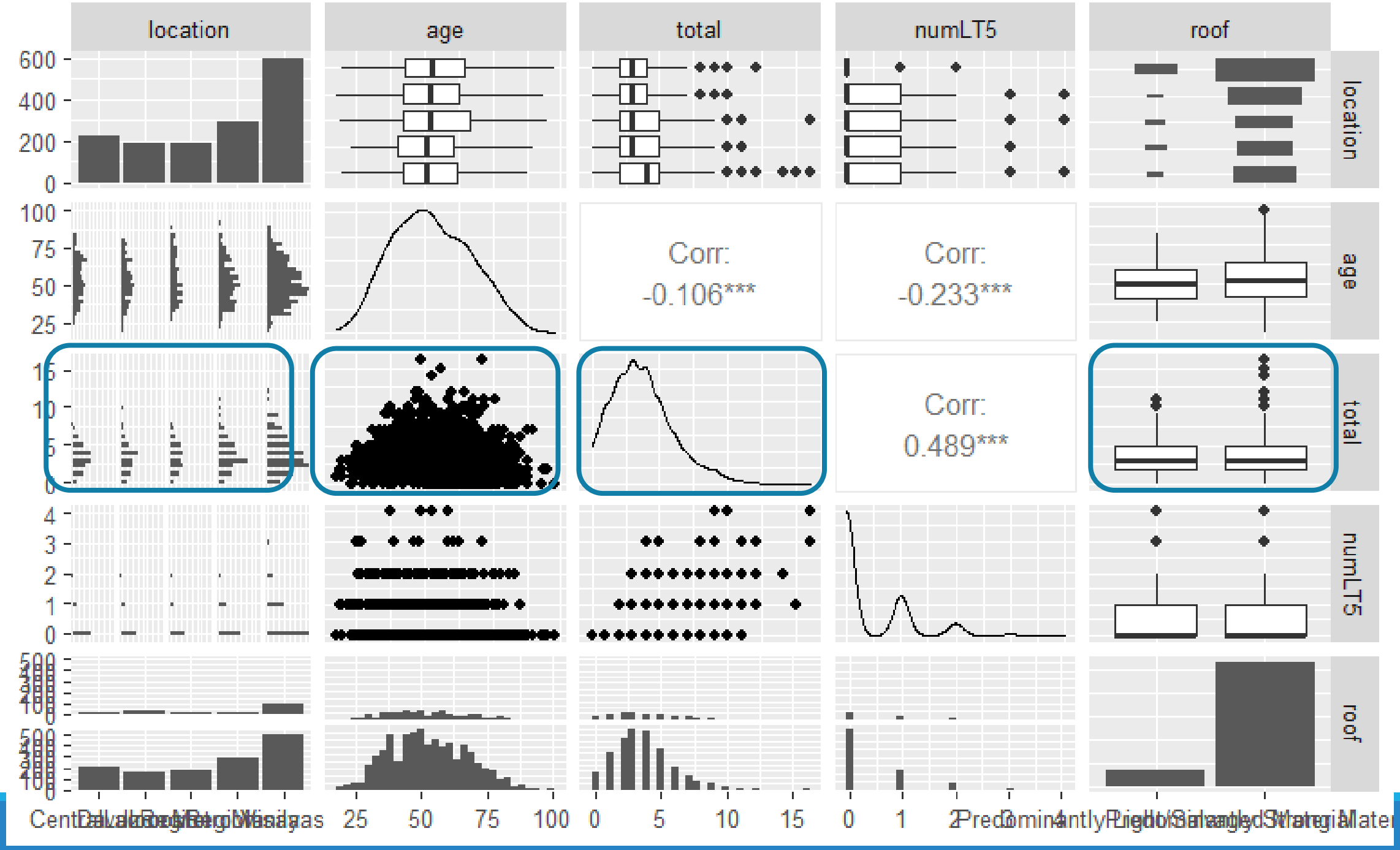
numLT5 = the number in the household under 5 years of age

total = the number of people in the household other than the head (RESPONSE VARIABLE)

roof = the type of roof in the household (either Predominantly Light/Salvaged Material, or Predominantly Strong Material, where stronger material can sometimes be used as a proxy for greater wealth)

location = where the house is located (Central Luzon, Davao Region, Ilocos Region, Metro Manila, or Visayas)

