Topics in Multivariate Analysis

APSTA GE-2004

Lecture 6 - Chi Square Test and Regression

3/1/2022

Outline

Welcome! Today, we'll cover the following:

- Chi Square Test
- Framing Chi Square as regression
- Overdispersion and Exposure

Reading:

RAOS Ch 9.1-9.2; Ch 10.1-10.4; Ch 11.2; Ch 12.4, 12.5; Ch 15
Estimating Generalized Linear Models for Count Data with rstanarm by Gabry and Goodrich
RAOS Appendix A and B

Chi Square Test

Example 1

Comparing several populations using two-way tables

Do men and women participate in sports for the same reasons?

One goal for sports participants is **social comparison** – the desire to win or to do better than other people

Another is **mastery** – the desire to improve one's skills or to try one's best

A study on why students participate in sports collected data from 67 male and 67 female undergraduates at a large university*

Each student was *classified into one of four categories*:

- High social comparison – High mastery (**HSC–HM**)
- High social comparison – Low mastery (**HSC–LM**)
- Low social comparison – High mastery (**LSC–HM**)
- Low social comparison – Low mastery (**LSC–LM**)

Observed counts for sports goals

Goal	Female	Male	Total
HSC-HM	14	31	45
HSC-LM	7	18	25
LSC-HM	21	5	26
LSC-LM	25	13	38
Total	67	67	134

* Example from Intro to Practice of Statistics, 3rd Edition pg. 625

Two-way table

The rows and columns of a two-way table represent values of two categorical variables

A two-way table with *r* rows and *c* columns contains r × c cells

In this example, the objective is to compare men and women:

- the column variable describes which population an observation comes from
- the row variable is a categorical response variable, type of sports goal

It is not always the case that one direction of the table identifies populations to be compared. Two-way tables can display observations on any two categorical variables

Observed counts for sports goals

Goal	Female	Male	Total
HSC-HM	**14**	**31**	45
HSC-LM	**7**	**18**	25
LSC-HM	**21**	**5**	26
LSC-LM	**25**	**13**	38
Total	67	67	134

Two-way table

The rows and columns of a two-way table represent values of two categorical variables

A two-way table with *r* rows and *c* columns contains r × c cells

In this example, the objective is to compare men and women:

- the column variable describes which population an observation comes from
- the row variable is a categorical response variable, type of sports goal

It is not always the case that one direction of the table identifies populations to be compared. Two-way tables can display observations on any two categorical variables

```
> # Observed counts for sports goals
> goal <- c( rep("HSC-HM", 14), rep("HSC-LM",  7), rep("LSC-HM", 21), rep("LSC-LM", 25),
+            rep("HSC-HM", 31), rep("HSC-LM", 18), rep("LSC-HM",  5), rep("LSC-LM", 13) )
>
> sex  <- c( rep("Female", 67), rep("Male", 67) )
>
> # Create table based on categorical variables
> table(goal,sex)
        sex
goal     Female Male
  HSC-HM     14   31
  HSC-LM      7   18
  LSC-HM     21    5
  LSC-LM     25   13
>
> # Add row and column totals
> addmargins( table(goal,sex) )
        sex
goal     Female Male Sum
  HSC-HM     14   31  45
  HSC-LM      7   18  25
  LSC-HM     21    5  26
  LSC-LM     25   13  38
  Sum        67   67 134
```

Describing relations in a two-way table

To describe relations between the variables, we compute and compare *percents*

The count in each cell can be viewed as a percent of:

- the **grand** total (*joint* distribution)
- the **row** total (*conditional* distribution)
- the **column** total (*conditional* distribution)

You must decide which percents are most appropriate in a specific problem

In this example, we are interested in the influence of sex on the distribution of sports goals. To compare the sexes, we examine the *column* percents

The data reveal something interesting: *it appears these females and males have different goals when they participate in recreational sports*

Column percents for sports goals

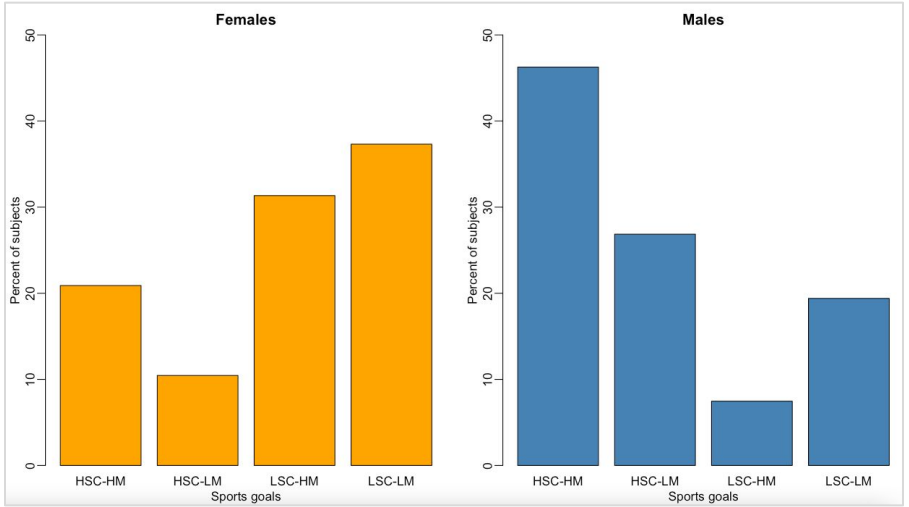
Goal	Female	Male
HSC-HM	**21**	**46**
HSC-LM	**10**	**27**
LSC-HM	**31**	**7**
LSC-LM	**37**	**19**
Total	100	100

Describing relations in a two-way table

The differences between the distributions of female and male sports goals in the sample appear to be large

A statistical test (and later a regression) will tell us whether or not these differences can be plausibly attributed to chance

Specifically, if the two population distributions were the same, how likely is it that a sample would show differences as large or larger than those displayed in the bar plots



The null hypothesis: no association

Null hypothesis:
The null hypothesis H_O of interest in a two-way table is: there is *no association* between the row variable and the column variable

In the sports goals example, this null hypothesis says that sex and sports goals *are not related*

Alternative hypothesis:
The alternative hypothesis H_a is that *there is an association* between these two variables

The alternative hypothesis H_a does not specify any particular direction for the association. We cannot describe H_a as one-sided or two-sided, because it includes all of the many kinds of association that are possible

In the sports goals example, this alternative hypothesis says that sex and sports goals *are related* in some way

Expected cell counts

To test the null hypothesis in r × c tables, we compare the *observed* cell counts with **expected cell counts** calculated under the assumption that the null hypothesis is true:

expected cell count = row total × column total / N

In this example, 33.58% of all respondents (female and male) are in the HSC–HM group. If the null hypothesis of no sex difference in sports goals is true, we expect this overall percentage to apply to both men and women. So, we expect 33.58% of the 67 females in the study to be in this group: 0.3358 * 67 = 22.5

The other expected counts are calculated similarly

In this example, the number of males is the same as the number of females, so the expected counts for males are the same as the expected counts for females

A numerical summary of the comparison between *observed* and *expected* counts will be our test statistic

```
> # Add row and column totals
> addmargins( table(goal,sex) )
        sex
goal     Female Male Sum
  HSC-HM     14   31  45
  HSC-LM      7   18  25
  LSC-HM     21    5  26
  LSC-LM     25   13  38
  Sum        67   67 134
>
> round( addmargins(table(goal,sex)) / sum(table(goal,sex)) * 100 , 2 )
        sex
goal     Female   Male    Sum
  HSC-HM  10.45  23.13  33.58
  HSC-LM   5.22  13.43  18.66
  LSC-HM  15.67   3.73  19.40
  LSC-LM  18.66   9.70  28.36
  Sum     50.00  50.00 100.00
```

```
> # Expected cell counts in Females column "by hand": row total * column total / total number of observations
> round( margin.table( table(goal,sex) , margin = 1 ) * margin.table(table(goal,sex), margin=2 )[1] / sum( table(goal,sex) ) , digits=1 )
goal
HSC-HM HSC-LM LSC-HM LSC-LM
  22.5   12.5   13.0   19.0
>
> # Expected cell counts in Males column "by hand": row total * column total / total number of observations
> round( margin.table( table(goal,sex) , margin = 1 ) * margin.table(table(goal,sex), margin=2 )[2] / sum( table(goal,sex) ) , digits=1 )
goal
HSC-HM HSC-LM LSC-HM LSC-LM
  22.5   12.5   13.0   19.0
>
> # Expected cell counts for all cells with gmodels::CrossTable
> CrossTable( table(goal,sex) , digits = 1, expected = TRUE , prop.r = FALSE , prop.c = FALSE , prop.t = FALSE , prop.chisq = FALSE )

   Cell Contents
|-------------------------|
|                       N |
|              Expected N |
|-------------------------|

Total Observations in Table:  134

             | sex
        goal |    Female |      Male | Row Total |
-------------|-----------|-----------|-----------|
      HSC-HM |        14 |        31 |        45 |
             |      22.5 |      22.5 |           |
-------------|-----------|-----------|-----------|
      HSC-LM |         7 |        18 |        25 |
             |      12.5 |      12.5 |           |
-------------|-----------|-----------|-----------|
      LSC-HM |        21 |         5 |        26 |
             |      13.0 |      13.0 |           |
-------------|-----------|-----------|-----------|
      LSC-LM |        25 |        13 |        38 |
             |      19.0 |      19.0 |           |
-------------|-----------|-----------|-----------|
Column Total |        67 |        67 |       134 |
-------------|-----------|-----------|-----------|
```

The chi-square test

To test the H_O of no association, we use a statistic that compares the set of *observed* counts with the set of *expected* counts

First, take the difference between each *observed* count and its corresponding *expected* count, and square these values so they are all 0 or positive. A large difference means less if it comes from a cell we think will have a large count, so divide each squared difference by the expected count, a kind of standardization. Finally, sum over all cells

The result is the *chi-square statistic*:

X^2: ∑(observed count – expected count)^2 / expected count

where "observed" represents an observed sample count, "expected" represents the expected count for the same cell, and the sum is over all r × c cells in the table

If the expected counts and the observed counts are very different, a large value of X^2 will result. So large values of X^2 provide evidence against the null hypothesis

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(goal,sex) ) )

        Pearson's Chi-squared test

data:  table(goal, sex)
X-squared = 24.898, df = 3, p-value = 1.622e-05
```

```
# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )
```

```
   Cell Contents
|-------------------------|
|                       N |
|              Expected N |
|-------------------------|


