Generalized Linear Models

"The trouble with normal is that it always gets worse"

Samuel Robinson, Ph.D.

Sept 29, 2023

Part 1: The exponential family

Outline

- Meet (some of) the exponential family!

Christmas gifts for the nerds in your life

Outline

- Meet (some of) the exponential family!
  - Normal

Christmas gifts for the nerds in your life

Outline

- Meet (some of) the exponential family!
  - Normal
  - Binomial

Christmas gifts for the nerds in your life

Outline

- Meet (some of) the exponential family!
    - Normal
    - Binomial
    - Poisson

Christmas gifts for the nerds in your life

Outline

- Meet (some of) the exponential family!
  - Normal
  - Binomial
  - Poisson
  - Beta-Binomial

Christmas gifts for the nerds in your life

Outline

- Meet (some of) the exponential family!
  - Normal
  - Binomial
  - Poisson
  - Beta-Binomial
  - Negative Binomial

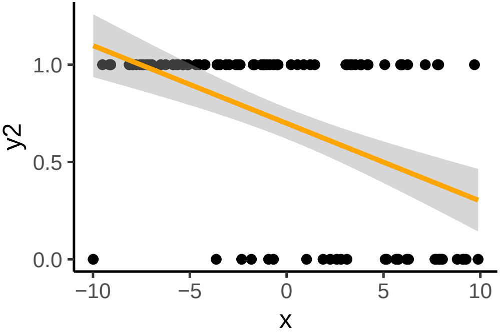
Christmas gifts for the nerds in your life

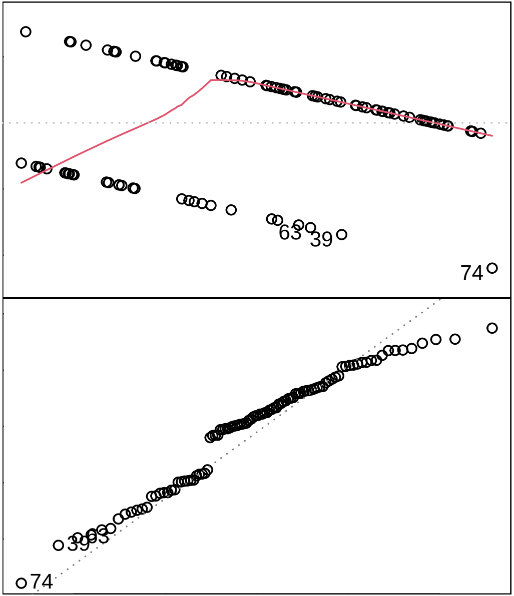
Outline

- Meet (some of) the exponential family!
  - Normal
  - Binomial
  - Poisson
  - Beta-Binomial
  - Negative Binomial
- "Play time"

Christmas gifts for the nerds in your life

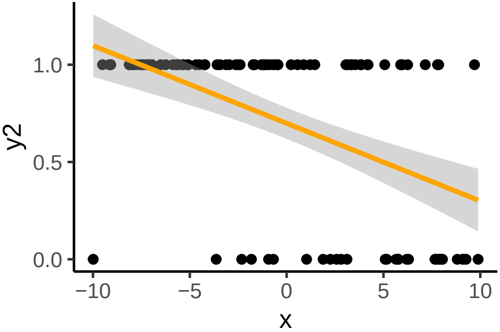
Problem: not everything is normal

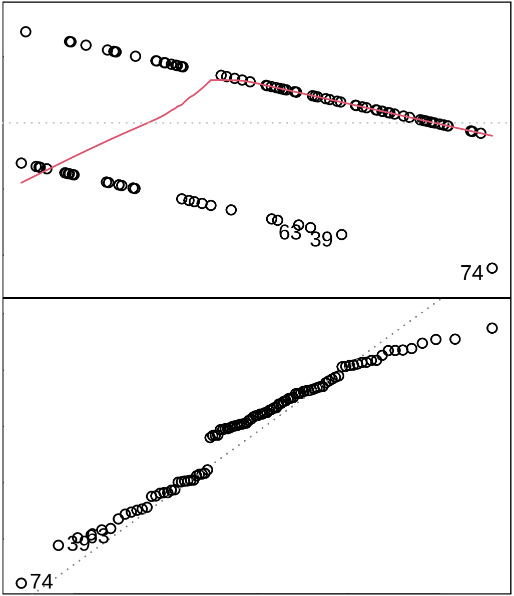




- Some types of data can never be transformed to make the residuals normal

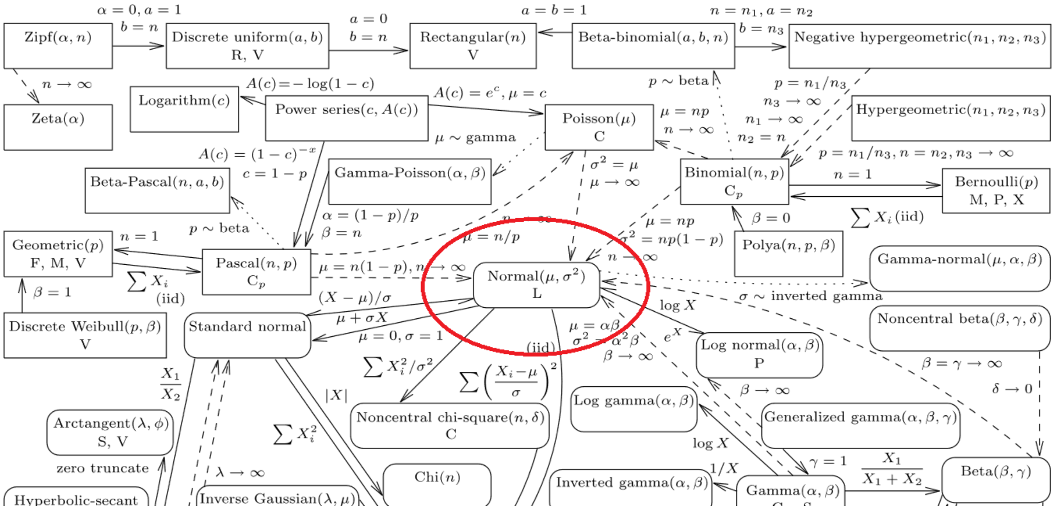
Problem: not everything is normal





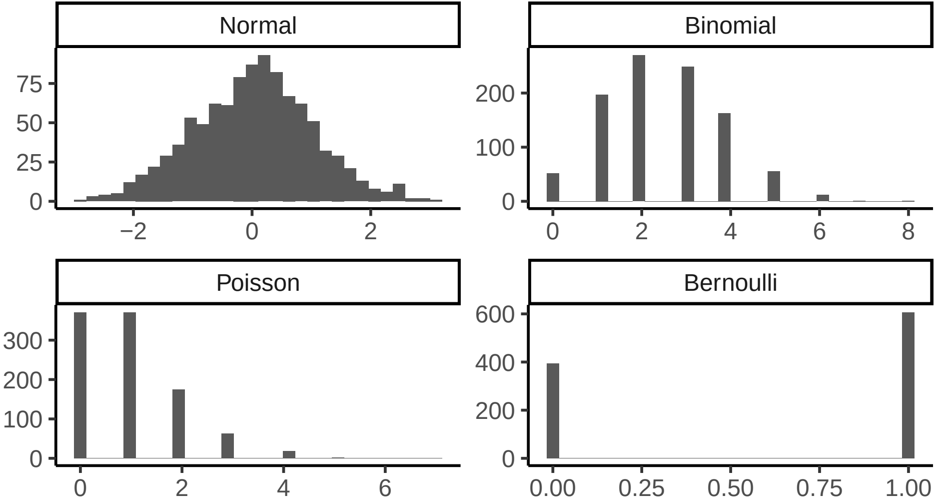
- Some types of data can never be transformed to make the residuals normal
- Solution: **use the distribution that generates the data!**

But how do I know which distribution to use?



And if thou gaze long into an abyss, the abyss will also gaze into thee - F. Nietzsche

Let's take a look at some *common* ones!



Time to meet the Exponential family!

The Normal Distribution (aka *Gaussian*)

- Imagine many random + and - numbers added together

A Galton Board in action:

The Normal Distribution (aka *Gaussian*)

- Imagine many random $+$ and - numbers added together
- If you do this *many* times:

A Galton Board in action:

The Normal Distribution (aka *Gaussian*)

- Imagine many random + and - numbers added together
- If you do this *many* times:
  - Most cancel out (somewhere around 0)

A Galton Board in action:

The Normal Distribution (aka *Gaussian*)

- Imagine many random + and - numbers added together
- If you do this *many* times:
  - Most cancel out (somewhere around 0)
  - Few are far away from 0 (tails of distribution)

A Galton Board in action:

The Normal Distribution (aka *Gaussian*)

- Imagine many random + and - numbers added together
- If you do this *many* times:
    - Most cancel out (somewhere around 0)
    - Few are far away from 0 (tails of distribution)
- Common in nature, because of many small + and - factors adding together

A Galton Board in action:

The Normal Distribution (aka *Gaussian*)

- Imagine many random + and - numbers added together
- If you do this *many* times:
  - Most cancel out (somewhere around 0)
  - Few are far away from 0 (tails of distribution)
- Common in nature, because of many small + and - factors adding together
  - e.g. Height is driven by many sets of genes

A Galton Board in action:

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

- Probability distribution function (PDF) for the Normal distribution

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

- Probability distribution function (PDF) for the Normal distribution
- Tells you about the probability of getting some number *given* μ and σ

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

- Probability distribution function (PDF) for the Normal distribution
- Tells you about the probability of getting some number *given* μ and σ

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

- Probability distribution function (PDF) for the Normal distribution
- Tells you about the probability of getting some number *given* μ and σ

Example: what is the probability of getting a 4, if the mean is 5 and SD is 1?

$$p(4|5,1) = \frac{1}{1\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{4-5}{1})^2}$$
$$= \sim 0.24$$

In R, this is easy:

The Normal Distribution - scary math!

- 2 parameters: mean (μ) and standard deviation (σ)

$$p(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

- Probability distribution function (PDF) for the Normal distribution
- Tells you about the probability of getting some number *given* μ and σ

Example: what is the probability of getting a 4, if the mean is 5 and SD is 1?

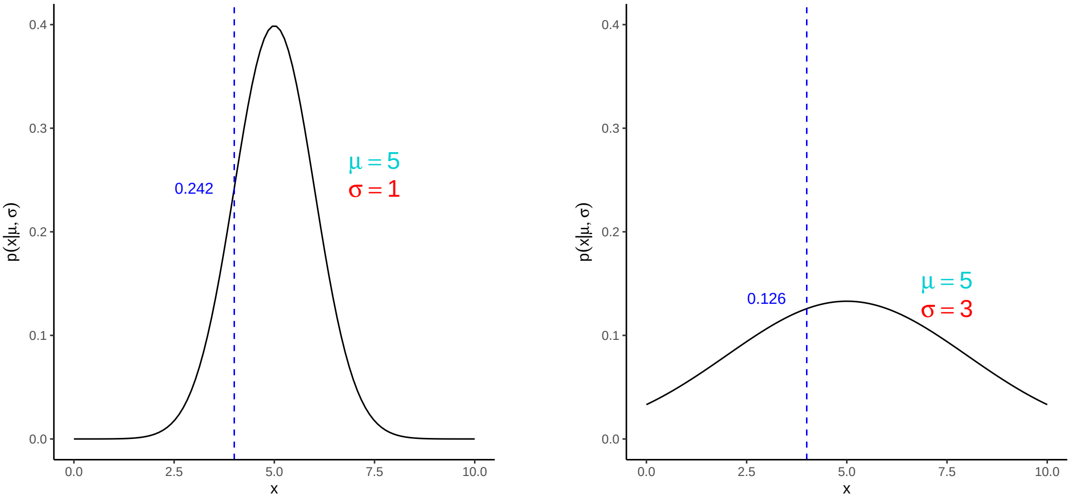
$$p(4|5, 1) = \frac{1}{1\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{4-5}{1})^2}$$
$$= \sim 0.24$$

In R, this is easy:

```
#d stands for "density"
dnorm(x=4,mean=5,sd=1)
```

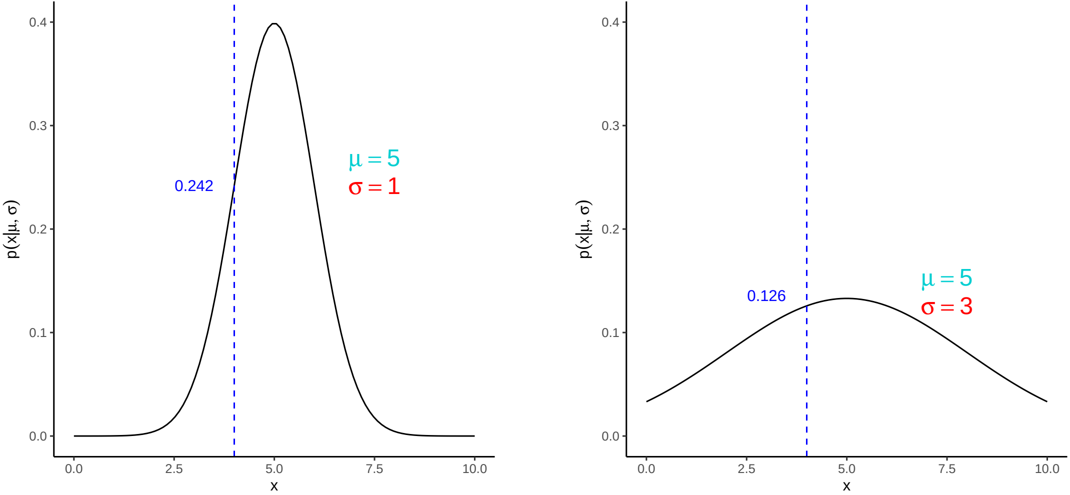
```
## [1] 0.2419707
```

The Normal Distribution



- Probability of x changes with μ and σ

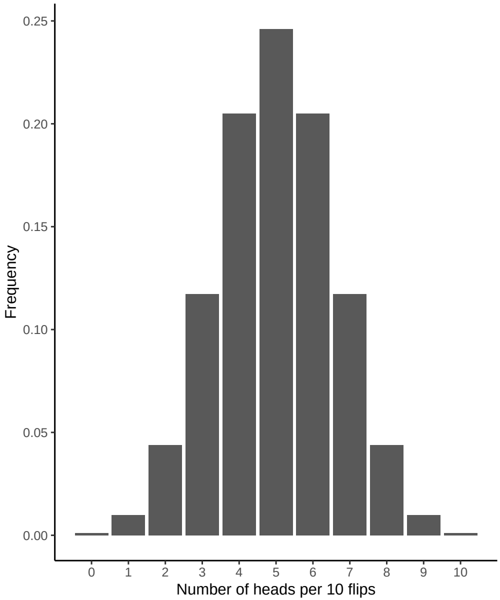
The Normal Distribution



- Probability of x changes with μ and σ
- Left: $\sigma = 1$, Right: $\sigma = 3$

