Accounting for Overdispersion

Plot Pearson residuals vs. fitted values

```r
nhat <- fitted( fit )
plot( nhat, presid^2, xlab="Fitted mean response",
     ylab="Squared Pearson residuals" )
abline( h=1, col="red", lwd=2 )
sfit <- loess( presid^2 ~ nhat )
lines( sort(sfit$x), sfit$fitted[order(sfit$x)], col="blue", lwd=2 )
abline(h=phihat, lty=2, col="red", lwd=2)
```

10 20 30 40
0 1 2 3 4 5 6

Fitted mean response
Squared Pearson residuals

Again, it looks as though the smoother is consistently above the y=1 line, indicating overdispersion.
Overdispersion in GLMs

Definition

- **Overdispersion** describes the situation when the actual $\text{Var}(Y_i)$ exceeds the specified GLM variance $\text{V}(\mu_i)$

- Examples:
  - For (grouped) "binomial" data with $Y_i = \sum_{j=1}^{n_i} Y_{ij}$, $Y_{ij} \sim \text{Bernoulli}(\mu_i)$, we may have that $\text{Var}(Y_i) > n_i \mu_i (1 - \mu_i)$
  - For count data we may have that $\text{Var}(Y_i) > \mu_i$
Overdispersion in GLMs

How does overdispersion arise?

- In (grouped) "binomial" data, response $Y_i$ is

$$Y_i = \sum_{j=1}^{n_i} Y_{ij},$$

the total number of success in $n_i$ Bernoulli trials

- In "Poisson" count data, $Y_i$ is the number of events observed over a set of exposure intervals

- In both cases, independence is assumed
  - Among trials in the binomial case
  - Among small intervals in the Poisson case

- Assumption is that trials in the same group of $n_i$ trials are no more alike one another than trials in different groups
### Example 1: Correlated binary outcomes

- Suppose that $\text{corr}(Y_{ij}, Y_{ik}) = \rho$ and that $E(Y_{ij}) = \mu_i$
- Then $\text{Var}(Y_{ij}) = \mu_i(1 - \mu_i)$
- Now, consider the mean and variance of $Y_i = \sum_{j=1}^{n_i} Y_{ij}$
## Overdispersion in GLMs

### Example 1: Correlated binary outcomes

- Conclusions:
  1. \( E(Y_i) = n_i \mu_i \)
  2. \( \text{Var}(Y_i) = n_i \mu_i (1 - \mu_i) \{1 + (n_i - 1) \rho \} \)

- The implications of this are
  1. If \( \rho > 0 \), \( \text{Var}(Y_i) \) is larger than the binomial variance
  2. If \( \rho = 0 \), \( \text{Var}(Y_i) \) has binomial variance
  3. If \( \rho < 0 \), \( \text{Var}(Y_i) \) is smaller than the binomial variance
Example 2: Population heterogeneity in "Poisson" count data

- Suppose there exists a binary covariate, $Z_i$ with $\Pr[Z_i = 1] = \pi$, and that

$$Y_i | Z_i = 0 \sim \text{Poisson}(\lambda_0)$$
$$Y_i | Z_i = 1 \sim \text{Poisson}(\lambda_1)$$

- Then the mean and variance of $Y_i$ are given by
Overdispersion in GLMs

Example 2: Population heterogeneity in "Poisson" count data

- Conclusions:
  
  1. \( \text{E}(Y_i) = \pi \lambda_1 + (1 - \pi)\lambda_0 = \mu \)
  
  2. \( \text{Var}(Y_i) = \mu + (\lambda_1 - \lambda_0)^2 \pi (1 - \pi) \)

- Therefore, if we do not observe \( Z_i \) then this omitted factor leads to increased variation!

- Thus overdispersion can arise when groups or observations are not independent
  
  - This is often due to lack of adjustment for (possibly unmeasured) covariates
Overdispersion in GLMs

Implications of overdispersion?

► What are the implications of overdispersion?