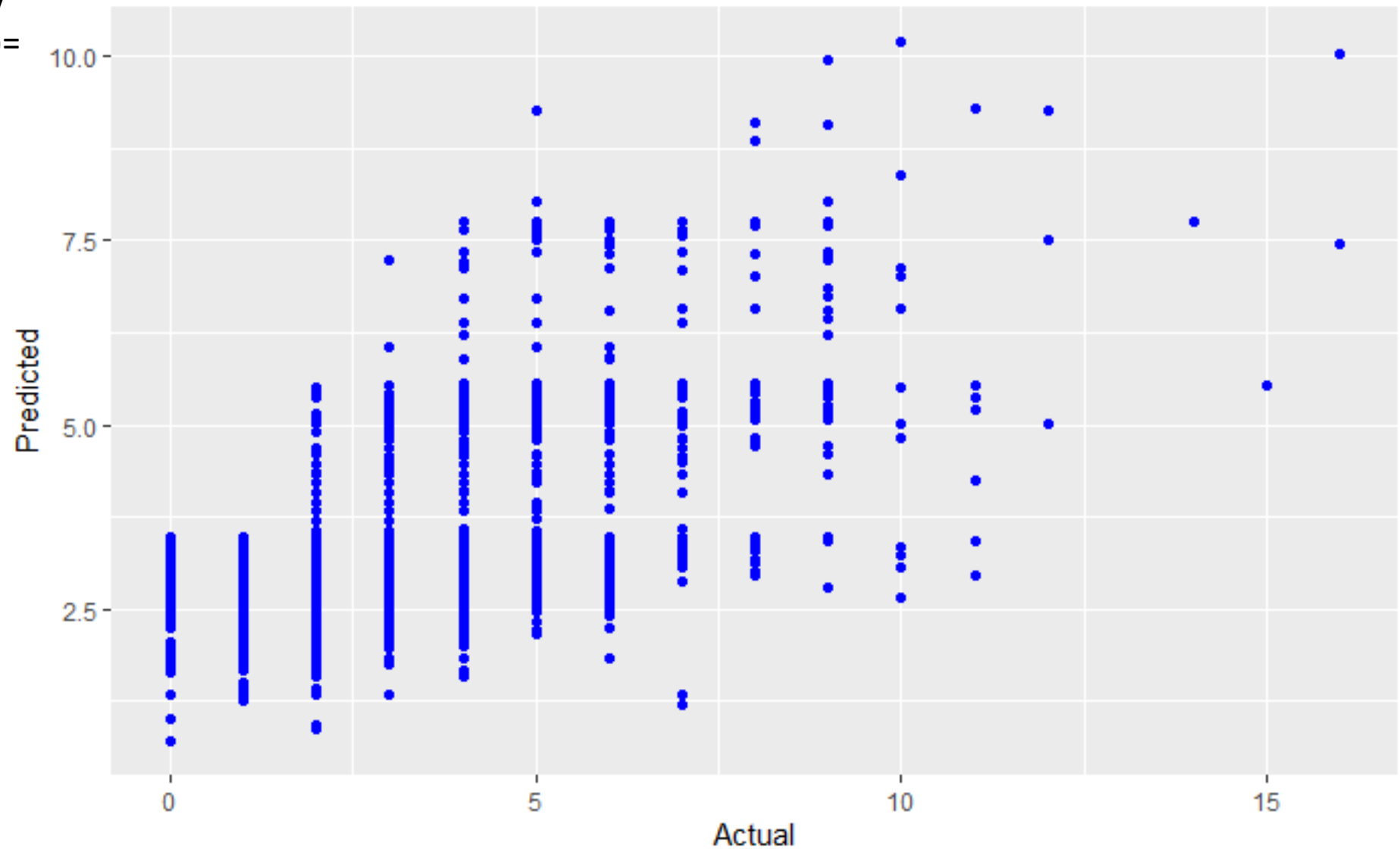

location
age
total
numLT5
roof
Corr:
-0.106***
Corr:
-0.233***
Corr:
0.489***


location
age
total
numLT5
roof
Corr:
-0.106***
Corr:
-0.233***
Corr:
0.489***

```
model.pois <- glm(total ~ poly(age,2)+poly(numLT5,2),
family="poisson", data=fhh.dat)
summary(model.pois)
```

```
Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       1.24801    0.01418  87.982  < 2e-16 ***
poly(age, 2)1    -0.59694    0.60295  -0.990    0.322
poly(age, 2)2    -7.70603    0.62645 -12.301  < 2e-16 ***
poly(numLT5, 2)1 10.45595    0.45854  22.803  < 2e-16 ***
poly(numLT5, 2)2 -1.69510    0.38958  -4.351 1.35e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 2362.5  on 1499  degrees of freedom
Residual deviance: 1720.4  on 1495  degrees of freedom
AIC: 6103.3
```

Pseudo R^2 = 1-Deviance(full)/Deviance(null)= 0.2718



Negative Binomial Regression

Negative binomial

What happens if the conditional variance is bigger than the conditional mean? This is called "overdispersion". We can use the Negative binomial in this case….

Model: $\log(\mu_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_k x_k + \varepsilon_i$

Extra parameter for "overdispersion" (we can test if there is overdispersion…..the dispersion parameter in SAS is alpha; in R it is theta)

To test overdispersion:

- H_0: Poisson is appropriate
- H_A: Negative Binomial is appropriate

R provides information for Wald test (can also perform Likelihood Ratio test)

More Negative Binomial

Some notes:

- In most algorithms, variance of estimators is calculated using the Hessian matrix (inverse of the second derivatives). If you see that that the Hessian is singular, you need to respecify model.
- If the algorithm does not converge, you need to respecify the model.
- Careful of potential multicollinearity.

Negative binomial is NOT recommended for small samples.

Negative binomial example

The data (Medicare) is a cross-sectional data set from health economics. There are a total of 4406 individuals, aged 66 and over, who are covered by Medicare, a public insurance program. Originally obtained from the US National Medical Expenditure Survey (NMES) for 1987/88. The variables we will be focusing on are:

Ofp – number of physicians office visits

Hosp – number of hospital stays

Health – self-perceived health status

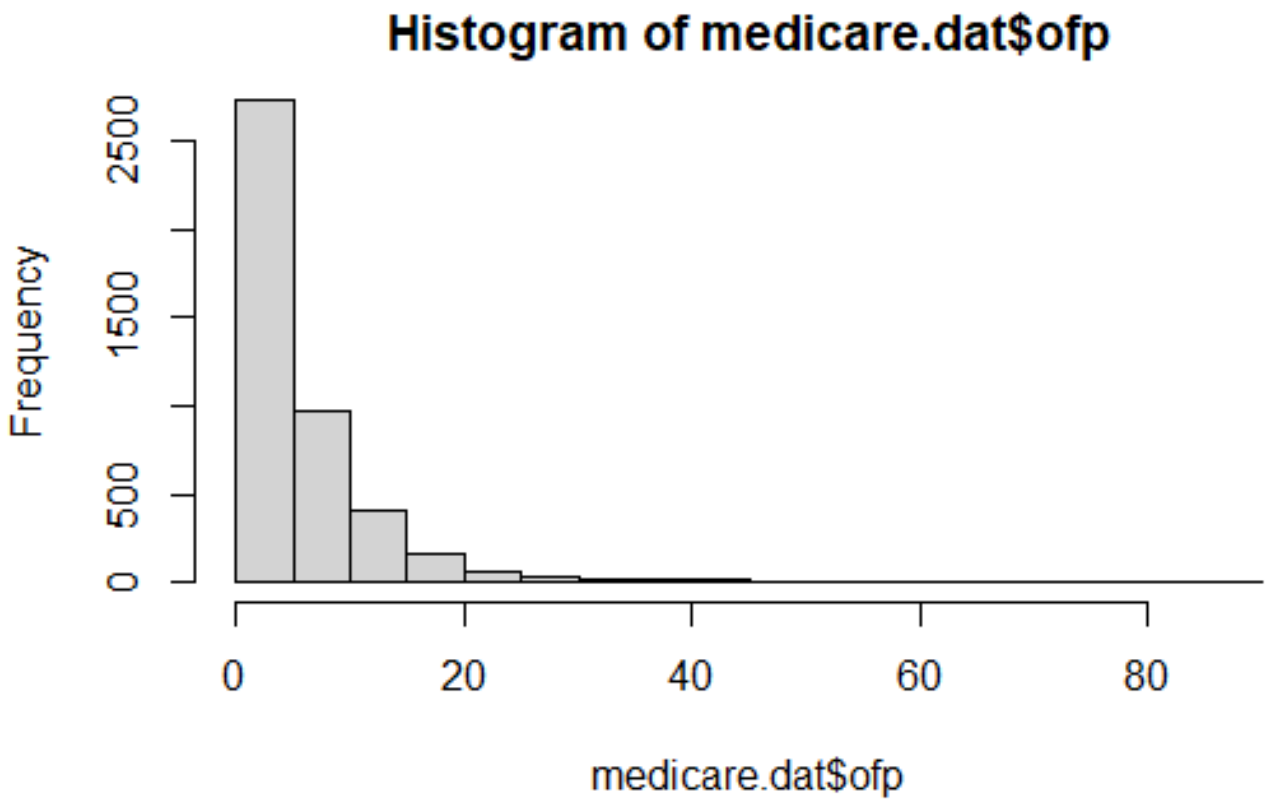
Numchron – number of chronic conditions

Gender –gender

School – number of years of education

Privins – indicator variable for private insurance

EDA


Histogram of medicare.dat$ofp
Frequency
0
500
1500
2500
0
20
40
60
80
medicare.dat$ofp

```
model.ngbin<-glm.nb(ofp ~ factor(health) +
factor(gender)+factor(privins)+hosp+numchron+school, link=log, data=dat)
```