Total Observations in Table:  134
```

goal	sex: Female	sex: Male	Row Total
HSC-HM	14 22.5	31 22.5	45
HSC-LM	7 12.5	18 12.5	25
LSC-HM	21 13.0	5 13.0	26
LSC-LM	25 19.0	13 19.0	38
Column Total	67	67	134

The chi-square test

The *chi-square statistic*:

X^2: ∑(observed count – expected count)^2 / expected count

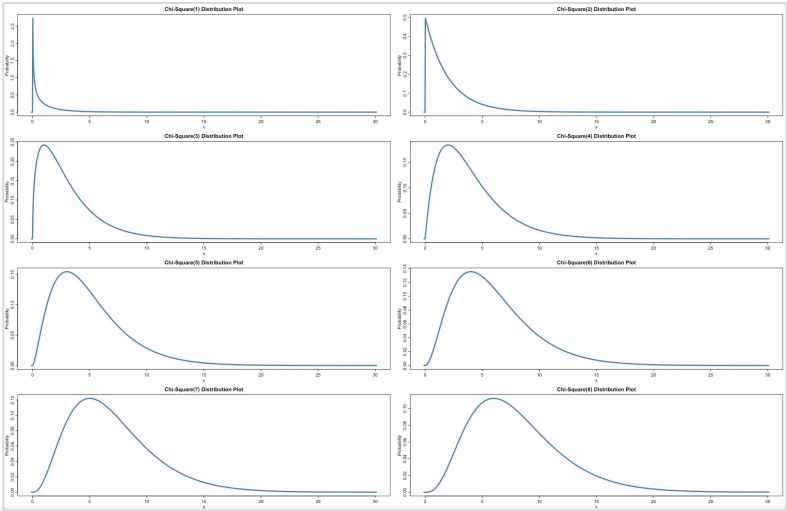
where “observed” represents an observed sample count, “expected” represents the expected count for the same cell, and the sum is over all r × c cells in the table

Like the t distributions, X^2 distributions form a family described by a single parameter, the degrees of freedom

If H_O is true, the X^2 statistic has approximately a X^2 distribution with **(r – 1)(c – 1) degrees of freedom**

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(goal,sex) ) )

        Pearson's Chi-squared test

data:  table(goal, sex)
X-squared = 24.898, df = 3, p-value = 1.622e-05
```

```
# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )
```



The chi-square test

The *chi-square statistic*:

X^2: ∑(observed count – expected count)^2 / expected count

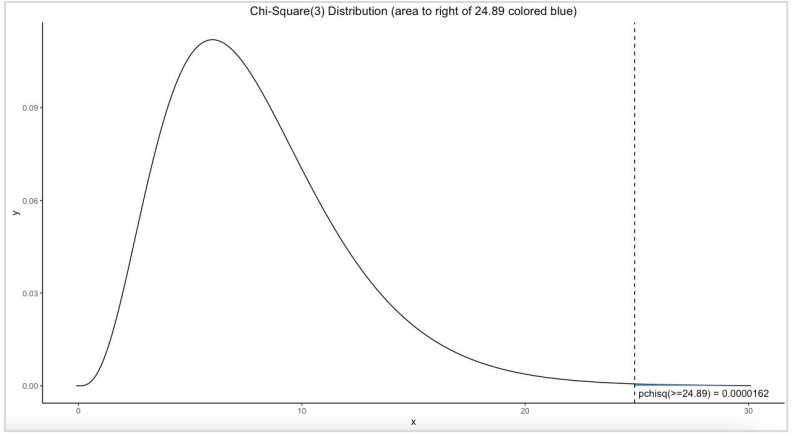
where "observed" represents an observed sample count, "expected" represents the expected count for the same cell, and the sum is over all r × c cells in the table

Like the t distributions, X^2 distributions form a family described by a single parameter, the degrees of freedom

If H_O is true, the X^2 statistic has approximately a X^2 distribution with **(r – 1)(c – 1) degrees of freedom**

The *p-value* for the chi-square test is: **$P(\chi^2 \geq X^2)$**

```r
# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )


# Degrees of freedom
( degrees_of_freedom <- (nrow(table(goal,sex)) - 1) * (ncol(table(goal,sex)) - 1) )


# Critical values for this chi-square distribution
( crit10  <- qchisq( 0.90  , df = 3 ) )
( crit05  <- qchisq( 0.95  , df = 3 ) )
( crit025 <- qchisq( 0.975 , df = 3 ) )
( crit01  <- qchisq( 0.99  , df = 3 ) )
( crit001 <- qchisq( 0.999 , df = 3 ) )
```



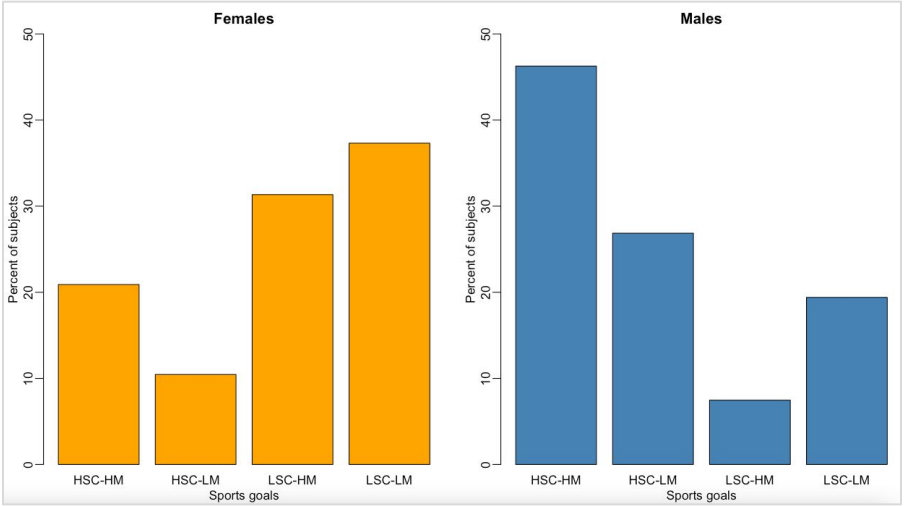
The chi-square test

The test indicates that the male and female distributions are not the same, but it does not say how they differ

Always combine the test with a description that shows what kind of relationship is present

From the side-by-side bar plots, , we see that the percent of males in each of the HSC goal classes is more than twice the percent of females. The pattern is reversed for the LSC goal classes

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(goal,sex) ) )

        Pearson's Chi-squared test

data:  table(goal, sex)
X-squared = 24.898, df = 3, p-value = 1.622e-05
```



Framing chi square as classical regression

Data structure

- The cells of the table indicate the frequency with which each combination occurred in the sample
- Each respondent fell in one and only one cell of the table
- The data to be predicted are the cell counts
- The predictors are the nominal variables
- This structure is analogous to two-way analysis of variance (ANOVA), which also had two nominal predictors, but had metric values in each cell instead of a single count

Observed counts for sports goals

Goal	Female	Male	Total
HSC-HM	14	31	45
HSC-LM	7	18	25
LSC-HM	21	5	26
LSC-LM	25	13	38
Total	67	67	134

Poisson exponential model

We can refer to the model we will use to describe these data as Poisson exponential, because the noise distribution is a Poisson distribution and the inverse-link function is exponential

Motivation 1

One way to motivate the model is to start with the two-way ANOVA model for combining nominal predictors, and find a way to map the predicted value to count data

The predicted μ from ANOVA can be any value from negative to positive infinity, but frequencies are non-negative

Therefore, we must transform the ANOVA predictions to non-negative values, while preserving order. A natural way to do this, mathematically, is with the exponential transformation

But this transformation only gets us to a continuous predicted value, not to the probability of discrete counts. A natural candidate for the needed likelihood distribution is the Poisson, which takes a non-negative λ and gives a probability for each integer from zero to infinity

Motivation 2

A different motivation starts by treating the cell counts as representative of underlying cell probabilities, and then asking whether the two nominal variables contribute independent influences to the cell probabilities

For example, in the table there's a particular marginal probability a respondent's sports goal is HSC-HM, and a particular marginal probability a respondent is female. If sports goal and sex are independent, then the joint probability of HSC-HM and female is the product of the marginal probabilities

The attributes of sports goal and sex are independent if that relationship holds for every cell in the table

Exponential link function

To check for independence of attributes, we need to estimate the marginal probabilities of the attributes

Denote the marginal (i.e., total) count of the *r*th row as y_r, and the marginal count of the *c*th column as y_c . Then the marginal proportions are y_r/N and y_c/N (where N is the total of the entire table)

If the attributes are independent, then the *predicted* joint probability, $\hat{p}(r,c)$, should equal the product of the marginal probabilities, which means

$\hat{p}(r,c) = p(r) * p(c)$

$\hat{y}_{r,c}/N = y_r/N * y_c/N$

Because the models we deal with involve *additive* combinations, not multiplicative combinations, we convert the multiplicative expression into an *additive* expression by using the facts:

log(a*b) = log(a) + log(b)

exp(ln(x)) = x

From multiplicative to additive:

$$\hat{y}_{r,c}/N = y_r/N * y_c/N$$

$$\hat{y}_{r,c} = 1/N * y_r * y_c$$

$$\underbrace{\hat{y}_{r,c}}_{\lambda_{r,c}} = \exp(\underbrace{\log(1/N)}_{\beta_0} + \underbrace{\log(y_r)}_{\beta_r} + \underbrace{\log(y_c)}_{\beta_c})$$

Extended with interaction terms, the model of the cell tendencies is:

$$\lambda_{r,c} = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

where $\sum\beta_r = 0$, $\sum\beta_c = 0$, $\sum\beta_{r,c} = 0$ for all c, and $\sum\beta_{r,c} = 0$ for all r

If we're interested in violations of independence, then *our interest is on the magnitudes of the* $\boldsymbol{\beta_{r,c}}$ *interaction terms, and specifically on meaningful interaction contrasts*

Poisson noise distribution

The value of $\boldsymbol{\lambda_{r,c}}$ is a cell *tendency*, not a predicted count

In particular, the value of $\lambda_{r,c}$ can be any non-negative real value, but counts can only be integers

What we need is a likelihood function that maps the parameter value $\lambda_{r,c}$ to the probabilities of possible counts

The Poisson distribution is a natural choice:

$$p(y|\lambda) = \lambda^y \exp(-\lambda) / y!$$

where y is a non-negative integer and λ is is a non-negative real number

The mean of the Poisson distribution is λ and, importantly, the variance is also λ (i.e. the standard deviation is $\sqrt{\lambda}$)

Additive model:

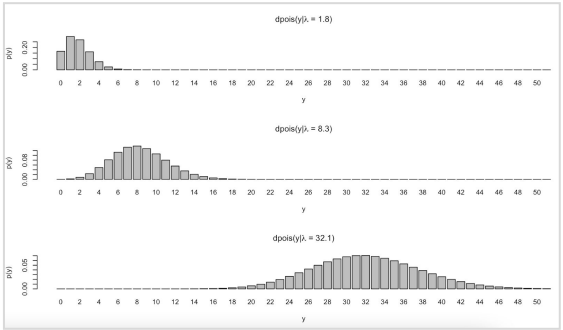
$$\hat{y}_{r,c} = \exp(\underbrace{\log(1/N)}_{\beta_0} + \underbrace{\log(y_r)}_{\beta_r} + \underbrace{\log(y_c)}_{\beta_c})$$

$$\lambda_{r,c}$$

We will use the Poisson distribution as the likelihood function for modeling the probability of the observed count, $\mathbf{y_{r,c}}$, given the mean, $\boldsymbol{\lambda_{r,c}}$

The idea is that each particular *r*, *c* combination has an underlying average rate of occurrence, $\lambda_{r,c}$

We collect data for a period of time, during which we happen to observed particular frequencies, $y_{r,c}$, of each combination. *From the observed frequencies, we infer the underlying average rates*



Framing chi square as classical regression in R

Now let's see how to recast the chi square test as a linear regression model in R:

```
# Minimal model (expected frequency)
m0.1 <- glm(count ~ 1, family = poisson(link = "log"),
            data = sports)
tidy(m0.1)
exp( tidy(m0.1) %>% pull(estimate) )

# 2. Additive model (expected frequencies)
m0.2 <- glm(count ~ 1 + goal + gender, family = poisson(link = "log"),
            data = sports)
tidy(m0.2)
# reference category: HSC-HM and Female
exp( tidy(m0.2) %>% filter(term == "(Intercept)") %>% pull(estimate) )