► If overdispersion is not accounted for in modeling, inferences could be wrong

► Incorrect standard errors
► Invalid hypothesis tests
► Invalid confidence intervals

► Often, a large deviance (for goodness-of-fit tests) that cannot be accounted for by other lack of model fit may be evidence of overdispersion (recall the melanoma data)
Overdispersion in GLMs

Notes on overdispersion

1. Overdispersion does not (cannot) occur with independent binary observations, so it is usually a problem in other models (eg. count data)

2. Underdispersion is not seen very often (eg. when there is negative correlation within groups of binomial data)
Detecting Overdispersion

Pearson residuals

- Suppose we assume a variance function $V(\mu)$
  - If we believe we have (truly) binomial data we would assume $V(\mu_i) = n_i\mu_i(1 - \mu_i)$
  - If we believe we have (truly) Poisson data we would assume $V(\mu_i) = \mu_i$

- If the model (mean model and variance function) is correct, then
  $$\text{Var}(Y_i) \approx \text{Var}(Y_i - \hat{\mu}_i) \approx V(\hat{\mu}_i)(1 - h_{ii})$$

  or
  $$\text{Var}(e_{i,P}) = \text{Var}\left(\frac{Y_i - \hat{\mu}_i}{\sqrt{\text{Var}(\hat{\mu}_i)(1 - h_{ii})}}\right) \approx 1$$
Detecting Overdispersion

Pearson residuals

• However, if the data are overdispersed, such that $V^*(\mu_i) = \phi V(\mu_i)$, then

$$\text{Var}(e_{i,P}) = \text{Var} \left( \frac{Y_i - \hat{\mu}_i}{\sqrt{\text{Var}(\hat{\mu}_i)(1 - h_{ii})}} \right)$$

$$\approx \frac{\phi V(\mu_i)(1 - h_{ii})}{V(\mu_i)(1 - h_{ii})} = \phi$$
Detecting Overdispersion

Pearson residuals

- Therefore the expected value of the squared Pearson residual is

\[
E(e_{i,P}^2) = \text{Var}(e_{i,P}) + [E(e_{i,P})]^2
\]

\[
= \text{Var}(e_{i,P}) + 0 \quad \text{(correct mean model)}
\]

\[
\approx \phi
\]

- So, as we have seen before, we can plot (and smooth) the squared Pearson residuals against the fitted outcomes, \(\hat{\mu}_i\), to examine evidence of overdispersion.
Detecting Overdispersion

Pearson residuals

▶ In addition, the Pearson Chi-square statistic given by

\[ \chi^2_P = \sum_{i=1}^{n} e_{i,P}^2, \sim \chi^2_{n-p} \]

under no overdispersion (\( \phi = 1 \)), where \( p \) is the number of parameters in the mean model

▶ Therefore, an estimate of the dispersion parameter \( \phi \) is given by

\[ \hat{\phi} = \sum_{i=1}^{n} e_{i,P}^2/(n - p) \]
Accounting for Overdispersion

Pearson residuals

▶ If overdispersion exists, then all parameter standard error estimates will be off by a factor of $\phi^{1/2}$

▶ Thus, one natural way to adjust for overdispersion is to scale all standard errors by the square-root of the estimated overdispersion parameter $\hat{\phi}^{1/2}$
Accounting for Overdispersion

Example

Example: Correlated binary outcomes

- For the binomial model, we assume $V(\mu_i) = n_i \mu_i(1 - \mu_i)$
- But actually, $V^*(\mu_i) = \text{Var}(Y_i) = n_i \mu_i(1 - \mu_i)\{1 + (n_i - 1)\rho\}$
- If the total number of observations (and correlation) is roughly constant across all groups, the the value $1 + (n_i - 1)\rho$ will be approximately constant across all groups
- Thus we could propose a simpler variance model and assume

  $$\text{Var}(Y_i) = \phi n_i \mu_i(1 - \mu_i)$$

- Then all standard errors in this model could be scaled using $\hat{\phi}$ from above
## Accounting for Overdispersion