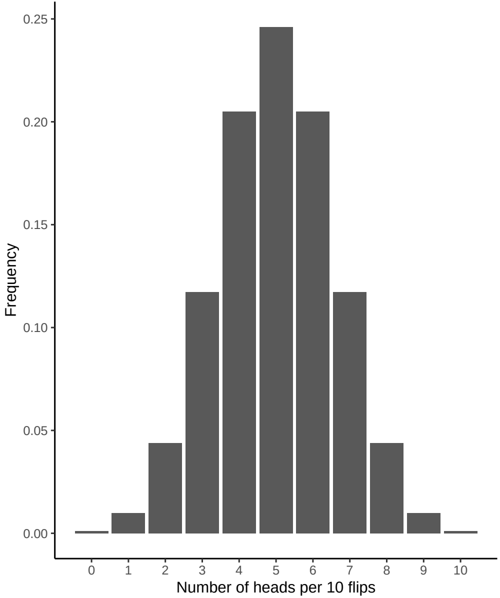
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all



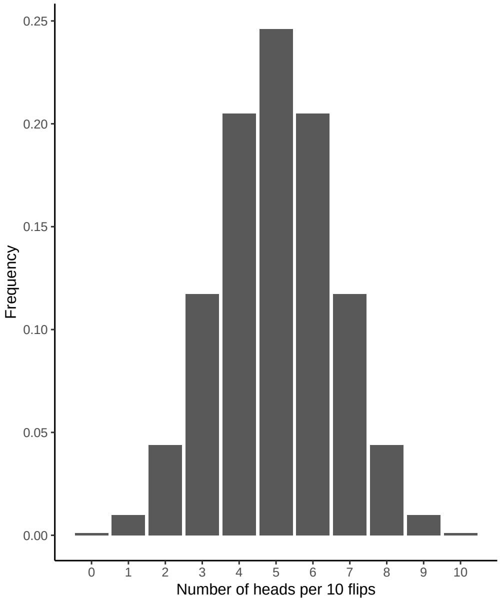
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:



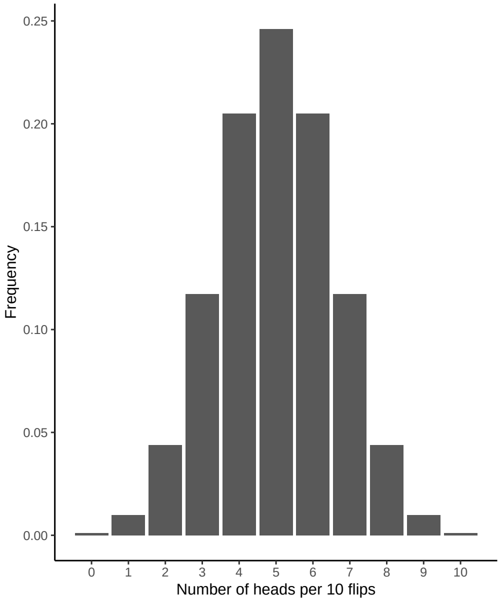
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:
  - Most will be about 5 heads/tails



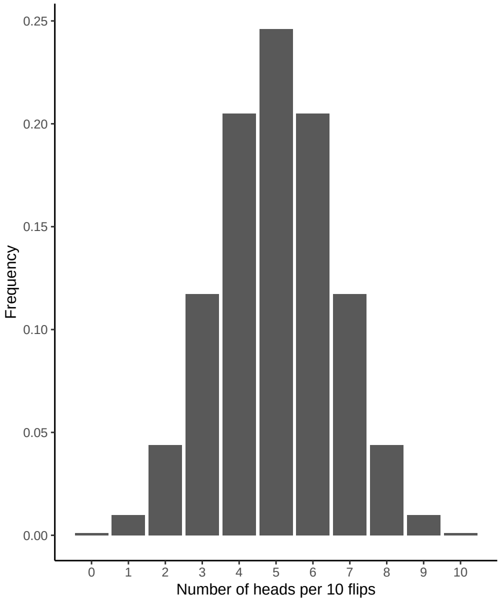
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:
  - Most will be about 5 heads/tails
  - Few will be 1 head, 9 tails (or reverse)



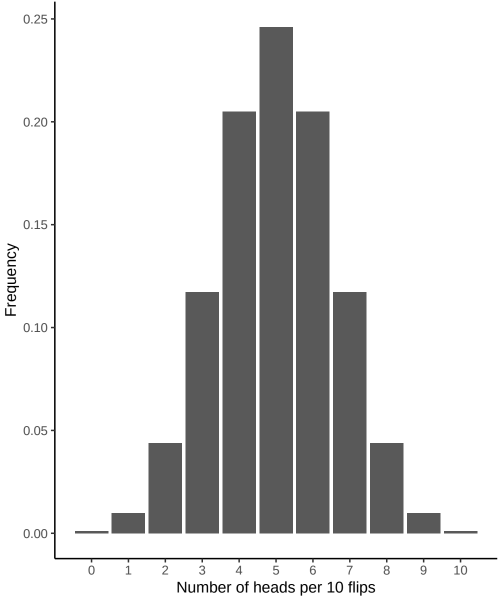
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:
    - Most will be about 5 heads/tails
    - Few will be 1 head, 9 tails (or reverse)
- Common in nature where outcomes are binary



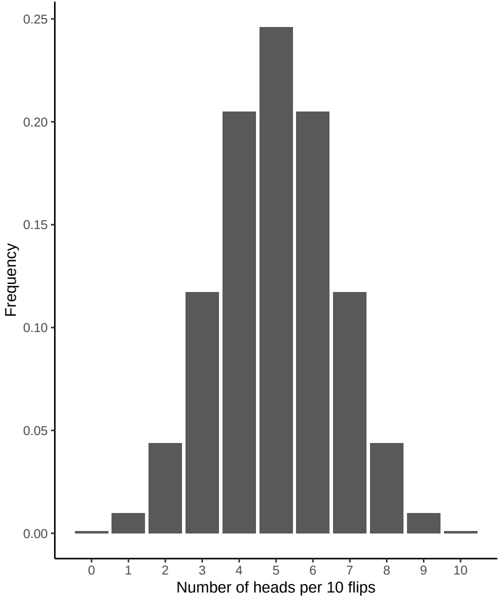
The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:
  - Most will be about 5 heads/tails
  - Few will be 1 head, 9 tails (or reverse)
- Common in nature where outcomes are binary
  - e.g. 10 seeds from a plant, how many will germinate?



The Binomial Distribution

- Imagine you have 10 coins, and you flip them all
- If you do this *many* times:
  - Most will be about 5 heads/tails
  - Few will be 1 head, 9 tails (or reverse)
- Common in nature where outcomes are binary
  - e.g. 10 seeds from a plant, how many will germinate?
- If N $=$ 1, this is called a *Bernoulli trial*



The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...

$$p(x|\phi, N) = \binom{N}{x}\phi^{x}(1-\phi)^{N-x}$$

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x} \phi^{x} (1 - \phi)^{N-x}$$

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x}\phi^{x}(1-\phi)^{N-x}$$

- Probability mass function (PMF); density = continuous

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x} \phi^{x} (1 - \phi)^{N-x}$$

- Probability mass function (PMF); density = continuous
- Tells you about the probability of getting x "successes" *given* ϕ and N

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus…
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x} \phi^x (1-\phi)^{N-x}$$

- Probability mass function (PMF); density = continuous
- Tells you about the probability of getting x "successes" *given* ϕ and N

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x} \phi^x (1 - \phi)^{N-x}$$

- Probability mass function (PMF); density = continuous
- Tells you about the probability of getting x "successes" *given* ϕ and N

Example: what is the probability of getting 4 successes, if ϕ is 0.25 and N is 15?

$$p(4|0.25, 15) = \binom{15}{4} 0.25^4 (1 - 0.25)^{15-4}$$

$$= \sim 0.23$$

The Binomial Distribution - scary math!

- 1 parameter: probability of success (ϕ), plus...
- Number of "coin flips" (N)

$$p(x|\phi, N) = \binom{N}{x} \phi^x (1-\phi)^{N-x}$$

- Probability mass function (PMF); density = continuous
- Tells you about the probability of getting x "successes" *given* ϕ and N

Example: what is the probability of getting 4 successes, if ϕ is 0.25 and N is 15?

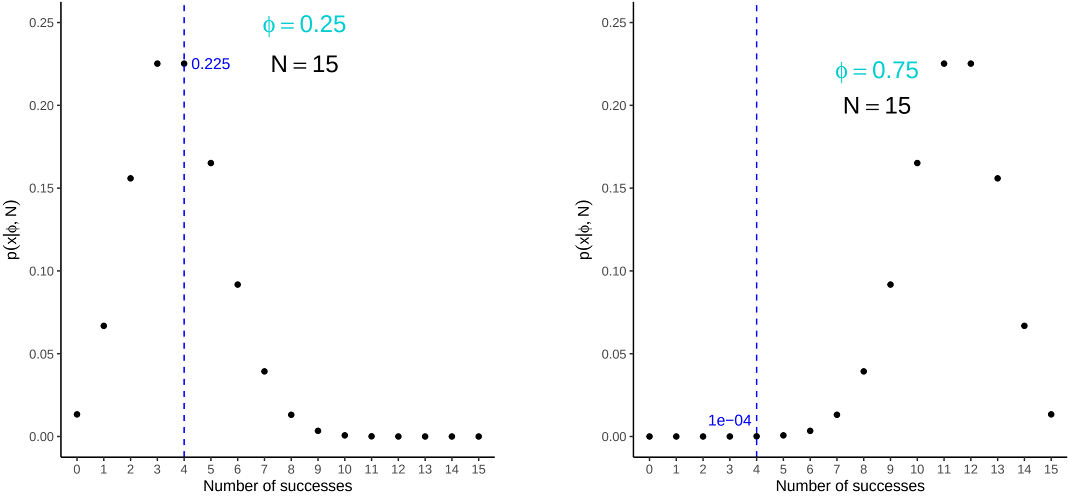
$$p(4|0.25, 15) = \binom{15}{4} 0.25^4 (1-0.25)^{15-4}$$
$$= \sim 0.23$$

In R, this is easy:

```
dbinom(x=4,size=15,prob=0.25)
```

```
## [1] 0.2251991
```

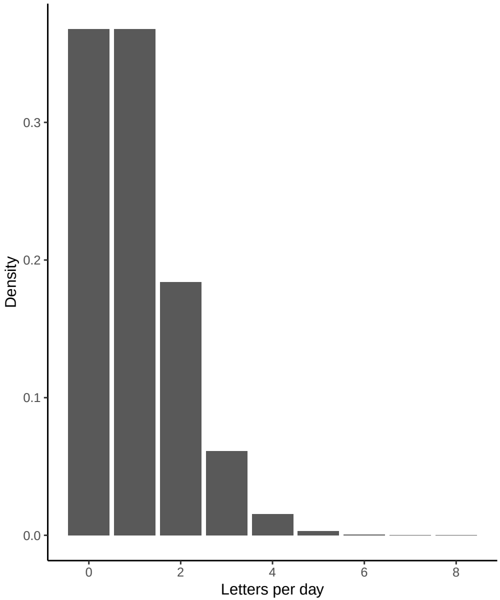
The Binomial Distribution



- Probability of x “successes” changes with ϕ and N

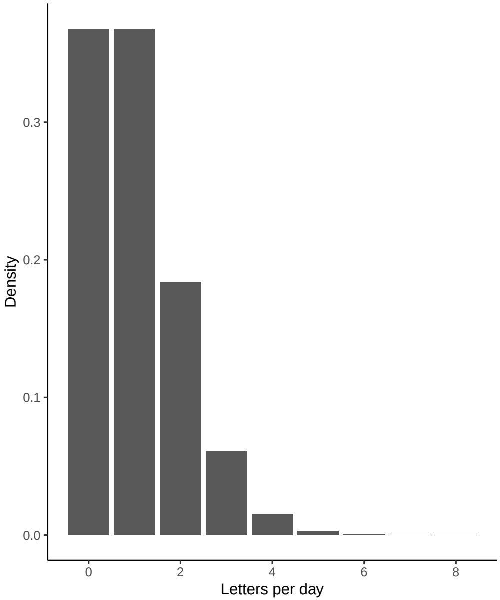
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)



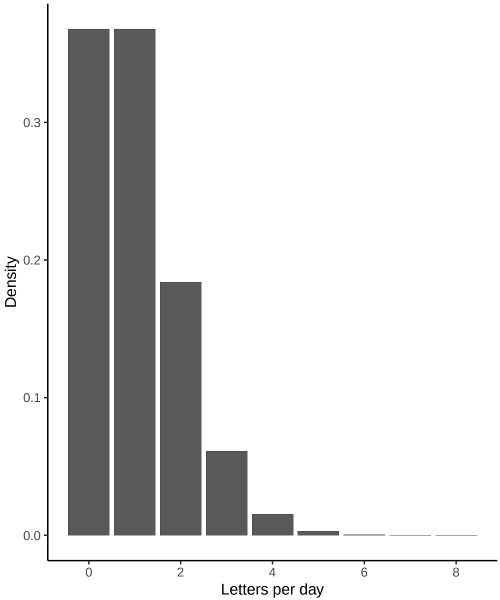
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:



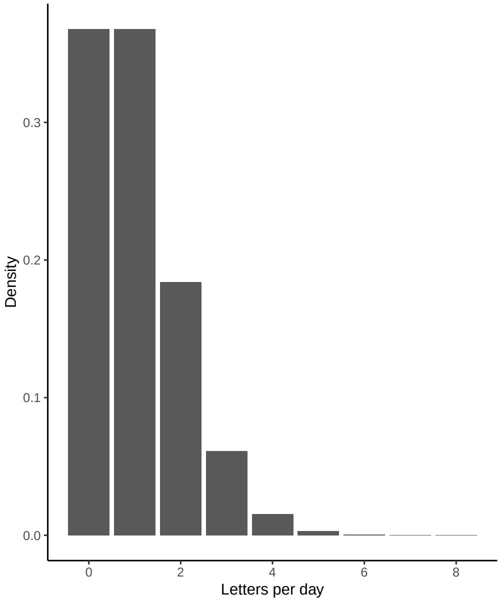
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:
    - Most days, you'll get 0 or maybe 1 letter



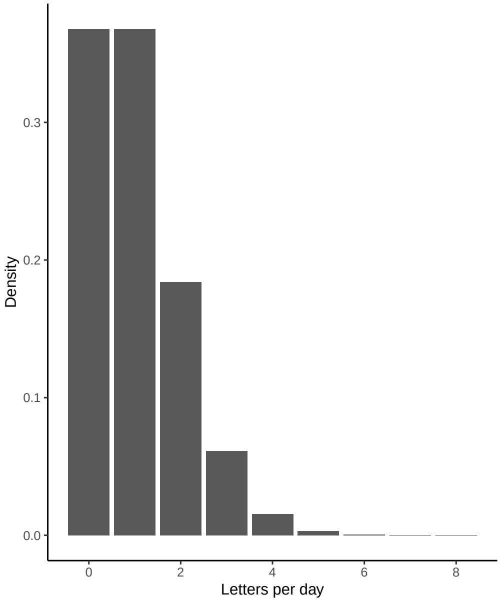
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:
  - Most days, you'll get 0 or maybe 1 letter
  - On some rare days, you'll get 3 or 4 letters



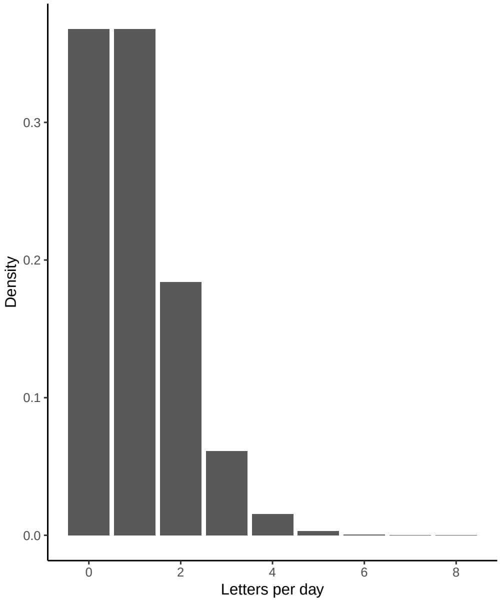
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:
  - Most days, you'll get 0 or maybe 1 letter
  - On some rare days, you'll get 3 or 4 letters
- Common in nature where rare events are measured over time/space:



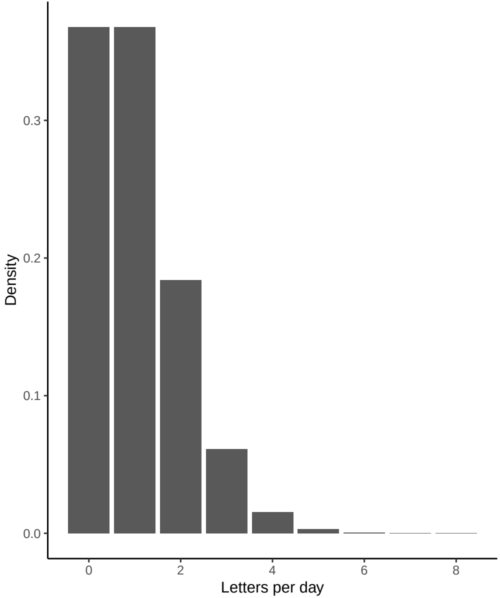
The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:
  - Most days, you'll get 0 or maybe 1 letter
  - On some rare days, you'll get 3 or 4 letters
- Common in nature where rare events are measured over time/space:
  - e.g. Number of bugs caught in a net (per sweep)



The Poisson Distribution

- Imagine a rare event (e.g. getting a non-junk mail letter)
- If you record the number of events every day:
  - Most days, you'll get 0 or maybe 1 letter
  - On some rare days, you'll get 3 or 4 letters
- Common in nature where rare events are measured over time/space:
  - e.g. Number of bugs caught in a net (per sweep)
- Equivalent to Binomial distribution, where N is unknown



The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

- Probability mass function (PMF)

The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

- Probability mass function (PMF)
- Tells you about the probability of getting x counts *given* λ

The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

- Probability mass function (PMF)
- Tells you about the probability of getting x counts *given* λ

The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^{x} e^{-\lambda}}{x!}$$

- Probability mass function (PMF)
- Tells you about the probability of getting x counts *given* λ

Example: what is the probability of getting 2 counts, if λ is 1?

$$\begin{aligned} p(2|1) &= \frac{1^{2} e^{-1}}{2!} \\ &= \sim 0.18 \end{aligned}$$

The Poisson Distribution - scary math!

- 1 parameter: rate parameter (λ)

$$p(x|\lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

- Probability mass function (PMF)
- Tells you about the probability of getting x counts *given* λ

Example: what is the probability of getting 2 counts, if λ is 1?

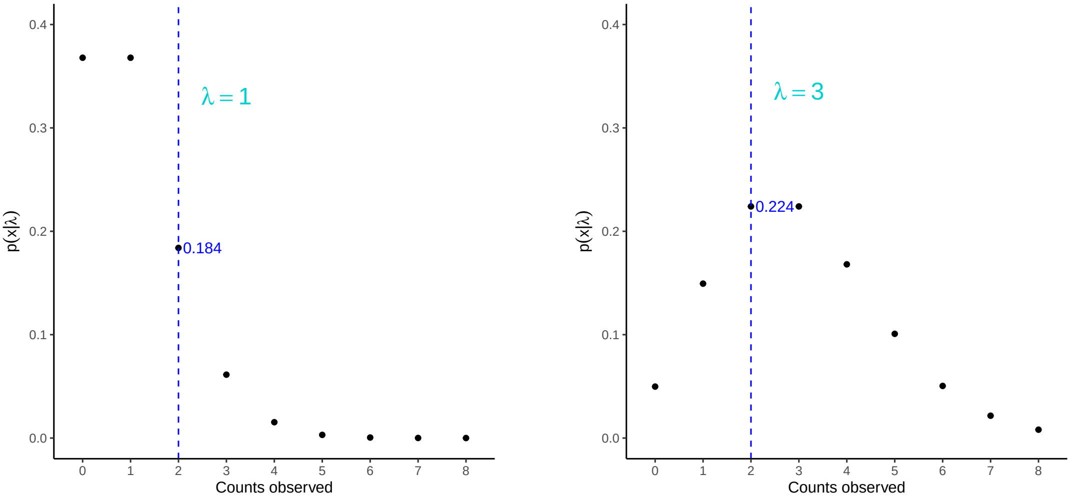
$$p(2|1) = \frac{1^2 e^{-1}}{2!}$$
$$= \sim 0.18$$

In R, this is easy:

```
dpois(x=2,lambda=1)
```

```
## [1] 0.1839397
```

The Poisson Distribution



- Probability of x counts changes with λ

More complications:

- The Normal distribution has a parameter for the mean and SD, but...

More complications:

- The Normal distribution has a parameter for the mean and SD, but...
- What about the Binomial and Poisson distributions?

More complications:

- The Normal distribution has a parameter for the mean and SD, but...
- What about the Binomial and Poisson distributions?
    - Binomial: mean $= Np$, SD $= \sqrt{Np(1-p)}$

More complications:

- The Normal distribution has a parameter for the mean and SD, but...
- What about the Binomial and Poisson distributions?
  - Binomial: mean $= Np$, SD $= \sqrt{Np(1-p)}$
  - Poisson: mean $= \lambda$, SD $= \sqrt{\lambda}$

More complications:

- The Normal distribution has a parameter for the mean and SD, but...
- What about the Binomial and Poisson distributions?
  - Binomial: mean $= Np$, SD $= \sqrt{Np(1-p)}$
  - Poisson: mean $= \lambda$, SD $= \sqrt{\lambda}$
- What if our data have additional variance?

More complications:

- The Normal distribution has a parameter for the mean and SD, but...
- What about the Binomial and Poisson distributions?
  - Binomial: mean $= Np$, SD $= \sqrt{Np(1-p)}$
  - Poisson: mean $= \lambda$, SD $= \sqrt{\lambda}$
- What if our data have additional variance?
  - *Beta Binomial* and *Negative Binomial* distributions

The Beta Binomial Distibution

- Many "coin-flip" processes have longer tails than standard Binomial

The Beta Binomial Distibution

- Many “coin-flip” processes have longer tails than standard Binomial
  - e.g. numbers of males/females in families

The Beta Binomial Distibution

- Many "coin-flip" processes have longer tails than standard Binomial
  - e.g. numbers of males/females in families
- Beta-binomial adds additional dispersion to coin flip process

The Beta Binomial Distibution

- Many “coin-flip” processes have longer tails than standard Binomial
  - e.g. numbers of males/females in families
- Beta-binomial adds additional dispersion to coin flip process
- 2 parameters: ϕ and s (if s is large, similar to Binomial)

The Beta Binomial Distibution

- Many "coin-flip" processes have longer tails than standard Binomial
  - e.g. numbers of males/females in families
- Beta-binomial adds additional dispersion to coin flip process
- 2 parameters: ϕ and s (if s is large, similar to Binomial)
  - Also requires: N

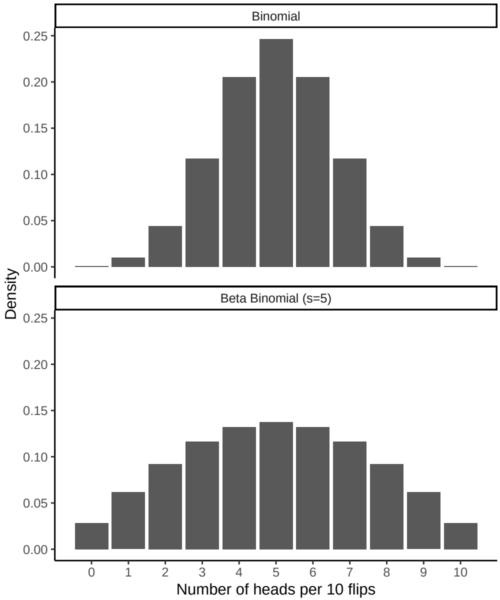
The Beta Binomial Distibution

- Many "coin-flip" processes have longer tails than standard Binomial
    - e.g. numbers of males/females in families
- Beta-binomial adds additional dispersion to coin flip process
- 2 parameters: ϕ and s (if s is large, similar to Binomial)
    - Also requires: N

The Beta Binomial Distibution

- Many "coin-flip" processes have longer tails than standard Binomial
  - e.g. numbers of males/females in families
- Beta-binomial adds additional dispersion to coin flip process
- 2 parameters: ϕ and s (if s is large, similar to Binomial)
  - Also requires: N

```
#Extra distributions
library(rmutil)
dbetabinom(x,m=phi,size=N,s=5)
```



The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution

The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution
- Negative Binomial is similar to a Poisson, but can have longer tails

The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution
- Negative Binomial is similar to a Poisson, but can have longer tails
- Also called: *Polya* distibution (`nbinom2` in many GLM commands)

The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution
- Negative Binomial is similar to a Poisson, but can have longer tails
- Also called: *Polya* distibution (`nbinom2` in many GLM commands)
- Parameters: μ and θ (if θ is large, close to Poisson)

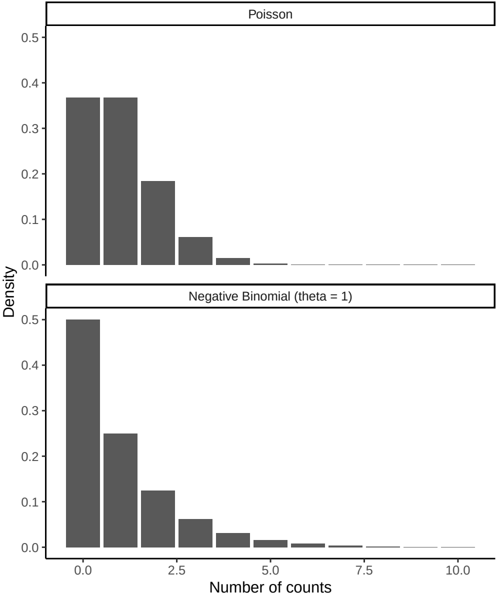
The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution
- Negative Binomial is similar to a Poisson, but can have longer tails
- Also called: *Polya* distibution (`nbinom2` in many GLM commands)
- Parameters: μ and θ (if θ is large, close to Poisson)

The Negative Binomial Distribution

- Unfortunately, *almost nothing* in ecology actually follows a Poisson distribution
- Negative Binomial is similar to a Poisson, but can have longer tails
- Also called: *Polya* distibution (`nbinom2` in many GLM commands)
- Parameters: μ and θ (if θ is large, close to Poisson)

```
#size = theta parameter
dnbinom(x,mu,size=1)
```



Summary of Common "Starter" Distributions

- Continuous data, spanning - or $+$ numbers:

Summary of Common “Starter” Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)

Summary of Common "Starter" Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data

Summary of Common “Starter” Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data
  - Poisson, Negative Binomial

Summary of Common “Starter” Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data
  - Poisson, Negative Binomial
- Count data of successes *and* failures

Summary of Common “Starter” Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data
  - Poisson, Negative Binomial
- Count data of successes *and* failures
  - Binomial, Beta Binomial

Summary of Common "Starter" Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data
  - Poisson, Negative Binomial
- Count data of successes *and* failures
  - Binomial, Beta Binomial

Summary of Common "Starter" Distributions

- Continuous data, spanning - or + numbers:
  - Normal (transformed or regular)
- Count data
  - Poisson, Negative Binomial
- Count data of successes *and* failures
  - Binomial, Beta Binomial

These are by *no means* the only useful distributions, but are fairly common

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)
- Weight of seeds from a plant

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)
- Weight of seeds from a plant
- Occupied/unoccupied nest sites

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)
- Weight of seeds from a plant
- Occupied/unoccupied nest sites
- Chemical concentrations

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)
- Weight of seeds from a plant
- Occupied/unoccupied nest sites
- Chemical concentrations
- Size of trees (DBH or height)

First challenge (Part 1)

Let's say that you've collected data at 2 different sites. Which distributions would you start with for the following data?

