

# Poisson Regression

Outline of this lecture :

- When to use Poisson regression
- How the parameters are estimated
- How to fit Poisson regression model to data with R, interpret output
- How to test goodness of fit of model
- How to run ROBUST Poisson regression (to adjust for heterogeneity)

## R functions and packages used in this lecture

- Function *stats* of library *fields* for obtaining descriptive statistics in a nicer format
- *glm* for Poisson regression (used earlier for logistic regression)
- *waldtest* of library *lmtest* for testing overall model fit
- *coefest* of library *sandwich* for adjusting for heterogeneity (robust version of Poisson regression) – also needs *lmtest* .
- *predict* function used to predict using the fitted model

- Poisson distribution can be thought of as the binomial distribution BIN(n,p) in the limiting case when number of trials n is very large, p is very small (i.e., we are looking at the occurrence of a rare event) in such a way that the mean  $np$  stays a constant :  $np = \lambda > 0$ .

- Variance of binomial =  $np(1-p) = np - np^2$  will approach  $\lambda - \lambda p$  or  $\lambda$ . In other words, for a Poisson random variable:

MEAN = VARIANCE =  $\lambda$

The probability distribution of  $Y \sim POI(\lambda)$  is given by:

$$f(y; \lambda) = \frac{e^{-\lambda} \lambda^y}{y!}, y = 0, 1, 2, \dots; \lambda > 0$$

Poisson Regression Model:

Sample =  $\{(X_{1i}, X_{2i}, \dots, X_{pi}, Y_i), i = 1, 2, \dots, n\}$

$$Y_i \sim POI(\lambda_i), \lambda_i = \lambda(X_i, \beta)$$

Three commonly used functions are:

$$\lambda(X_i, \beta) = \beta_0 + \beta_1 X_{1i} + \dots + \beta_p X_{pi}$$

$$\lambda(X_i, \beta) = e^{\beta_0 + \beta_1 X_{1i} + \dots + \beta_p X_{pi}}$$

$$\lambda(X_i, \beta) = \ln(\beta_0 + \beta_1 X_{1i} + \dots + \beta_p X_{pi})$$

Likelihood function of the sample is:

$$L(\beta_0, \beta_1, \dots, \beta_P; Y_1, Y_2, \dots, Y_n) = \prod_{i=1}^n \frac{e^{-\lambda(X_i, \beta)} \lambda(X_i, \beta)^{Y_i}}{Y_i!}$$

Numerical search method is used to find  $\beta_0, \beta_1, \dots, \beta_P$  so that the likelihood function is maximized.

Model deviance for Poisson regression

$$= -2 \left[ \sum_{i=1}^n Y_i \ln\left(\frac{\hat{\lambda}_i}{Y_i}\right) + \sum_{i=1}^n (Y_i - \hat{\lambda}_i) \right]$$

Deviance residual for the i-th case is

$$dev_i = \begin{cases} + \sqrt{-2Y_i \ln\left(\frac{\hat{\lambda}_i}{Y_i}\right) - 2(Y_i - \hat{\lambda}_i)}, & Y_i - \hat{\lambda}_i > 0 \\ - \sqrt{-2Y_i \ln\left(\frac{\hat{\lambda}_i}{Y_i}\right) - 2(Y_i - \hat{\lambda}_i)}, & Y_i - \hat{\lambda}_i < 0 \end{cases}$$

Example: The data file poissonreg.csv has data on student's gender, math score, language/arts score and days absent (DV).

```
school2 <- read.csv("K:/DataMining/Data/poissonreg.csv",header=TRUE)
attach(school2)
```

```
# compute summary stats of DV and lvs – use library fields and function stats
# to get a nicer output
# no need to compute descriptives of id and school, so remove these columns
sch <- school2[,3:7]
stats(sch)
```

	male	math	langarts	daysatt	daysabs
N	316	316	316	316	316
mean	0.487342	48.751	50.064	74.658	5.81
Std.Dev.	0.500633	17.881	17.939	11.467	7.449
min	0	1.007	1.007	13	0
Q1	0	37.725	40.15	70	1
median	0	48.944	50	76	3
Q3	1	61.044	61.044	84	8
max	1	98.993	98.993	86	45
missing values	0	0	0	0	0

```
pr1<-glm(daysabs~math+langarts+male, family=poisson)
summary(pr1)
```