### Example 1: Beetle Data

- Study designed to test the effectiveness of ethylene oxide (EO) as a fumigant against the grain beetle
- Batches of about 30 beetles were exposed to various concentrations of EO for 1 hour
- Groups were then placed in glass tubes with food
- After several days, the number of affected (dead) beetles were counted
- Possible source for overdispersion?
Accounting for Overdispersion

Example 1: Beetle Data

▶ A quick look at the data...

```r
> source("http://www.ics.uci.edu/~dgillen/Stat211/Handouts/Stat211Functions.q")
> beetles <- read.table("http://www.ics.uci.edu/~dgillen/Stat211/Data/beetles.txt")
> beetles
   concentration affected exposed
1        24.8       23       30
2        24.6       30       30
3        23.0       29       31
4        21.0       22       30
5        20.6       23       26
6        18.2        7       27
7        16.8       12       31
8        15.8       17       30
9        14.7       10       31
10       10.8        0       24
```

• Again, it looks as though the smoother is consistently above the y=1 line, indicating overdispersion.
Accounting for Overdispersion

Example 1: Beetle Data

First, let’s start by fitting a basic logistic regression model adjusting for (log) concentration

> fit1 <- glm( affected/exposed ~ I( log(concentration) ),
   data=beetles, family=binomial, weights=exposed )
> summary( fit1 )

Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| (Intercept) | -17.8670    | 2.2690  | -7.874   | 3.43e-15 *** |
| I(log(concentration)) | 6.2654     | 0.7781  | 8.052    | 8.15e-16 *** |

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 138.001 on 9 degrees of freedom
Residual deviance: 36.444 on 8 degrees of freedom
AIC: 68.017
Accounting for Overdispersion

Example 1: Beetle Data

- Plot fitted vs observed probabilities

```r
> #
> ##### Plot observed vs. fitted logit
> #
> etaobs <- log( (beetles$affected+.5) / (beetles$exposed-beetles$affected+.5) )
> etahat <- predict( fit1 )
> plot( log( beetles$concentration ), etaobs,
>       xlab="log( Concentration )", ylab="Observed/Fitted Logit" )
> lines( log( beetles$concentration ), etahat, lwd=2, col="red" )
```
A look the model fit

\begin{itemize}
  \item Again, it looks as though the smoother is consistently above the \( y=1 \) line, indicating overdispersion
\end{itemize}
Accounting for Overdispersion

Example 1: Beetle Data

- Conclusions

- Relatively good fit...
  - General trend is captured
  - No outliers
  - No influential points
Accounting for Overdispersion

Example 1: Beetle Data

Let's consider the deviance goodness-of-fit test

```r
> ##
> ##### Deviance GOF test
> ##
> beetles <- beetles[ order(beetles$concentration), ]
> beetles$grp <- 1:10
> beetles

<table>
<thead>
<tr>
<th>concentration</th>
<th>affected</th>
<th>exposed</th>
<th>grp</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10.8</td>
<td>0</td>
<td>24</td>
</tr>
<tr>
<td>9</td>
<td>14.7</td>
<td>10</td>
<td>31</td>
</tr>
<tr>
<td>8</td>
<td>15.8</td>
<td>17</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>16.8</td>
<td>12</td>
<td>31</td>
</tr>
<tr>
<td>6</td>
<td>18.2</td>
<td>7</td>
<td>27</td>
</tr>
<tr>
<td>5</td>
<td>20.6</td>
<td>23</td>
<td>26</td>
</tr>
<tr>
<td>4</td>
<td>21.0</td>
<td>22</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>23.0</td>
<td>29</td>
<td>31</td>
</tr>
<tr>
<td>2</td>
<td>24.6</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>1</td>
<td>24.8</td>
<td>23</td>
<td>30</td>
</tr>
</tbody>
</table>
```

- Again, it looks as though the smoother is consistently above the y=1 line, indicating overdispersion.
Accounting for Overdispersion