- Insects caught in a trap (per day)
- Weight of seeds from a plant
- Occupied/unoccupied nest sites
- Chemical concentrations
- Size of trees (DBH or height)
- Number of male and female bats

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*
  - `rnorm(n,mean,sd)`

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*
  - `rnorm(n,mean,sd)`
- Size of trees (DBH or height):*log-Normal*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*
  - `rnorm(n,mean,sd)`
- Size of trees (DBH or height):*log-Normal*
  - `exp(rnorm(n,mean,sd))`

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*
  - `rnorm(n,mean,sd)`
- Size of trees (DBH or height): *log-Normal*
  - `exp(rnorm(n,mean,sd))`
- Number of male and female bats: *Binomial or Beta Binomial*

First challenge (Part 2)

Now that you've figured out which distribution, try simulating some data from each "site", and plot it!

- Insects caught in a trap (per day): *Poisson or NB*
  - `rpois(n,lambda)` or `rnbinom(n,mu,size)`
- Weight of seeds: *Normal*
  - `rnorm(n,mean,sd)`
- Occupied/unoccupied nest sites: *Binomial*
  - `rbinom(n, 1, prob)` aka. *Bernoulli* distribution
- Chemical concentrations in a pond: *Normal*
  - `rnorm(n,mean,sd)`
- Size of trees (DBH or height): *log-Normal*
  - `exp(rnorm(n,mean,sd))`
- Number of male and female bats: *Binomial or Beta Binomial*
  - `rbinom(n, size, prob)` or `rbetabinom(n,size,m,s)`

Part 2: Maximum likelihood and GLMs

Outline

- Maximum likelihood

Outline

- Maximum likelihood
  - A way to think about data

Outline

- Maximum likelihood
    - A way to think about data
    - Likelihood vs Probability

Outline

- Maximum likelihood
  - A way to think about data
  - Likelihood vs Probability
- Generalized linear models

Outline

- Maximum likelihood
  - A way to think about data
  - Likelihood vs Probability
- Generalized linear models
  - Link functions

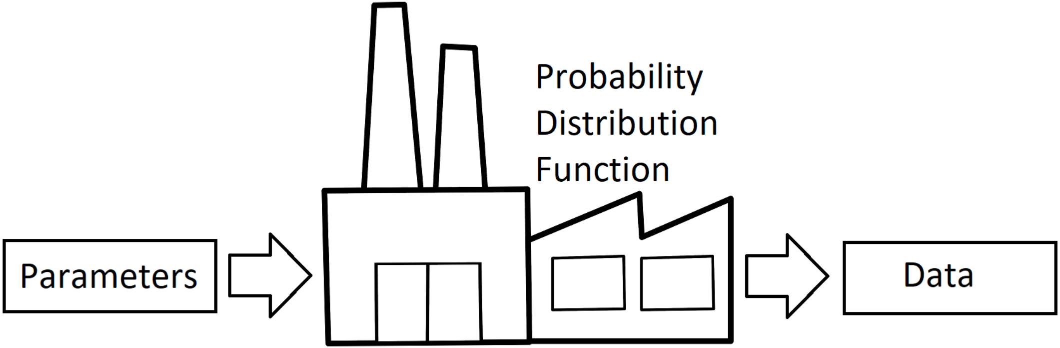
Outline

- Maximum likelihood
  - A way to think about data
  - Likelihood vs Probability
- Generalized linear models
  - Link functions
  - Predictors -> Linear model

How is our data made?

Making data can be thought of as a *factory*

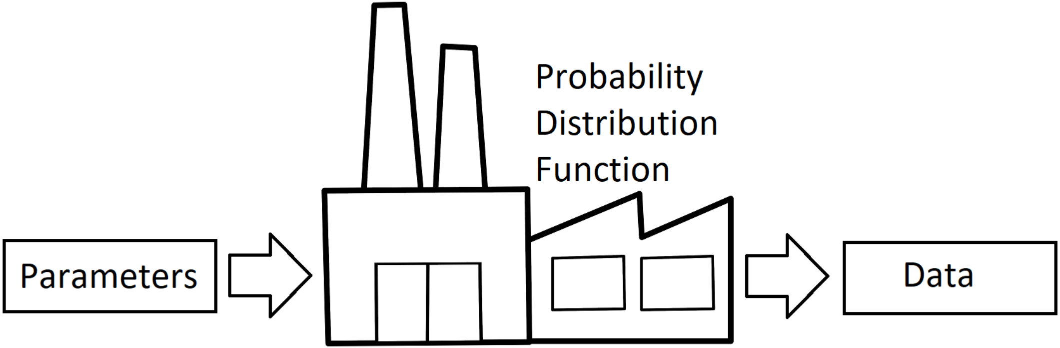
- Input: **parameters** (things that guide the process)



How is our data made?

Making data can be thought of as a *factory*

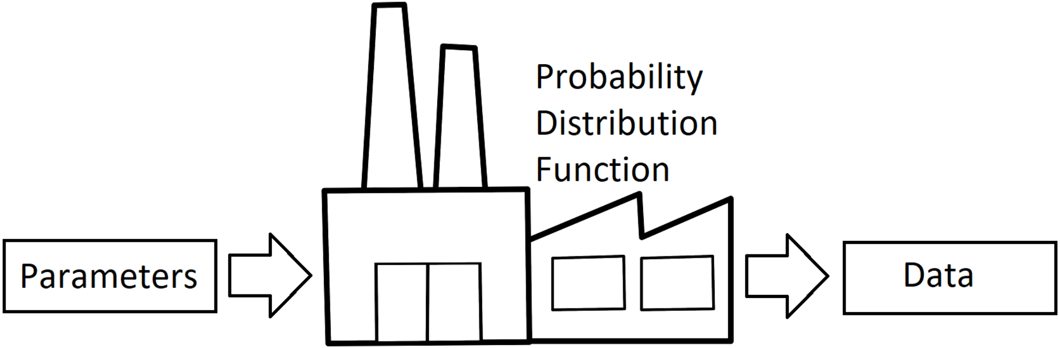
- Input: **parameters** (things that guide the process)
- Process: **probability function**



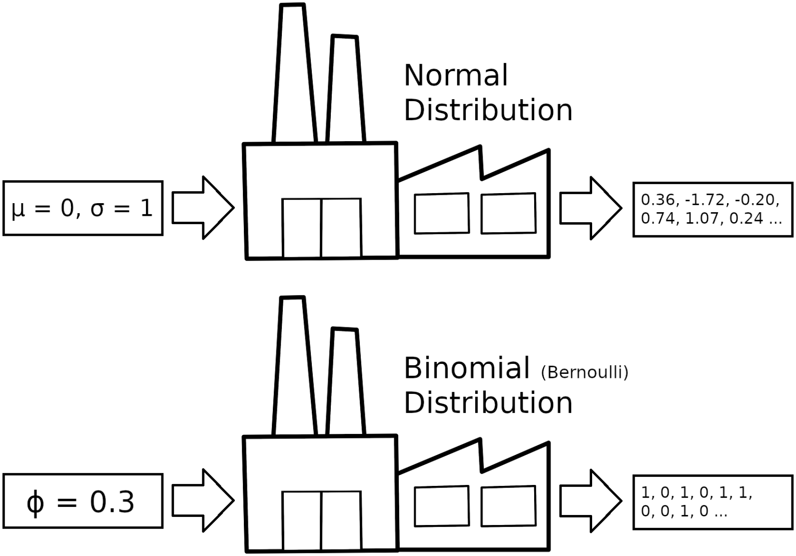
How is our data made?

Making data can be thought of as a *factory*

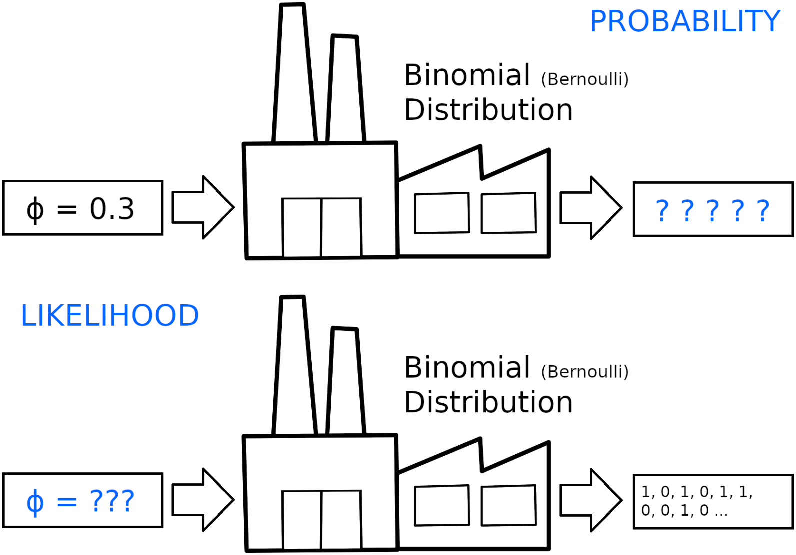
- Input: **parameters** (things that guide the process)
- Process: **probability function**
- Output: **data** (things made by the process)



Examples


μ = 0, σ = 1
Normal
Distribution
0.36, -1.72, -0.20,
0.74, 1.07, 0.24 ...
ϕ = 0.3
Binomial (Bernoulli)
Distribution
1, 0, 1, 0, 1, 1,
0, 0, 1, 0 ...

Likelihood vs Probability



Likelihood vs Probability (cont.)

Probability and likelihood both use the same PDF

- "I know that $\phi = 0.3$. What is the chance of getting 2 heads and a tail?"

```
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Likelihood vs Probability (cont.)

Probability and likelihood both use the same PDF

- "I know that $\phi = 0.3$. What is the chance of getting 2 heads and a tail?"

```
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Likelihood vs Probability (cont.)

Probability and likelihood both use the same PDF

- "I know that $\phi = 0.3$. What is the chance of getting 2 heads and a tail?"

```
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

- "I got 2 heads and a tail. What is the likelihood that $\phi = 0.3$?"

```
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Likelihood vs Probability (cont.)

Probability and likelihood both use the same PDF

- "I know that $\phi = 0.3$. What is the chance of getting 2 heads and a tail?"

```r
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

- "I got 2 heads and a tail. What is the likelihood that $\phi = 0.3$?"

```r
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Likelihood vs Probability (cont.)

Probability and likelihood both use the same PDF

- "I know that $\phi = 0.3$. What is the chance of getting 2 heads and a tail?"

```r
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

- "I got 2 heads and a tail. What is the likelihood that $\phi = 0.3$?"

```r
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Since we're (mostly) collecting data and trying to guess parameters from it, are we dealing with *probability* or *likelihood*?

Likelihood vs Probability (cont.)

Let's see how *likelihood* changes with different values of ϕ:

```r
#phi = 0.3
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

```
## [1] 0.063
```

Likelihood vs Probability (cont.)

Let's see how *likelihood* changes with different values of ϕ:

```r
#phi = 0.3
dbinom(1,1,0.3)*dbinom(1,1,0.3)*dbinom(0,1,0.3)
```

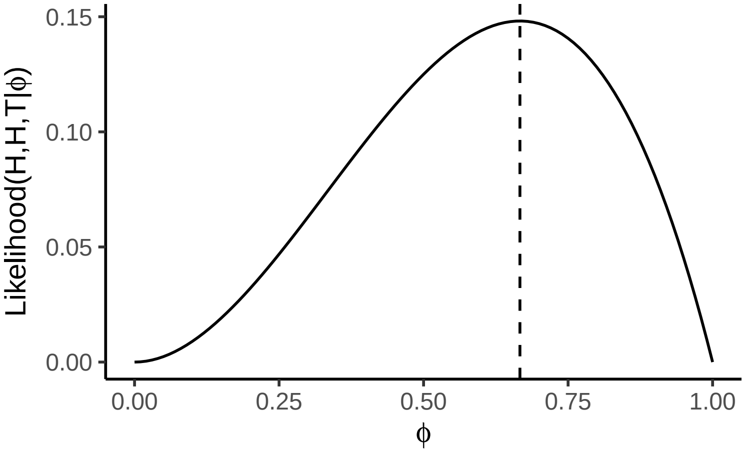
```
## [1] 0.063
```

```r
#phi = 0.7
dbinom(1,1,0.7)*dbinom(1,1,0.7)*dbinom(0,1,0.7)
```

```
## [1] 0.147
```

Likelihood of $\phi = 0.7$ is higher, i.e. $\phi = 0.7$ matches our data *better*

Likelihood



The best match (**maximum likelihood** value) is at $\phi = 0.666$ (2 heads out of 3 flips)

Generalized Linear Models

glm() will fit a model like this, and find the ML solution

```
dat <- data.frame(flips=c(1,1,0)) #Data (2 heads, 1 tail)
mod1 <- glm(flips~1,data=dat,family='binomial') #Note family specification
summary(mod1)
```

```
##
## Call:
## glm(formula = flips ~ 1, family = "binomial", data = dat)
##
## Deviance Residuals:
##       1        2        3
##  0.9005   0.9005  -1.4823
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.6931     1.2247   0.566    0.571
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.8191  on 2  degrees of freedom
## Residual deviance: 3.8191  on 2  degrees of freedom
## AIC: 5.8191
##
## Number of Fisher Scoring iterations: 4
```

Generalized Linear Models

`glm()` will fit a model like this, and find the ML solution

```r
dat <- data.frame(flips=c(1,1,0)) #Data (2 heads, 1 tail)
mod1 <- glm(flips~1,data=dat,family='binomial') #Note family specification
summary(mod1)
```

