Contents

1 Introduction

Introduction

In this lecture we discuss the problem of overdispersion in logistic and Poisson regression, and how to include it in the modeling process.

2 The Problem of Overdispersion

2.1 Relevant Distributional Characteristics

Distributional Characteristics

In models based on the normal distribution, the mean μ and variance σ^2 are mathematically independent. The variance σ^2 can, theoretically, take on any value relative to μ.

However, with binomial or Poisson distributions, means and variances are not independent. The binomial random variable X, the number of successes in N independent trials, has mean $\mu = Np$, and variance $\sigma^2 = Np(1-p) = (1-p)\mu$. The binomial sample proportion, $\hat{p} = X/N$, has mean p and variance $p(1-p)/N$.

The Poisson distribution has a variance equal to its mean, μ.

Distributional Characteristics

Consequently, if we observe a set of observations x_i that truly are realizations of a Poisson random variable X, these observations should show a sample variance that is reasonably close to their sample mean.

In a similar vein, if we observe a set of sample proportions $\hat{p}_i$, each based on N_i independent observations, and our model is that they all represent samples in a situation where p remains stable, then the variation of the $\hat{p}_i$ should be consistent with the formula $p(1-p)/N_i$.

2.2 Observing Overdispersion in Practice

Observing Overdispersion Overdispersed Proportions

There are numerous reasons why overdispersion can occur in practice. Let's consider sample proportions based on the binomial.

Suppose we hypothesize that the support enjoyed by President Obama is constant across 5 midwestern states. That is, the proportion of people in the populations of those states who would answer "Yes" to a particular question is constant.

We perform opinion polls by randomly sampling 200 people in each of the 5 states.

Observing Overdispersion Overdispersed Proportions

We observe the following results: Wisconsin 0.285, Michigan 0.565, Illinois 0.280, Iowa 0.605, Minnesota .765. An unbiased estimate of the average proportion in these states can be obtained by simply averaging the 5 proportions, since each was based on a sample of size $N = 200$.

Using R, we obtain:

```
> data ← c(0.285,0.565,0.280,0.605,.765)
> mean(data)
```

```
[1] 0.5
```

Observing Overdispersion Overdispersed Proportions

These proportions have a mean of 0.50. They also show considerable variability.

Is the variability of these proportions consistent with our binomial model, which states that they are all representative of a constant proportion p?

There are several ways we might approach this question, some involving brute force statistical simulation, others involving the use of statistical theory. Recall that sample proportions based on $N = 200$ *independent* observations should show a variance of $p(1-p)/N$. We can estimate this quantity in this case as

```
> 0.50*(1-0.50)/200
```

```
[1] 0.00125
```

Observing Overdispersion Overdispersed Proportions

On the other hand, these 5 sample proportions show a variance of

```
> var(data)
```

```
[1] 0.045025
```

The variance ratio is

```
> variance.ratio = var(data) / (0.50*(1-0.50)/200)
> variance.ratio
```

```
[1] 36.02
```

The variance of the proportions is 36.02 times as large as it should be. There are several statistical tests we could perform to assess whether this variance ratio is statistically significant, and they all reject the null hypothesis that the actual variance ratio is 1.

Observing Overdispersion Overdispersed Proportions

As an example, we could look at the residuals of the 5 sample proportions from their fitted value of .50. The residuals are:

```
> residuals ← data - mean(data)
> residuals
```

```
[1] -0.215  0.065 -0.220  0.105  0.265
```

Each residual can be converted to a standardized residual z-score by dividing by its estimated standard deviation.

```
> standardized.residuals ← residuals / sqrt(0.50*(1-0.50)/200)
```

We can then generate a χ^2 statistic by taking the sum of squared residuals. The statistic has the value

```
> chi.square ← sum(standardized.residuals^2)
> chi.square
```

```
[1] 144.08
```

Observing Overdispersion Overdispersed Proportions

We have to subtract one degree of freedom because we estimated p from the mean of the proportions. Our χ^2 statistic can be compared to the χ^2 distribution with 4 degrees of freedom. The 2-sided $p - value$ is

```
> 2*(1-pchisq(chi.square,4))
```

```
[1] 0
```

Observing Overdispersion Overdispersed Proportions

Our sample proportions show overdispersion. Why?

The simplest explanation in this case is that they are not samples from a population with a constant proportion p. That is, there is heterogeneity of support for Obama across these 5 states.

Can you think of another reason why a set of proportions might show overdispersion? (C.P.)