Example 1: Beetle Data

First fit the saturated model

> fit.sat <- glm( affected/exposed ~ factor( grp ), data=beetles,
>                  family=binomial, weights=exposed )
> summary( fit.sat )

Deviance Residuals:
  [1] 0 0 0 0 0 0 0 0 0 0

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   -25.92  52727.26  -0.000492    1
factor(grp)2  25.18  52727.26   0.000478    1
factor(grp)3  26.19  52727.26   0.000497    1
factor(grp)4  25.46  52727.26   0.000483    1
factor(grp)5  24.87  52727.26   0.000472    1
factor(grp)6  27.96  52727.26    0.001     1
factor(grp)7  26.94  52727.26    0.001     1
factor(grp)8  28.60  52727.26    0.001     1
factor(grp)9  52.06  74375.59    0.001     1
factor(grp)10 27.11  52727.26    0.001     1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1.3800e+02 on 9 degrees of freedom
Residual deviance: 5.3207e-10 on 0 degrees of freedom
AIC: 47.573

Number of Fisher Scoring iterations: 22
Accounting for Overdispersion

Notes on this fit

- The parameter estimates appear to be unstable
  - This is because we have two groups with 0 or all ‘successes’
  - Can’t trust the parameter estimates, but can we still perform a goodness-of-fit test?
Accounting for Overdispersion

Aside: Goodness-of-fit test with 0 or all successes

- As we have seen, when we fit the saturated model, problems arise:
  1. Some observations are dropped
  2. Some parameter estimates are unstable

- However, for a goodness-of-fit test, as long as the current model does not also suffer from these problems, we can still handle this

- Recall that the log-likelihood contribution for grouped binomial data is given by

\[
\log(L(y; \mu)) = \sum_{i=1}^{N} \left[ \log \left( \frac{n_i}{r_i} \right) + r_i \log(\hat{\mu}_i) + (n_i - r_i) \log(1 - \hat{\mu}_i) \right]
\]
Accounting for Overdispersion

Aside: Goodness-of-fit test with 0 or all successes

- Thus, the contribution to the likelihood for groups with 0 or all successes is 0

- Therefore, if these observations are dropped (in the saturated model), the likelihood computed without them will be the same as that with them in the saturated model

- RESULT: We can trust the likelihood reported for the saturated model
Accounting for Overdispersion

Back to the GOF test

- Perform the deviance (LR test) in R using my defined `lrtest` function:

```r
> lrtest(fit1, fit.sat)
Assumption: Model 1 nested within Model 2

<table>
<thead>
<tr>
<th>Resid. Df</th>
<th>Resid. Dev</th>
<th>Dev Df</th>
<th>Deviance</th>
<th>pValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>36.444</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.000</td>
<td>8</td>
<td>36.444</td>
</tr>
</tbody>
</table>
```
Accounting for Overdispersion

Notes on the GOF test

- Why 8 degrees of freedom?
  - Recall that the degrees of freedom for the Deviance test is $N - p$, where $N$=# binomial observations, $p$=# parameters in the current model

- Conclusion: Deviance of 36.44 on 8 degrees of freedom...bad lack of fit?!?
Accounting for Overdispersion

Investigation of overdispersion

- Or, is it just overdispersion? Let’s investigate...

```r
> ##
> ######  Fit the original model
> ##
> fit1 <- glm( affected/exposed ~ I( log(concentration) ), data=beetles,
> family=binomial, weights=exposed )
> summary( fit1 )

Deviance Residuals:
  Min 1Q Median 3Q Max
-3.3512 -1.3534 0.1293 1.6558 2.5115

Coefficients:    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -17.8670   2.2690  -7.874 3.43e-15 ***
I(log(concentration)) 6.2654   0.7781   8.052 8.15e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

(Dispersion parameter for binomial family taken to be 1)

> ##
> ###### Calculate Pearson’s estimate of phi
> ##
> presid <- residuals( fit1, type="pearson" )
> pChi2 <- sum( presid^2 )
> phihat <- pChi2 / (10-2)

> cbind( pChi2, phihat )
pChi2   phihat
[1,] 33.43404 4.179256
```
Investigation of overdispersion

- Thus we estimate that the variance is about 4-times larger than what we would expect if the data were truly binomial.