```
##
## Call:
## glm(formula = flips ~ 1, family = "binomial", data = dat)
##
## Deviance Residuals:
##       1        2        3
##  0.9005   0.9005  -1.4823
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.6931     1.2247   0.566    0.571
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.8191  on 2  degrees of freedom
## Residual deviance: 3.8191  on 2  degrees of freedom
## AIC: 5.8191
##
## Number of Fisher Scoring iterations: 4
```

Wait... our estimate should be 0.666 (2/3), not 0.693!

Link functions

- Some parameters of PDFs have *limits*

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$
  - Poisson: $0 < \lambda$

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$
  - Poisson: $0 < \lambda$
- GLMs use *link functions* to map values onto the appropriate parameter range

Link functions

- Some parameters of PDFs have *limits*
    - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
    - Binomial: $0 < \phi < 1$
    - Poisson: $0 < \lambda$
- GLMs use *link functions* to map values onto the appropriate parameter range
    - Normal: Identity (i.e. $\times 1$)

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$
  - Poisson: $0 < \lambda$
- GLMs use *link functions* to map values onto the appropriate parameter range
  - Normal: Identity (i.e. $\times 1$)
  - Binomial: Logit

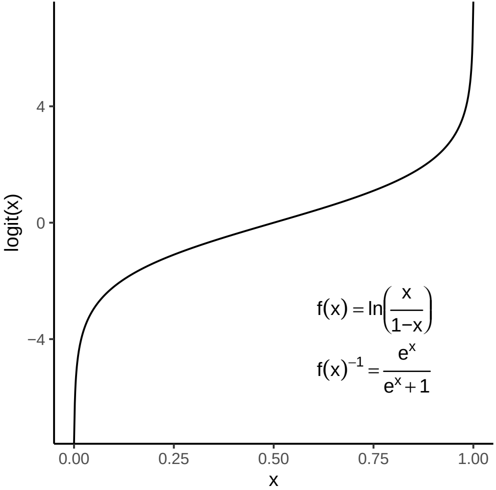
Link functions

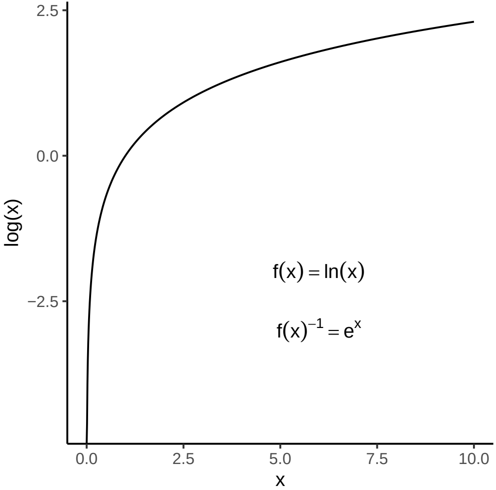
- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$
  - Poisson: $0 < \lambda$
- GLMs use *link functions* to map values onto the appropriate parameter range
  - Normal: Identity (i.e. $\times 1$)
  - Binomial: Logit
  - Poisson/NB: Log

Link functions

- Some parameters of PDFs have *limits*
  - Normal: $-\infty < \mu < \infty$, $0 < \sigma$
  - Binomial: $0 < \phi < 1$
  - Poisson: $0 < \lambda$
- GLMs use *link functions* to map values onto the appropriate parameter range
  - Normal: Identity (i.e. $\times 1$)
  - Binomial: Logit
  - Poisson/NB: Log
- *logit*$(0.693) = 0.666$, so the GLM actually got it right!

What do these functions look like?

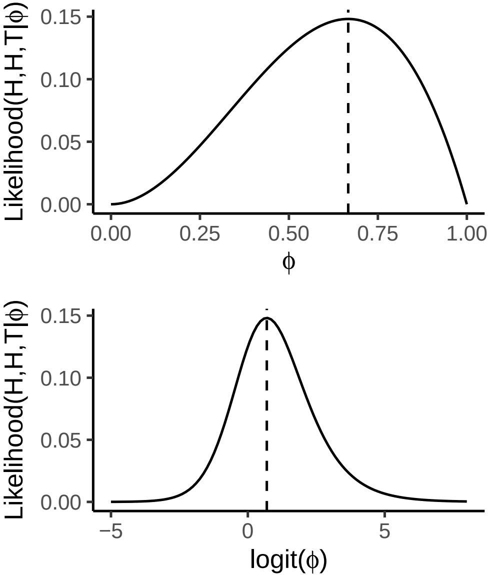




- These functions map parameter values from the appropriate range (0-1 or 0-∞) onto $-\infty$ to $+\infty$

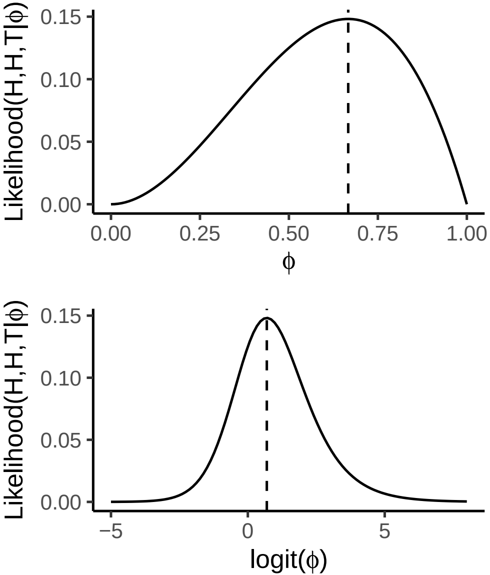
Why do we bother with these link function?

- Likelihood functions are not symmetrical on the regular scale



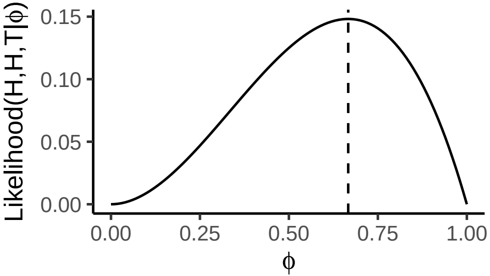
Why do we bother with these link function?

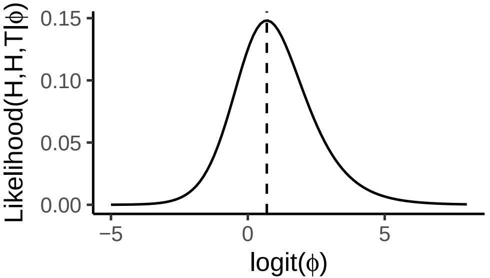
- Likelihood functions are not symmetrical on the regular scale
- On the link-scale, they are closer to a normal distribution



Why do we bother with these link function?

- Likelihood functions are not symmetrical on the regular scale
- On the link-scale, they are closer to a normal distribution
- Makes it easier for R to find the ML estimate (and confidence intervals)





How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$
$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$

$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

1. Expected value (ϕ) fed through a link function

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$

$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

1. Expected value (ϕ) fed through a link function
2. Data is fit to a non-normal probability function

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$
$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

1. Expected value (ϕ) fed through a link function
2. Data is fit to a non-normal probability function

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$

$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

1. Expected value (ϕ) fed through a link function
2. Data is fit to a non-normal probability function

$$logit(\hat{\phi}) = b_0 + b_1 x_1 ... + b_i x_i$$

$$flips \sim Binomial(\hat{\phi})$$

How do linear models fit into this?

- Usually we aren't interested in finding only a single parameter ϕ.
- Solution: ϕ becomes a *linear* function of the predictors
- Remember: simple linear models take the form:

$$\hat{y} = b_0 + b_1 x_1 ... + b_i x_i$$
$$y \sim Normal(\hat{y}, \sigma)$$

- Generalized linear models are similar, except that:

1. Expected value (ϕ) fed through a link function
2. Data is fit to a non-normal probability function

$$logit(\hat{\phi}) = b_0 + b_1 x_1 ... + b_i x_i$$
$$flips \sim Binomial(\hat{\phi})$$

Instead of finding ϕ, R finds the coefficients (b_0, b_1 ... b_i) that create ϕ

How do I fit GLMs in R?

Syntax and model output is very similar to `lm`

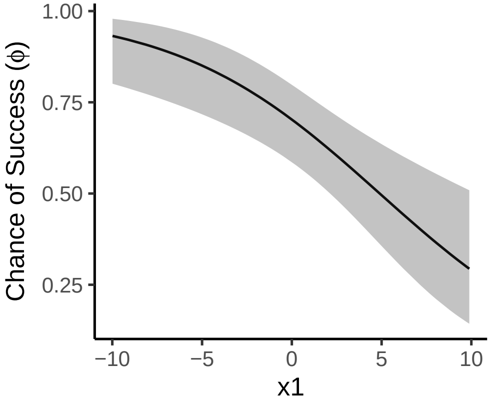
```r
# y ~ x, where x is the predictor of y
mod_binomial <- glm(y2 ~ x1 + x2 , data = d1, family = 'binomial') #Fit a binomial GLM
```

