Statistical Modeling, Winter 2025

Poisson GLM

From the binomial to the Poisson

The basic binomial model follows the form

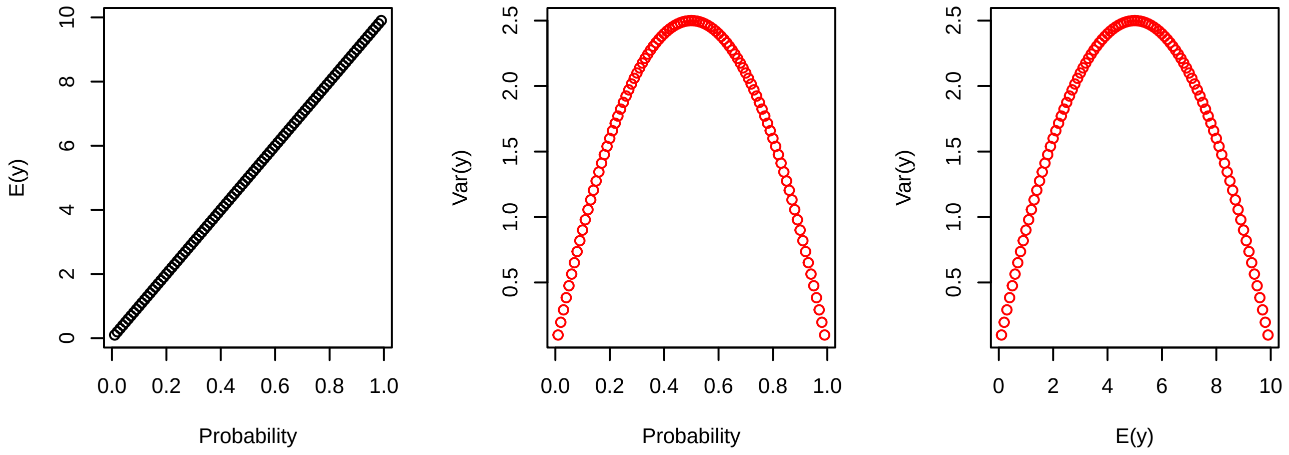
$$y \sim \text{Binomial}(n, p)$$

where y is some count variable, n is the number of trials, and p is the probability a given trial was a 1, which is sometimes termed a *success*. Therefore, we use the binomial when we know the number of successful counts, as well as the number of trials. The mean (E(y)) and variance (Var(y)) are given by:

$$\text{E}(y) = np$$
$$\text{Var}(y) = np(1 - p)$$

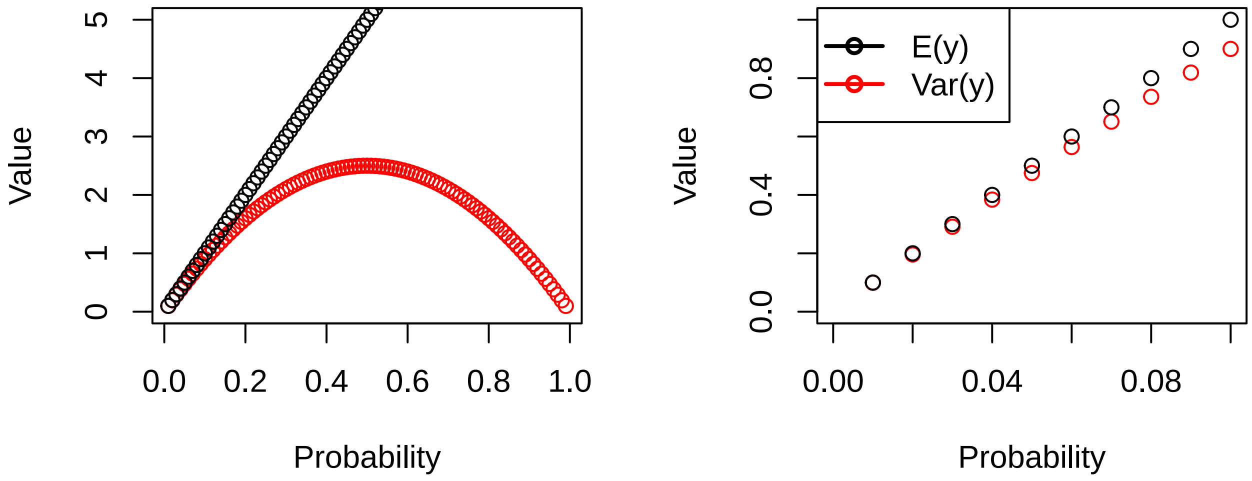
The consequence of these relationships is that the mean and variance are not independent, and that the variance is maximized when p = 0.5. Below is some code to demonstrate this:

```r
n <- 10; p_grid <- seq(0.01, 0.99, by = 0.01)
mean_y <- n*p_grid; var_y <- (n*p_grid)*(1-p_grid)
par(mfrow = c(1,3))
plot(p_grid, mean_y, type = "b", col = "black", xlab = "Probability", ylab = "E(y)")
plot(p_grid, var_y, type = "b", col = "red", xlab = "Probability", ylab = "Var(y)")
plot(mean_y, var_y, type = "b", col = "red", xlab = "E(y)", ylab = "Var(y)")
```



If we plot the mean and variance on the same plot (left), and then zoom in (right):

```r
par(mfrow = c(1,2))
plot(p_grid, var_y, type = "b", col = "red", ylim = c(0, 5),
     xlab = "Probability", ylab = "Value")
points(p_grid, mean_y, type = "p", col = "black")
plot(p_grid, var_y, type = "p", col = "red",  xlim = c(0, 0.1), ylim = c(0, 1),
     xlab = "Probability", ylab = "Value")
points(p_grid, mean_y, type = "p", col = "black")
legend("topleft", legend = c("E(y)", "Var(y)"), pch = 21, col = c("black", "red"), lwd = 2)
```



Notice that at low values of p ($p < 0.1$; right panel), the mean and variance are very similar. So if n is large (in this case not very large!), and the probability of success is low, we expect the mean and variance to be the same.

We can use the following example (McElreath 2020) to illustrate this:

- You own a monastery with 1000 monks who copy manuscripts (n = 1000)
- On average, 1 manuscript is finished each day (E(y) = 1)
- The probability that a monk finishes a manuscript is 1/1000 (p = 1/1000)

We can simulate the expected number of manuscripts according to a binomial:

```r
y <- rbinom(n = 1e5, size = 1000, prob = 1/1000)
c(mean(y), var(y))
```

```
[1] 1.001890 1.005516
```

The mean and variance are nearly identical, and these conditions (small p, big n). results in a special shape of the binomial - the Poisson.

Poisson regression

You might be asking yourself what the point of all that was - good question. The Poisson is useful because it allows us to model binomial events for which the number of trials is unknown or uncountably large. Following up on the earlier monks example:

1. You acquire a new monastery, but you don't know how many monks there are (n = ?)
2. It produces, on average, 2 manuscripts per day ($\lambda = 2$)
3. Now we can infer the distribution of numbers of manuscripts per day

The Poisson distribution gives the distribution of the number of discrete events/individiuals/counts etc., in a defined sample, if each event is independent of all others. The most common definition of the Poisson has only one parameter:

$$y \sim \text{Poisson}(\lambda)$$

where λ is the average density or event rate. When dealing with Poisson data, $\mu = \sigma^2$ - and the single parameter λ encapsulates both. Therefore, values well above the mean are highly improbable. Since λ is constrained to be positive, we typically use the log link. The basic Poisson regression model is

$$\begin{aligned} y_i &\sim \text{Poisson}(\lambda_i) \\ \log(\lambda_i) &= \alpha + \beta x_i \\ \lambda_i &= \exp(\alpha + \beta x_i) \end{aligned}$$