Coefficients:

```
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)               0.929257   0.054591  17.022  < 2e-16 ***
factor(health)excellent -0.341807   0.060924  -5.610 2.02e-08 ***
factor(health)poor        0.305013   0.048511   6.288 3.23e-10 ***
factor(gender)male       -0.126488   0.031216  -4.052 5.08e-05 ***
factor(privins)yes        0.224402   0.039464   5.686 1.30e-08 ***
hosp                      0.217772   0.020176  10.793  < 2e-16 ***
numchron                  0.174916   0.012092  14.466  < 2e-16 ***
school                    0.026815   0.004394   6.103 1.04e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Theta: 1.2066
Std. Err.: 0.0336

Can we simplify from Negative Binomial to Poisson?

H_0: Scale is 1 (simplifies to Poisson)

H_A: Scale is different than 1 (overdispersed - need to keep Negative Binomial)

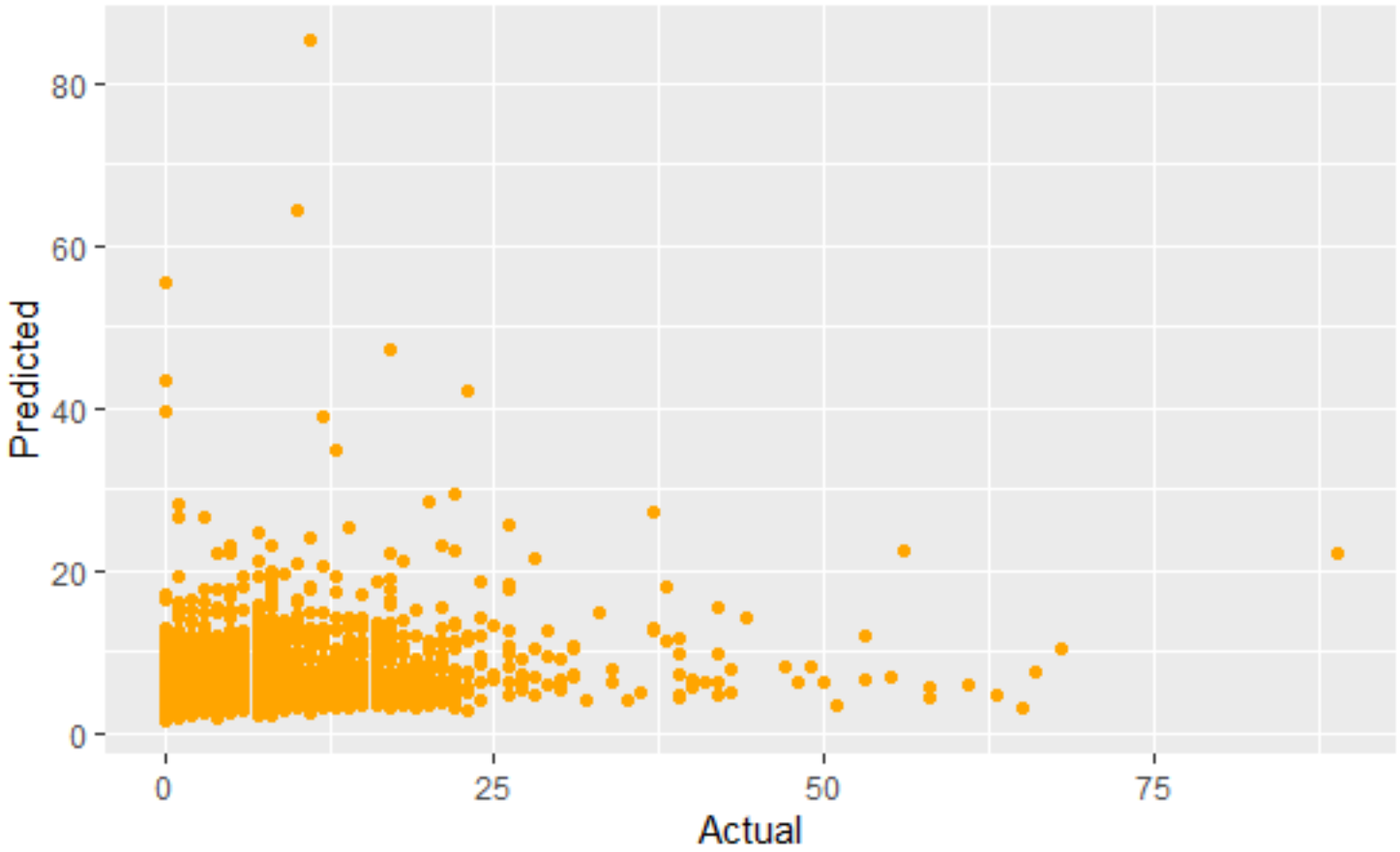
<u>Wald Test:</u>

Theta: 1.2066 Std. Err.: 0.0336 Wald Statistic = (1.2066-1)/0.0336 = 6.149 (p-value very small!!)

<u>Likelihood ratio test: χ^2 with 1 df (this one is BETTER)</u>