Call:

```
glm(formula = daysabs ~ math + langarts + male, family =
poisson)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-4.0117	-2.5455	-1.1014	0.8956	11.0374

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	2.687666	0.072651	36.994	< 2e-16 ***
math	-0.003523	0.001821	-1.934	0.0531 .
langarts	-0.012152	0.001835	-6.623	3.52e-11 ***
male	-0.400921	0.048412	-8.281	< 2e-16 ***

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

STATS24x7.com© 2010 ADI-NV, INC

9

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 2409.8 on 315 degrees of freedom  
Residual deviance: 2234.5 on 312 degrees of freedom  
AIC: 3103.9

Change in deviance from null (intercept only) model and the full model is a measure of model fit .

We can formally test how well model fits the data by using Wald's test (need the R-package **lmtest**)

STATS24x7.com© 2010 ADI-NV, INC

10

```
# install package lmtest and then load it
library(lmtest)
waldtest(pr1, test="Chisq")
```

Wald test

Model 1: daysabs ~ math + langarts + male

Model 2: daysabs ~ 1

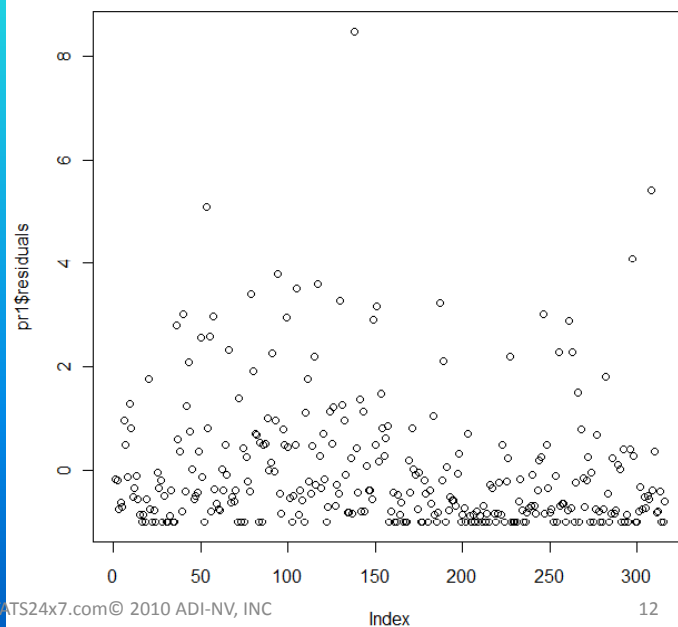
	Res.Df	Df	Chisq	Pr(>Chisq)
1	312			
2	315	-3	176.24	< 2.2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Hence the fitted model is doing significantly better than the null model.

```
# Index plot of residuals
plot(pr1$residuals)
```



```
# install and load R package sandwich (needs lmtest as well)
library(lmtest)
library(sandwich)
coeftest(pr1, vcov=sandwich)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	2.6876659	0.2177980	12.3402	< 2.2e-16
math	-0.0035232	0.0076252	-0.4621	0.644044
langarts	-0.0121521	0.0052867	-2.2986	0.021527
male	-0.4009209	0.1393705	-2.8767	0.004019
---				

Standard errors of estimates are quite different now, with math turning out to be not significant.

We next run Poisson regression dropping math term.

13

```
pr2 <- glm(daysabs~langarts+male, family=poisson)
summary(pr2)
Call:
glm(formula = daysabs ~ langarts + male, family = poisson)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-4.297	-2.510	-1.123	0.869	10.495

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	2.646976	0.069776	37.935	<2e-16 ***
langarts	-0.014670	0.001293	-11.342	<2e-16 ***
male	-0.409353	0.048219	-8.489	<2e-16 ***

Null deviance: 2409.8 on 315 degrees of freedom

Residual deviance: 2238.3 on 313 degrees of freedom

AIC: 3105.7

Robust version of standard errors:

```
coeftest(pr2, cov=sandwich)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	2.6469765	0.0697764	37.9351	< 2.2e-16
langarts	-0.0146700	0.0012934	-11.3418	< 2.2e-16
male	-0.4093528	0.0482192	-8.4894	< 2.2e-16

```
> waldtest(pr2, test = "Chisq")
```

Wald test

Model 1: daysabs ~ langarts + male

Model 2: daysabs ~ 1

Res.Df	Df	Chisq	Pr(>Chisq)
1	313		
2	315	-2	172.17

< 2.2e-16 \*\*\*

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Once we have our final Poisson regression model, we can use it to predict values of days absent.

Example: predict # of days for a student with math score = 75, language/arts score = 85.

R-code:

```
male<-c(0, 1)
```

```
langarts<-c(85, 85)
```

```
new<-cbind(male, langarts)
```

```
fitted<-predict(pr2, newdata=data.frame(new), type="response")
```

```
> fitted
```

```
      1      2  
4.696072 2.693048
```