

# Project

1) Find a dataset (assuming you don't have one already)

Route 1: Look for papers (Google scholar)

Route 2: Look for datasets (Dryad)

**Proposal is due in 2 weeks:  
12 Feb 2359h**

2) Do some reading of the paper to see what is currently known in that field

This is your Background

3) Come up with a research question

Use the variables in the dataset; ask a different question from the paper

Ask yourself: what type of variables do you have? Therefore what type of analysis would I have to use? (You will have a better idea of what I mean after today's lecture)

4) Write the Abstract (for the proposal based on the above). See the sample provided in Canvas.

# Basic Statistical Concepts and Tests

Lecture 3

**LSM3257**

AY22/23; Sem 2 | Ian Z.W. Chan



# Summary (Learning Objectives)

## Basic statistical concepts

- Variable types:

  - Properties: Continuous vs. Categorical vs. Discrete

  - Function: Explanatory vs. Response

- Describing data: Mean, Variance, df, SD, SE, Confidence Intervals

- Modelling data with Distributions: continuous and discrete


## Basic statistical tests

- Experimental design

- Test assumptions: **Independent and Identically Distributed**, normality, equal variances; testing for normality and equal variances

- Tests: based on number and type of explanatory and response variables

- Other considerations: Corrections for multiple comparisons, Power analysis



# Part 1: Basic Concepts

## Variable types

Categorical vs. Continuous vs. Discrete

X vs. Y: Explanatory vs. Response

# Types of variables (properties): categorical vs. continuous vs. discrete

**Categorical:** Variables with a finite number of groups

- Ordinal: categories with a logical order, e.g. Clothing size
- Nominal: categories without a logical order, e.g. Ethnic group

**Continuous:** Variables with infinite number of values between any two values

- E.g. Length, Height, Weight, Area

**Discrete:** Numeric variables that have a countable number of values between any two values

- Can be treated as categorical or continuous based on the **number of different values**
- **Size of the values** also affects the type of tests you should use (more on this later)
- E.g. Number of students

# Types of variables (function): explanatory vs. response

## X-axis

Predict or explain differences in another variable

Example: pollution levels in a land plot

Many names:

**Explanatory** variable

Predictor variable

**Independent** variable

Control variable

Regressor

## Y-axis

The variable to be predicted or explained

Example: biodiversity in that plot

**Response** variable

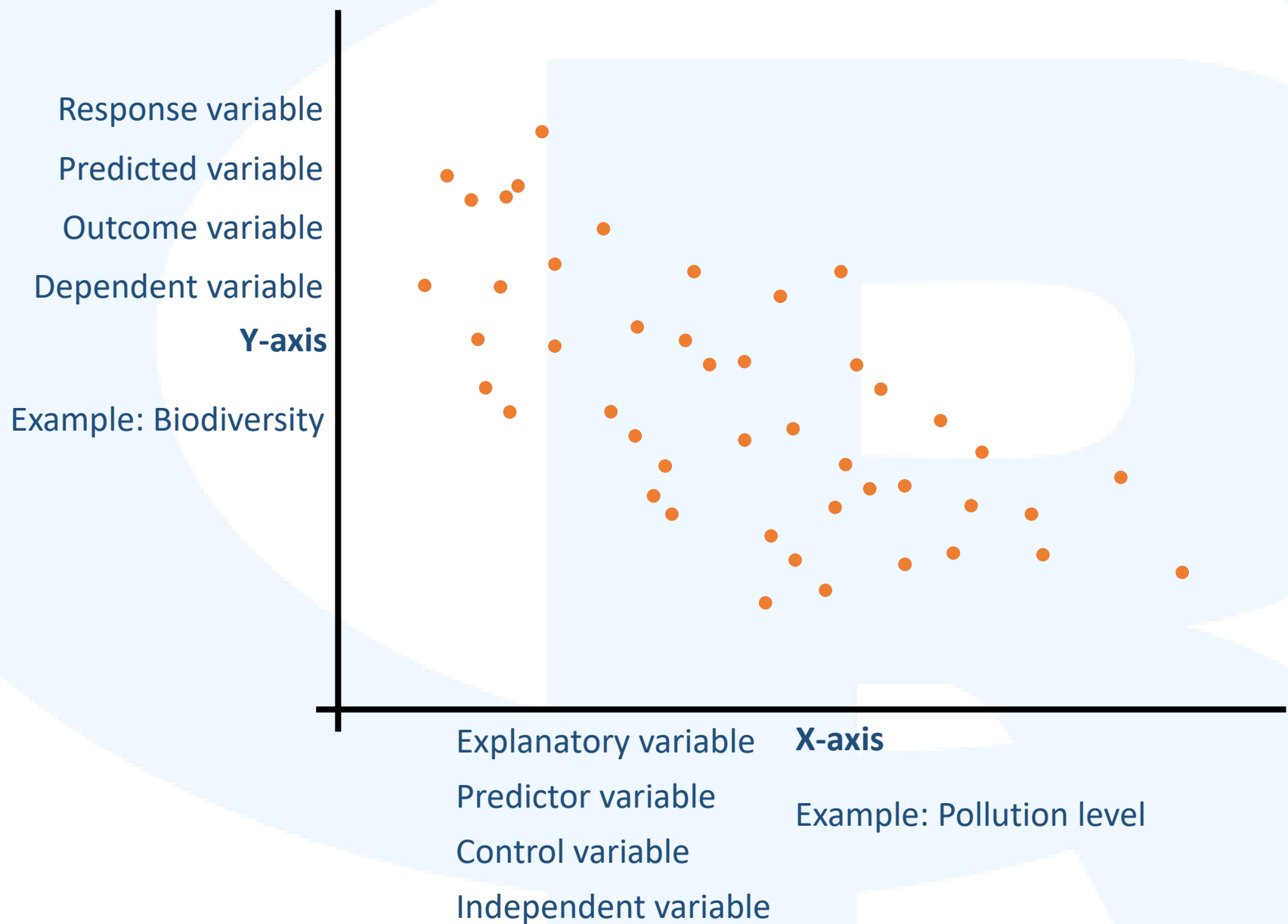
Predicted variable

**Dependent** variable

Outcome variable

Regressand

# Types of variables: explanatory vs. response





# Describing data



# Mean

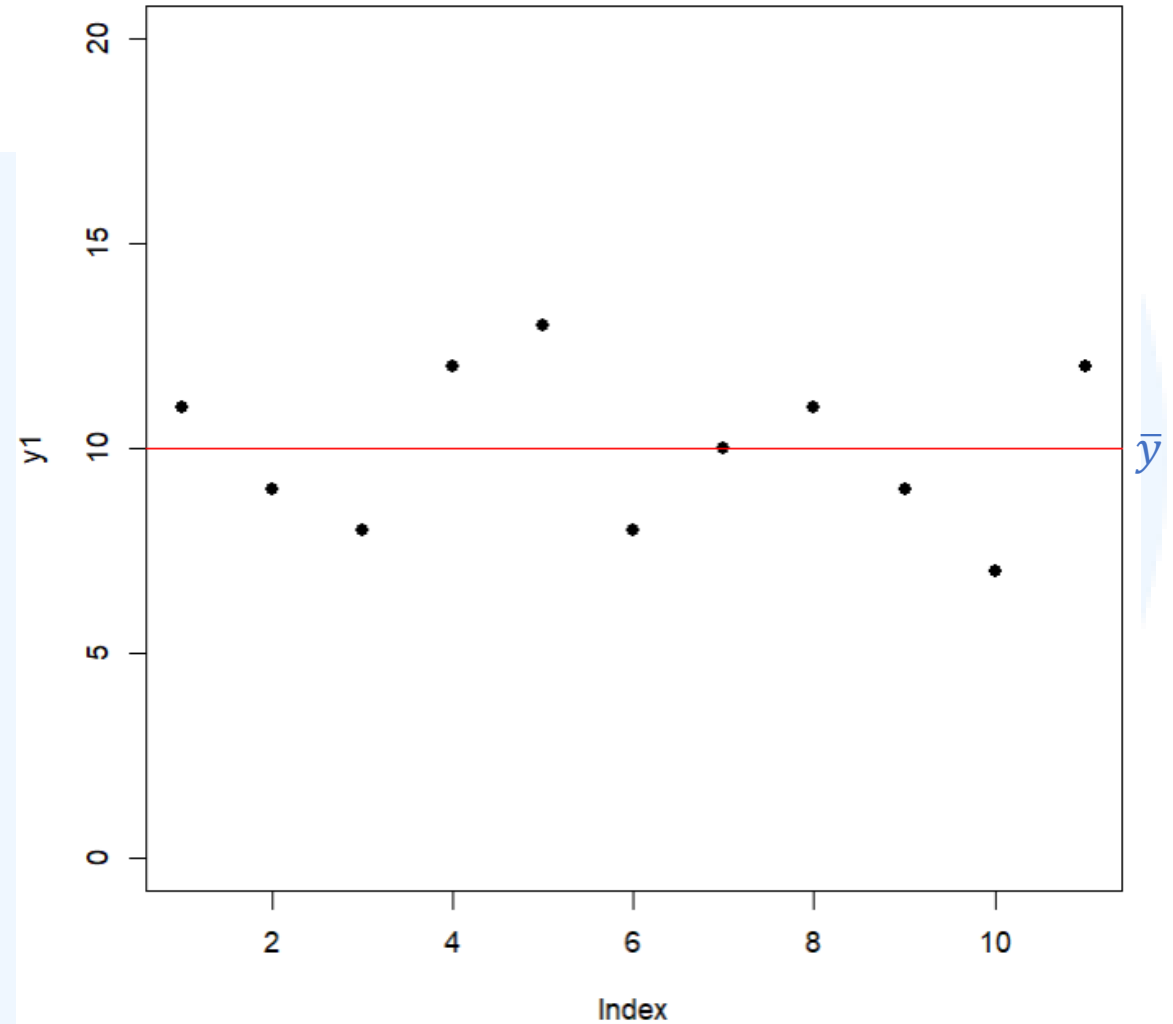
#Imagine these data:

```
y1=c(11,9,8,12,13,8,10,11,9,7,12)
plot(y1,ylim=c(0,20),pch=16)
```

How do we describe the data?:

1) The central (**Mean**) value

```
#Plot mean line
abline(h=mean(y1),col="red")
```



This is great! It gives us an idea of what the values are like.

But that doesn't tell us about how spread out the datapoints are, so...

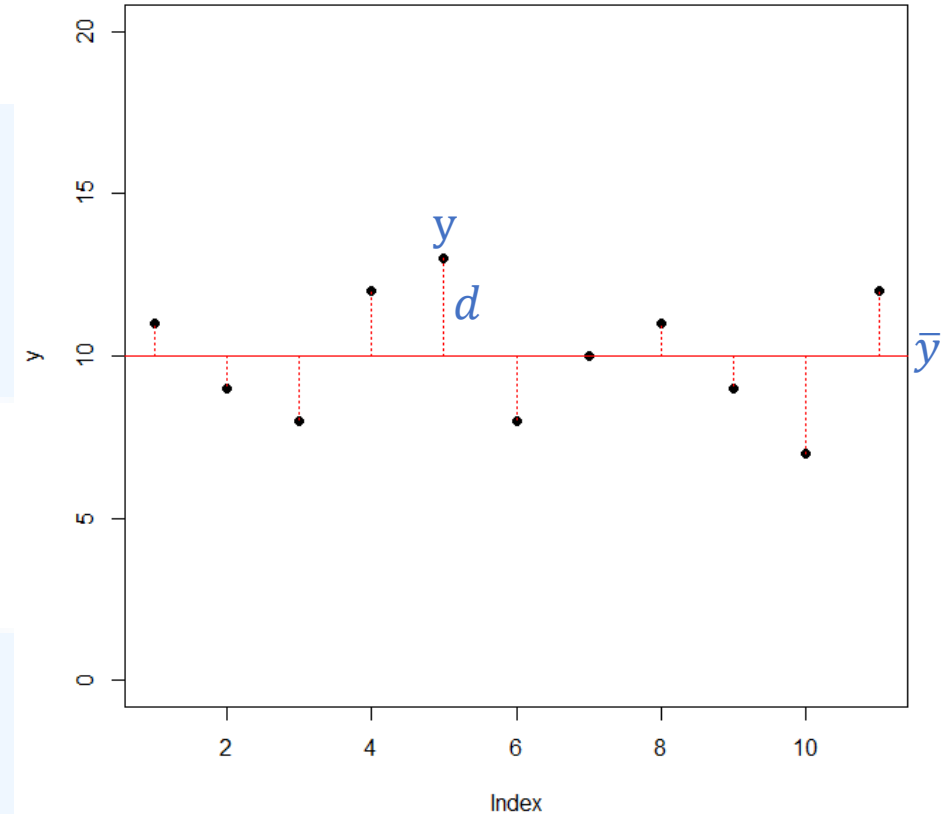
# Variance

## 2) The Variance

Variability is probably the most important measure of data in statistical analyses.

- More variability in the data = more uncertainty with our results (the values of the parameters estimated) = less ability to distinguish between competing hypothesis (e.g. null vs. alternative).

Conclusion: **high variability is bad.**



One way to measure variability is to calculate the distance between each point and the mean.

```
#Plot distances to the mean
```

```
segments(x0=seq(1,11),x1=seq(1,11),y0=mean(y1),y1=y1,col="red",lty=3)
```

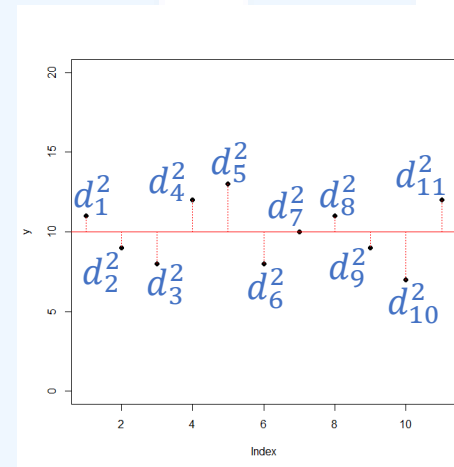
## Variance—how to calculate?

For the entire dataset, we would sum all the deviations ( $d$ ) between each point ( $y$ ) and the mean ( $\bar{y}$ ):  $\sum d = \sum y - \bar{y}$ .

Problem: positive and negative values will cancel out.

- Solution 1: use absolute values instead. (Not very elegant)
- Solution 2: use squared values instead:

$$\sum d^2 = \sum (y - \bar{y})^2 \text{ (this is the famous “sum of squares”)}$$



Problem: the more values we have, the larger the sum of squares becomes.

- Solution: use the mean of the sum of squares (this is *almost* the Variance).

Problem (small one): we could not calculate the sum of squares before knowing the value of  $\bar{y}$ . This had to be estimated from the data and we need to account for that first. We need to learn a very important concept: **degrees of freedom**.

# Variance—degrees of freedom

**Degrees of freedom (df):** the number of values in a dataset with the “freedom to vary”.

## Example

$n = 3$

--	--	--

mean = 5

The first number has the freedom to be anything:

4		
---	--	--

mean = 5

The second number still has the freedom to be anything:

4	5	
---	---	--

mean = 5

The third number has no freedom, it **MUST** be 6!