# 3. Saturated model (equal to observed frequencies)
m0.3 <- glm(count ~ 1 + goal + gender + goal:gender,
            family = poisson(link = "log"),
            data = sports)
tidy(m0.3)
# reference category: HSC-HM and Female
# equal to observed frequency, 14
exp( tidy(m0.3) %>% filter(term == "(Intercept)") %>% pull(estimate) )
```

Untransformed coefficients (on logarithmic scale)

	Minimal model		Additive model		Saturated model	
Predictors	*Log-Mean*	*p*	*Log-Mean*	*p*	*Log-Mean*	*p*
(Intercept)	2.82 (0.09)	**<0.001**	3.11 (0.17)	**<0.001**	2.64 (0.27)	**<0.001**
goal [HSC-LM]			-0.59 (0.25)	**0.018**	-0.69 (0.46)	0.134
goal [LSC-HM]			-0.55 (0.25)	**0.026**	0.41 (0.35)	0.240
goal [LSC-LM]			-0.17 (0.22)	0.443	0.58 (0.33)	0.082
gender [Male]			-0.00 (0.17)	1.000	0.79 (0.32)	**0.014**
goal [HSC-LM] * gender [Male]					0.15 (0.55)	0.786
goal [LSC-HM] * gender [Male]					-2.23 (0.59)	**<0.001**
goal [LSC-LM] * gender [Male]					-1.45 (0.47)	**0.002**
Observations	8		8		8	
R^2 Nagelkerke	0.000		0.656		1.000	

Lack of independence is captured by the model's interaction terms.

Framing chi square as classical regression in R

Now let's see how to recast the chi square test as a linear regression model in R:

```
# Minimal model (expected frequency)
m0.1 <- glm(count ~ 1, family = poisson(link = "log"),
            data = sports)
tidy(m0.1)
exp( tidy(m0.1) %>% pull(estimate) )

# 2. Additive model (expected frequencies)
m0.2 <- glm(count ~ 1 + goal + gender, family = poisson(link = "log"),
            data = sports)
tidy(m0.2)
# reference category: HSC-HM and Female
exp( tidy(m0.2) %>% filter(term == "(Intercept)") %>% pull(estimate) )

# 3. Saturated model (equal to observed frequencies)
m0.3 <- glm(count ~ 1 + goal + gender + goal:gender,
            family = poisson(link = "log"),
            data = sports)
tidy(m0.3)
# reference category: HSC-HM and Female
# equal to observed frequency, 14
exp( tidy(m0.3) %>% filter(term == "(Intercept)") %>% pull(estimate) )
```

Transformed (exponentiated) coefficients (multiplicative effects)

	Minimal model		Additive model		Saturated model	
Predictors	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*
(Intercept)	16.75 (1.45)	**<0.001**	22.50 (3.88)	**<0.001**	14.00 (3.74)	**<0.001**
goal [HSC-LM]			0.56 (0.14)	**0.018**	0.50 (0.23)	0.134
goal [LSC-HM]			0.58 (0.14)	**0.026**	1.50 (0.52)	0.240
goal [LSC-LM]			0.84 (0.19)	0.443	1.79 (0.60)	0.082
gender [Male]			1.00 (0.17)	1.000	2.21 (0.71)	**0.014**
goal [HSC-LM] * gender [Male]					1.16 (0.64)	0.786
goal [LSC-HM] * gender [Male]					0.11 (0.06)	**<0.001**
goal [LSC-LM] * gender [Male]					0.23 (0.11)	**0.002**
Observations	8		8		8	
R^2 Nagelkerke	0.000		0.656		1.000	

Lack of independence is captured by the model's interaction terms.

From Poisson regression coefficients to cell tendencies

The coefficients can be combined to calculate the expected cell tendencies and to retrodict the counts in each of the cells:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\mathbf{X}_i\boldsymbol{\beta})$$

Female HSC-HM = exp(`(Intercept)`)

Female HSC-LM = exp(`(Intercept)` + `goalHSC-LM`)

Female LSC-HM = exp(`(Intercept)` + `goalLSC-HM`)

Female LSC-LM = exp(`(Intercept)` + `goalLSC-LM`)

Male HSC-HM = exp(`(Intercept)` + `genderMale`)

Male HSC-LM = exp(`(Intercept)` + `goalHSC-LM` + `genderMale` + `goalHSC-LM:genderMale`)

Male LSC-HM = exp(`(Intercept)` + `goalLSC-HM` + `genderMale` + `goalLSC-HM:genderMale`)

Male LSC-LM = exp(`(Intercept)` + `goalLSC-LM` + `genderMale` + `goalLSC-LM:genderMale`)

```
> # 3. Saturated model (equal to observed frequencies)
> m0.3 <- glm(count ~ 1 + goal + gender + goal:gender,
+     family = poisson(link = "log"),
+     data = sports)
> tidy(m0.3)
# A tibble: 8 × 5
  term                 estimate std.error statistic  p.value
  <chr>                   <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)             2.64      0.267     9.87  5.37e-23
2 goalHSC-LM             -0.693     0.463    -1.50  1.34e- 1
3 goalLSC-HM              0.405     0.345     1.18  2.40e- 1
4 goalLSC-LM              0.580     0.334     1.74  8.24e- 2
5 genderMale              0.795     0.322     2.47  1.36e- 2
6 goalHSC-LM:genderMale   0.150     0.550     0.272 7.86e- 1
7 goalLSC-HM:genderMale  -2.23      0.593    -3.76  1.68e- 4
8 goalLSC-LM:genderMale  -1.45      0.470    -3.08  2.04e- 3
>
> # Exponentiated fitted values from the saturated model equal the observed counts
> matrix( sapply(augment(m0.3)$.fitted, FUN = exp) ,
+         nrow = 4 , ncol = 2 , byrow = TRUE )
     [,1] [,2]
[1,]   14   31
[2,]    7   18
[3,]   21    5
[4,]   25   13
>
> # Observed counts
> Xsq$observed
        sex
goal     Female Male
  HSC-HM     14   31
  HSC-LM      7   18
  LSC-HM     21    5
  LSC-LM     25   13
```

Interpreting Poisson regression coefficients

The coefficients can be exponentiated and treated as multiplicative effects:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

Intercept:

The coefficient estimate for the Intercept (2.64) gives the prediction (on the logarithmic scale) if $X_{i1} = 0$ and $X_{i2} = 0$. Exponentiating this value (exp(2.64) = 14) produces the cell tendency, $\lambda_{r,c}$, for the Female HSC-HM cell in the table

goalHSC-LM:

The coefficient estimate for goalHSC-LM (–0.693) is the expected difference in y (on the logarithmic scale). The multiplicative decrease is $e^{-0.693} = 0.5$, which means (exp(2.64 – 0.693) = 14 * 0.5 = 7) is the cell tendency, $\lambda_{r,c}$, for the Female HSC-HM cell in the table. The calculations for Female LSC-HM and Female LSC-LM are similar

genderMale:

The coefficient estimate for genderMale (0.795) tells us the predictive difference for the Male HSC-HM cell in the table, which means (exp(2.64 + 0.795) = 14 * 2.2 = 31) is the cell tendency, $\lambda_{r,c}$, for the Male HSC-HM cell in the table

goalHSC-LM:genderMale:

The coefficient estimate for goalHSC-LM:genderMale (0.15) tells us the predictive difference for the Male HSC-LM cell in the table. The multiplicative increase is $e^{0.15} = 1.16$, which means (exp(2.64 – 0.693 + 0.795 + 0.15) = 14 * 0.5 * 2.2 * 1.16 = 18) is the cell tendency, $\lambda_{r,c}$, for the Male HSC-LM cell in the table. The calculations for Male LSC-HM and Male LSC-LM are similar

```
> tidy(m0.3)
# A tibble: 8 × 5
  term                  estimate std.error statistic  p.value
  <chr>                    <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)              2.64      0.267      9.87  5.37e-23
2 goalHSC-LM              -0.693     0.463     -1.50  1.34e- 1
3 goalLSC-HM               0.405     0.345      1.18  2.40e- 1
4 goalLSC-LM               0.580     0.334      1.74  8.24e- 2
5 genderMale               0.795     0.322      2.47  1.36e- 2
6 goalHSC-LM:genderMale    0.150     0.550      0.272 7.86e- 1
7 goalLSC-HM:genderMale   -2.23      0.593     -3.76  1.68e- 4
8 goalLSC-LM:genderMale   -1.45      0.470     -3.08  2.04e- 3
```

	Minimal model		Additive model		Saturated model	
Predictors	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*
(Intercept)	16.75 (1.45)	**<0.001**	22.50 (3.88)	**<0.001**	14.00 (3.74)	**<0.001**
goal [HSC-LM]			0.56 (0.14)	**0.018**	0.50 (0.23)	0.134
goal [LSC-HM]			0.58 (0.14)	**0.026**	1.50 (0.52)	0.240
goal [LSC-LM]			0.84 (0.19)	0.443	1.79 (0.60)	0.082
gender [Male]			1.00 (0.17)	1.000	2.21 (0.71)	**0.014**
goal [HSC-LM] * gender [Male]					1.16 (0.64)	0.786
goal [LSC-HM] * gender [Male]					0.11 (0.06)	**<0.001**
goal [LSC-LM] * gender [Male]					0.23 (0.11)	**0.002**
Observations	8		8		8	
R^2 Nagelkerke	0.000		0.656		1.000	

Framing chi square as Bayesian regression

Framing chi square as Bayesian regression in R

Now let's see how to recast the chi square test
as a linear regression model in R:

```
# Estimate Bayesian version with stan_glm
# 3. Saturated model (equal to observed frequencies)
b0.3 <- stan_glm(count ~ 1 + goal + gender + goal:gender,
                 family = poisson(link = "log"), data = sports)
tidy(b0.3)
```

```
> tidy(b0.3)
# A tibble: 8 x 3
  term                  estimate std.error
  <chr>                    <dbl>     <dbl>
1 (Intercept)              2.64      0.270
2 goalHSC-LM              -0.748     0.470
3 goalLSC-HM               0.383     0.351
4 goalLSC-LM               0.554     0.339
5 genderMale               0.787     0.332
6 goalHSC-LM:genderMale    0.209     0.556
7 goalLSC-HM:genderMale   -2.26      0.628
8 goalLSC-LM:genderMale   -1.45      0.475
```

Lack of independence is captured by the model's interaction terms.

Untransformed coefficients (on logarithmic scale)

Predictors	**Classical model** *Log-Mean*	**Bayesian model** *Log-Mean*
(Intercept)	2.64 (0.27)	2.64 (0.27)
goal: HSC-LM	-0.69 (0.46)	-0.75 (0.47)
goal: LSC-HM	0.41 (0.35)	0.38 (0.35)
goal: LSC-LM	0.58 (0.33)	0.55 (0.34)
gender: Male	0.79 (0.32)	0.79 (0.33)
goalHSC-LM:genderMale	0.15 (0.55)	0.21 (0.56)
goalLSC-HM:genderMale	-2.23 (0.59)	-2.26 (0.63)
goalLSC-LM:genderMale	-1.45 (0.47)	-1.45 (0.47)

Transformed (exponentiated) coefficients (multiplicative effects)

Predictors	**Classical model** *Incidence Rate Ratios*	**Bayesian model** *Incidence Rate Ratios*
(Intercept)	14.00 (3.74)	14.00 (3.74)
goal: HSC-LM	0.50 (0.23)	0.47 (0.22)
goal: LSC-HM	1.50 (0.52)	1.47 (0.51)
goal: LSC-LM	1.79 (0.60)	1.74 (0.57)
gender: Male	2.21 (0.71)	2.20 (0.71)
goalHSC-LM:genderMale	1.16 (0.64)	1.23 (0.66)
goalLSC-HM:genderMale	0.11 (0.06)	0.10 (0.06)
goalLSC-LM:genderMale	0.23 (0.11)	0.23 (0.11)

From Poisson regression coefficients to cell tendencies

The coefficients can be combined to calculate the expected cell tendencies and to retrodict the counts in each of the cells:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\mathbf{X}_i\boldsymbol{\beta})$$

Female HSC-HM = exp(`(Intercept)`)

Female HSC-LM = exp(`(Intercept)` + `goalHSC-LM`)

Female LSC-HM = exp(`(Intercept)` + `goalLSC-HM`)

Female LSC-LM = exp(`(Intercept)` + `goalLSC-LM`)

Male HSC-HM = exp(`(Intercept)` + `genderMale`)

Male HSC-LM = exp(`(Intercept)` + `goalHSC-LM` + `genderMale` + `goalHSC-LM:genderMale`)

Male LSC-HM = exp(`(Intercept)` + `goalLSC-HM` + `genderMale` + `goalLSC-HM:genderMale`)

Male LSC-LM = exp(`(Intercept)` + `goalLSC-LM` + `genderMale` + `goalLSC-LM:genderMale`)