```
ll.test=-2*(as.numeric(logLik(model.pois2))-as.numeric(logLik(model.negbin)))
pchisq(ll.test,1,lower.tail=F)
0
```

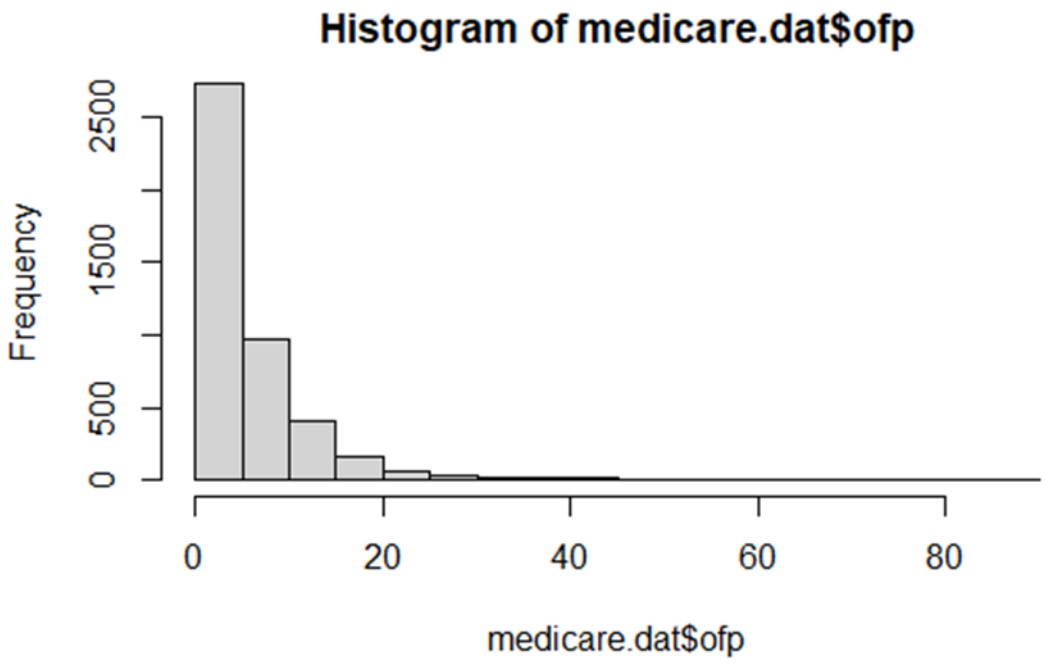
Pseudo R2 =
0.1217



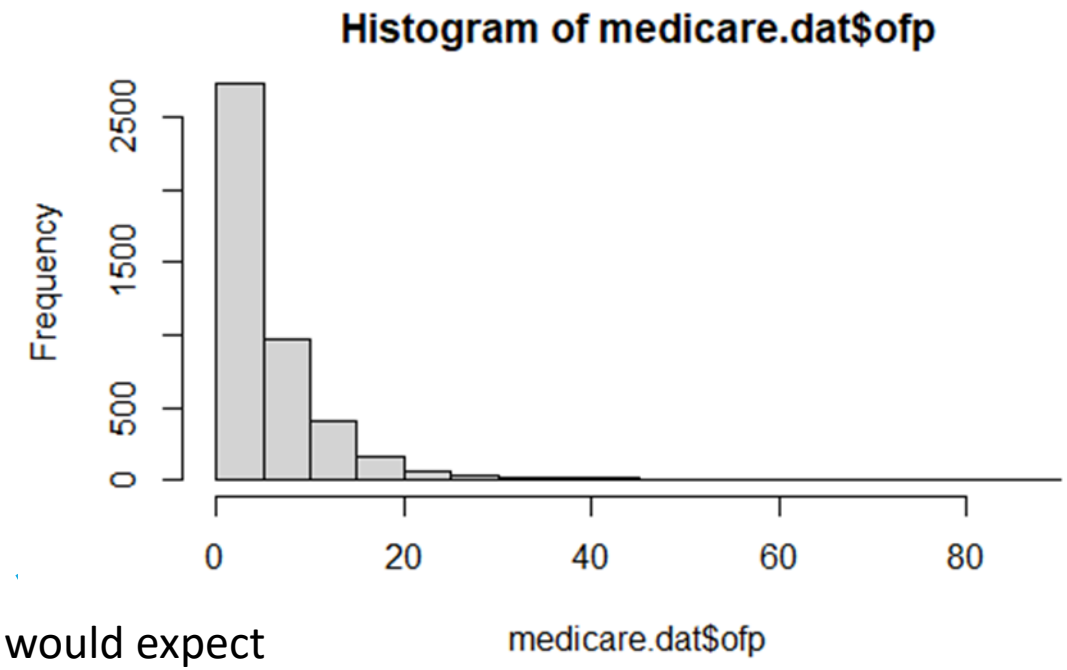
Zero-inflated Poisson regression

Some count data have A LOT of zeros

Both the Poisson and Negative binomial can be fit with ‘Zero-inflated’ models



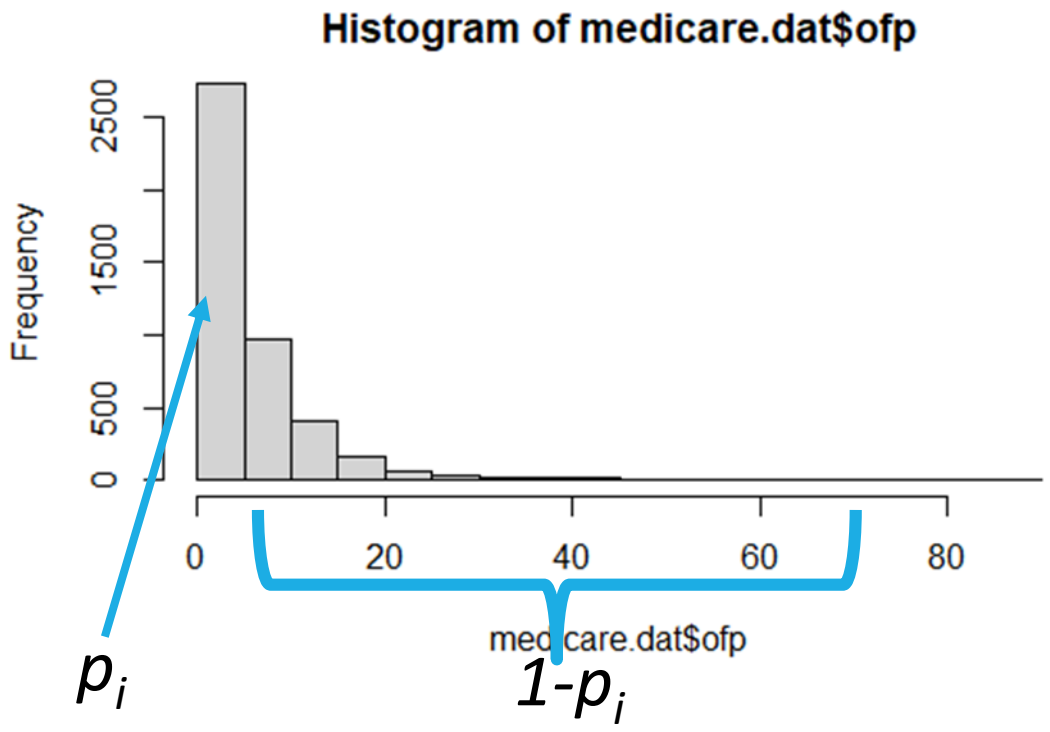
Zero-inflated Poisson



More zeros than one would expect with a Poisson distribution

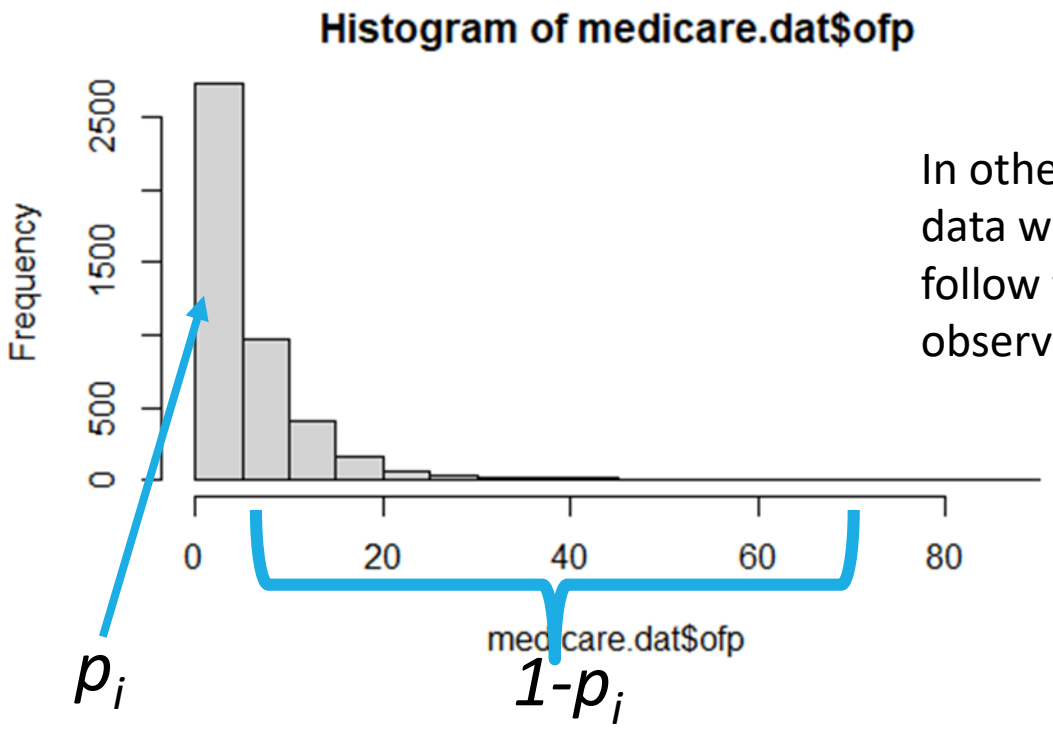
Zero-inflated Poisson

Since there are more zeros than one would expect with a Poisson distribution, we need to model this extra amount of zeros



Zero-inflated Poisson

Since there are more zeros than one would expect with a Poisson distribution, we need to model this extra amount of zeros



In other words, we model the what percent of the data we expect to be at 0 and what percent would follow the Poisson (keep in mind that some of the observations at zero are from the Poisson distribution)

Zero-inflated Poisson

There are two pieces that need to be modeled:

- Extra zero's - Predict having 0's versus not having 0's (binary outcome)
- The Poisson regression

The first part of the model fits a Logistic regression (predict 0 versus having a "count"…this is treated as binary)

The second part models the count data as Poisson with mean λ

Zero-inflated Poisson

There are two pieces that need to be modeled:

- Extra zero's - Predict having 0's versus not having 0's (binary outcome)
- The Poisson regression

The first part of the model fits a Logistic regression (predict 0 versus having a "count"…this is treated as binary)

CAN USE VARIABLES TO PREDICT THIS PART

The second part models the count data as Poisson with mean λ

Zero-inflated Poisson

There are two pieces that need to be modeled:

- Extra zero's - Predict having 0's versus not having 0's (binary outcome)
- The Poisson regression

The first part of the model fits a Logistic regression (predict 0 versus having a "count"…this is treated as binary)

The second part models the count data as Poisson with mean λ

CAN USE VARIABLES TO PREDICT THIS PART