This dataset of  $n = 3$  had  $df = 2$  (two numbers can vary freely).

In general: we will have  $n-1$  degrees of freedom if we estimate one parameter (in this case, the mean) from the sample.

### Why do we use df instead of $n$ ?

To get a truer estimate of our uncertainty, we need to account for the fact that we actually only had  $n-1$  “free” datapoints: dividing by a smaller number gives you larger uncertainty.

### Read here:

<https://medium.com/@dlectus/degrees-of-freedom-simply-explained-a96cafa3b39f>

# Variance & Standard Deviation

We can now calculate the variance:

$$\text{Variance} = \frac{\text{sum of squares}}{\text{degrees of freedom}} = \frac{\sum (y - \bar{y})^2}{n - 1}$$

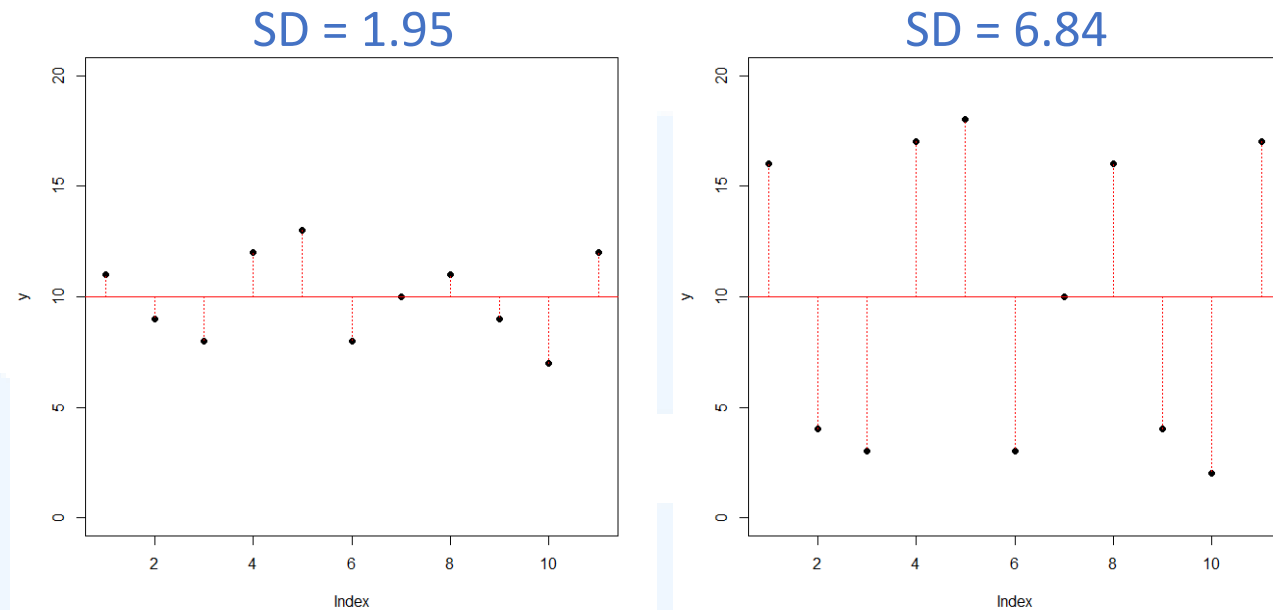
Last step: we take the square root of the Variance to convert Variance back to the original units – this is known as the **Standard Deviation (SD)**:  $\text{Variance} = \text{SD}^2$

2 datasets

Same Mean

Different Variance

Which mean looks more reliable?



Understanding Variance will be important later on when choosing statistical tests

# Standard Error

Standard error (SE) ≠ Standard deviation (SD)

- SD measures the distance of the individual datapoints from the sample mean
- SE measures **how likely the sample mean you have is the true population mean**

SE is thus a measure of the unreliability of your mean. It would get smaller as you increase sample size (you're more confident with bigger sample size) and get bigger as the Variance goes up, therefore:

$$SE_{\bar{y}} = \sqrt{\frac{SD^2}{n}}$$

When reporting results in your papers, you would write something like:

“The height of the species was 3.0 cm ± 0.25 (mean + S.E., n = 35)”

# Confidence Intervals

Confidence intervals show the likely range in which the sample mean would fall if you repeated your sampling multiple times.

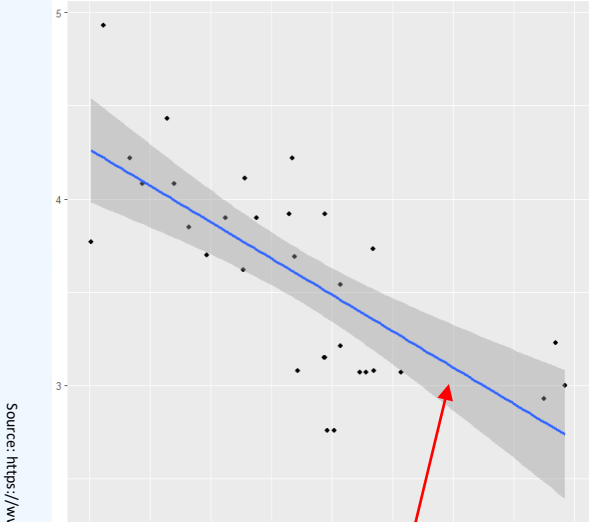
It is a measure of the (un)reliability of your model and would:

- Increase with SD
- Decrease with n
- Increase with confidence level

How to calculate?:

- Old school, checked in tables like this...