```
> # Coefficients from the saturated model (on logarithmic scale)
> matrix( round(b0.3$coefficients,2) , nrow = 4 , ncol = 2 , byrow = TRUE )
      [,1]  [,2]
[1,]  2.64 -0.75
[2,]  0.38  0.55
[3,]  0.79  0.21
[4,] -2.26 -1.45
>
> # Exponentiated coefficients from the saturated model (multiplicative effects)
> matrix( round(exp(b0.3$coefficients),2) , nrow = 4 , ncol = 2 , byrow = TRUE )
      [,1] [,2]
[1,] 14.00 0.47
[2,]  1.47 1.74
[3,]  2.20 1.23
[4,]  0.10 0.23
>
> # Exponentiated fitted values from the saturated model equal the observed counts
> matrix( round(b0.3$fitted) , nrow = 4 , ncol = 2 , byrow = TRUE )
     [,1] [,2]
[1,]   14   31
[2,]    7   18
[3,]   21    5
[4,]   24   12
>
> # Observed counts
> Xsq$observed
        sex
goal     Female Male
  HSC-HM     14   31
  HSC-LM      7   18
  LSC-HM     21    5
  LSC-LM     25   13
```

Interpreting Poisson regression coefficients

The coefficients can be exponentiated and treated as multiplicative effects:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

Intercept:

The coefficient estimate for the Intercept (2.64) gives the prediction (on the logarithmic scale) if $X_{i1} = 0$ and $X_{i2} = 0$. Exponentiating this value (exp(2.64) = 14) produces the cell tendency, $\lambda_{r,c}$, for the Female HSC-HM cell in the table

goalHSC-LM:

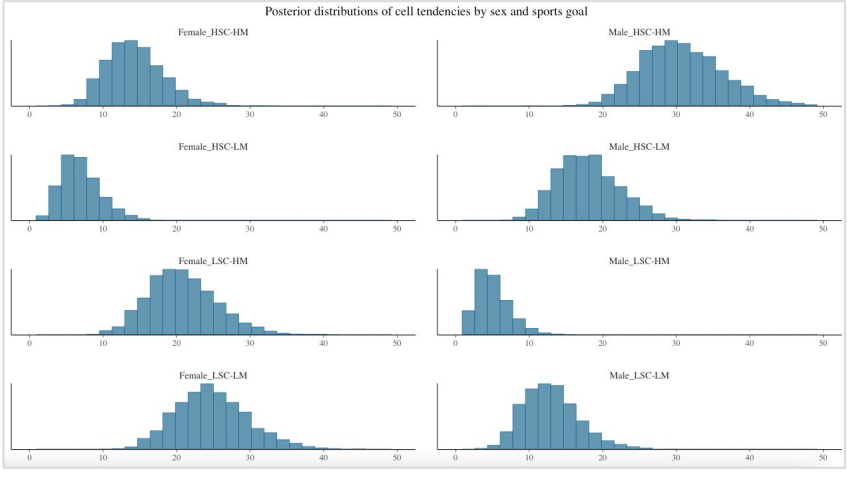
The coefficient estimate for goalHSC-LM (–0.693) is the expected difference in y (on the logarithmic scale). The multiplicative decrease is $e^{-0.693} = 0.5$, which means (exp(2.64 – 0.693) = 14 * 0.5 = 7) is the cell tendency, $\lambda_{r,c}$, for the Female HSC-HM cell in the table. The calculations for Female LSC-HM and Female LSC-LM are similar

genderMale:

The coefficient estimate for genderMale (0.795) tells us the predictive difference for the Male HSC-HM cell in the table, which means (exp(2.64 + 0.795) = 14 * 2.2 = 31) is the cell tendency, $\lambda_{r,c}$, for the Male HSC-HM cell in the table

goalHSC-LM:genderMale:

The coefficient estimate for goalHSC-LM:genderMale (0.15) tells us the predictive difference for the Male HSC-LM cell in the table. The multiplicative increase is $e^{0.15} = 1.16$, which means (exp(2.64 – 0.693 + 0.795 + 0.15) = 14 * 0.5 * 2.2 * 1.16 = 18) is the cell tendency, $\lambda_{r,c}$, for the Male HSC-LM cell in the table. The calculations for Male LSC-HM and Male LSC-LM are similar




```
# Posterior distributions of cell tendencies convey uncertainty
plot_title <- ggtitle("Posterior distributions of cell tendencies by sex and sports goal")

as.data.frame(b0.3) %>%
  mutate(`Female_HSC-HM` = exp(`(Intercept)`),
         `Female_HSC-LM` = exp(`(Intercept)` + `goalHSC-LM`),
         `Female_LSC-HM` = exp(`(Intercept)` + `goalLSC-HM`),
         `Female_LSC-LM` = exp(`(Intercept)` + `goalLSC-LM`),
         `Male_HSC-HM` = exp(`(Intercept)` + `genderMale`),
         `Male_HSC-LM` = exp(`(Intercept)` + `goalHSC-LM` + `genderMale` + `goalHSC-LM:genderMale`),
         `Male_LSC-HM` = exp(`(Intercept)` + `goalLSC-HM` + `genderMale` + `goalLSC-HM:genderMale`),
         `Male_LSC-LM` = exp(`(Intercept)` + `goalLSC-LM` + `genderMale` + `goalLSC-LM:genderMale`) ) %>%
  mcmc_hist(pars = c("Female_HSC-HM", "Male_HSC-HM", "Female_HSC-LM", "Male_HSC-LM",
                     "Female_LSC-HM", "Male_LSC-HM", "Female_LSC-LM", "Male_LSC-LM"),
            facet_args = list(nrow=4, ncol=2)) +
  xlim(0,50) +
  plot_title +
  theme(plot.title=element_text(hjust=0.5),
        plot.subtitle=element_text(hjust=0.5))
```

Example 2

Examining independence using two-way tables

Is smoking associated with socioeconomic (SES) status?

In a study of heart disease in male federal employees, researchers classified 356 volunteers according to their socioeconomic status (SES) and their smoking habits

There were three categories of SES: *high*, *middle*, and *low*

Individuals were asked whether they were *current* smokers, *former* smokers, or had *never* smoked, producing three categories for smoking habits as well

The researchers suspected that SES helps explain smoking, so in this situation SES is the explanatory variable and smoking is the response variable

Observed counts for smoking and SES

Smoking	High	Middle	Low	Total
Current	51	22	43	116
Former	92	21	28	141
Never	68	9	22	99
Total	211	52	93	356

* Example from Intro to Practice of Statistics, 3rd Edition pg. 635

Two-way table

The rows and columns of a two-way table represent values of two categorical variables

A two-way table with *r* rows and *c* columns contains r × c cells

In this example, the objective is to compare the conditional distributions of the response variable for each value of the explanatory variable. That is, *compare the column percents that give the conditional distribution of smoking for each SES category*

The two-way table in this example does not compare several populations. Instead, it arises by classifying observations on a single population in two ways

Observed counts for smoking and SES

Smoking	High	Middle	Low	Total
Current	**51**	**22**	**43**	116
Former	**92**	**21**	**28**	141
Never	**68**	**9**	**22**	99
Total	211	52	93	356

Two-way table

The rows and columns of a two-way table represent values of two categorical variables

A two-way table with *r* rows and *c* columns contains r × c cells

In this example, the objective is to compare the conditional distributions of the response variable for each value of the explanatory variable. That is, *compare the column percents that give the conditional distribution of smoking for each SES category*

The two-way table in this example does not compare several populations. Instead, it arises by classifying observations on a single population in two ways

```
> # Observed counts for smoking habits and socioeconomic status (ses)
> smoking <- c( rep("Current", 51), rep("Former",  92), rep("Never", 68),
+               rep("Current", 22), rep("Former",  21), rep("Never", 9),
+               rep("Current", 43), rep("Former",  28), rep("Never", 22) )
>
> ses  <- factor( c( rep("High", 211), rep("Middle",  52), rep("Low", 93) ) ,
+                 levels = c("High", "Middle", "Low") )
>
> # Create table based on categorical variables
> table(smoking,ses)
         ses
smoking   High Middle Low
  Current   51     22  43
  Former    92     21  28
  Never     68      9  22
>
> # Add row and column totals
> addmargins( table(smoking,ses) )
         ses
smoking   High Middle Low Sum
  Current   51     22  43 116
  Former    92     21  28 141
  Never     68      9  22  99
  Sum      211     52  93 356
```

Describing relations in a two-way table

To describe relations between the variables, we compute and compare *percents*

The count in each cell can be viewed as a percent of:

- the **grand** total (*joint* distribution)
- the **row** total (*conditional* distribution)
- the **column** total (*conditional* distribution)

You must decide which percents are most appropriate in a specific problem

In this example, we are interested in the influence of SES on the distribution of smoking habits. To compare the SESs, we examine the *column* percents

The data reveal something interesting: *it appears there is a negative association between smoking and SES: higher-SES people tend to smoke less*

Column percents for smoking and SES

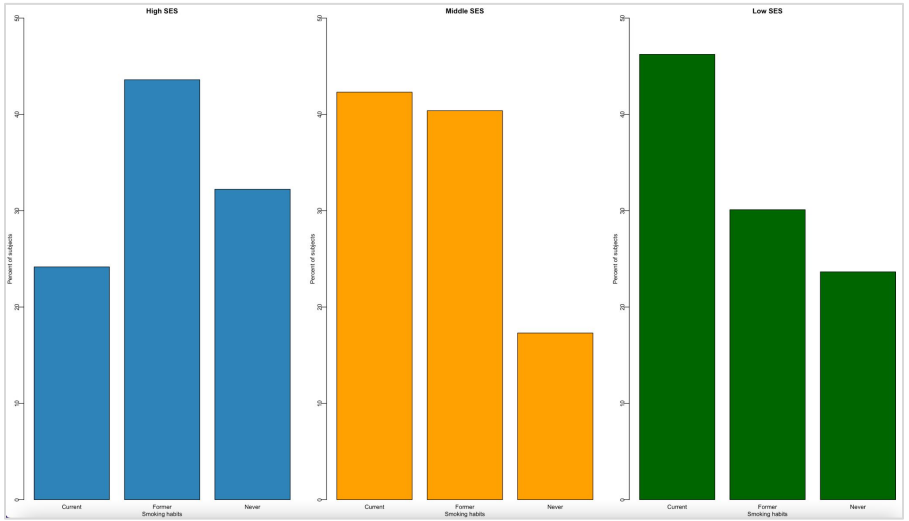
Smoking	High	Middle	Low	Total
Current	**24.2**	**42.3**	**46.2**	32.6
Former	**43.6**	**40.4**	**30.1**	39.6
Never	**32.2**	**17.3**	**23.7**	27.8
Total	100.0	100.0	100.0	100.0

Describing relations in a two-way table

The differences between the distributions of smoking by socioeconomic status in the sample appear to be large

A statistical test (and later a regression) will tell us whether or not these differences can be plausibly attributed to chance

Specifically, is the SES-smoking relationship in the sample sufficiently strong for us to conclude that it is due to a relationship between these two variables in the underlying population and not merely to chance



The null hypothesis: no association

Null hypothesis:
The null hypothesis H_O of interest in a two-way table is: there is *no association* between the row variable and the column variable

In the smoking habits – SES example, this null hypothesis says that smoking habits and SES *are not related*

Alternative hypothesis:
The alternative hypothesis H_a is that *there is an association* between these two variables

The alternative hypothesis H_a does not specify any particular direction for the association. We cannot describe H_a as one-sided or two-sided, because it includes all of the many kinds of association that are possible

In the smoking habits – SES example, this alternative hypothesis says that smoking habits and SES *are related* in some way

Expected cell counts

To test the null hypothesis in r × c tables, we compare the *observed* cell counts with **expected cell counts** calculated under the assumption that the null hypothesis is true:

expected cell count = row total × column total / N

In this example, 32.58% of all respondents (high, middle, & low SES) are in the Current smoker group. If the null hypothesis of no SES difference in smoking habits is true, we expect this overall percentage to apply to all SES categories. So, we expect 32.58% of the 211 high SES people in the study to be in this group: 0.3258 * 211 = 68.8

The other expected counts are calculated similarly

In this example, the number of males is the same as the number of females, so the expected counts for males are the same as the expected counts for females

A numerical summary of the comparison between *observed* and *expected* counts will be our test statistic