- Let's try to refit the model accounting for overdispersion, assuming the variance function

\[ V(\mu_i) = \phi n_i \mu_i (1 - \mu_i) \]
Accounting for Overdispersion

Accounting for overdispersion via quasi-likelihood

- To do this in R, we use the family `quasibinomial`:

```r
> ##
> ##### Fit the model accounting for overdispersion (quasi-likelihood)
> ##
> fit.quasi <- glm( affected/exposed ~ I( log(concentration) ),
>                   data=beetles, family=quasibinomial, weights=exposed )
> summary( fit.quasi )

Deviance Residuals:
Min 1Q Median 3Q Max
-3.3512 -1.3534 0.1293 1.6558 2.5115

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.867 4.639 -3.852 0.00486 **
I(log(concentration)) 6.265 1.591 3.939 0.00430 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

(Dispersion parameter for quasibinomial family taken to be 4.179255)

Null deviance: 138.001 on 9 degrees of freedom
Residual deviance: 36.444 on 8 degrees of freedom
AIC: NA

*Note that 1.591 = \sqrt{4.179256} \times .7781
Accounting for Overdispersion

Example 2: Melanoma data

- Recall: We were fitting the rate of melanoma events as a function of age and latitude

```r
> ##
> ##### Read in the melanoma data
> ##
> ##
> gail <- read.table( "/Volumes/LACIE/Teaching/Stat211/Data/gail.txt" )
```
Accounting for Overdispersion

Example 2: Melanoma data

- Our analysis with age and latitude included as indicator variables was:

```r
> #
> ##### Read in the melanoma data
> #
> gail <- read.table("/Volumes/LACIE/Teaching/Stat211/Data/gail.txt")
> #
> ##### Fit factored latitude model
> #
> gail$ageg <- relevel(gail$ageg, ref="<35 years")
> gail$latitude <- relevel(gail$latitude, ref="Northern")
> fit <- glm(inccases ~ factor(ageg) + factor(latitude), data=gail,
>          family=poisson, offset=log(persyrs))
> summary(fit)
```

Coefficients:

```
                         Estimate Std. Error   z value  Pr(>|z|)  
(Intercept)            -10.53053   0.07922  -132.924 < 2e-16 ***
factor(ageg)35-44 years  1.66230   0.09798   16.966 < 2e-16 ***
factor(ageg)45-54 years  1.87735   0.09359   20.059 < 2e-16 ***
factor(ageg)55-64 years  2.04075   0.09694   21.051 < 2e-16 ***
factor(ageg)65-74 years  2.18933   0.10796   20.279 < 2e-16 ***
factor(ageg)>=75 years   2.70761   0.10983   24.654 < 2e-16 ***
factor(latitude)Middle   0.52285   0.06924   7.551  4.32e-14 ***
factor(latitude)Southern 0.80803   0.07091  11.394  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 1
(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1196.49 on 53 degrees of freedom
Residual deviance:  67.63 on 46 degrees of freedom
```
Accounting for Overdispersion

Example 2: Melanoma data

- In the above analysis, we are assuming $V(\mu_i) = \mu_i$
- It is not uncommon (common?) to find overdispersion in such count
- Let’s consider the Pearson statistic and a diagnostic plot of the squared Pearson residuals versus the fitted counts

```
> ##
> ##### Pearson’s scale
> ##
> presid <- residuals( fit, type="pearson" )
> pChi2 <- sum( presid^2 )
> pihat <- pChi2 / fit$df.residual
>
> cbind( pChi2, pihat )
    pChi2    pihat
[1,] 65.65801 1.427348
```
Accounting for Overdispersion

Example 2: Melanoma data

- Notice that the Pearson statistic is 1.47 indicating some overdispersion.
- What about the squared residual plot...