```
##
## Call:
## glm(formula = y2 ~ x1 + x2, family = "binomial", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0050  -0.9493   0.3924   0.8336   1.6806
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.81748    0.25851   3.162 0.001565 **
## x1          -0.17576    0.04871  -3.608 0.000309 ***
## x2           0.30193    0.09950   3.034 0.002410 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 129.49  on 99  degrees of freedom
## Residual deviance: 102.98  on 97  degrees of freedom
## AIC: 108.98
##
## Number of Fisher Scoring iterations: 4
```

Dispersion and deviance will be discussed later...

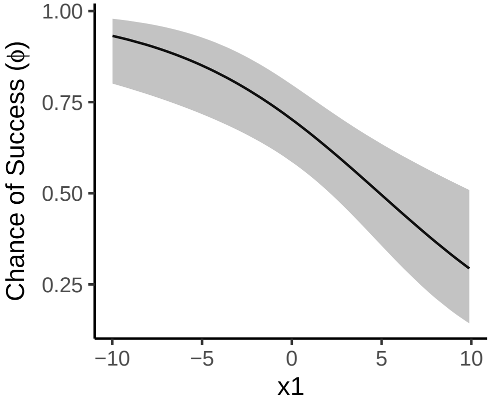
How do I get partial effects plots?

- `crPlot` (from `car`) and `ggpredict` (`ggeffects`) work with fitted `glm` models:



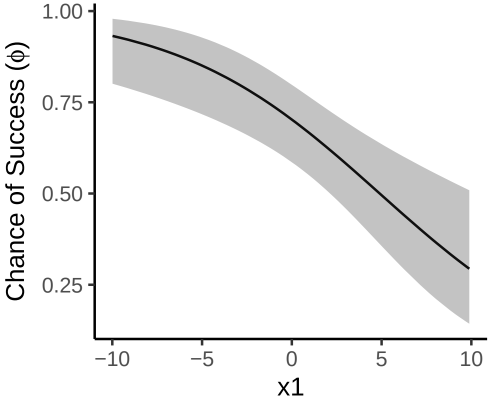
How do I get partial effects plots?

- `crPlot` (from `car`) and `ggpredict` (`ggeffects`) work with fitted `glm` models:
- Why is the line not straight? Why are the confidence intervals not symmetrical?



How do I get partial effects plots?

- `crPlot` (from `car`) and `ggpredict` (`ggeffects`) work with fitted `glm` models:
- Why is the line not straight? Why are the confidence intervals not symmetrical?
- Answer: the model is *linear* on the link scale, but *nonlinear* on the data scale



Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).

Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).
- Fit a model to each of the datasets (found here in `beeDatGLM.csv`) using a GLM

Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).
- Fit a model to each of the datasets (found here in `beeDatGLM.csv`) using a GLM
  - `beeCounts` should be modeled using a Poisson GLM, and `batPres` should use a Binomial GLM

Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).
- Fit a model to each of the datasets (found here in `beeDatGLM.csv`) using a GLM
  - `beeCounts` should be modeled using a Poisson GLM, and `batPres` should use a Binomial GLM
  - Terms to include: `city` and `size` (no interaction)

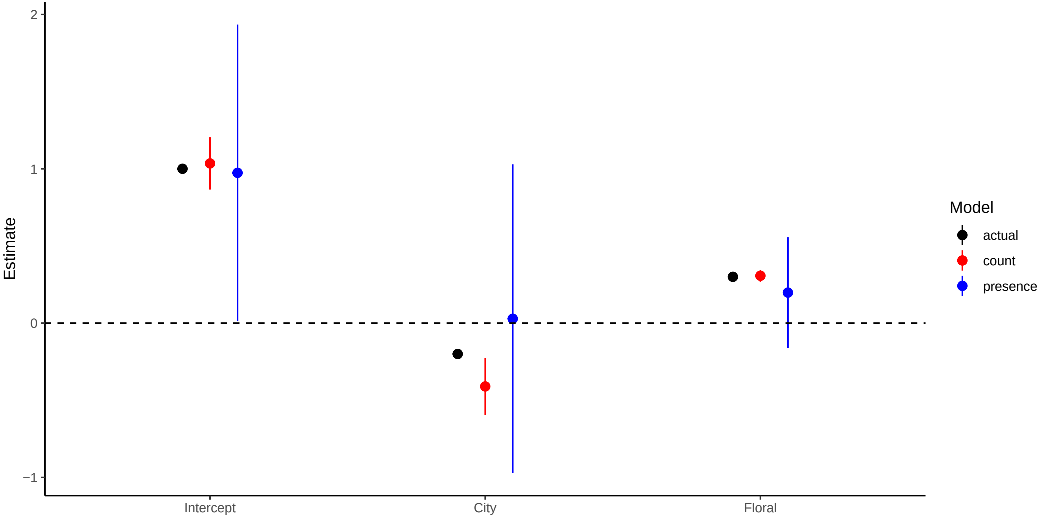
Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).
- Fit a model to each of the datasets (found here in `beeDatGLM.csv`) using a GLM
  - `beeCounts` should be modeled using a Poisson GLM, and `batPres` should use a Binomial GLM
  - Terms to include: `city` and `size` (no interaction)
- How do the models look? Compare the coefficients and see if they are different

Second challenge

- Dr. *Paulo Malpern* (Paul Galpern's evil nemesis) sent 2 people out to check out some bee habitats in Edmonton and Calgary. Counted bees at each site, but the other one was really lazy, and just recorded "bees or no bees" (1 or 0).
- Fit a model to each of the datasets (found here in `beeDatGLM.csv`) using a GLM
  - `beeCounts` should be modeled using a Poisson GLM, and `batPres` should use a Binomial GLM
  - Terms to include: `city` and `size` (no interaction)
- How do the models look? Compare the coefficients and see if they are different
  - Bonus: make a partial regression plot of terms in the Poisson GLM

Model results



Part 3: Models behaving badly

Motivation

- Are my model results reliable?

Motivation

- Are my model results reliable?
  - Residual checks

Motivation

- Are my model results reliable?
    - Residual checks
    - Overdispersion

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation
- Model selection - which terms should I use?

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation
- Model selection - which terms should I use?
  - log-likelihood, χ^2 tests, and AIC

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation
- Model selection - which terms should I use?
  - log-likelihood, χ^2 tests, and AIC
- Other things

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation
- Model selection - which terms should I use?
  - log-likelihood, χ^2 tests, and AIC
- Other things
  - Binomial GLMs with >1 trial

Motivation

- Are my model results reliable?
  - Residual checks
  - Overdispersion
  - Zero-inflation
- Model selection - which terms should I use?
  - log-likelihood, χ^2 tests, and AIC
- Other things
  - Binomial GLMs with >1 trial
  - Offsets in count models

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related
2. Generating process is valid

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related
2. Generating process is valid
3. Variance is constant

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related
2. Generating process is valid
3. Variance is constant

- "Regular" residuals don't work this way for GLMs!

Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related
2. Generating process is valid
3. Variance is constant

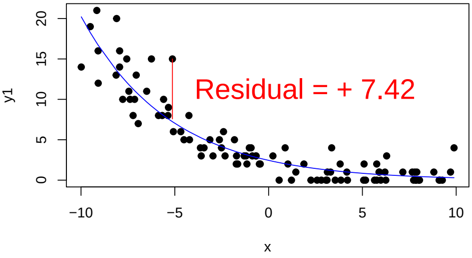
- “Regular” residuals don’t work this way for GLMs!

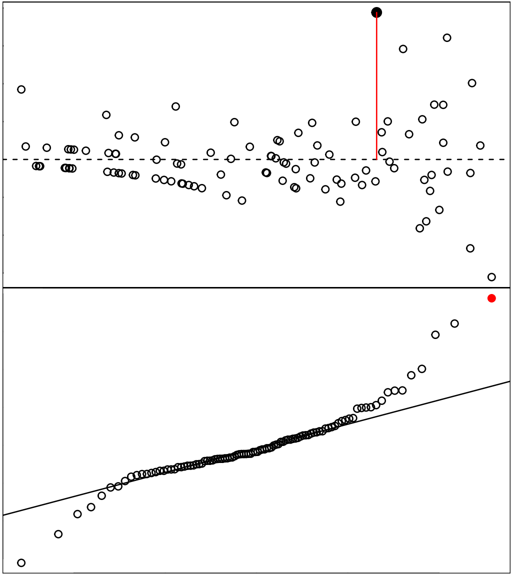
Problem 1: Residual checks

- In LMs, residual checks are used to make sure that:

1. Terms are linearly related
2. Generating process is valid
3. Variance is constant

- "Regular" residuals don't work this way for GLMs!





There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals
- **Deviance residuals**

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals
- **Deviance residuals**

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals
- **Deviance residuals**

Deviance residuals use *likelihood*:

$$r_{dev} = sign(y - \hat{y})\sqrt{2(log(L(y|\theta_s)) - log(L(y|\theta))))}$$

- This may look scary, but R does this all for you!

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals
- **Deviance residuals**

Deviance residuals use *likelihood*:

$$r_{dev} = sign(y - \hat{y})\sqrt{2(log(L(y|\theta_s)) - log(L(y|\theta))))}$$

- This may look scary, but R does this all for you!
- These are analogous to regular residuals in LMs

There are *many* kinds of residuals!

In addition to *response* (regular) residuals there are:

- Working residuals
- Pearson residuals
- **Deviance residuals**

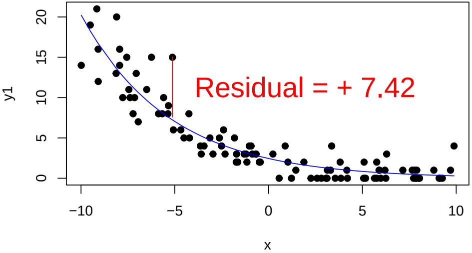
Deviance residuals use *likelihood*:

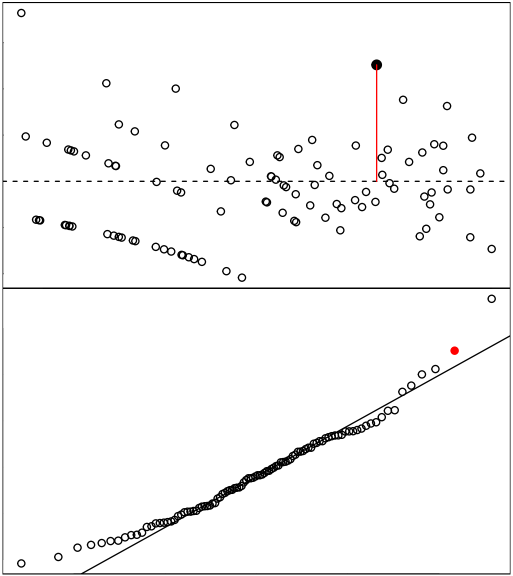
$$r_{dev} = sign(y - \hat{y})\sqrt{2(log(L(y|\theta_s)) - log(L(y|\theta))))}$$

- This may look scary, but R does this all for you!
- These are analogous to regular residuals in LMs
- For more about the different kinds of residuals, see here

Solution: use deviance residuals for GLMs

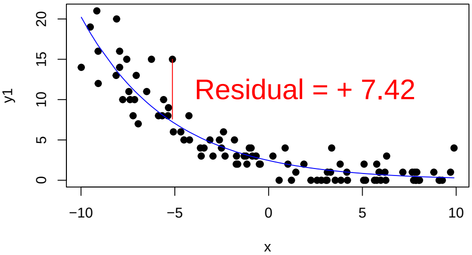
- Residuals from GLMs will never be as "pretty" as those from LMs

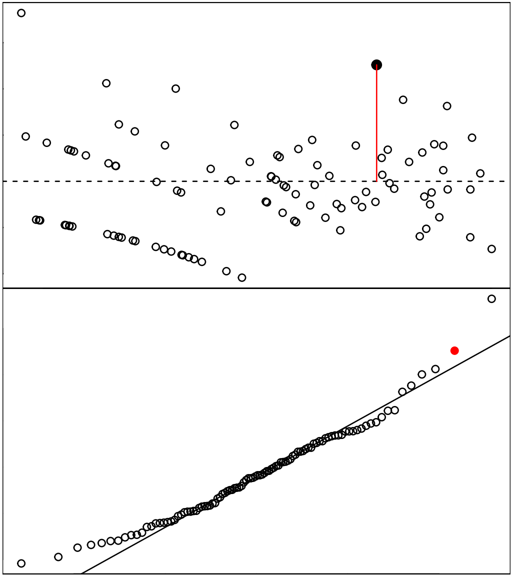




Solution: use deviance residuals for GLMs

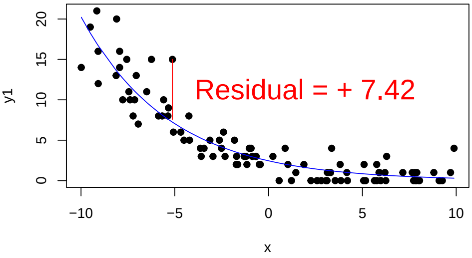
- Residuals from GLMs will never be as "pretty" as those from LMs
- *Especially* true for:

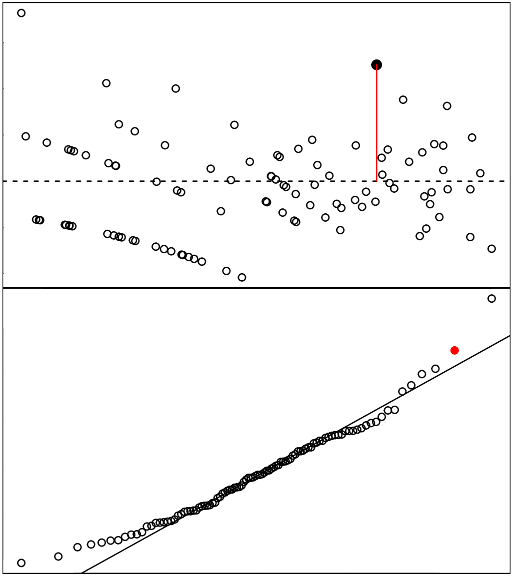




Solution: use deviance residuals for GLMs

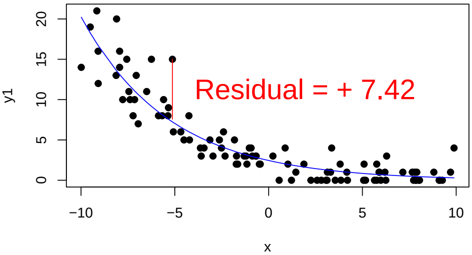
- Residuals from GLMs will never be as "pretty" as those from LMs
- *Especially* true for:
  - Binomial GLMs

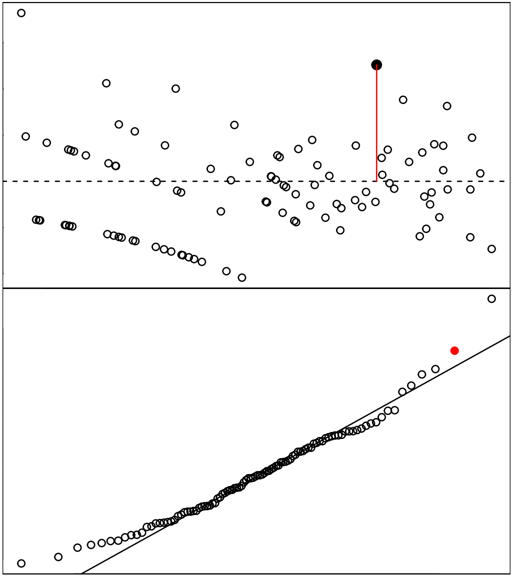




Solution: use deviance residuals for GLMs

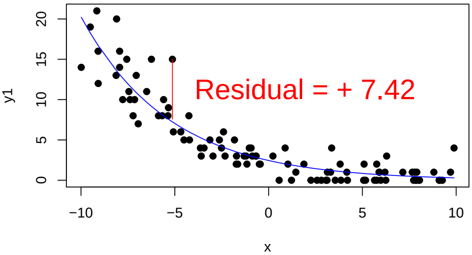
- Residuals from GLMs will never be as "pretty" as those from LMs
- *Especially* true for:
    - Binomial GLMs
    - Poisson/Negative Binomial GLMs with many zeros

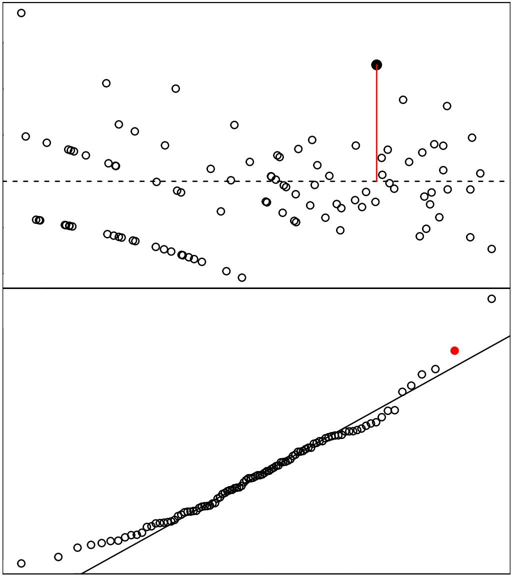




Solution: use deviance residuals for GLMs

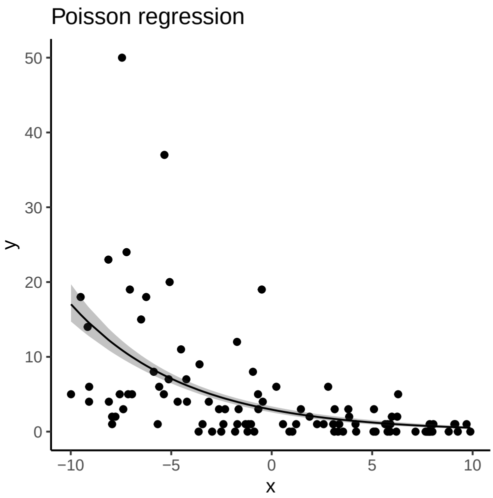
- Residuals from GLMs will never be as "pretty" as those from LMs
- *Especially* true for:
  - Binomial GLMs
  - Poisson/Negative Binomial GLMs with many zeros
- Next week we will deal with *simulation testing* residuals





Problem 2: Overdispersion

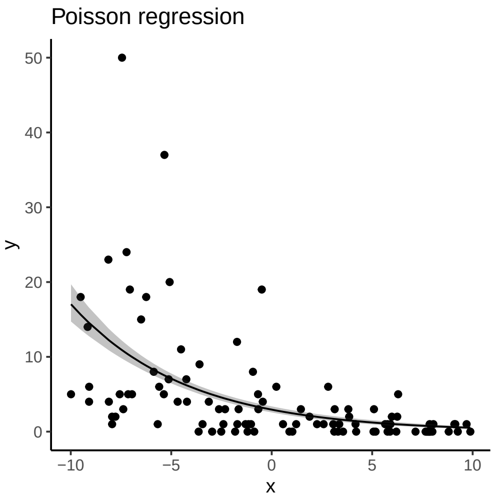
- Binomial and Poisson families have **no** variance term (e.g. *SD*).



Example: data are much more variable than the predictions from the model

Problem 2: Overdispersion

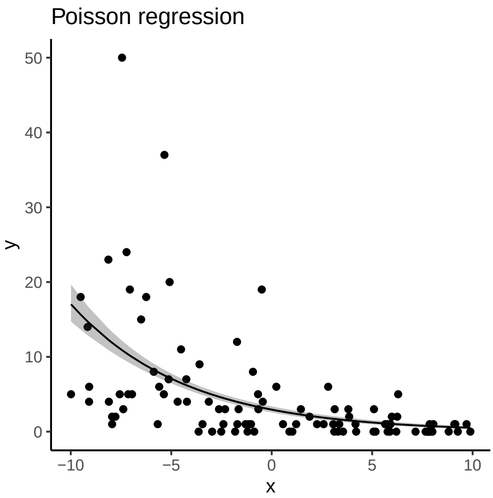
- Binomial and Poisson families have **no** variance term (e.g. *SD*).
- Sometimes this assumption doesn't work! (Very common for Poisson models)



Example: data are much more variable than the predictions from the model

Problem 2: Overdispersion

- Binomial and Poisson families have **no** variance term (e.g. *SD*).
- Sometimes this assumption doesn't work! (Very common for Poisson models)
- Strong overdispersion biases SEs, meaning that p-values are useless



Example: data are much more variable than the predictions from the model

Problem 2: Overdispersion