```
> # Add row and column totals
> addmargins( table(smoking,ses) )
         ses
smoking   High Middle Low Sum
  Current   51     22  43 116
  Former    92     21  28 141
  Never     68      9  22  99
  Sum      211     52  93 356
>
> round( addmargins(table(smoking,ses)) / sum(table(smoking,ses)) * 100 , 2 )
         ses
smoking     High Middle    Low    Sum
  Current  14.33   6.18  12.08  32.58
  Former   25.84   5.90   7.87  39.61
  Never    19.10   2.53   6.18  27.81
  Sum      59.27  14.61  26.12 100.00
```

```
> # Expected cell counts in High SES column "by hand": row total * column total / total number of observations
> round( margin.table( table(smoking,ses) , margin = 1 ) * margin.table(table(smoking,ses), margin=2 )[1] / sum( table(smoking,ses) ) , digits=1 )
smoking
Current  Former   Never
   68.8    83.6    58.7
> # Expected cell counts in Middle SES column "by hand": row total * column total / total number of observations
> round( margin.table( table(smoking,ses) , margin = 1 ) * margin.table(table(smoking,ses), margin=2 )[2] / sum( table(smoking,ses) ) , digits=1 )
smoking
Current  Former   Never
   16.9    20.6    14.5
> # Expected cell counts in Low SES column "by hand": row total * column total / total number of observations
> round( margin.table( table(smoking,ses) , margin = 1 ) * margin.table(table(smoking,ses), margin=2 )[3] / sum( table(smoking,ses) ) , digits=1 )
smoking
Current  Former   Never
   30.3    36.8    25.9
> # Expected cell counts for all cells with gmodels::CrossTable
> # https://stackoverflow.com/questions/34214787/is-there-an-r-function-to-get-a-table-of-expected-counts/34214881
> CrossTable( table(smoking,ses) , digits = 1, expected = TRUE , prop.r = FALSE , prop.c = FALSE , prop.t = FALSE , prop.chisq = FALSE )


   Cell Contents
|-------------------------|
|                       N |
|              Expected N |
|-------------------------|


Total Observations in Table:  356


             | ses
      smoking |      High |    Middle |       Low | Row Total |
--------------|-----------|-----------|-----------|-----------|
      Current |        51 |        22 |        43 |       116 |
              |      68.8 |      16.9 |      30.3 |           |
--------------|-----------|-----------|-----------|-----------|
       Former |        92 |        21 |        28 |       141 |
              |      83.6 |      20.6 |      36.8 |           |
--------------|-----------|-----------|-----------|-----------|
        Never |        68 |         9 |        22 |        99 |
              |      58.7 |      14.5 |      25.9 |           |
--------------|-----------|-----------|-----------|-----------|
 Column Total |       211 |        52 |        93 |       356 |
--------------|-----------|-----------|-----------|-----------|
```

The chi-square test

To test the H_O of no association, we use a statistic that compares the set of *observed* counts with the set of *expected* counts

First, take the difference between each *observed* count and its corresponding *expected* count, and square these values so they are all 0 or positive. A large difference means less if it comes from a cell we think will have a large count, so divide each squared difference by the expected count, a kind of standardization. Finally, sum over all cells

The result is the *chi-square statistic*:

X^2: ∑(observed count – expected count)^2 / expected count

where "observed" represents an observed sample count, "expected" represents the expected count for the same cell, and the sum is over all r × c cells in the table

If the expected counts and the observed counts are very different, a large value of X^2 will result. So large values of X^2 provide evidence against the null hypothesis

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(smoking,ses) ) )

        Pearson's Chi-squared test

data:  table(smoking, ses)
X-squared = 18.51, df = 4, p-value = 0.0009808
```

```
# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )
```

```
   Cell Contents
|-------------------------|
|                       N |
|              Expected N |
|-------------------------|


Total Observations in Table:  356
```

smoking	ses: High	ses: Middle	ses: Low	Row Total
Current	51 68.8	22 16.9	43 30.3	116
Former	92 83.6	21 20.6	28 36.8	141
Never	68 58.7	9 14.5	22 25.9	99
Column Total	211	52	93	356

The chi-square test

The *chi-square statistic*:

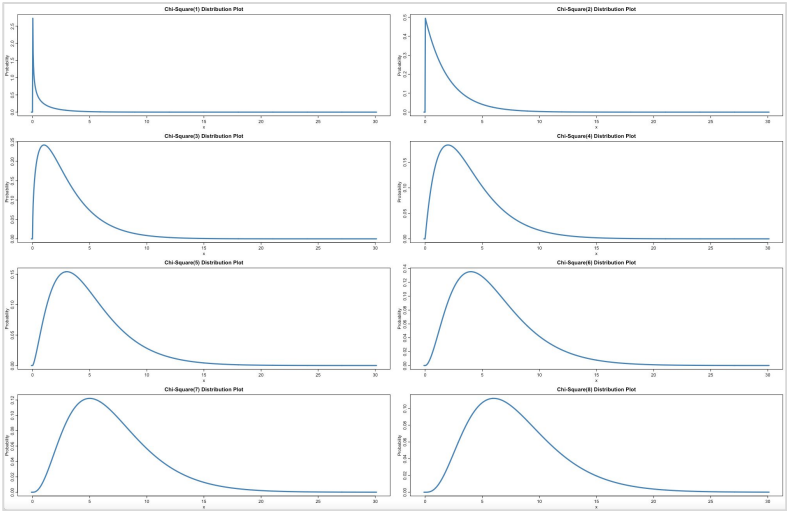
X^2: ∑(observed count – expected count)^2 / expected count

where “observed” represents an observed sample count, “expected” represents the expected count for the same cell, and the sum is over all r × c cells in the table

Like the t distributions, X^2 distributions form a family described by a single parameter, the degrees of freedom

If H_O is true, the X^2 statistic has approximately a X^2 distribution with **(r – 1)(c – 1) degrees of freedom**

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(smoking,ses) ) )

        Pearson's Chi-squared test

data:  table(smoking, ses)
X-squared = 18.51, df = 4, p-value = 0.0009808


# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )
```



The chi-square test

The *chi-square statistic*:

X^2: ∑(observed count – expected count)^2 / expected count

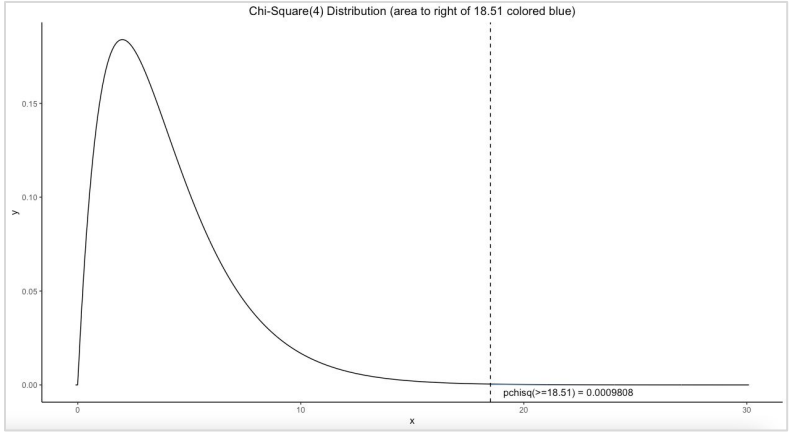
where "observed" represents an observed sample count, "expected" represents the expected count for the same cell, and the sum is over all r × c cells in the table

```
# Chi-Square Statistic "by hand"
sum( (Xsq$observed - Xsq$expected)^2 / Xsq$expected )


# Degrees of freedom
( degrees_of_freedom <- (nrow(table(smoking,ses)) - 1) * (ncol(table(smoking,ses)) - 1) )


# Critical values for this chi-square distribution
( crit10  <- qchisq( 0.90  , df = 4 ) )
( crit05  <- qchisq( 0.95  , df = 4 ) )
( crit025 <- qchisq( 0.975 , df = 4 ) )
( crit01  <- qchisq( 0.99  , df = 4 ) )
( crit001 <- qchisq( 0.999 , df = 4 ) )
```

Like the t distributions, X^2 distributions form a family described by a single parameter, the degrees of freedom

If H_O is true, the X^2 statistic has approximately a X^2 distribution with **(r – 1)(c – 1) degrees of freedom**

The *p-value* for the chi-square test is: **$P(\chi^2 \geq X^2)$**



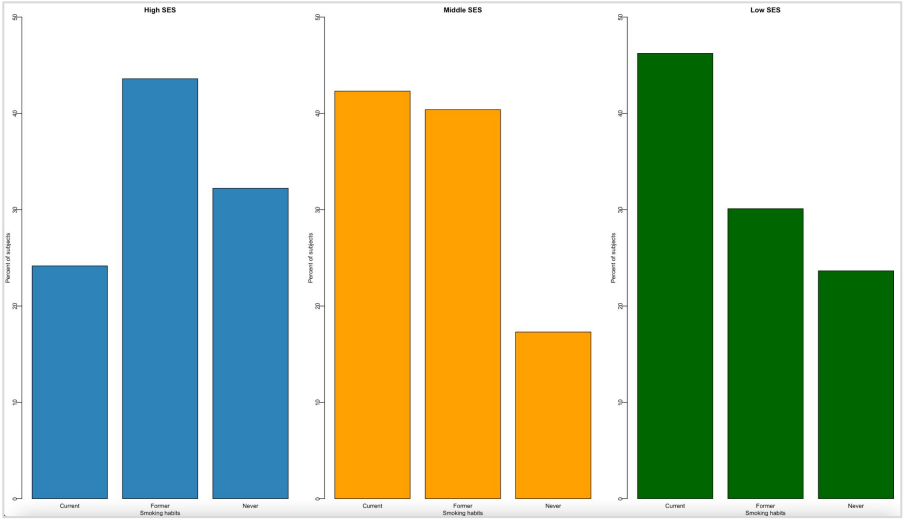
The chi-square test

The test indicates that the distributions of smoking habits across SES categories are not the same, but it does not say how they differ

Always combine the test with a description that shows what kind of relationship is present

From the side-by-side bar plots, we see that the percent of current smokers is higher among people with low and middle SES than among people with high SES. Similarly, the percent of people who have never smoked is higher among people with high SES than among people with low and middle SES