```r
> ##
> ##### Plot Pearson residuals vs. fitted values
> ##
> nhat <- fitted( fit )
> plot( nhat, presid^2, xlab="Fitted mean response",
> ylab="Squared Pearson residuals" )
> abline( h=1, col="red", lwd=2 )
> sfit <- loess( presid^2 ~ nhat )
> lines( sort(sfit$x), sfit$fitted[order(sfit$x)], col="blue", lwd=2 )
> abline(h=phihat, lty=2, col="red", lwd=2)
```
Pearson Residual Plot for Melanoma Data

- Again, it looks as though the smoother is consistently above the y=1 line, indicating overdispersion.
Accounting for Overdispersion

Example 2: Melanoma data

Let’s fit the model accounting for overdispersion (use family=quasipoisson)

```r
> ##
> ##### Fit quasi-Poisson model
> ##
> fit1.quasi <- glm( inccases ~ factor( ageg ) + factor( latitude ),
>                    data=gail, family=quasipoisson, offset=log(persyrs) )
> summary( fit1.quasi )

Deviance Residuals:
    Min      1Q  Median      3Q     Max
-2.8244  -0.6210  -0.1859   0.6028   2.4101

Coefficients:          Estimate Std. Error t value Pr(>|t|)
(Intercept)          -10.53053   0.09465  -111.260  < 2e-16 ***
factor(ageg)35-44 years  1.66230   0.11705    14.201  < 2e-16 ***
factor(ageg)45-54 years  1.87735   0.11182    16.790  < 2e-16 ***
factor(ageg)55-64 years  2.04075   0.11582    17.620  < 2e-16 ***
factor(ageg)65-74 years  2.18933   0.12898    16.974  < 2e-16 ***
factor(ageg)><=75 years  2.70761   0.13121    20.636  < 2e-16 ***
factor(latitude)Middle  0.52285   0.08272     6.320   9.57e-08 ***
factor(latitude)Southern 0.80803   0.08472    9.537   1.80e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for quasipoisson family taken to be 1.427348)

Null deviance: 1196.49  on 53  degrees of freedom
Residual deviance: 67.63  on 46  degrees of freedom
```
Accounting for Overdispersion

Notes on the last fit

▶ Notice that the model fit is exactly the same, but that the standard errors are simply rescaled by $\hat{\phi}^{1/2}$

▶ Doesn’t make much of a difference in this model
Accounting for Overdispersion

### Likelihood ratio tests

- What about performing likelihood ratio tests?
  - Need to account for the scale parameter $\hat{\phi}$!
  - This includes GOF tests based upon the deviance!

- Use a LRT to test the effect of latitude
  - First, fit the reduced model

```r
> #
> #### Fit reduced age model
> #
> fit2.quasi <- glm( inccases ~ factor( ageg ), data=gail,
>                   family=quasipoisson, offset=log(persyrs) )
> summary( fit2.quasi )

Coefficients:                Estimate Std. Error t value Pr(>|t|)
(Intercept)                  -10.1746    0.1499  -67.889  < 2e-16 ***
factor(ageg)35-44 years      1.6684     0.2092   7.975  2.40e-10 ***
factor(ageg)45-54 years      1.8630     0.1998   9.323  2.39e-12 ***
factor(ageg)55-64 years      2.0124     0.2069   9.725  6.25e-13 ***
factor(ageg)65-74 years      2.1453     0.2304   9.313  2.47e-12 ***
factor(ageg)75+ years        2.6421     0.2341  11.287  4.16e-15 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

(Dispersion parameter for quasipoisson family taken to be 4.559725)

Null deviance: 1196.49 on 53 degrees of freedom
Residual deviance: 205.64 on 48 degrees of freedom
```
Accounting for Overdispersion

Notes on the last fit

► Now perform the LRT...