t Table											
cum. prob one-tail two-tails											
	t <sub>.50</sub>	t <sub>.75</sub>	t <sub>.90</sub>	t <sub>.95</sub>	t <sub>.98</sub>	t <sub>.99</sub>	t <sub>.995</sub>	t <sub>.998</sub>	t <sub>.999</sub>	t <sub>.9995</sub>	t <sub>.9998</sub>
	1.00	0.50	0.40	0.30	0.20	0.10	0.05	0.025	0.01	0.005	0.001
df											
1	0.000	1.000	1.376	1.963	3.078	6.314	12.71	31.82	63.66	318.31	636.62
2	0.000	0.816	1.061	1.386	1.886	2.920	4.303	6.965	9.925	22.327	31.599
3	0.000	0.765	0.978	1.250	1.638	2.353	3.182	4.541	5.841	10.215	12.924
4	0.000	0.741	0.941	1.190	1.533	2.132	2.776	3.747	4.604	7.173	8.610
5	0.000	0.727	0.920	1.156	1.476	2.015	2.571	3.365	4.032	5.893	6.869
6	0.000	0.718	0.906	1.134	1.440	1.943	2.447	3.143	3.707	5.208	5.959
7	0.000	0.711	0.896	1.119	1.415	1.895	2.365	2.998	3.499	4.785	5.408
8	0.000	0.706	0.889	1.108	1.397	1.860	2.306	2.896	3.355	4.501	5.041
9	0.000	0.703	0.883	1.100	1.383	1.833	2.262	2.821	3.250	4.297	4.781
10	0.000	0.700	0.879	1.093	1.372	1.812	2.228	2.764	3.169	4.144	4.587
11	0.000	0.697	0.876	1.088	1.363	1.796	2.201	2.718	3.106	4.025	4.437
12	0.000	0.695	0.873	1.083	1.356	1.782	2.179	2.681	3.055	3.930	4.318
13	0.000	0.694	0.870	1.079	1.350	1.771	2.160	2.650	3.012	3.852	4.221
14	0.000	0.692	0.868	1.076	1.345	1.761	2.145	2.624	2.977	3.787	4.140
15	0.000	0.691	0.866	1.074	1.341	1.753	2.131	2.602	2.947	3.733	4.073
16	0.000	0.690	0.865	1.071	1.337	1.746	2.120	2.583	2.921	3.686	4.015
17	0.000	0.689	0.863	1.069	1.333	1.740	2.110	2.567	2.898	3.646	3.965
18	0.000	0.688	0.862	1.067	1.330	1.734	2.101	2.552	2.878	3.610	3.922
19	0.000	0.688	0.861	1.066	1.328	1.729	2.093	2.539	2.861	3.579	3.883
20	0.000	0.687	0.860	1.064	1.325	1.725	2.086	2.528	2.845	3.552	3.850
21	0.000	0.686	0.859	1.063	1.323	1.721	2.080	2.518	2.831	3.527	3.819
22	0.000	0.686	0.858	1.061	1.321	1.717	2.074	2.508	2.819	3.505	3.792
23	0.000	0.685	0.858	1.060	1.319	1.714	2.069	2.500	2.807	3.485	3.768
24	0.000	0.685	0.857	1.059	1.318	1.711	2.064	2.492	2.797	3.467	3.745
25	0.000	0.684	0.856	1.058	1.316	1.708	2.060	2.485	2.787	3.450	3.725
26	0.000	0.684	0.856	1.058	1.315	1.706	2.056	2.479	2.779	3.435	3.707
27	0.000	0.684	0.855	1.057	1.314	1.703	2.052	2.473	2.771	3.421	3.690
28	0.000	0.683	0.855	1.056	1.313	1.701	2.048	2.467	2.763	3.408	3.674
29	0.000	0.683	0.854	1.055	1.311	1.699	2.045	2.462	2.756	3.396	3.659
30	0.000	0.683	0.854	1.055	1.310	1.697	2.042	2.457	2.750	3.385	3.646
40	0.000	0.681	0.851	1.050	1.303	1.684	2.021	2.423	2.704	3.307	3.551
60	0.000	0.679	0.848	1.045	1.296	1.671	2.000	2.390	2.660	3.232	3.460
80	0.000	0.678	0.846	1.043	1.292	1.664	1.990	2.374	2.639	3.195	3.416
100	0.000	0.677	0.845	1.042	1.290	1.660	1.984	2.364	2.626	3.174	3.390
1000	0.000	0.675	0.842	1.037	1.282	1.646	1.962	2.330	2.581	3.098	3.300
Z	0.000	0.674	0.842	1.036	1.282	1.645	1.960	2.326	2.576	3.090	3.291
	0%	50%	60%	70%	80%	90%	95%	98%	99%	99.8%	99.9%
	Confidence Level										



Source: <https://www.sjsu.edu/faculty/gerstman/StatPrimer/t-table.pdf>

The shaded area is the confidence interval of the blue line

Question: why does it get bigger towards the ends?

# Confidence Intervals

## How to calculate?:

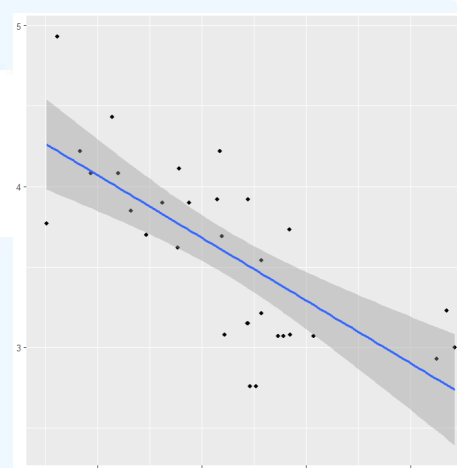
- Now we can easily calculate and plot them...

```
library(ggplot2)
f=ggplot(mtcars,aes(x=wt,col=drat))
f+geom_point()+geom_smooth(method=lm,level=0.95)
#default 95%

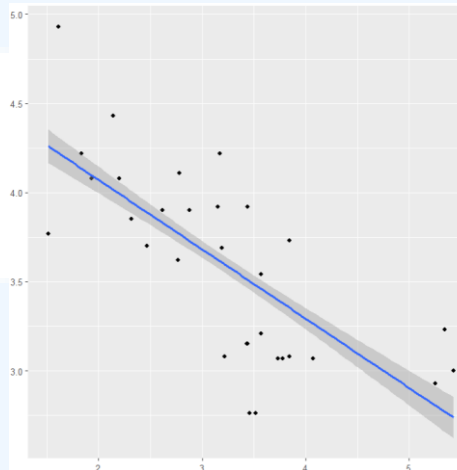
f+geom_point()+geom_smooth(method=lm,level=0.5)
#50% confidence level, smaller interval!

f+geom_point()+geom_smooth(method=lm,level=0.9999)
#99.99% confidence level, larger interval!
```

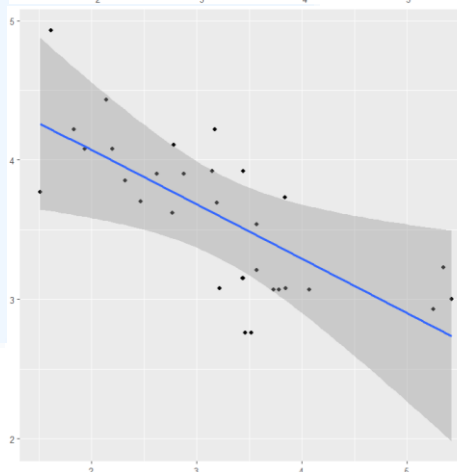
95%



50%



99.99%



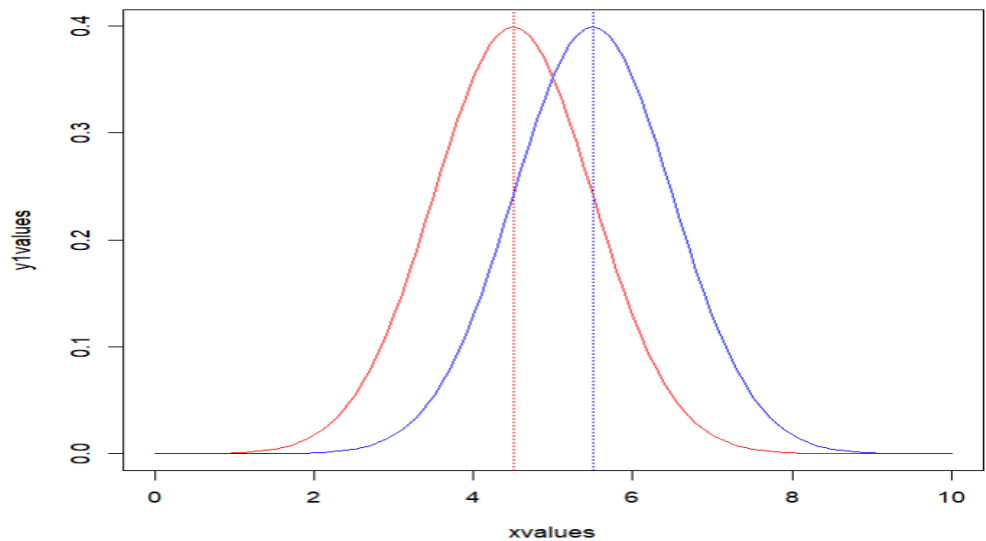
**Debugging Time!**

Can you spot the error in this slide?