```
model.zpois2<-zeroinfl(ofp ~
factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')
```

```
model.zpois2<-zeroinfl(ofp ~
factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')
```

These factors are for the Poisson regression part

```
model.zpois2<-zeroinfl(ofp ~

factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')
```

```
model.zpois2<-zeroinfl(ofp ~

factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')
```

These factors are for the Logistic regression part

model.zpois2<-zeroinfl(ofp ~ factor(health) + factor(gender)+factor(privins)+hosp+numchron+school| factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')

Count model coefficients (poisson with log link):

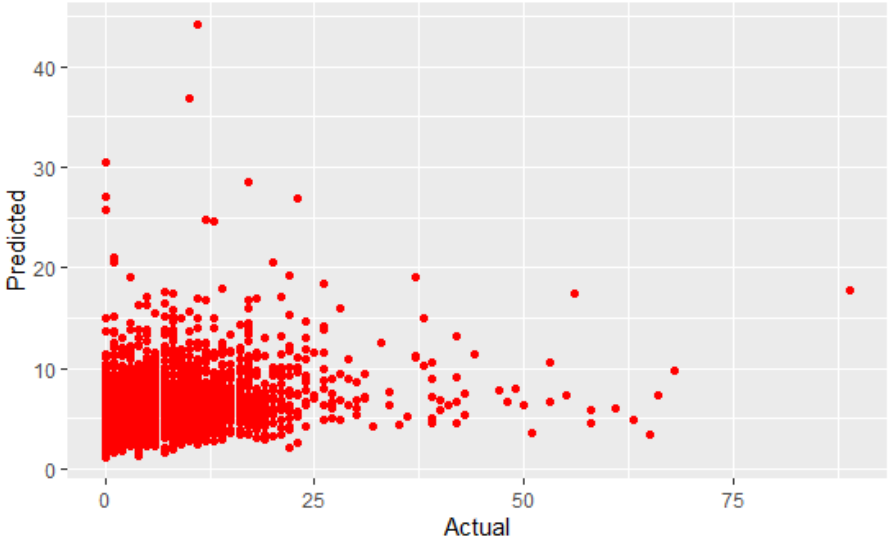
```
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)               1.405600   0.024179  58.134  < 2e-16 ***
factor(health)excellent -0.307366   0.031265  -9.831  < 2e-16 ***
factor(health)poor        0.253416   0.017706  14.313  < 2e-16 ***
factor(gender)male       -0.062352   0.013056  -4.776 1.79e-06 ***
factor(privins)yes        0.080533   0.017147   4.697 2.65e-06 ***
hosp                      0.159014   0.006060  26.240  < 2e-16 ***
numchron                  0.101846   0.004721  21.573  < 2e-16 ***
school                    0.019169   0.001873  10.232  < 2e-16 ***