```
##
## Call:
## glm(formula = y1 ~ x, family = "poisson", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0843  -0.9460  -0.1897   0.5333   3.6416
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.89455    0.07818   11.44   <2e-16 ***
## x           -0.21145    0.01174  -18.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 564.27  on 99  degrees of freedom
## Residual deviance: 106.20  on 98  degrees of freedom
## AIC: 362.01
##
## Number of Fisher Scoring iterations: 5
```

- In Poisson or Binomial models, Residual deviance $\div$ Degrees of Freedom should be ~ 1

Problem 2: Overdispersion

```
##
## Call:
## glm(formula = y1 ~ x, family = "poisson", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0843  -0.9460  -0.1897   0.5333   3.6416
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.89455    0.07818   11.44   <2e-16 ***
## x           -0.21145    0.01174  -18.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 564.27  on 99  degrees of freedom
## Residual deviance: 106.20  on 98  degrees of freedom
## AIC: 362.01
##
## Number of Fisher Scoring iterations: 5
```

- In Poisson or Binomial models, Residual deviance $\div$ Degrees of Freedom should be ~ 1
- Residual deviance is the sum of all deviance from the model

Problem 2: Overdispersion

```
##
## Call:
## glm(formula = y1 ~ x, family = "poisson", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0843  -0.9460  -0.1897   0.5333   3.6416
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.89455    0.07818   11.44   <2e-16 ***
## x           -0.21145    0.01174  -18.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 564.27  on 99  degrees of freedom
## Residual deviance: 106.20  on 98  degrees of freedom
## AIC: 362.01
##
## Number of Fisher Scoring iterations: 5
```

- In Poisson or Binomial models, Residual deviance $\div$ Degrees of Freedom should be ~ 1
- Residual deviance is the sum of all deviance from the model
- This model looks OK ($106.2 \div 98 = 1.08$)

Problem 2: Overdispersion

```
##
## Call:
## glm(formula = y2 ~ x, family = "poisson", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.1009  -1.7543  -0.8805   0.4796   8.6102
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.07897    0.06871   15.70   <2e-16 ***
## x           -0.17581    0.01069  -16.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 851.96  on 99  degrees of freedom
## Residual deviance: 501.98  on 98  degrees of freedom
## AIC: 735.46
##
## Number of Fisher Scoring iterations: 5
```

- This model does **not** look OK ($501.98 \div 98 = 5.12$)

Problem 2: Overdispersion

```
##
## Call:
## glm(formula = y2 ~ x, family = "poisson", data = d1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.1009  -1.7543  -0.8805   0.4796   8.6102
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.07897    0.06871   15.70   <2e-16 ***
## x           -0.17581    0.01069  -16.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 851.96  on 99  degrees of freedom
## Residual deviance: 501.98  on 98  degrees of freedom
## AIC: 735.46
##
## Number of Fisher Scoring iterations: 5
```

- This model does **not** look OK ($501.98 \div 98 = 5.12$)
- Generated using Negative Binomial, but fit to Poisson

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data
  - e.g. Very short observation period

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data
  - e.g. Very short observation period
- Leaving out an important term

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data
  - e.g. Very short observation period
- Leaving out an important term
  - e.g. An important *interaction* term was omitted

[1] Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data
  - e.g. Very short observation period
- Leaving out an important term
  - e.g. An important *interaction* term was omitted
- Random effects[1]not accounted for

[1]Random effects discussed later

Causes

Overdispersion can be caused by different things:

- Using the wrong probability distribution
  - e.g. Poisson, but should be Negative Binomial
- Lots of zeros in count data
  - e.g. Very short observation period
- Leaving out an important term
  - e.g. An important *interaction* term was omitted
- Random effects[1]not accounted for
  - e.g. Data collected at different sites, but ignored

[1]Random effects discussed later

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
    1. Fixed effects

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
    1. Fixed effects
    2. Random effects

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
   1. Fixed effects
   2. Random effects
2. Try distributions that account for overdispersion

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
    1. Fixed effects
    2. Random effects
2. Try distributions that account for overdispersion
    1. Negative Binomial, Beta Binomial, Zero-inflated Poisson[2]

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
   1. Fixed effects
   2. Random effects
2. Try distributions that account for overdispersion
   1. Negative Binomial, Beta Binomial, Zero-inflated Poisson[2]
   2. Quasi-binomial[2] and quasi-poisson[2]

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
    1. Fixed effects
    2. Random effects
2. Try distributions that account for overdispersion
    1. Negative Binomial, Beta Binomial, Zero-inflated Poisson[2]
    2. Quasi-binomial[2] and quasi-poisson[2]
    3. Transform counts to presence/absence

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
    1. Fixed effects
    2. Random effects
2. Try distributions that account for overdispersion
    1. Negative Binomial, Beta Binomial, Zero-inflated Poisson[2]
    2. Quasi-binomial[2] and quasi-poisson[2]
    3. Transform counts to presence/absence
3. Lower your expectations, and use a lower critical p-value (e.g. 0.01 instead of 0.05)

[2] These can be annoying to deal with, so avoid if possible

Solutions for overdispersion

Try the following (in this order):

1. Consider terms that may have been left out
   1. Fixed effects
   2. Random effects
2. Try distributions that account for overdispersion
   1. Negative Binomial, Beta Binomial, Zero-inflated Poisson[2]
   2. Quasi-binomial[2] and quasi-poisson[2]
   3. Transform counts to presence/absence
3. Lower your expectations, and use a lower critical p-value (e.g. 0.01 instead of 0.05)
4. Design a better study :(

[2]These can be annoying to deal with, so avoid if possible

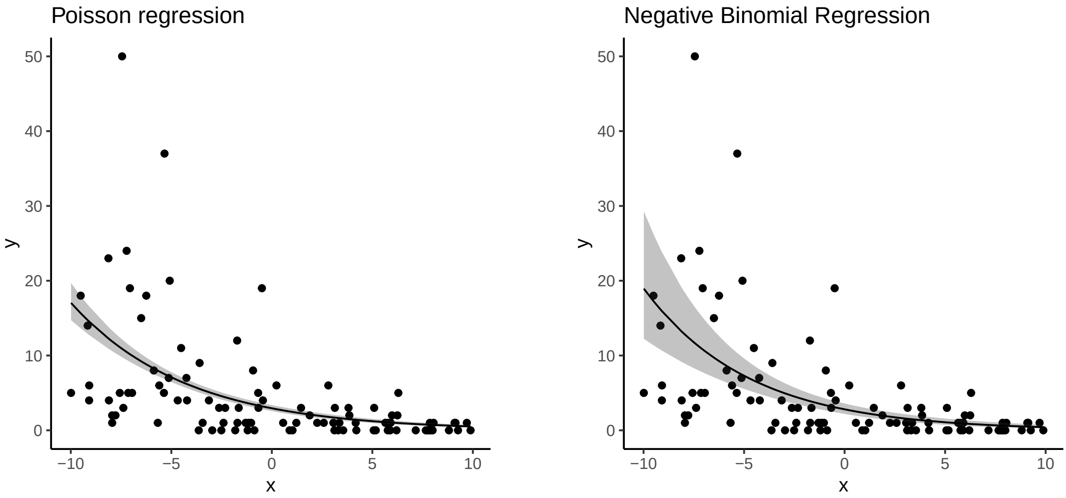
Negative Binomial Regression

```r
library(MASS) #Required for NB models
m3 <- glm.nb(y2~x,data=d1)
summary(m3)
```

- Model no longer indicates overdispersion!

```
##
## Call:
## glm.nb(formula = y2 ~ x, data = d1, init.theta = 1.075023363,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9829  -1.0492  -0.3989   0.3831   2.4289
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.03037    0.12281   8.390   <2e-16 ***
## x           -0.19131    0.02222  -8.609   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.075) family taken to be 1)
##
##     Null deviance: 182.31  on 99  degrees of freedom
## Residual deviance: 103.87  on 98  degrees of freedom
## AIC: 458.65
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  1.075
##           Std. Err.:  0.216
##
##  2 x log-likelihood:  -452.653
```

Negative Binomial Regression (cont.)



Zero-inflation: drunk monks

A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)

Zero-inflation: drunk monks

A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)
2. Some days they decide to try out the beer that's been brewing in the cellar! No manuscripts get made on those days.

Zero-inflation: drunk monks

A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)
2. Some days they decide to try out the beer that's been brewing in the cellar! No manuscripts get made on those days.
3. The number of manuscripts made (per day) follows a *zero-inflated Poisson distribution*

Zero-inflation: drunk monks

A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)
2. Some days they decide to try out the beer that's been brewing in the cellar! No manuscripts get made on those days.
3. The number of manuscripts made (per day) follows a *zero-inflated Poisson distribution*

Zero-inflation: drunk monks

A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)
2. Some days they decide to try out the beer that's been brewing in the cellar! No manuscripts get made on those days.
3. The number of manuscripts made (per day) follows a *zero-inflated Poisson distribution*

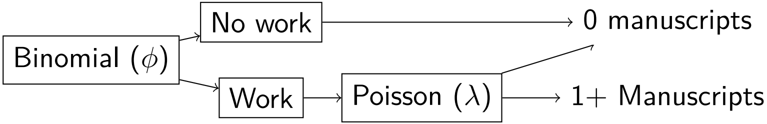
This is *mixture* of a Poisson and a Binomial:

Zero-inflation: drunk monks

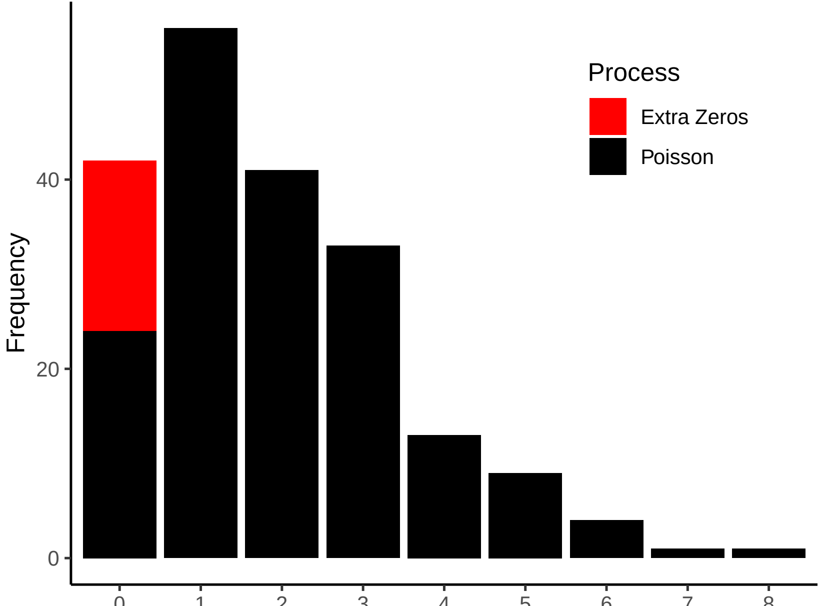
A way to think about this model:

1. Monks at a monastery make copies of manuscripts. Most days they make very few (0 or 1), but occasionally they make many (2-5)
2. Some days they decide to try out the beer that's been brewing in the cellar! No manuscripts get made on those days.
3. The number of manuscripts made (per day) follows a *zero-inflated Poisson distribution*

This is *mixture* of a Poisson and a Binomial:

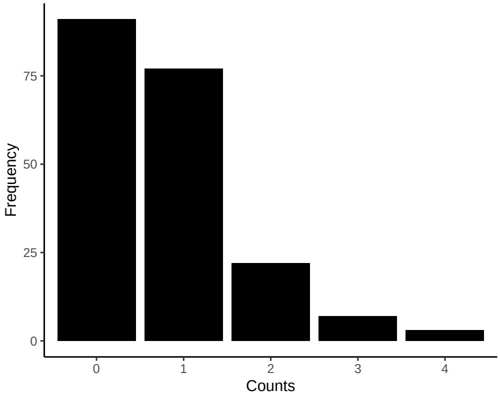


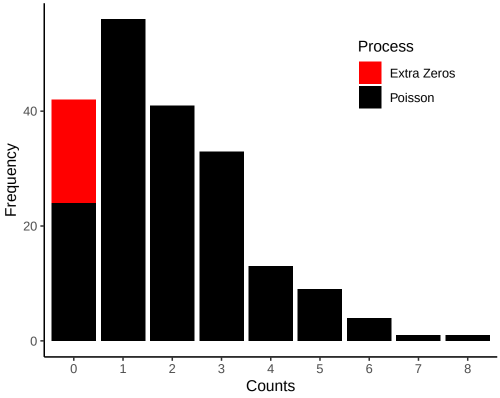
Zero-inflation: graphical model



Problem: hard to fit

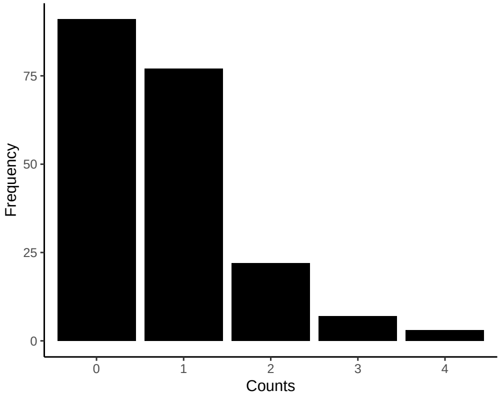
- Hard for R to tell the difference between ZIP/ZINB, and a Poisson/NB with a low mean (λ).

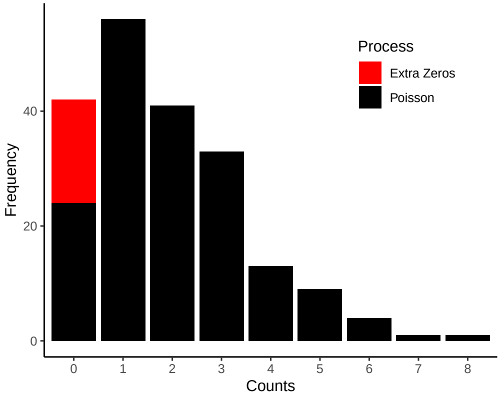




Problem: hard to fit

- Hard for R to tell the difference between ZIP/ZINB, and a Poisson/NB with a low mean (λ).
- This needs a lot of data in order to work! Consider longer sampling periods in order to reduce zeros





Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**

[3] "My God, it's full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is “not significant” doesn’t mean it should be dropped out!

[3] "My God, it’s full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is "not significant" doesn't mean it should be dropped out!
  - Just because a term is "significant" doesn't mean it should be left in!

[3] "My God, it's full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is “not significant” doesn’t mean it should be dropped out!
  - Just because a term is “significant” doesn’t mean it should be left in!
  - Avoid selecting models based on R^2 or AIC alone. Avoid stargazing[3](hunting around for “better” p-values or AIC scores)

[3]"My God, it's full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is “not significant” doesn’t mean it should be dropped out!
  - Just because a term is “significant” doesn’t mean it should be left in!
  - Avoid selecting models based on R^2 or AIC alone. Avoid stargazing[3](hunting around for “better” p-values or AIC scores)
- To test whether terms are important in predicting your data (similar to), use *likelihood-ratio tests*

[3]"My God, it's full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is “not significant” doesn’t mean it should be dropped out!
  - Just because a term is “significant” doesn’t mean it should be left in!
  - Avoid selecting models based on R^2 or AIC alone. Avoid stargazing[3](hunting around for “better” p-values or AIC scores)
- To test whether terms are important in predicting your data (similar to), use *likelihood-ratio tests*
  - `drop1(model,test='Chisq')`

[3]"My God, it's full of stars!" -2001, A Space Odyssey

Model selection

How many terms should be in my model?

- Same principle as in regular linear models: **what do you think the process is?**
  - Just because a term is "not significant" doesn't mean it should be dropped out!
  - Just because a term is "significant" doesn't mean it should be left in!
  - Avoid selecting models based on R^2 or AIC alone. Avoid stargazing[3](hunting around for "better" p-values or AIC scores)
- To test whether terms are important in predicting your data (similar to), use *likelihood-ratio tests*
  - `drop1(model,test='Chisq')`
  - AIC tests usually say the same thing as LR tests

[3] "My God, it's full of stars!" -2001, A Space Odyssey

Other useful things about GLMs!

- Binomial GLMs with >1 trial

Other useful things about GLMs!

- Binomial GLMs with $>$1 trial
- Offsets in count models

Other useful things about GLMs!

- Binomial GLMs with $>$1 trial
- Offsets in count models
- R^2 for GLMs

Other useful things about GLMs!

- Binomial GLMs with $>$1 trial
- Offsets in count models
- R^2 for GLMs
- Partial effects plots

Binomial GLMs with $>$1 trial

- If you're measuring single "success/failures", 1s and 0s are used

Binomial GLMs with $>$1 trial

- If you're measuring single "success/failures", 1s and 0s are used
- If multiple trials occur, R requires counts of successes and failures

Binomial GLMs with >1 trial

- If you're measuring single "success/failures", 1s and 0s are used
- If multiple trials occur, R requires counts of successes and failures
- Example: "I counted male and female critters at different sites. Does temperature affect sex ratios?"

Binomial GLMs with $>$1 trial

- If you're measuring single "success/failures", 1s and 0s are used
- If multiple trials occur, R requires counts of successes and failures
- Example: "I counted male and female critters at different sites. Does temperature affect sex ratios?"

Binomial GLMs with >1 trial

- If you're measuring single "success/failures", 1s and 0s are used
- If multiple trials occur, R requires counts of successes and failures
- Example: "I counted male and female critters at different sites. Does temperature affect sex ratios?"