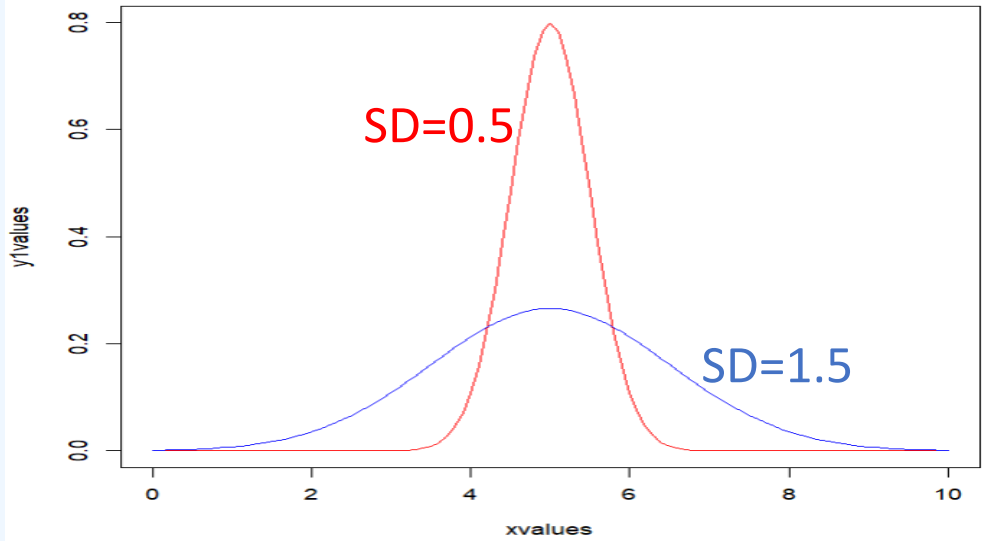


# 4 moments of distributions

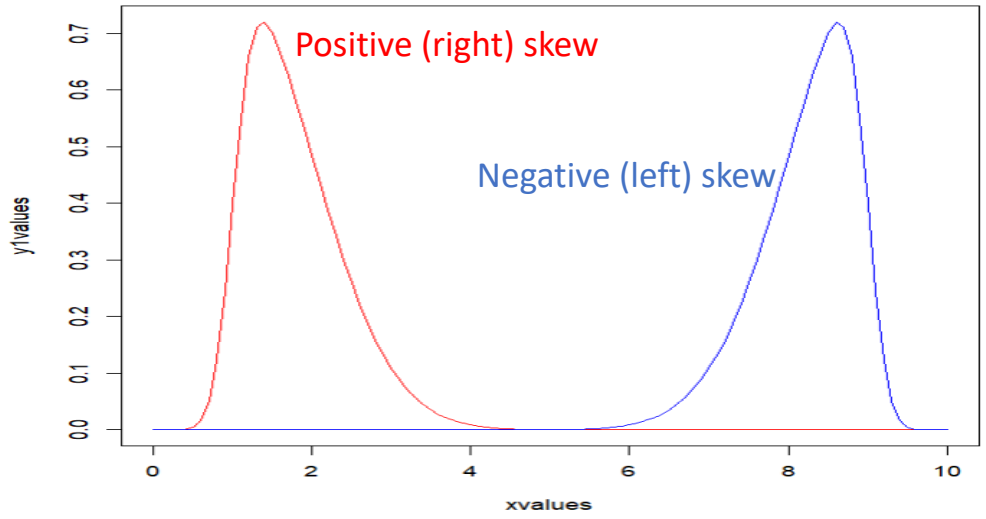
## Moment 1: Mean



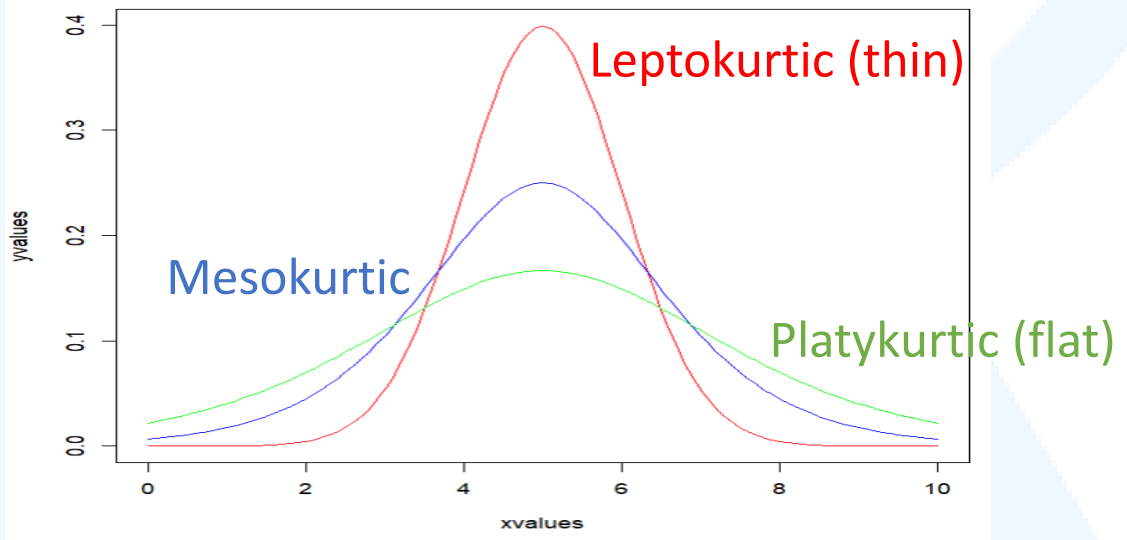
## Moment 2: Variance



## Moment 3: Skew



## Moment 4: Kurtosis





# Modelling data

Distributions

# Distributions

Mathematical functions that are used to model the values of your variables: different functions have different parameters and can take different shapes

- 2 types: **Continuous distributions** (for continuous data) and **Discrete distributions** (for categorical data)
- No distribution will fit your data perfectly; we always choose the best

Some analyses assume that your data follow certain distributions

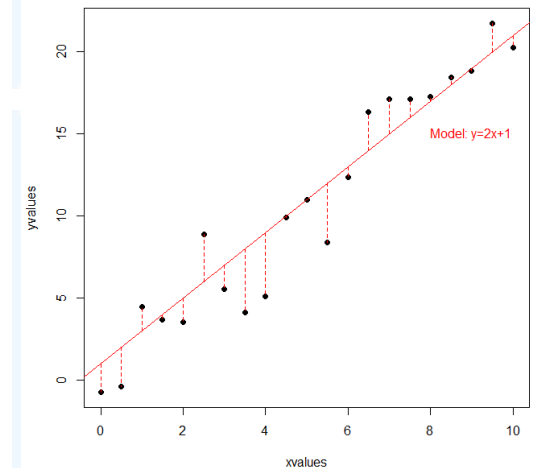
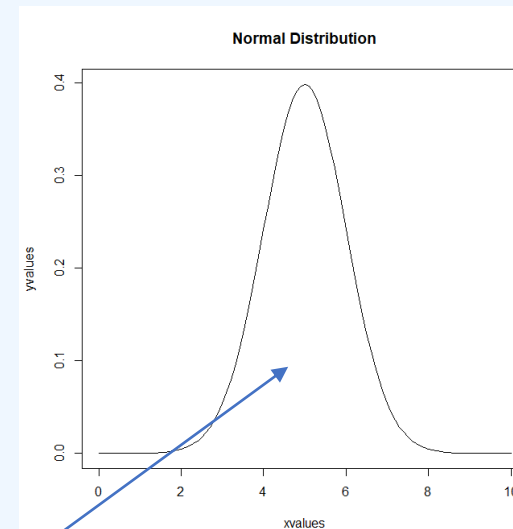
- If this is not true, then you cannot use the analysis

## Example: Normal distribution

- Easy example: height and weight measurements
- More abstract example: the errors in a dataset

In Normal distributions:

- Most values are near the mean
- Values are symmetrical about the mean



# Continuous distributions

## Normal Distribution

Model data that is symmetrical about the mean.

The most common because of the Central Limit Theorem.