Zero-inflation model coefficients (binomial with logit link):
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        -0.05937    0.14040  -0.423 0.672392
factor(gender)male  0.41806    0.08920   4.687 2.77e-06 ***
factor(privins)yes -0.75373    0.10211  -7.381 1.57e-13 ***
hosp               -0.30669    0.09121  -3.363 0.000772 ***
numchron           -0.53972    0.04419 -12.212  < 2e-16 ***
school             -0.05560    0.01218  -4.564 5.02e-06 ***
```

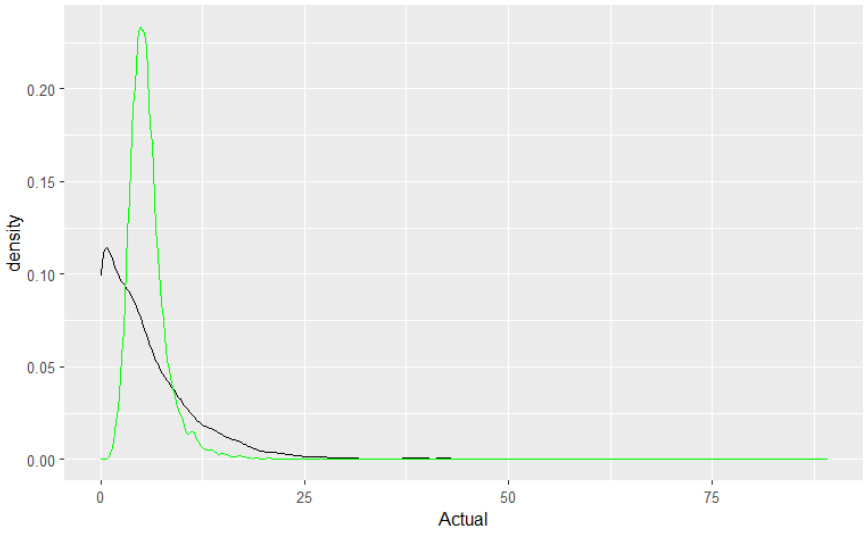
```
model.zpois2<-zeroinfl(ofp ~ factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='poisson')
```

Count model coefficients (poisson with log link):

```
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.405600   0.024179  58.134  < 2e-16 ***
factor(health)excellent -0.307366   0.031265  -9.831  < 2e-16 ***
factor(health)poor       0.253416   0.017706  14.313  < 2e-16 ***
factor(gender)male      -0.062352   0.013056  -4.776 1.79e-06 ***
factor(privins)yes       0.080533   0.017147   4.697 2.65e-06 ***
hosp                     0.159014   0.006060  26.240  < 2e-16 ***
numchron                 0.101846   0.004721  21.573  < 2e-16 ***
school                   0.019169   0.001873  10.232  < 2e-16 ***
```