```r
> lrtest( fit2.quasi, fit1.quasi )

Assumption: Model 1 nested within Model 2

Resid. Df Resid. Dev Df Deviance pValue
1    48  205.641
2    46   67.630  2 138.011  0
```
Accounting for Overdispersion

Notes on the last fit

- The test is comparing the unscaled likelihood values, which overstates the test statistic because it does not account for overdispersion.

- To account for overdispersion, we need to divide the LRT statistic by $\hat{\phi}$.

```r
> # Rescale the LRT statistic
> lrStat <- 138.011 / phihat
> lrStat
[1] 96.6905

> pchisq( lrStat, df=2, lower.tail=FALSE )
[1] 1.009077e-21
```
Accounting for Overdispersion

Notes and conclusions on the LRT

- **Conclusion:**

- **Note:** We use the estimated scale parameter from the largest model being entertained! (why?)
  - When performing a deviance GOF test, use the scale parameter estimated from the current model
Accounting for Overdispersion

Robust variance estimator

- Question: What if a simple scale parameter doesn’t adequately correct the variance function mispecification
  - Different group sizes in the correlated binary data case
  - The missing covariate count data case

- What could we do in linear regression using weighted least squares if our weights weren’t ‘correct’?
## Accounting for Overdispersion

### Robust variance estimator for the melanoma data

- Let's compare the Poisson, scaled, and robust standard errors for the melanoma data

- First the Poisson (model-based) fit

```r
# Poisson model

fit <- glm( inccases ~ factor( ageg ) + factor( latitude ), data=gail, 
            family=poisson, offset=log(persyrs) )

summary( fit )
```

### Deviance Residuals:

```
Min 1Q Median 3Q Max
-2.8244 -0.6210 -0.1859 0.6028 2.4101
```

### Coefficients:

```
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -10.53053  0.07922 -132.924  < 2e-16 ***
factor(ageg)35-44 years 1.66230  0.09798   16.966  < 2e-16 ***
factor(ageg)45-54 years 1.87735  0.09359   20.059  < 2e-16 ***
factor(ageg)55-64 years 2.04075  0.09694   21.051  < 2e-16 ***
factor(ageg)65-74 years 2.18933  0.10796   20.279  < 2e-16 ***
factor(ageg)>75 years 2.70761  0.10983   24.654  < 2e-16 ***
factor(latitude)Middle 0.52285  0.06924    7.551 4.32e-14 ***
factor(latitude)Southern 0.80803  0.07091   11.394  < 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: 1196.49  on 53  degrees of freedom
Residual deviance: 67.63  on 46  degrees of freedom
Accounting for Overdispersion

Robust variance estimator for the melanoma data

> Now the scaled (quasi-likelihood) fit

> #
> #
> fit1.quasi <- glm( inccases ~ factor( ageg ) + factor( latitude ),
>                   data=gail, family=quasipoisson, offset=log(persyrs) )
> summary( fit1.quasi )

Deviance Residuals:
Min 1Q Median 3Q Max
-2.8244 -0.6210 -0.1859 0.6028 2.4101

Coefficients:

|                | Estimate | Std. Error | t value | Pr(>|t|) |
|----------------|----------|------------|---------|----------|
| (Intercept)    | -10.53053| 0.09465    | -111.260| < 2e-16 *** |
| factor(ageg)35-44 years | 1.66230 | 0.11705    | 14.201  | < 2e-16 *** |
| factor(ageg)45-54 years  | 1.87735 | 0.11182    | 16.790  | < 2e-16 *** |
| factor(ageg)55-64 years  | 2.04075 | 0.11582    | 17.620  | < 2e-16 *** |
| factor(ageg)65-74 years  | 2.18933 | 0.12898    | 16.974  | < 2e-16 *** |
| factor(ageg)75+ years   | 2.70761 | 0.13121    | 20.636  | < 2e-16 *** |
| factor(latitude)Middle   | 0.52285 | 0.08272    | 6.320   | 9.57e-08 *** |
| factor(latitude)Southern | 0.80803 | 0.08472    | 9.537   | 1.80e-12 *** |