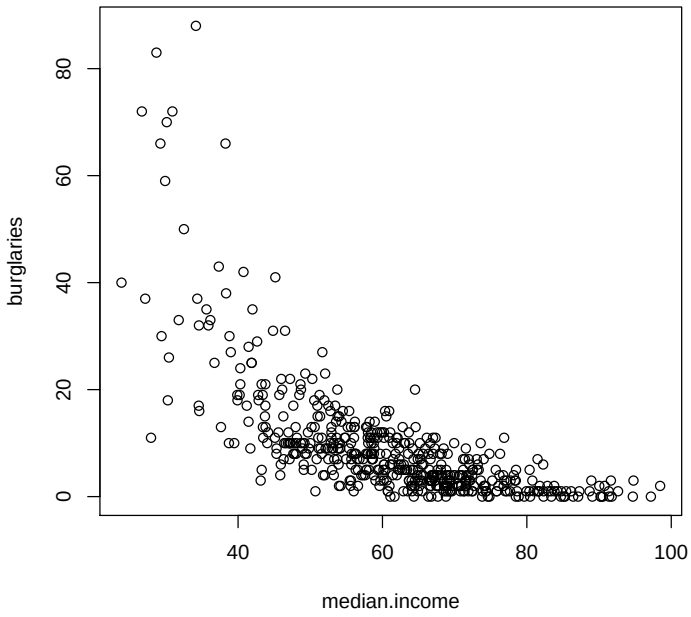
How about underdispersion? (C.P.)

Overdispersed Counts

Since counts are free to vary over the integers, they obviously can show a variance that is either substantially greater or less than their mean, and thereby show overdispersion or underdispersion relative to what is specified by the Poisson model.

As an example, suppose we examine the impact of the median income (in thousands) of families in a neighborhood on the number of burglaries per month. Load the *burglary.txt* data file, then plot `burglaries` as a function of `median.income`. These data represent burglary counts for 500 metropolitan and suburban neighborhoods.

```
> plot(median.income,burglaries)
```



Assessing Overdispersion

Let's examine some data for evidence of overdispersion. First, we'll grab scores corresponding to a `median.income` between 59 and 61.

```
> test.data ← burglaries[median.income > 59 & median.income < 61]
> var(test.data)
```

```
[1] 22.53846
```

```
> mean(test.data)
```

```
[1] 7.333333
```

```
> var(test.data) / mean(test.data)
```

```
[1] 3.073427
```

The variance for these data is more than 3 times as large as the mean.

Assessing Overdispersion

Let's try another region of the plot.

```
> test.data ← burglaries[median.income > 39 & median.income < 41]
> var(test.data)
```

```
[1] 97.14286
```

```
> mean(test.data)
```

```
[1] 21.85714
```

```
> var(test.data) / mean(test.data)
```

```
[1] 4.444444
```

Assessing Overdispersion

The data show clear evidence of overdispersion. Let's fit a standard Poisson model to the data.

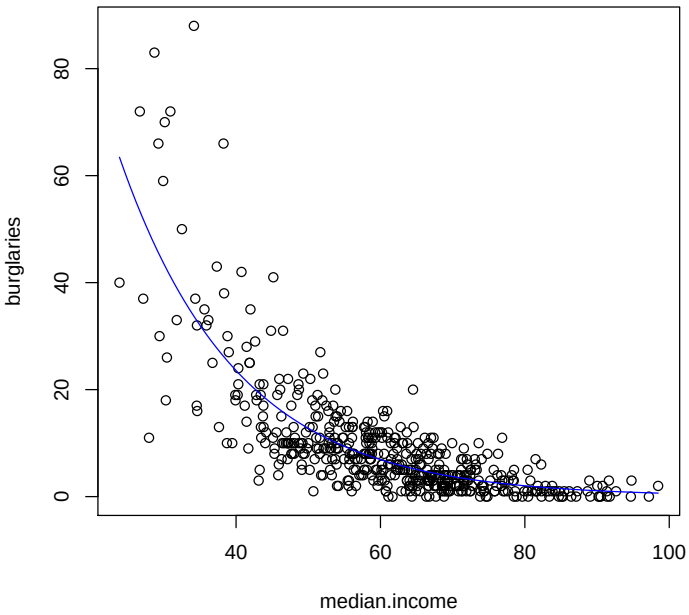
```
> standard.fit ← glm(burglaries ~ median.income, family = "poisson")
> summary(standard.fit)
```

```
Call:
glm(formula = burglaries ~ median.income, family = "poisson")

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-6.6106  -1.2794  -0.2884   0.9102   7.7649

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    5.612422   0.055996  100.23   <2e-16 ***
median.income -0.061316   0.001091  -56.19   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 4721.4  on 499  degrees of freedom
Residual deviance: 1452.6  on 498  degrees of freedom
AIC: 3196.4

Number of Fisher Scoring iterations: 5
```