```
Zero-inflation model coefficients (binomial with logit link):
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         -0.05937    0.14040  -0.423 0.672392
factor(gender)male   0.41806    0.08920   4.687 2.77e-06 ***
factor(privins)yes  -0.75373    0.10211  -7.381 1.57e-13 ***
hosp                -0.30669    0.09121  -3.363 0.000772 ***
numchron            -0.53972    0.04419 -12.212  < 2e-16 ***
school              -0.05560    0.01218  -4.564 5.02e-06 ***
```


Predicted
Actual
0
10
20
30
40
0
25
50
75


density
Actual
0.00
0.05
0.10
0.15
0.20
0
25
50
75

Zero-inflated Negative binomial

Zero-inflated Negative Binomial

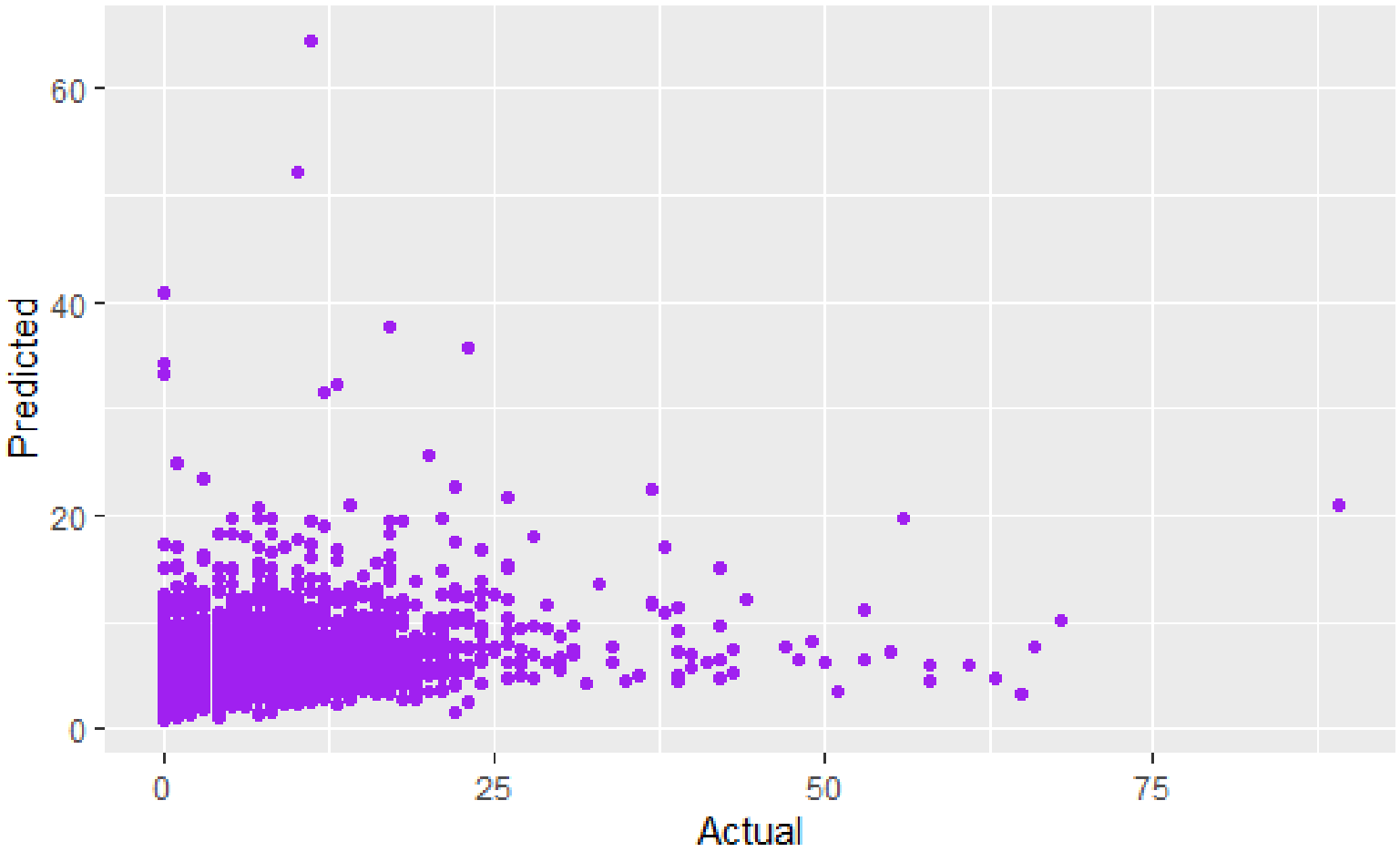
Same idea as zero-inflated Poisson, except we now have overdispersion

Can again use Log-Likelihood test to see if overdispersion is needed

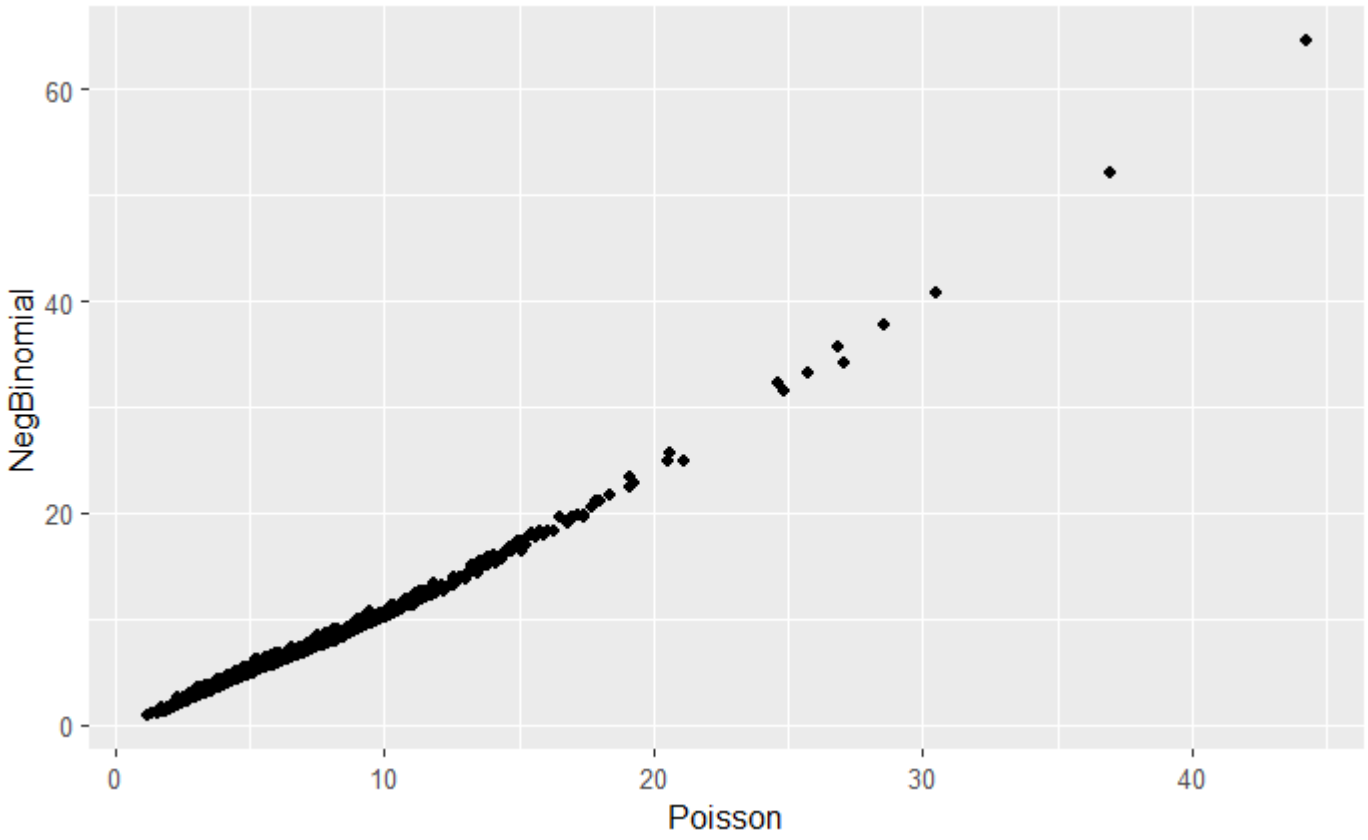
```
model.znbin2<-zeroinfl(ofp ~ factor(health) + factor(gender)+factor(privins)+hosp+numchron+school|
factor(gender)+factor(privins)+hosp+numchron+school,data=dat,dist='negbin')
```

Count model coefficients (negbin with log link):

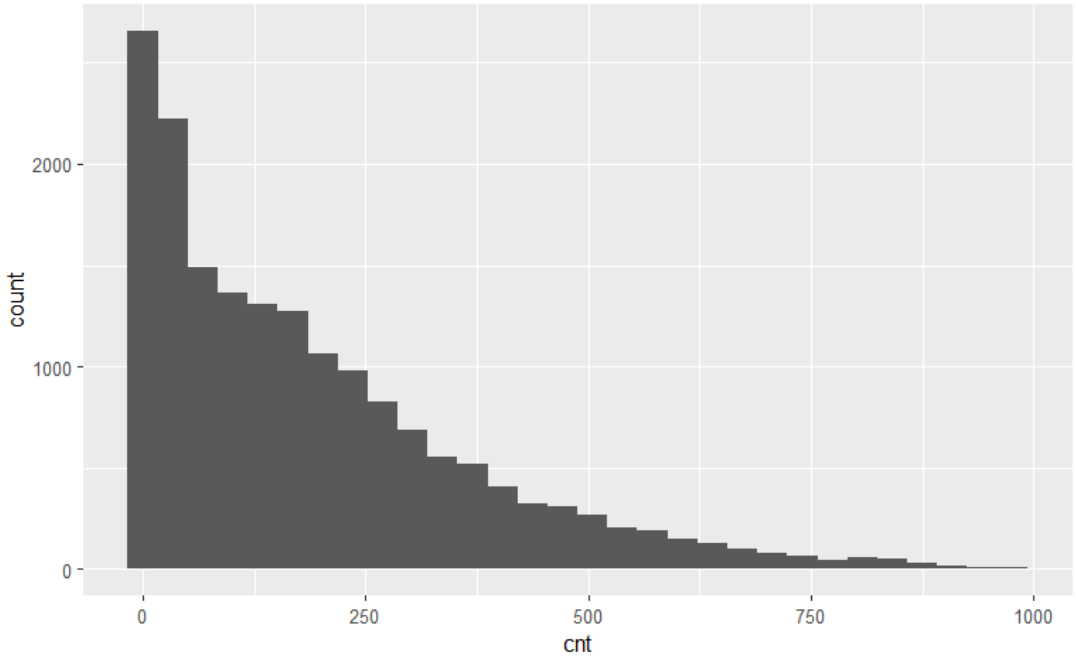
```
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.193716   0.056661  21.068  < 2e-16 ***
factor(health)excellent -0.319339   0.060405  -5.287 1.25e-07 ***
factor(health)poor       0.285133   0.045093   6.323 2.56e-10 ***
factor(gender)male      -0.080277   0.031024  -2.588  0.00967 **
factor(privins)yes       0.125865   0.041588   3.026  0.00247 **
hosp                     0.201477   0.020360   9.896  < 2e-16 ***
numchron                 0.128999   0.011931  10.813  < 2e-16 ***
school                   0.021423   0.004358   4.916 8.82e-07 ***
Log(theta)               0.394144   0.035035  11.250  < 2e-16 ***

