Data Set

- The table below shows the average numbers of days absent by program type and seems to suggest that program type is a good candidate for predicting the number of days absent, our outcome variable, because the mean value of the outcome appears to vary by **prog**.

Negative Binomial Regression with R

```
with(dat, tapply(daysabs, prog, function(x) {
sprintf("M (SD) = %1.2f (%1.2f)", mean(x), sd(x))
}))
##                  General
## "M (SD) = 10.65 (8.20)"
##                 Academic
## "M (SD) = 6.93 (7.45)"
##               Vocational
## "M (SD) = 2.67 (3.73)"
```

Data Set

- The variances within each level of **prog** are higher than the means within some of the levels.
- These are the conditional means and variances. These differences suggest that over-dispersion is present and that a Negative Binomial model would be appropriate.

Negative Binomial Regression with R

Negative binomial regression analysis
We will use the glm.nb function from the MASS package to estimate a negative binomial regression.

```
summary(m1 <- glm.nb(daysabs ~ math + prog,
 data = negbinom))
##
## Call:
## glm.nb(formula = daysabs ~ math + prog,
        data = dat, init.theta = 1.032713156,
##     link = log)
##
```

Negative Binomial Regression with R

- R first displays the call and the deviance residuals.
- Next, we see the regression coefficients for each of the variables, along with standard errors, z-scores, and p-values.

Negative Binomial Regression with R

```
Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     2.61527    0.19746   13.24  < 2e-16 ***
math           -0.00599    0.00251   -2.39    0.017 *
progAcademic   -0.44076    0.18261   -2.41    0.016 *
progVocational -1.27865    0.20072   -6.37  1.9e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Negative Binomial Regression with R

- The variable **math** has a coefficient of -0.006, which is statistically significant.
- This means that for each one-unit increase in **math**, the expected log count of the number of days absent decreases by 0.006.
- The indicator variable shown as **progAcademic** is the expected difference in log count between group 2 and the reference group (prog=1).

Negative Binomial Regression with R

- The expected log count for level 2 of prog is 0.44 lower than the expected log count for level 1.
- The indicator variable for **progVocational** is the expected difference in log count between group 3 and the reference group.

Negative Binomial Regression with R

- The expected log count for level 3 of prog is 1.28 lower than the expected log count for level 1.
- To determine if prog itself, overall, is statistically significant, we can compare a model with and without **prog**.
- The reason it is important to fit separate models, is that unless we do, the overdispersion parameter is held constant.

Negative Binomial Regression with R

```
m2 <- update(m1, . ~ . - prog)
anova(m1, m2)
## Likelihood ratio tests of Negative Binomial Models
##
## Response: daysabs
##            Model  theta Resid. df    2 x log-lik.   Test
## 1           math 0.8559       312           -1776
## 2    math + prog 1.0327       310           -1731 1 vs 2
##      Pr(Chi)
## 1
## 2 1.652e-10
```

Negative Binomial Regression with R

- The two degree-of-freedom chi-square test indicates that prog is a statistically significant predictor of daysabs.
- The null deviance is calculated from an intercept-only model with 313 degrees of freedom.
- Then we see the residual deviance, the deviance from the full model.
- We are also shown the AIC and 2*log likelihood.

Negative Binomial Regression with R

- The theta parameter shown is the dispersion parameter.

Negative Binomial Regression with R

```
## (Dispersion parameter for Negative Binomial(1.033) famil
##
##     Null deviance: 427.54  on 313  degrees of freedom
## Residual deviance: 358.52  on 310  degrees of freedom
## AIC: 1741
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  1.033
##           Std. Err.:  0.106
##
##  2 x log-likelihood:  -1731.258
```

Checking model assumption

- As we mentioned earlier, negative binomial models assume the conditional means are not equal to the conditional variances.
- This inequality is captured by estimating a dispersion parameter (not shown in the output) that is held constant in a Poisson model.
- Thus, the Poisson model is actually nested in the negative binomial model.
- We can then use a likelihood ratio test to compare these two and test this model assumption.
- To do this, we can run our model as a Poisson,

Negative Binomial Regression with R

Confidence Intervals

We can get the confidence intervals for the coefficients by profiling the likelihood function.

```
(est <- cbind(Estimate = coef(m1), confint(m1)))
## Waiting for profiling to be done...
##                   Estimate   2.5 %    97.5 %
## (Intercept)       2.615265  2.2421  3.012936
## math             -0.005993 -0.0109 -0.001067
## progAcademic     -0.440760 -0.8101 -0.092643
## progVocational   -1.278651 -1.6835 -0.890078
```

Incidence Rate Ratios

- We might be interested in looking at incident rate ratios rather than coefficients.
- To do this, we can exponentiate our model coefficients.
- The same applies to the confidence intervals.

Negative Binomial Regression with R

```
exp(est)
##                   Estimate  2.5 %  97.5 %
## (Intercept)        13.6708 9.4127 20.3470
## math                0.9940 0.9892  0.9989
## progAcademic        0.6435 0.4448  0.9115
## progVocational      0.2784 0.1857  0.4106
```

Negative Binomial Regression with R

- The output above indicates that the incident rate for prog = 2 is 0.64 times the incident rate for the reference group (prog = 1).
- Likewise, the incident rate for prog = 3 is 0.28 times the incident rate for the reference group holding the other variables constant.
- The percent change in the incident rate of **daysabs** is a 1% decrease for every unit increase in math.

Negative Binomial Regression with R

- The form of the model equation for negative binomial regression is the same as that for Poisson regression.
- The log of the outcome is predicted with a linear combination of the predictors:

$$ln(\widehat{daysabs_i}) = Intercept + b_1(prog_i = 2) + b_2(prog_i = 3) + b_3 math_i$$

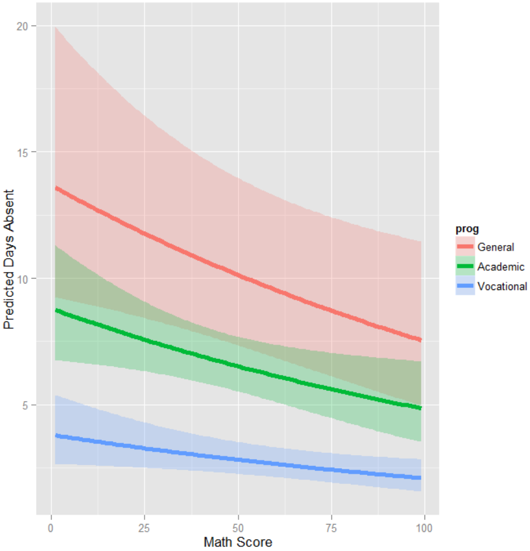
Negative Binomial Regression with R

$$\widehat{daysabs}_i = e^{Intercept + b_1(prog_i=2) + b_2(prog_i=3) + b_3 math_i}$$

$$= e^{Intercept} e^{b_1(prog_i=2)} e^{b_2(prog_i=3)} e^{b_3 math_i}$$

Negative Binomial Regression with R

- The coefficients have an additive effect in the $ln(y)$ scale and the IRR have a multiplicative effect in the y scale.
- The dispersion parameter in negative binomial regression does not effect the expected counts, but it does effect the estimated variance of the expected counts.


Predicted Days Absent
20
15
10
5
0
25
50
75
100
Math Score
prog
General
Academic
Vocational