- Many distributions converge to it with large enough sample size.
- Most classical statistical methods (e.g. parametric tests) are based on this distribution.

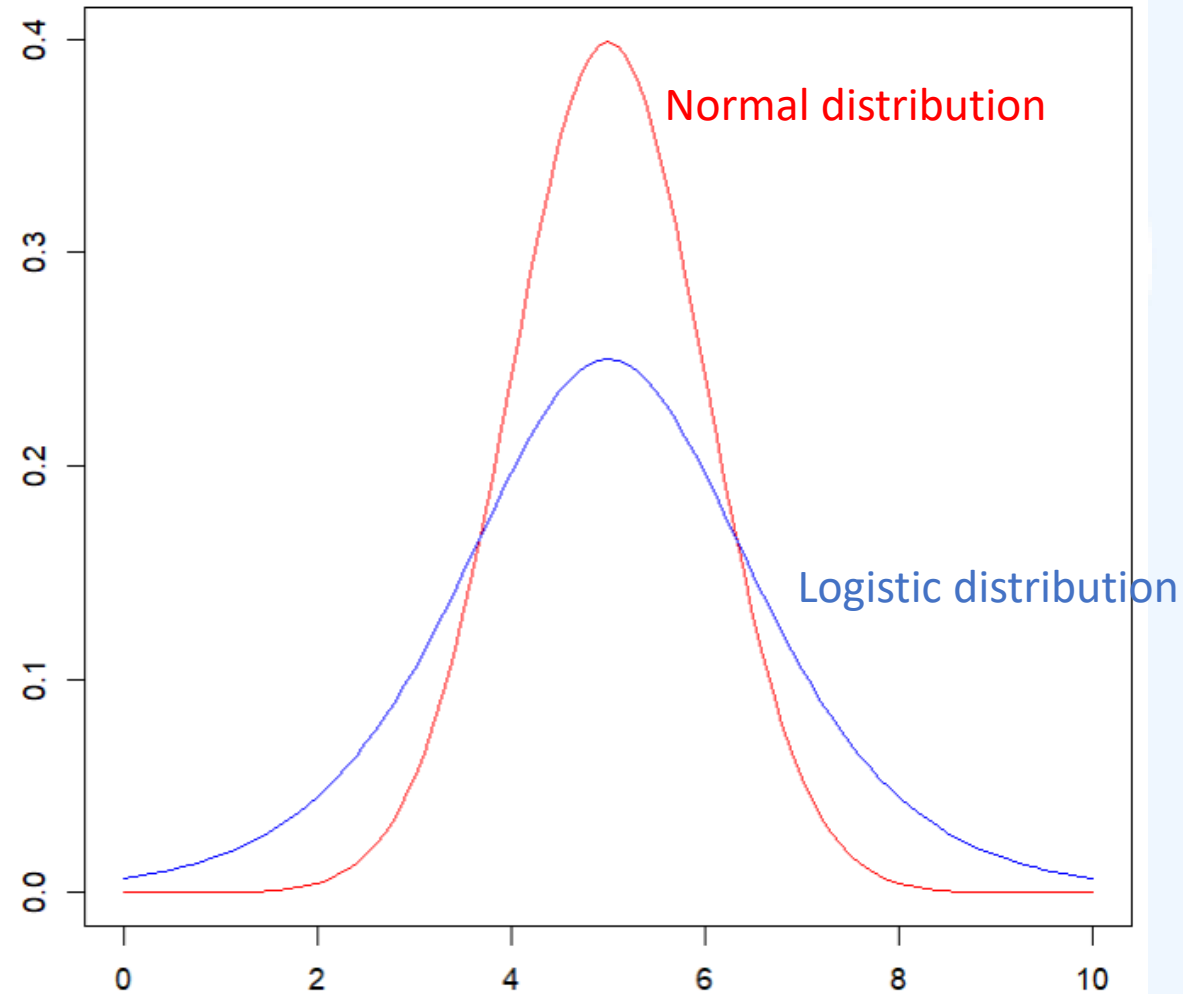
Mean and variance are two parameters that are not linked.

```
#Plot a Normal distribution (not using ggplot because it needs a dataframe)
xvalues=seq(0,10,length=101)
yvalues=dnorm(xvalues,mean=5,sd=1)
plot(xvalues,yvalues,type="l",main="Normal Distribution")
#Try changing the mean and sd values to see what happens
```

# Continuous distributions

## Logistic Distribution

Similar to a Normal distribution but with fatter tails (more kurtosis)



# Continuous distributions

## Gamma distribution

Model data that is always positive and skewed.

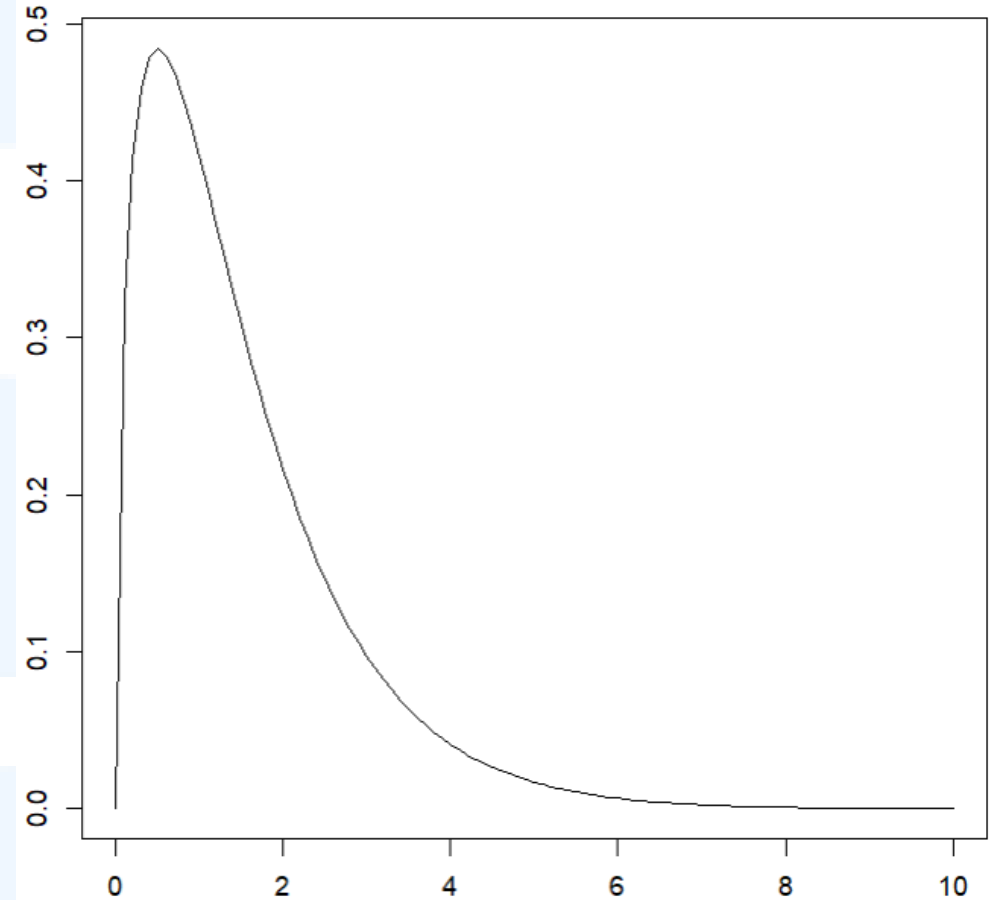
- Continuous counterpart to the negative binomial.

Good to model datasets where variance is much greater than the mean.