```
#Number of females and males are in 2 separate columns in d1
glm(cbind(females,males) ~ temp, family='binomial',data = d1)
```

This will correctly account for different numbers of critters ("trials") at each site

Offsets in count models

"I counted critters for different lengths of time at each site. Does temperature affect counts?"

- Count models use integers only, so you can't just do: $counts \div hours$

```
#hours = observation time at each site (must be log-transformed)
glm(counts ~ offset(log(hours)) + temp, family='poisson',data = d1)
```

This will return estimates that have been scaled to a 1-hour observation time

Offsets in count models

"I counted critters for different lengths of time at each site. Does temperature affect counts?"

- Count models use integers only, so you can't just do: $counts \div hours$
- Solution: use *offsets* to deal with different observation times

```
#hours = observation time at each site (must be log-transformed)
glm(counts ~ offset(log(hours)) + temp, family='poisson',data = d1)
```

This will return estimates that have been scaled to a 1-hour observation time

Offsets in count models

"I counted critters for different lengths of time at each site. Does temperature affect counts?"

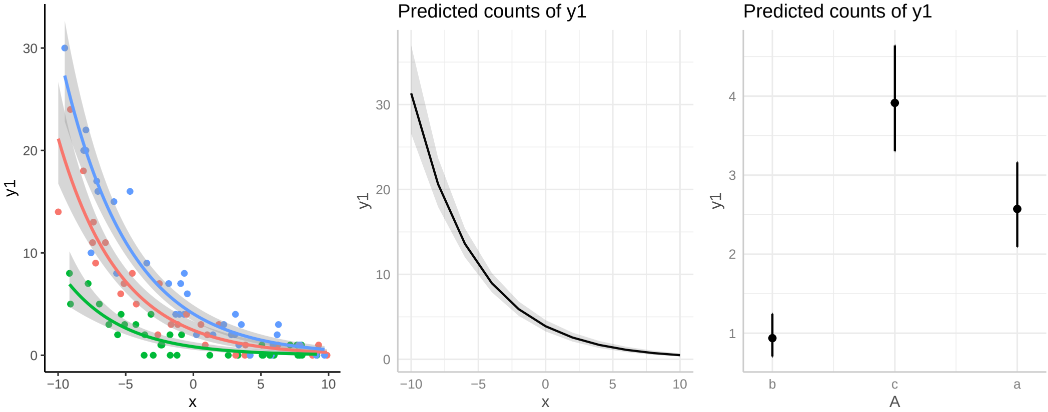
- Count models use integers only, so you can't just do: *counts* ÷ *hours*
- Solution: use *offsets* to deal with different observation times
    - Predictor with a slope fixed at 1

```r
#hours = observation time at each site (must be log-transformed)
glm(counts ~ offset(log(hours)) + temp, family='poisson',data = d1)
```

This will return estimates that have been scaled to a 1-hour observation time

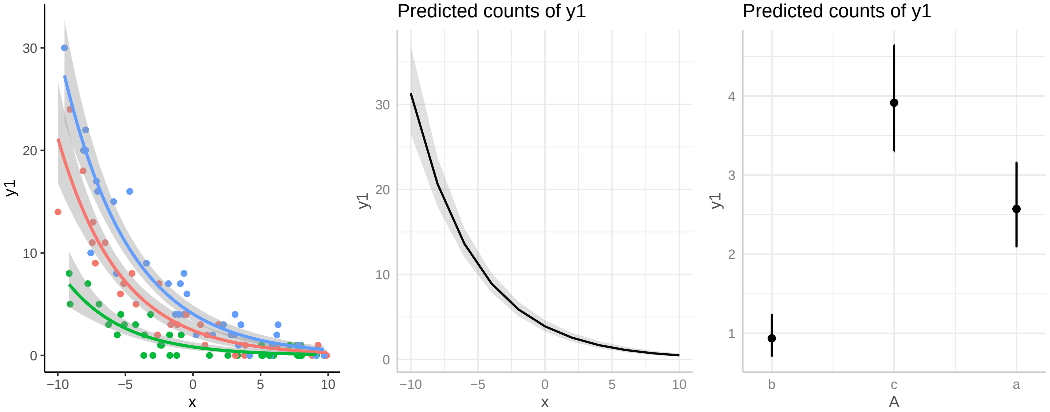
Partial effects plots

- `library(ggeffects)` and `library(effects)` work for partial effects plots, but. . .



Partial effects plots

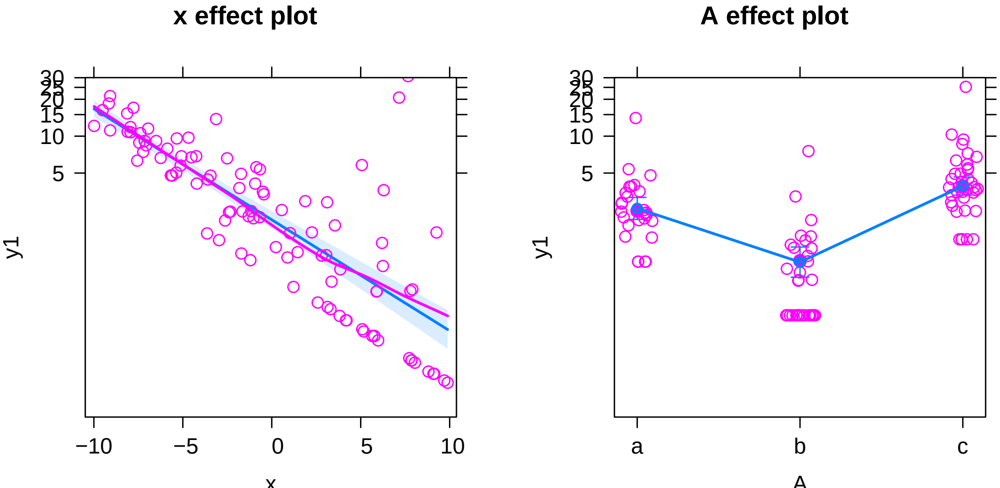
- `library(ggeffects)` and `library(effects)` work for partial effects plots, but. . .
- Residuals are tricky to display, unless you plot them on the link scale



Partial effects plots

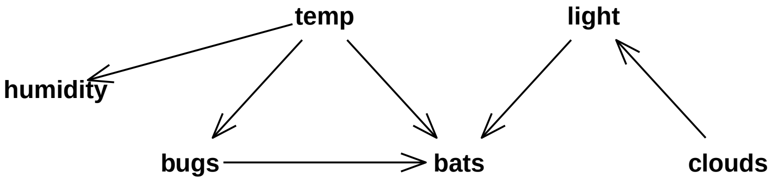
- Plots from `effects` use *working residuals* (not on the link scale)

```r
library(effects)
plot(allEffects(m4,residuals=TRUE))
```



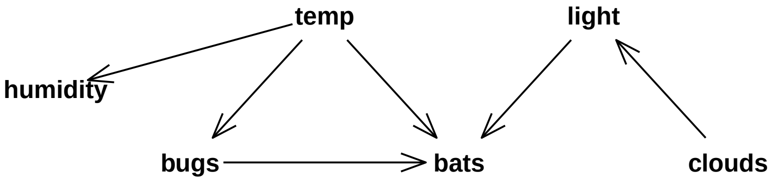
Third challenge

- Remember that bat data from last week? (Found here in `batDat.csv`). We used a `lm` last week to fit it, but it actually came from a `glm`



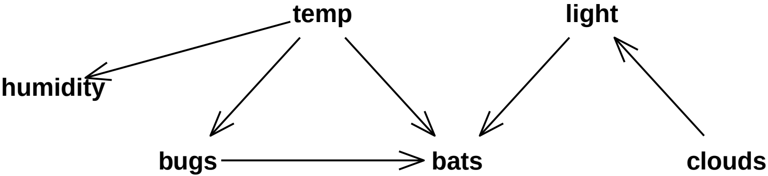
Third challenge

- Remember that bat data from last week? (Found here in `batDat.csv`). We used a `lm` last week to fit it, but it actually came from a `glm`
- Fit a `glm` to those data, check how the model fits, and make some partial effects plots

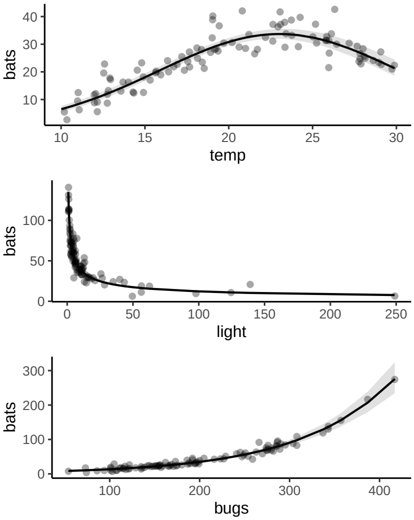


Third challenge

- Remember that bat data from last week? (Found here in `batDat.csv`). We used a `lm` last week to fit it, but it actually came from a `glm`
- Fit a `glm` to those data, check how the model fits, and make some partial effects plots
- Recall: this is the "true" causal relationship here (no interactions)



Solution:




```
##
## Call:
## glm(formula = bats ~ poly(temp, 2) + log(light) + bugs, family = "poisson",
##     data = batDat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1595  -0.6406  -0.0559   0.6176   2.0356
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     2.756999   0.077780   35.45   <2e-16 ***
## poly(temp, 2)1  3.253942   0.321972   10.11   <2e-16 ***
## poly(temp, 2)2 -2.765370   0.160287  -17.25   <2e-16 ***
## log(light)     -0.508112   0.011943  -42.54   <2e-16 ***
## bugs            0.009513   0.000346   27.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 8237.284  on 99  degrees of freedom
## Residual deviance:   91.794  on 95  degrees of freedom
## AIC: 645.77
##
## Number of Fisher Scoring iterations: 4
```

To do this week: update your models

- Some of you have non-normal data... time to update those models!

To do this week: update your models

- Some of you have non-normal data. . . time to update those models!
- For those who don't need to run GLMS, it's your lucky week!