---

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for quasipoisson family taken to be 1.427348)

Null deviance: 1196.49 on 53 degrees of freedom
Residual deviance: 67.63 on 46 degrees of freedom
Accounting for Overdispersion

Robust variance estimator for the melanoma data

To calculate the model robust (sandwich) variance estimate, we’ll need a function:

```r
##
## Function to compute robust se for glms
##
robust.se.glm <- function(glm.obj){
  ## Compute robust (sandwich) variance estimate
  if (is.matrix(glm.obj$x))
    xmat <- glm.obj$x
  else {
    mf <- model.frame(glm.obj)
    xmat <- model.matrix(terms(glm.obj), mf)
  }
  umat <- residuals(glm.obj, "working") * glm.obj$weights * xmat
  modelv <- summary(glm.obj)$cov.unscaled
  robust.cov <- modelv %*% t(umat) %*% umat %*% modelv

  ## Format the model output with p-values and CIs
  s <- summary(glm.obj)
  robust.se <- sqrt(diag(robust.cov))
  z <- glm.obj$coefficients/robust.se
  p <- 2*pnorm(-abs(z))
  ci95.lo <- glm.obj$coefficients - qnorm(.975) * robust.se
  ci95.hi <- glm.obj$coefficients + qnorm(.975) * robust.se
  rslt <- cbind(glm.obj$coefficients, robust.se, z, p, ci95.lo, ci95.hi)
  dimnames(rslt)[[2]] <- c("Robust SE", "ci95.lo", "ci95.hi")
  dimnames(rslt)[[2]][2] <- "Robust SE"
  rslt
}
```
Accounting for Overdispersion

Robust variance estimator for the melanoma data

Now, feed in the original Poisson model to compute the robust (sandwich) variance estimates:

```r
> robust.se.glm( fit )

Estimate Robust SE   z value Pr(>|z|) ci95.lo  ci95.hi
(Intercept)   -10.5305310 0.09605610 -109.628961 0.000000e+00 -10.7187975 -10.3422645
(ageg)35-44  1.6623027  0.10861721  15.304230  7.164471e-53  1.4494169  1.8751885
(ageg)45-54  1.8773496  0.10779888  17.415299  6.315809e-68  1.6660677  2.0886316
(ageg)55-64  2.0407484  0.12382243  16.481250  5.003928e-61  1.7980609  2.2834359
(ageg)65-74  2.1893285  0.12384816  17.677521  6.248370e-70  1.9465906  2.4320665
(ageg)>=75   2.7076059  0.14693347  18.427428  7.915738e-76  2.4196216  2.9955902
I(Middle)    0.5228485  0.08500064  6.151113  7.694129e-10  0.3562503  0.6894467
I(Southern)  0.8080259  0.07238818 11.162400  6.228668e-29  0.6661477  0.9499041
```

I(Middle)    0.5228485  0.08500064  6.151113  7.694129e-10  0.3562503  0.6894467
I(Southern)  0.8080259  0.07238818 11.162400  6.228668e-29  0.6661477  0.9499041
Accounting for Overdispersion

Pros and cons of robust se vs. quasi-likelihood

- The pros of using the robust standard error are that it does not assume a simple overdispersion variance function and will yield consistent standard errors.

- The cons of using the robust standard error are that it can be inefficient (relative to using the ‘correct’ variance function and that it requires a relatively large sample size for stability.
Accounting for Overdispersion

General Comments

1. Accounting for overdispersion is necessary to obtain valid tests and CIs

2. One should make every effort to ensure that excess variability is not due to incorrect functional form, model formulation, omitted interactions terms, etc.

3. LRTs need be scaled as well!

4. Robust standard errors are an alternative to accounting for mis-specified variance functions

5. "Model based" approaches for post-hoc fixes can also be used. . .More later!