E.g. distribution of waiting times until a certain number of events take place.

```
#Plot a Gamma distribution
xvalues=seq(0,10,length=101)
yvalues=dgamma(xvalues,shape=0.5)
plot(xvalues,yvalues,type="l",main="Gamma
Distribution")

#Try changing the shape value to see what
happens
```



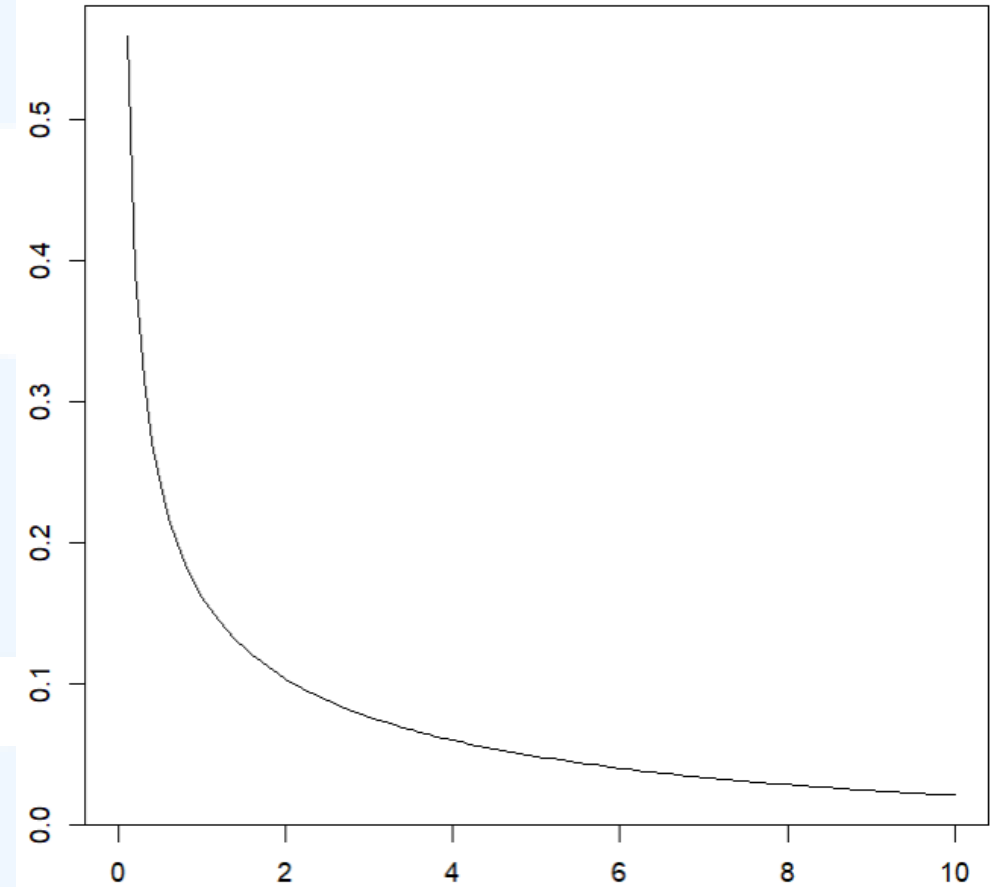
# Continuous distributions

## Exponential distribution

A special case of the Gamma.

Describes the distribution of waiting times for an event to happen given a constant probability per unit time of happening.

Useful to model inter-event times or lifetimes or to describe a distribution with highest probability close to zero.



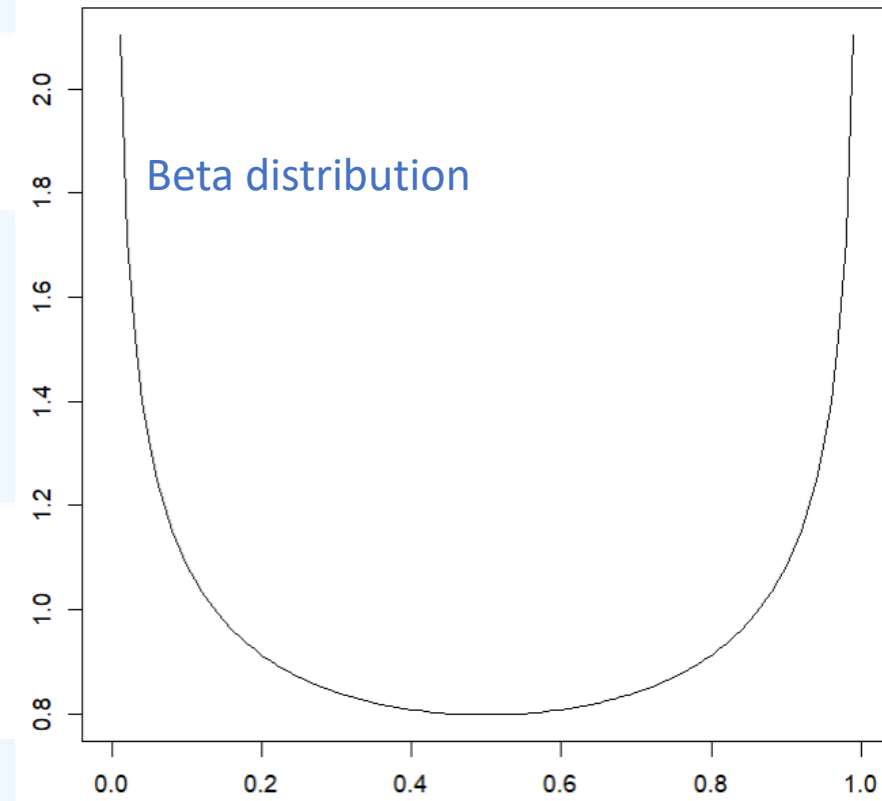
# Continuous distributions

## Beta distribution

Model data with a limited range (e.g. 0 to 1).

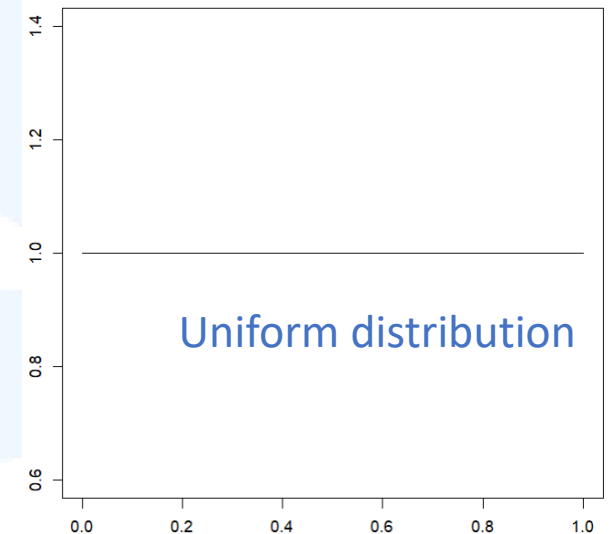
Good to model probabilities or proportions and data with peaks at both ends.

```
#Plot a Beta distribution  
xvalues=seq(0,1,length=101)  
yvalues=dbeta(xvalues,0.7,0.7)  
plot(xvalues,yvalues,type="l",main="Beta  
Distribution")
```



## Uniform:

Model data with a limited range (e.g. 0 to 1) where all outcomes are equally probable.





# Discrete distributions

## Binomial distribution

Counts the number of successes (or failures) in a given number of trials with **two possible outcomes**, each with a fixed probability

- Useful to model proportions
- Examples: flipping a coin, proportion of infected people in a population, extinction status.



If there is only one trial then it is called a **Bernoulli** distribution.

# Discrete distributions

## Poisson distribution

Distribution of counts (e.g. number of individuals, arrivals, events) where each event is independent of others. It does not have an upper limit (if you expect a ceiling in your counts you would choose the binomial).