```
> # Chi-Square Test
> ( Xsq <- chisq.test( table(smoking,ses) ) )

        Pearson's Chi-squared test

data:  table(smoking, ses)
X-squared = 18.51, df = 4, p-value = 0.0009808
```



Framing chi square as classical regression

Data structure

- The cells of the table indicate the frequency with which each combination occurred in the sample
- Each respondent fell in one and only one cell of the table
- The data to be predicted are the cell counts
- The predictors are the nominal variables
- This structure is analogous to two-way analysis of variance (ANOVA), which also had two nominal predictors, but had metric values in each cell instead of a single count

Observed counts for smoking and SES

Smoking	High	Middle	Low	Total
Current	51	22	43	116
Former	92	21	28	141
Never	68	9	22	99
Total	211	52	93	356

Poisson exponential model

We can refer to the model we will use to describe these data as Poisson exponential, because the noise distribution is a Poisson distribution and the inverse-link function is exponential

Motivation 1

One way to motivate the model is to start with the two-way ANOVA model for combining nominal predictors, and find a way to map the predicted value to count data

The predicted μ from ANOVA can be any value from negative to positive infinity, but frequencies are non-negative

Therefore, we must transform the ANOVA predictions to non-negative values, while preserving order. A natural way to do this, mathematically, is with the exponential transformation

But this transformation only gets us to a continuous predicted value, not to the probability of discrete counts. A natural candidate for the needed likelihood distribution is the Poisson, which takes a non-negative λ and gives a probability for each integer from zero to infinity

Motivation 2

A different motivation starts by treating the cell counts as representative of underlying cell probabilities, and then asking whether the two nominal variables contribute independent influences to the cell probabilities

For example, in the table there's a particular marginal probability a respondent's smoking habit is Current, and a particular marginal probability a respondent's SES is High. If smoking habits and SES are independent, then the joint probability of Current and High is the product of the marginal probabilities

The attributes of smoking habit and SES are independent if that relationship holds for every cell in the table

Exponential link function

To check for independence of attributes, we need to estimate the marginal probabilities of the attributes

Denote the marginal (i.e., total) count of the *r*th row as y_r, and the marginal count of the *c*th column as y_c . Then the marginal proportions are y_r/N and y_c/N (where N is the total of the entire table)

If the attributes are independent, then the *predicted* joint probability, $\hat{p}(r,c)$, should equal the product of the marginal probabilities, which means

$$\hat{p}(r,c) = p(r) * p(c)$$

$$\hat{y}_{r,c}/N = y_r/N * y_c/N$$

Because the models we deal with involve *additive* combinations, not multiplicative combinations, we convert the multiplicative expression into an *additive* expression by using the facts:

$\log(a*b) = \log(a) + \log(b)$

$\exp(\ln(x)) = x$

From multiplicative to additive:

$$\hat{y}_{r,c}/N = y_r/N * y_c/N$$

$$\hat{y}_{r,c} = 1/N * y_r * y_c$$

$$\underbrace{\hat{y}_{r,c}}_{\lambda_{r,c}} = \exp(\underbrace{\log(1/N)}_{\beta_0} + \underbrace{\log(y_r)}_{\beta_r} + \underbrace{\log(y_c)}_{\beta_c})$$

Extended with interaction terms, the model of the cell tendencies is:

$$\lambda_{r,c} = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

where $\sum\beta_r = 0$, $\sum\beta_c = 0$, $\sum\beta_{r,c} = 0$ for all c, and $\sum\beta_{r,c} = 0$ for all r

If we're interested in violations of independence, then *our interest is on the magnitudes of the $\beta_{r,c}$ interaction terms, and specifically on meaningful interaction contrasts*

Poisson noise distribution

The value of $\lambda_{r,c}$ is a cell tendency, not a predicted count

In particular, the value of $\lambda_{r,c}$ can be any non-negative real value, but counts can only be integers

What we need is a likelihood function that maps the parameter value $\lambda_{r,c}$ to the probabilities of possible counts

The Poisson distribution is a natural choice:

$$p(y|\lambda) = \lambda^y \exp(-\lambda) / y!$$

where y is a non-negative integer and λ is is a non-negative real number

The mean of the Poisson distribution is λ and, importantly, the variance is also λ (i.e. the standard deviation is √λ)

Additive model:

$$\underbrace{\hat{y}_{r,c}}_{\lambda_{r,c}} = \exp(\underbrace{\log(1/N)}_{\beta_0} + \underbrace{\log(y_r)}_{\beta_r} + \underbrace{\log(y_c)}_{\beta_c})$$

We will use the Poisson distribution as the likelihood function for modeling the probability of the observed count, $y_{r,c}$, given the mean, $\lambda_{r,c}$

The idea is that each particular *r*, *c* combination has an underlying average rate of occurrence, $\lambda_{r,c}$

We collect data for a period of time, during which we happen to observed particular frequencies, $y_{r,c}$, of each combination. *From the observed frequencies, we infer the underlying average rates*

Framing chi square as classical regression in R

Now let's see how to recast the chi square test as a linear regression model in R:

```
# Minimal model (expected frequency)
m0.1 <- glm(count ~ 1, family = poisson(link = "log"),
            data = smoking_ses)
tidy(m0.1)
exp( tidy(m0.1) %>% pull(estimate) )

# 2. Additive model (expected frequencies)
m0.2 <- glm(count ~ 1 + smoking + ses, family = poisson(link = "log"),
            data = smoking_ses)
tidy(m0.2)
# reference category: Current and High
exp( tidy(m0.2) %>% filter(term == "(Intercept)") %>% pull(estimate) )

# 3. Saturated model (equal to observed frequencies)
m0.3 <- glm(count ~ 1 + smoking + ses + smoking:ses,
            family = poisson(link = "log"),
            data = smoking_ses)

tidy(m0.3)
# reference category: Current and High
# equal to observed frequency, 51
exp( tidy(m0.3) %>% filter(term == "(Intercept)") %>% pull(estimate) )
```

Untransformed coefficients (on logarithmic scale)

	Minimal model		Additive model		Saturated model	
Predictors	*Log-Mean*	*p*	*Log-Mean*	*p*	*Log-Mean*	*p*
(Intercept)	3.68 (0.05)	**<0.001**	4.23 (0.10)	**<0.001**	3.93 (0.14)	**<0.001**
smoking [Former]			0.20 (0.13)	0.119	0.59 (0.17)	**0.001**
smoking [Never]			-0.16 (0.14)	0.247	0.29 (0.19)	0.120
ses [Low]			-0.82 (0.12)	**<0.001**	-0.17 (0.21)	0.410
ses [Middle]			-1.40 (0.15)	**<0.001**	-0.84 (0.26)	**0.001**
smoking [Former] * ses [Low]					-1.02 (0.30)	**0.001**
smoking [Never] * ses [Low]					-0.96 (0.32)	**0.003**
smoking [Former] * ses [Middle]					-0.64 (0.35)	0.070
smoking [Never] * ses [Middle]					-1.18 (0.44)	**0.007**
Observations	9		9		9	
R^2 Nagelkerke	0.000		1.000		1.000	

Lack of independence is captured by the model's interaction terms.

Framing chi square as classical regression in R

Now let's see how to recast the chi square test as a linear regression model in R:

```
# Minimal model (expected frequency)
m0.1 <- glm(count ~ 1, family = poisson(link = "log"),
            data = smoking_ses)
tidy(m0.1)
exp( tidy(m0.1) %>% pull(estimate) )

# 2. Additive model (expected frequencies)
m0.2 <- glm(count ~ 1 + smoking + ses, family = poisson(link = "log"),
            data = smoking_ses)
tidy(m0.2)
# reference category: Current and High
exp( tidy(m0.2) %>% filter(term == "(Intercept)") %>% pull(estimate) )

# 3. Saturated model (equal to observed frequencies)
m0.3 <- glm(count ~ 1 + smoking + ses + smoking:ses,
            family = poisson(link = "log"),
            data = smoking_ses)

tidy(m0.3)
# reference category: Current and High
# equal to observed frequency, 51
exp( tidy(m0.3) %>% filter(term == "(Intercept)") %>% pull(estimate) )
```

Transformed (exponentiated) coefficients (multiplicative effects)

	Minimal model		Additive model		Saturated model	
Predictors	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*
(Intercept)	39.56 (2.10)	**<0.001**	68.75 (7.06)	**<0.001**	51.00 (7.14)	**<0.001**
smoking [Former]			1.22 (0.15)	0.119	1.80 (0.31)	**0.001**
smoking [Never]			0.85 (0.12)	0.247	1.33 (0.25)	0.120
ses [Low]			0.44 (0.05)	**<0.001**	0.84 (0.17)	0.410
ses [Middle]			0.25 (0.04)	**<0.001**	0.43 (0.11)	**0.001**
smoking [Former] * ses [Low]					0.36 (0.11)	**0.001**
smoking [Never] * ses [Low]					0.38 (0.12)	**0.003**
smoking [Former] * ses [Middle]					0.53 (0.19)	0.070
smoking [Never] * ses [Middle]					0.31 (0.13)	**0.007**
Observations	9		9		9	
R^2 Nagelkerke	0.000		1.000		1.000	

Lack of independence is captured by the model's interaction terms.

From Poisson regression coefficients to cell tendencies

The coefficients can be combined to calculate the expected cell tendencies and to retrodict the counts in each of the cells:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\mathbf{X_i\beta})$$

Current High = exp(`(Intercept)`)

Current Middle = exp(`(Intercept)` + `sesMiddle`)

Current Low = exp(`(Intercept)` + `sesLow`)

Former High = exp(`(Intercept)` + `smokingFormer`)

Former Middle = exp(`(Intercept)` + `smokingFormer` + `sesMiddle` + `smokingFormer:sesMiddle`)

Former Low = exp(`(Intercept)` + `smokingFormer` + `sesLow` + `smokingFormer:sesLow`)

Never High = exp(`(Intercept)` + `smokingNever`)

Never Middle = exp(`(Intercept)` + `smokingNever` + `sesMiddle` + `smokingNever:sesMiddle`)

Never Low = exp(`(Intercept)` + `smokingNever` + `sesLow` + `smokingNever:sesLow`)