Fitting the Overdispersed Poisson Model

```
> plot(median.income,burglaries)
> curve(exp(coef(standard.fit)[1] + coef(standard.fit)[2]*x),add=TRUE,col="blue")
```



The expected mean line, plotted with the coefficients from the model, looks like a nice fit to the data. However, the variance is several times the mean in this model, and since the standard errors are based on the assumption that the variance is equal to the mean, this creates a problem. The actual variance is several times what it should be, and so the standard errors printed by the program are underestimates.

Fitting the Overdispersed Poisson Model

It is not spelled out very clearly in Gelman & Hill , but there are two fairly standard ways of handling this in R. One way assumes simply that the conditional distribution is like the Poisson, but with the variance a constant multiple of the mean rather than being equal to the mean. This approach is used in `glm` by selecting `family="quasipoisson"`. Notice how the dispersion parameter is estimated, and the estimated standard errors from the Poisson fit are divided by the square root of this parameter to obtain the revised standard errors shown below.

```
> overdispersed.fit ← glm(burglaries ~ median.income,family="quasipoisson")
> summary(overdispersed.fit)
```

```
Call:
glm(formula = burglaries ~ median.income, family = "quasipoisson")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-6.6106  -1.2794  -0.2884   0.9102   7.7649

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    5.612422   0.096108   58.40   <2e-16 ***
median.income -0.061316   0.001873  -32.74   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 2.945783)

    Null deviance: 4721.4  on 499  degrees of freedom
Residual deviance: 1452.6  on 498  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

Fitting the Overdispersed Poisson Model

Another more sophisticated approach uses quasi-likelihood estimation to fit the negative binomial model, which assumes that the log means predicted from `median.income` are perturbed by random variation (having a gamma distribution). This random variation means that individual observations, for a given value of the predictors, can have different means, centered around $\boldsymbol{x}'\boldsymbol{\beta}$. This leaves the conditional mean line the same, but inflates the variance relative to that predicted by the Poisson. The variance inflation is not constant, however. In the negative binomial, there is an overdispersion parameter θ, but the variance and mean are related as follows:

$$\sigma^2 = \mu(1 + \mu/\theta) \tag{1}$$

Fitting the Overdispersed Poisson Model

We can fit the negative binomial model, using the MASS library function `glm.nb`. (Make sure the MASS library is loaded.)

```
> negative.binomial.fit <- glm.nb(burglaries ~ median.income)
> summary(negative.binomial.fit)
```

```
Call:
glm.nb(formula = burglaries ~ median.income, init.theta = 4.95678961145058,
    link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8813  -0.8490  -0.1922   0.6297   2.9637

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    5.57414    0.12042   46.29   <2e-16 ***
median.income -0.06060    0.00207  -29.27   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(4.9568) family taken to be 1)

    Null deviance: 1606.97  on 499  degrees of freedom
Residual deviance:  545.33  on 498  degrees of freedom
AIC: 2730.7

Number of Fisher Scoring iterations: 1


              Theta:  4.957
          Std. Err.:  0.550

 2 x log-likelihood:  -2724.713
```

Fitting the Overdispersed Poisson Model

In this case, the data were artificial. I created them according to the negative binomial model $\mu = -.06x + 5.5$, with overdispersion parameter $\theta = 5$.

As you can see, in this case `glm.nb` estimates were very close to the true values, and the χ^2 fit statistic of 545.33 fails to reach significance at the .05 level, meaning that the hypothesis of perfect fit cannot be rejected.

On the other hand, the `quasipoisson` family model fit, which assumes that the variance is a constant multiple of the mean, could not fit these data nearly as well. The deviance statistic of 1452.6 is much higher.

Fitting the Overdispersed Poisson Model

Consider an instructive case, when median.income is 30. In this case, the

mean and variance are actually

```
> m ← exp(-.06 * 30 + 5.5)
> v ← m * (1+m/5)
> m
```

```
[1] 40.44730
```

```
> v
```

```
[1] 367.6442
```

The `quasipoisson` fit estimates them as

```
> m ← exp(coef(overdispersed.fit)[1] + coef(overdispersed.fit)[2] * 30)
> v ← m * 2.945783
> m
```

```
(Intercept)
   43.50732
```

```
> v
```

```
(Intercept)
   128.1631
```