We can also think of the parameter λ as a rate, with a mean μ number of events that occur per unit time (or space) τ:

$$\begin{aligned} \log(\lambda_i) &= \log\left(\frac{\mu_i}{\tau_i}\right) \\ \log\left(\frac{\mu_i}{\tau_i}\right) &= \alpha + \beta x_i \end{aligned}$$

Which simplifies to:

$$\begin{aligned} \log(\mu_i) - \log(\tau_i) &= \alpha + \beta x_i \\ \log(\mu_i) &= \log(\tau_i) + \alpha + \beta x_i \end{aligned}$$

The term $\log(\tau_i)$ is called an *offset*, and permits the model to account for differences in the size or duration of sampling events. τ gets a column in the data, but is assumed to be measured without error and typically not treated as a random variable (i.e., there is no prior or likelihood for this term).

Exercise

1. Create a vector of x-values from -1 to 1. Now create a vector of y-values according to a linear regression with an intercept of 0 and slope of 2.

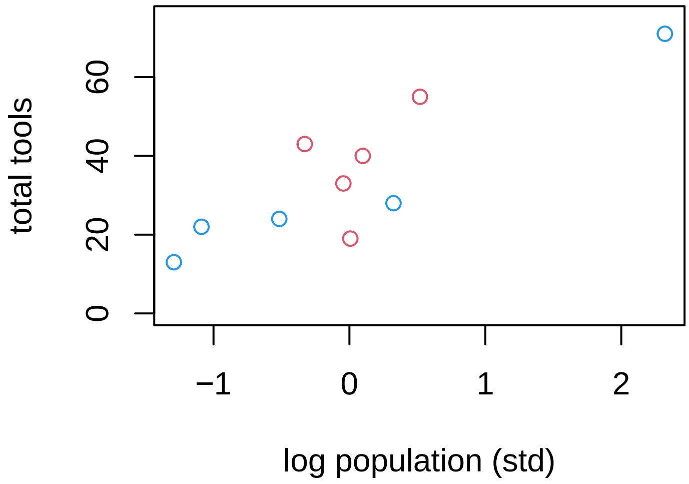
   a. Plot y-vector as a function of x-vector, and make a dashed line at 0 using `abline(h = 0, lty = 2)`.

   b. Make the same plot, but on the y-axis plot the `exp(y)`. This time make a dashed line with `abline(h = 1, lty = 2)`.

   c. What does this dashed line represent?

2. Load the oceanic tool complexity data in `rethinking` with `data(Kline)`.

```r
data(Kline)
d <- Kline
d$P <- scale(log(d$population))
d$contact_id <- ifelse( d$contact=="high" , 2 , 1 )
dat <- list(T = d$total_tools , P = d$P , C = d$contact_id )
plot( dat$P , dat$T , xlab="log population (std)" , ylab="total tools" ,
      col=ifelse( dat$C==1 , 4 , 2 ) , ylim=c(0,75)  )
```



For now, ignore the effect of `contact_id`. To start, you are going to fit this Poisson model to the data:

$$
\begin{aligned}
T_i &\sim \text{Poisson}(\lambda_i) \\
\log(\lambda_i) &= \alpha + \beta_P \log(\text{population}) \\
\alpha &\sim \text{Normal}(?, ?) \\
\beta_P &\sim \text{Normal}(?, ?)
\end{aligned}
$$

You will need to choose priors for α and β_P. But the exponential scaling makes it hard to intuit what your priors should be. As always, do a prior predictive check. Justify your priors for the intercept and slope. Run the model and interpret your results. Use the text and lecture slides for hints when you are stuck.

When you are feeling good about the fundamentals, work through the rest of the text/lecture code to (1) explore the effects of low- and high-contact islands on tool count, and (2) apply a scientific model to the tool data.

References

McElreath, Richard. 2020. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press.