```
> # 3. Saturated model (equal to observed frequencies)
> m0.3 <- glm(count ~ 1 + smoking + ses + smoking:ses,
+             family = poisson(link = "log"),
+             data = smoking_ses)
> tidy(m0.3)
# A tibble: 9 × 5
  term                    estimate std.error statistic   p.value
  <chr>                      <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept)                3.93      0.140    28.1    1.78e-173
2 smokingFormer              0.590     0.175     3.38   7.27e-  4
3 smokingNever               0.288     0.185     1.55   1.20e-  1
4 sesLow                    -0.171     0.207    -0.824  4.10e-  1
5 sesMiddle                 -0.841     0.255    -3.30   9.80e-  4
6 smokingFormer:sesLow      -1.02      0.299    -3.41   6.57e-  4
7 smokingNever:sesLow       -0.958     0.321    -2.98   2.84e-  3
8 smokingFormer:sesMiddle   -0.636     0.351    -1.81   7.02e-  2
9 smokingNever:sesMiddle    -1.18      0.437    -2.70   6.84e-  3
>
> # Exponentiated fitted values from the saturated model equal the observed counts
> matrix( sapply(augment(m0.3)$.fitted, FUN = exp) ,
+         nrow = 3 , ncol = 3 , byrow = TRUE )
     [,1] [,2] [,3]
[1,]   51   22   43
[2,]   92   21   28
[3,]   68    9   22
>
> # Observed counts
> Xsq$observed
         ses
smoking   High Middle Low
  Current   51     22  43
  Former    92     21  28
  Never     68      9  22
```

Interpreting Poisson regression coefficients

The coefficients can be exponentiated and treated as multiplicative effects:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

Intercept:

The coefficient estimate for the Intercept (3.93) gives the prediction (on the logarithmic scale) if $X_{i1} = 0$ and $X_{i2} = 0$. Exponentiating this value (exp(3.93) = 51) produces the cell tendency, $\lambda_{r,c}$, for the CurrentSmoker HighSES cell in the table

sesLow:

The coefficient estimate for sesLow (–0.171) is the expected difference in y (on the logarithmic scale). The multiplicative decrease is $e^{-0.171} = 0.84$, which means (exp(3.93 – 0.171) = 51 * 0.84 = 43) is the cell tendency, $\lambda_{r,c}$, for the CurrentSmoker LowSES cell in the table. The calculation for CurrentSmoker MiddleSES is similar

smokingNever:

The coefficient estimate for smokingNever (0.288) tells us the predictive difference for the NeverSmoker HighSES cell in the table, which means (exp(3.93 + 0.288) = 51 * 1.33 = 68) is the cell tendency, $\lambda_{r,c}$, for the NeverSmoker HighSES cell in the table

smokingNever:sesLow:

The coefficient estimate for smokingNever:sesLow (–0.958) tells us the predictive difference for the NeverSmoker LowSES cell in the table. The multiplicative decrease is $e^{-0.958} = 0.38$, which means (exp(3.93 + 0.288 – 0.171 – 0.958) = 51 * 1.33 * 0.84 * 0.38 = 22) is the cell tendency, $\lambda_{r,c}$, for the NeverSmoker LowSES cell in the table. The calculation for FormerSmoker and MiddleSES cells is similar

```
> tidy(m0.3)
# A tibble: 9 x 5
  term                      estimate std.error statistic   p.value
  <chr>                        <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept)                  3.93      0.140     28.1   1.78e-173
2 smokingFormer                0.590     0.175      3.38  7.27e-  4
3 smokingNever                 0.288     0.185      1.55  1.20e-  1
4 sesLow                      -0.171     0.207     -0.824 4.10e-  1
5 sesMiddle                   -0.841     0.255     -3.30  9.80e-  4
6 smokingFormer:sesLow        -1.02      0.299     -3.41  6.57e-  4
7 smokingNever:sesLow         -0.958     0.321     -2.98  2.84e-  3
8 smokingFormer:sesMiddle     -0.636     0.351     -1.81  7.02e-  2
9 smokingNever:sesMiddle      -1.18      0.437     -2.70  6.84e-  3
```

	Minimal model		Additive model		Saturated model	
Predictors	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*	*Incidence Rate Ratios*	*p*
(Intercept)	39.56 (2.10)	**<0.001**	68.75 (7.06)	**<0.001**	51.00 (7.14)	**<0.001**
smoking [Former]			1.22 (0.15)	0.119	1.80 (0.31)	**0.001**
smoking [Never]			0.85 (0.12)	0.247	1.33 (0.25)	0.120
ses [Low]			0.44 (0.05)	**<0.001**	0.84 (0.17)	0.410
ses [Middle]			0.25 (0.04)	**<0.001**	0.43 (0.11)	**0.001**
smoking [Former] * ses [Low]					0.36 (0.11)	**0.001**
smoking [Never] * ses [Low]					0.38 (0.12)	**0.003**
smoking [Former] * ses [Middle]					0.53 (0.19)	0.070
smoking [Never] * ses [Middle]					0.31 (0.13)	**0.007**
Observations	9		9		9	
R^2 Nagelkerke	0.000		1.000		1.000	

Framing chi square as Bayesian regression

Framing chi square as Bayesian regression in R

Now let's see how to recast the chi square test as a linear regression model in R:

```
# Estimate Bayesian version with stan_glm
# 3. Saturated model (equal to observed frequencies)
b0.3 <- stan_glm(count ~ 1 + smoking + ses + smoking:ses,
                 family = poisson(link = "log"), data = smoking_ses)
tidy(b0.3)
```

```
> tidy(b0.3)
# A tibble: 9 × 3
  term                   estimate std.error
  <chr>                     <dbl>     <dbl>
1 (Intercept)               3.93      0.137
2 smokingFormer             0.586     0.170
3 smokingNever              0.286     0.182
4 sesLow                   -0.177     0.197
5 sesMiddle                -0.858     0.251
6 smokingFormer:sesLow     -1.02      0.304
7 smokingNever:sesLow      -0.958     0.322
8 smokingFormer:sesMiddle  -0.629     0.353
9 smokingNever:sesMiddle   -1.20      0.444
```

Lack of independence is captured by the model's interaction terms.

Untransformed coefficients (on logarithmic scale)

	Classical model	Bayesian model
Predictors	*Log-Mean*	*Log-Mean*
(Intercept)	3.93 (0.14)	3.93 (0.14)
smoking: Former	0.59 (0.17)	0.59 (0.17)
smoking: Never	0.29 (0.19)	0.29 (0.18)
ses: Low	-0.17 (0.21)	-0.18 (0.20)
ses: Middle	-0.84 (0.26)	-0.86 (0.25)
smokingFormer:sesLow	-1.02 (0.30)	-1.02 (0.30)
smokingNever:sesLow	-0.96 (0.32)	-0.96 (0.32)
smokingFormer:sesMiddle	-0.64 (0.35)	-0.63 (0.35)
smokingNever:sesMiddle	-1.18 (0.44)	-1.20 (0.44)

Transformed (exponentiated) coefficients (multiplicative effects)

	Classical model	Bayesian model
Predictors	*Incidence Rate Ratios*	*Incidence Rate Ratios*
(Intercept)	51.00 (7.14)	50.74 (6.95)
smoking: Former	1.80 (0.31)	1.80 (0.30)
smoking: Never	1.33 (0.25)	1.33 (0.24)
ses: Low	0.84 (0.17)	0.84 (0.16)
ses: Middle	0.43 (0.11)	0.42 (0.11)
smokingFormer:sesLow	0.36 (0.11)	0.36 (0.11)
smokingNever:sesLow	0.38 (0.12)	0.38 (0.12)
smokingFormer:sesMiddle	0.53 (0.19)	0.53 (0.18)
smokingNever:sesMiddle	0.31 (0.13)	0.30 (0.13)

From Poisson regression coefficients to cell tendencies

The coefficients can be combined to calculate the expected cell tendencies and to retrodict the counts in each of the cells:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\mathbf{X}_i\boldsymbol{\beta})$$

Current High = exp(`(Intercept)`)

Current Middle = exp(`(Intercept)` + `sesMiddle`)

Current Low = exp(`(Intercept)` + `sesLow`)

Former High = exp(`(Intercept)` + `smokingFormer`)

Former Middle = exp(`(Intercept)` + `smokingFormer` + `sesMiddle` + `smokingFormer:sesMiddle`)

Former Low = exp(`(Intercept)` + `smokingFormer` + `sesLow` + `smokingFormer:sesLow`)

Never High = exp(`(Intercept)` + `smokingNever`)

Never Middle = exp(`(Intercept)` + `smokingNever` + `sesMiddle` + `smokingNever:sesMiddle`)

Never Low = exp(`(Intercept)` + `smokingNever` + `sesLow` + `smokingNever:sesLow`)

```
> # Coefficients from the saturated model (on logarithmic scale)
> matrix( round(b0.3$coefficients,2) , nrow = 3 , ncol = 3 , byrow = TRUE )
      [,1]  [,2]  [,3]
[1,]  3.93  0.59  0.29
[2,] -0.18 -0.86 -1.02
[3,] -0.96 -0.63 -1.20
>
> # Exponentiated coefficients from the saturated model (multiplicative effects)
> matrix( round(exp(b0.3$coefficients),2) , nrow = 3 , ncol = 3 , byrow = TRUE )
      [,1] [,2] [,3]
[1,] 50.74 1.80 1.33
[2,]  0.84 0.42 0.36
[3,]  0.38 0.53 0.30
>
> # Exponentiated fitted values from the saturated model equal the observed counts
> matrix( round(b0.3$fitted) , nrow = 3 , ncol = 3 , byrow = TRUE )
     [,1] [,2] [,3]
[1,]   51   22   43
[2,]   91   21   28
[3,]   68    9   22
>
> # Observed counts
> Xsq$observed
          ses
smoking    High Middle Low
  Current   51     22  43
  Former    92     21  28
  Never     68      9  22
```

Interpreting Poisson regression coefficients

The coefficients can be exponentiated and treated as multiplicative effects:

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\lambda_i = \exp(\beta_0 + \beta_r + \beta_c + \beta_{r,c})$$

Intercept:

The coefficient estimate for the Intercept (3.93) gives the prediction (on the logarithmic scale) if $X_{i1} = 0$ and $X_{i2} = 0$. Exponentiating this value (exp(3.93) = 51) produces the cell tendency, $\lambda_{r,c}$, for the CurrentSmoker HighSES cell in the table

sesLow:

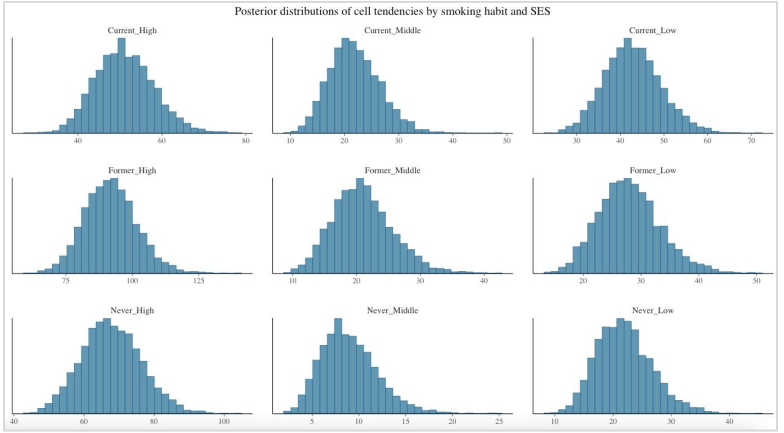
The coefficient estimate for sesLow (–0.171) is the expected difference in y (on the logarithmic scale). The multiplicative decrease is $e^{-0.171} = 0.84$, which means (exp(3.93 – 0.171) = 51 * 0.84 = 43) is the cell tendency, $\lambda_{r,c}$, for the CurrentSmoker LowSES cell in the table. The calculation for CurrentSmoker MiddleSES is similar

smokingNever:

The coefficient estimate for smokingNever (0.288) tells us the predictive difference for the NeverSmoker HighSES cell in the table, which means (exp(3.93 + 0.288) = 51 * 1.33 = 68) is the cell tendency, $\lambda_{r,c}$, for the NeverSmoker HighSES cell in the table

smokingNever:sesLow:

The coefficient estimate for smokingNever:sesLow (–0.958) tells us the predictive difference for the NeverSmoker LowSES cell in the table. The multiplicative decrease is $e^{-0.958} = 0.38$, which means (exp(3.93 + 0.288 – 0.171 – 0.958) = 51 * 1.33 * 0.84 * 0.38 = 22) is the cell tendency, $\lambda_{r,c}$, for the NeverSmoker LowSES cell in the table. The calculation for FormerSmoker and MiddleSES cells is similar