Zero-inflation model coefficients (binomial with logit link):
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       -0.04684    0.26855  -0.174  0.86154
factor(gender)male  0.64766    0.20011   3.236  0.00121 **
factor(privins)yes -1.17558    0.22012  -5.341 9.26e-08 ***
hosp              -0.80046    0.42081  -1.902  0.05715 .
numchron          -1.24790    0.17831  -6.999 2.59e-12 ***
school            -0.08378    0.02625  -3.191  0.00142 **
```

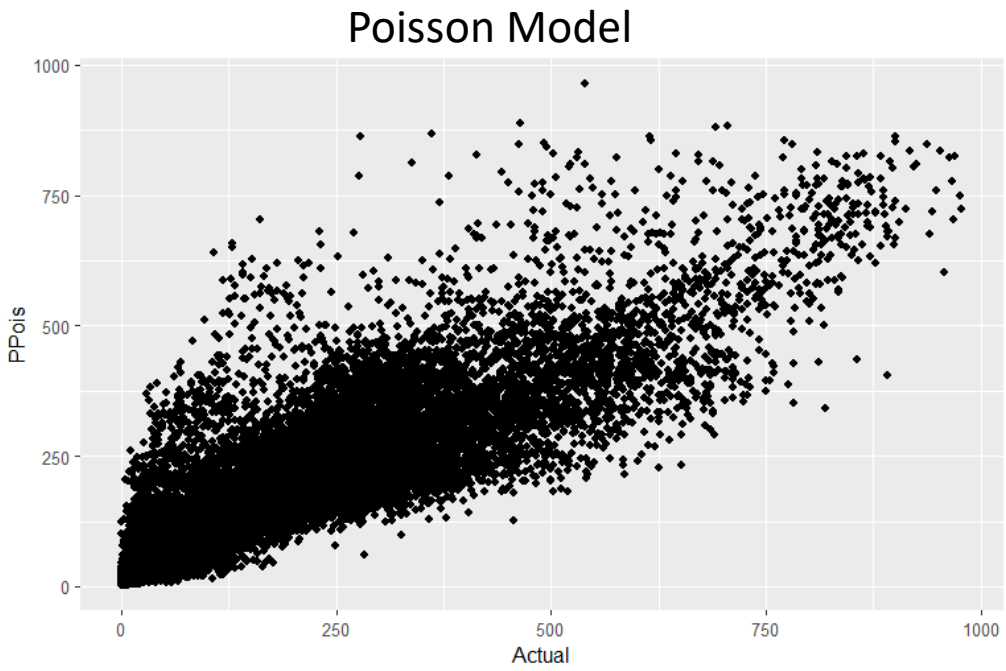

Predicted
60
40
20
0
0
25
50
75
Actual

How similar is the Poisson to the Negative Binomial…..



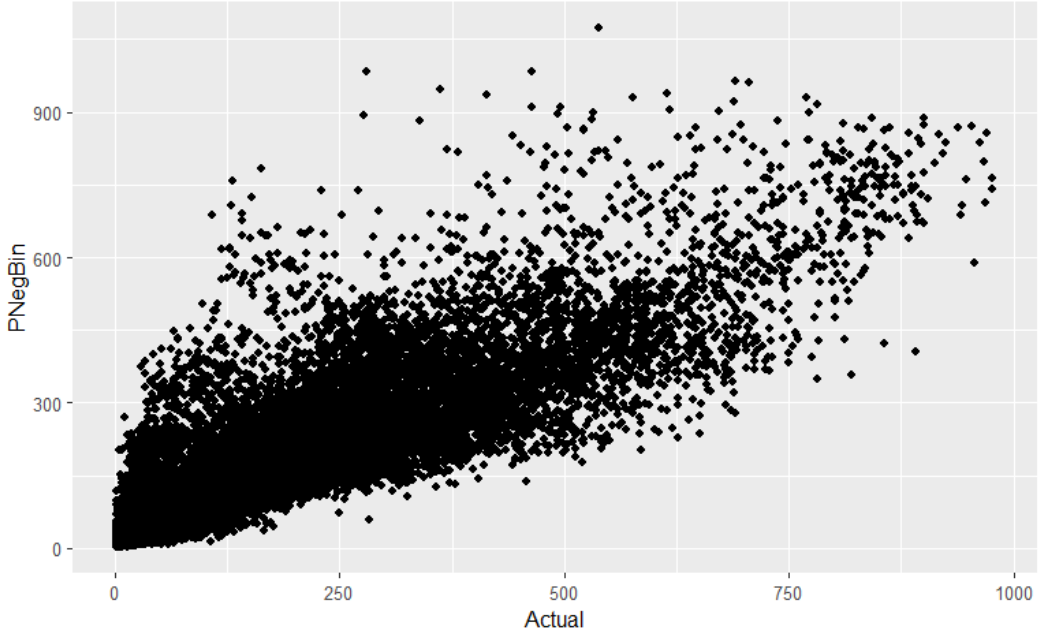
Bike data (from summer!)



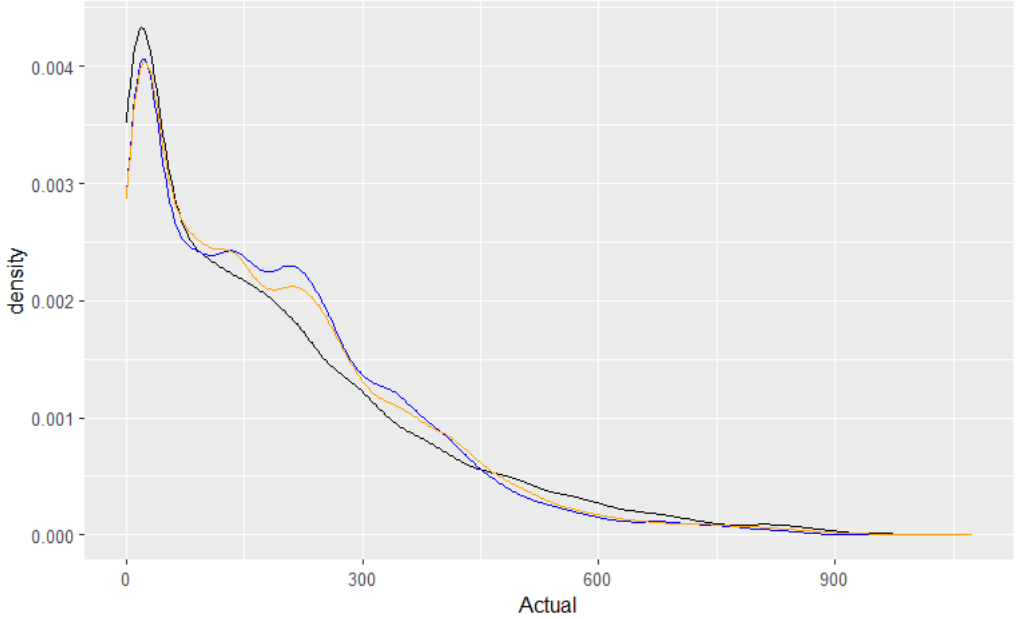


Bike data continued…

Negative Binomial model



Blue=Poisson
Orange=Negative Binomial



Thank you!
Happy counting….