```
# Posterior distributions of cell tendencies convey uncertainty
plot_title <- ggtitle("Posterior distributions of cell tendencies by smoking habit and SES")

as.data.frame(b0.3) %>%
  mutate(`Current_High` = exp(`(Intercept)`),
         `Current_Middle` = exp(`(Intercept)` + `sesMiddle`),
         `Current_Low` = exp(`(Intercept)` + `sesLow`),
         `Former_High` = exp(`(Intercept)` + `smokingFormer`),
         `Former_Middle` = exp(`(Intercept)` + `smokingFormer` + `sesMiddle` + `smokingFormer:sesMiddle`),
         `Former_Low` = exp(`(Intercept)` + `smokingFormer` + `sesLow` + `smokingFormer:sesLow`),
         `Never_High` = exp(`(Intercept)` + `smokingNever`),
         `Never_Middle` = exp(`(Intercept)` + `smokingNever` + `sesMiddle` + `smokingNever:sesMiddle`),
         `Never_Low` = exp(`(Intercept)` + `smokingNever` + `sesLow` + `smokingNever:sesLow`) ) %>%
  mcmc_hist(pars = c("Current_High", "Current_Middle", "Current_Low",
                     "Former_High", "Former_Middle", "Former_Low",
                     "Never_High", "Never_Middle", "Never_Low"),
            facet_args = list(nrow=3, ncol=3)) +
  #xlim(0,150) +
  plot_title +
  theme(plot.title=element_text(hjust=0.5),
        plot.subtitle=element_text(hjust=0.5))
```

Overdispersion and Exposure

Poisson regression

The simplest regression model for count data is:

$$y_i \sim \text{Poisson}(\lambda_i)$$

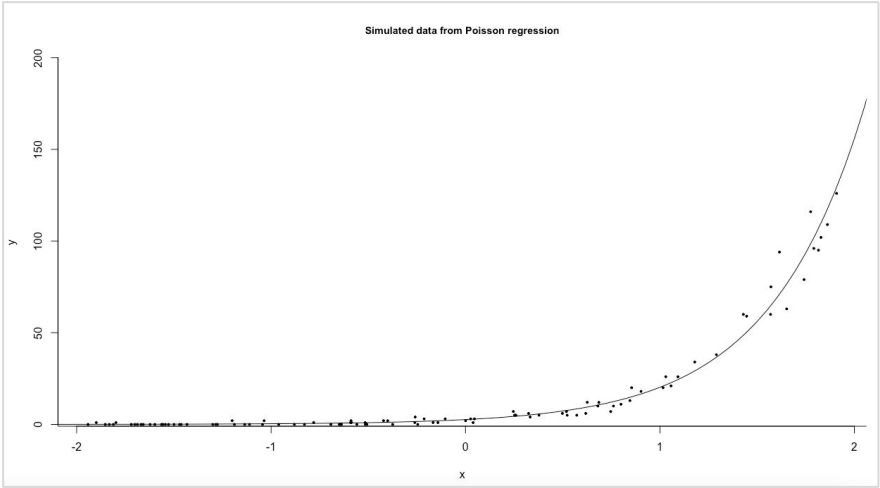
$$\lambda_i = \exp(\mathbf{X_i\beta})$$

So, the linear predictor $X_i\beta$ is the logarithm of the expected value of measurement y_i

Under the Poisson model, $sd(y_i) = \sqrt{E(y_i)}$; thus if the model accurately describes the data, we also have a sense of how much variation we would expect from the fitted curve

The Poisson distribution has its own internal scale of variation: unlike with the normal distribution, there is no `sigma` parameter to be fit. From the Poisson distribution, we expect variation on the order of $\sqrt{E(y_i)}$

For example, where expected number of counts is 10, prediction errors should be mostly in the range ± 3, where expected number of counts is 100, prediction errors should be mostly in the range ± 10, and so on



Negative binomial model for overdispersion

Overdispersion refers to data that show more variation than expected based on a fitted probability model

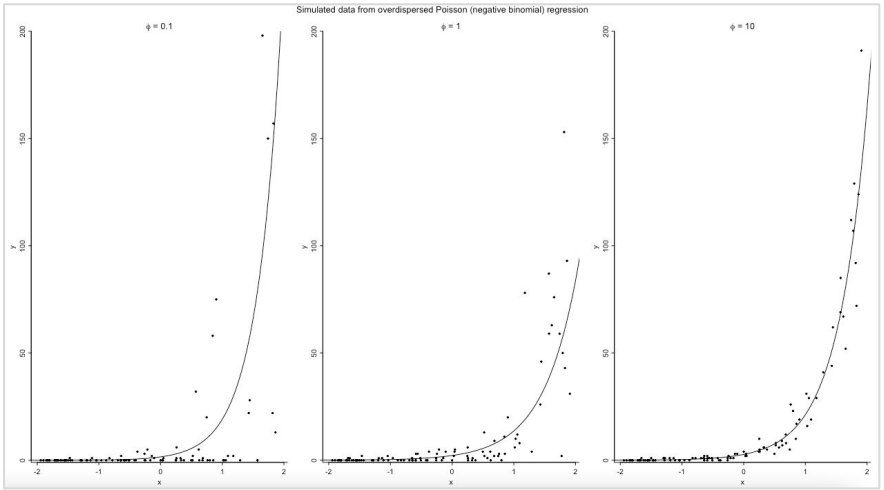
To generalize the Poisson model to allow for overdispersion, we use the *negative binomial* – a probability distribution that includes an additional "reciprocal dispersion" parameter ϕ so that

$$sd(y|x) = \sqrt{[\, E(y|x) + E(y|x)^2 / \phi \,]}$$

where ϕ is restricted to be positive, with lower values corresponding to more overdispersion, and the limit $\phi \to \infty$ representing the Poisson model (i.e. zero overdispersion)

To get a sense of what this model looks like, let's simulate three datasets, using the coefficients and the same values of the predictor x as on the previous slide, but replacing the Poisson with the negative binomial distribution with parameter ϕ set to 0.1, 1, or 10

In the plots to the right, the curve of E(y|x) is the same for all three models, but *the variation of y is much higher when ϕ is near zero*



Exposure

In most applications of count-data regression, there is a baseline or **exposure**, some value such as the average flow of vehicles that travel through the intersection in the traffic accidents example

We can model y_i as the number of cases in a process with **rate** θ_i and **exposure** u_i:

$$y_i \sim \text{negative binomial}(u_i\theta_i, \phi)$$

where $\theta_i = e^{\mathbf{X_i\beta}}$, and the expression includes Poisson regression as a special case of $\phi \to \infty$

The logarithm of the exposure, $\log(u_i)$, is called the **offset** in generalized linear model terminology

Putting the logarithm of the exposure into the model as an offset is equivalent to including it as a regression predictor, but with its coefficient fixed to the value 1

Another option is to include it as a predictor and let its coefficient be estimated from the data

```
# Roaches: Pest Management
# ROS pg. 268
# https://mc-stan.org/rstanarm/articles/count.html
data(roaches)
(n <- nrow(roaches))

# Set random seed for reproducability
SEED <- 3579

# Scale the number of roaches by 100
roaches$roach100 <- roaches$roach1 / 100
head(roaches)

# Negative-binomial model is over-dispersed compared to Poisson
fit_1 <- stan_glm(y ~ roach100 + treatment + senior,
                  family=neg_binomial_2, offset=log(exposure2),
                  data=roaches, seed=SEED)
prior_summary(fit_1)
print(fit_1, digits=2)

loo_1 <- loo(fit_1)
```

Comparing Poisson and Negative Binomial

We want to make inferences about the efficacy of a certain pest management system at reducing the number of roaches in urban apartments*

Predictors:

- Intercept
- Pre-treatment number of roaches
- Treatment indicator
- Indicator for whether the apartment is in a building restricted to elderly residents
- Because the number of days for which the roach traps were used is not the same for all apartments in the sample, let's include it as an *exposure*

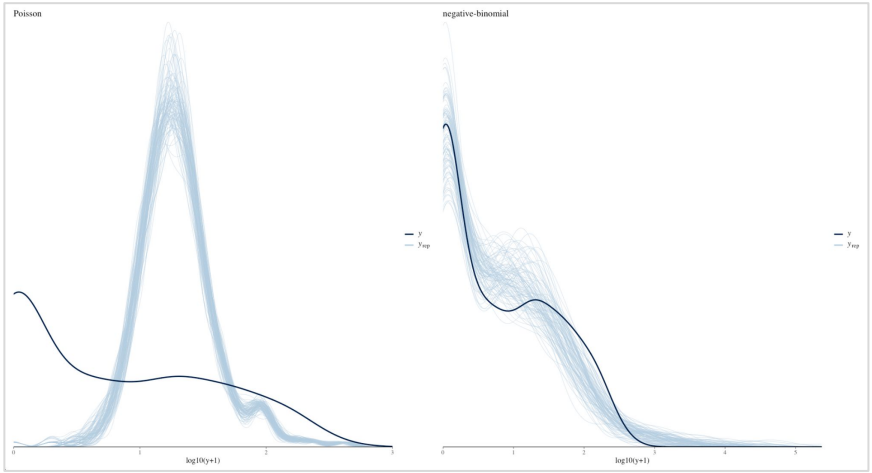
Posterior predictive checks comparing density plot of roaches data (dark line) to 100 predictive replications (light blue lines) of replicated data from
(a) a fitted Poisson model, and (b) a fitted negative binomial model

The *Poisson model shows strong lack of fit*, and the negative binomial model shows some lack of fit

When we comparing the models using the `loo` package we also see
a clear preference for the negative binomial model

```
# Poisson is a special case of negative-binomial
fit_2 <- stan_glm(y ~ roach100 + treatment + senior,
                  family=poisson, offset=log(exposure2),
                  data=roaches, seed=SEED)
prior_summary(fit_2)
print(fit_2, digits=2)
```

```
# Negative-binomial model is over-dispersed compared to Poisson
fit_1 <- stan_glm(y ~ roach100 + treatment + senior,
                  family=neg_binomial_2, offset=log(exposure2),
                  data=roaches, seed=SEED)
prior_summary(fit_1)
print(fit_1, digits=2)
```




```
> # Compare the two models
> loo_compare(loo_1, loo_2)
      elpd_diff se_diff
fit_1     0.0       0.0
fit_2 -5347.3     707.9
```

* Chapter 8.3 of Gelman and Hill (2007)

Appendix

Resources

Regression and Other Stories

Statistical Rethinking

Statistical rethinking with brms, ggplot2, and the tidyverse: Second edition

Bayes Rules!

Tidy Modeling with R

Doing Bayesian Data Analysis, Second edition

Doing Bayesian Data Analysis in brms and the tidyverse

rstanarm vignettes

bayesplot vignettes

R for Data Science

R Graphics Cookbook