The variance is equal to the mean and it **only applies to count data**.

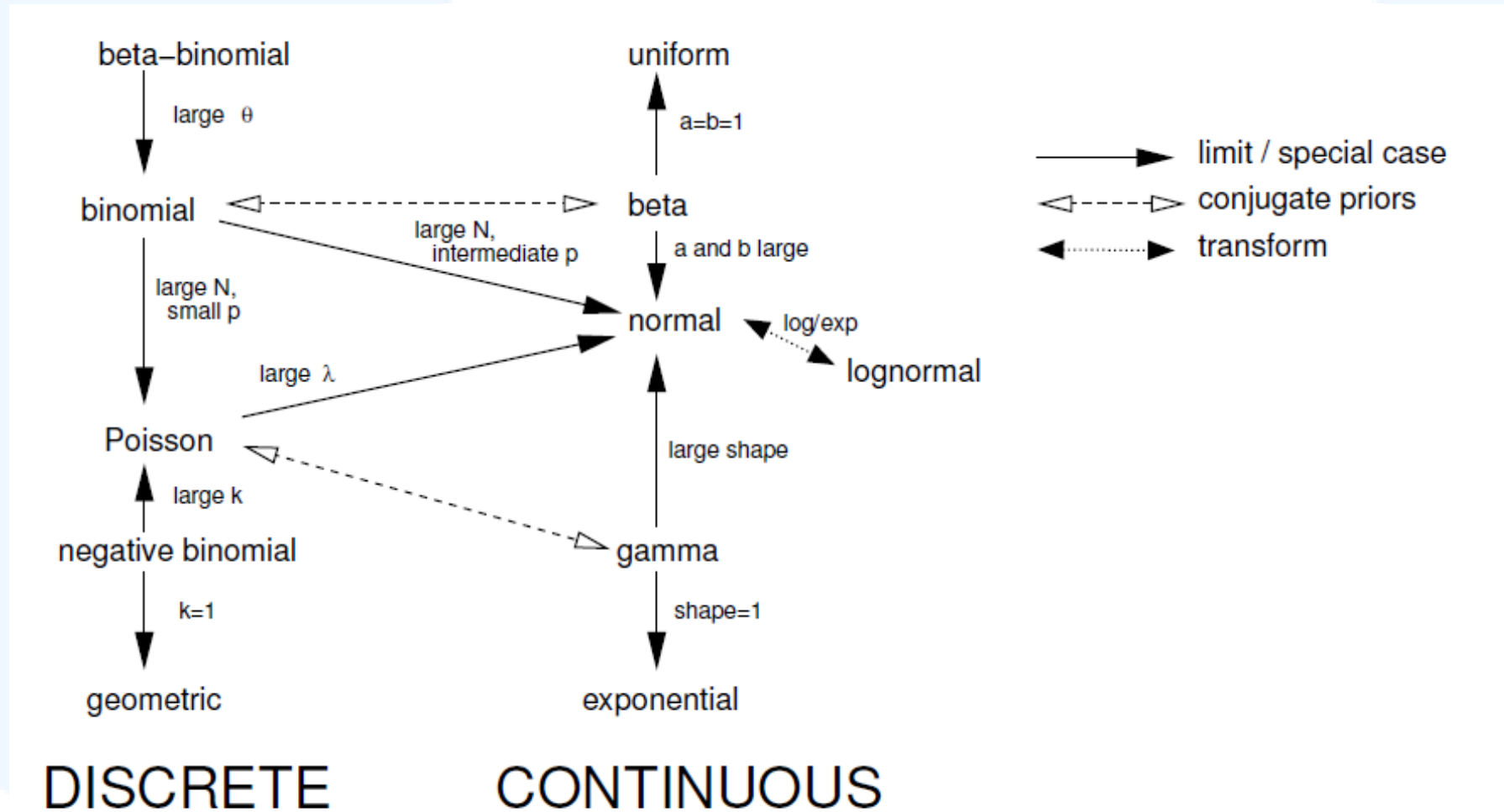
## Negative binomial distribution

Counts the number of failures before a predetermined number of successes takes place. The variance can be greater than the mean (unlike the Poisson) which is good for overdispersed data (more on this later).

Can be used to model birth-death processes.

If we are interested in obtaining one single success, it is called a **Geometric** distribution.

# The various distributions are mathematically related



Bolker B. (2007). Ecological Models and Data in R.

Full list of distributions in R:

[https://en.wikibooks.org/wiki/R\\_Programming/Probability\\_Distributions](https://en.wikibooks.org/wiki/R_Programming/Probability_Distributions)

# R functions with distributions – example using Normal distribution (`dnorm`)

Specify a value and get the density of the distribution at that value:

```
dnorm(x=5,mean=5,sd=1) #0.399: relative likelihood of a value being 5
```

Specify a value and get the cumulative density below it:

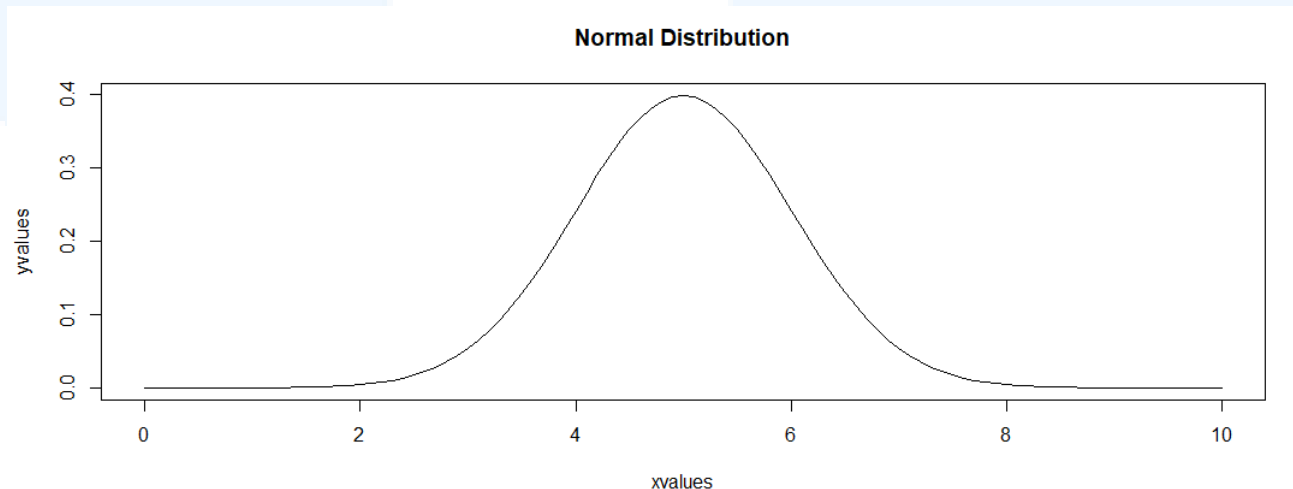
```
pnorm(q=4.75,mean=5,sd=1) #40% of the sample would be less than 4.75
```

Specify a cumulative density, get the value (opposite of `pnorm`):

```
qnorm(p=0.4,mean=5,sd=1) #4.75 is the value that 40% of observations would be less than
```

Generate a normally distributed sample with the stated parameters:

```
rnorm(n=100,mean=5,sd=1) #100 values, mean = 5, sd = 1
```



## Other distributions

Logistic: `dlogis`

Uniform: `dunif`

Gamma: `dgamma`

Exponential: `dexp`

Beta: `dbeta`



# Part 2: Basic Tests

## Experimental design

# Experimental design: When to use basic tests?

Good for:

- Controlled experiments where you can guarantee that **only one explanatory variable is changed**
- No need for complicated analyses when a simple test will do!

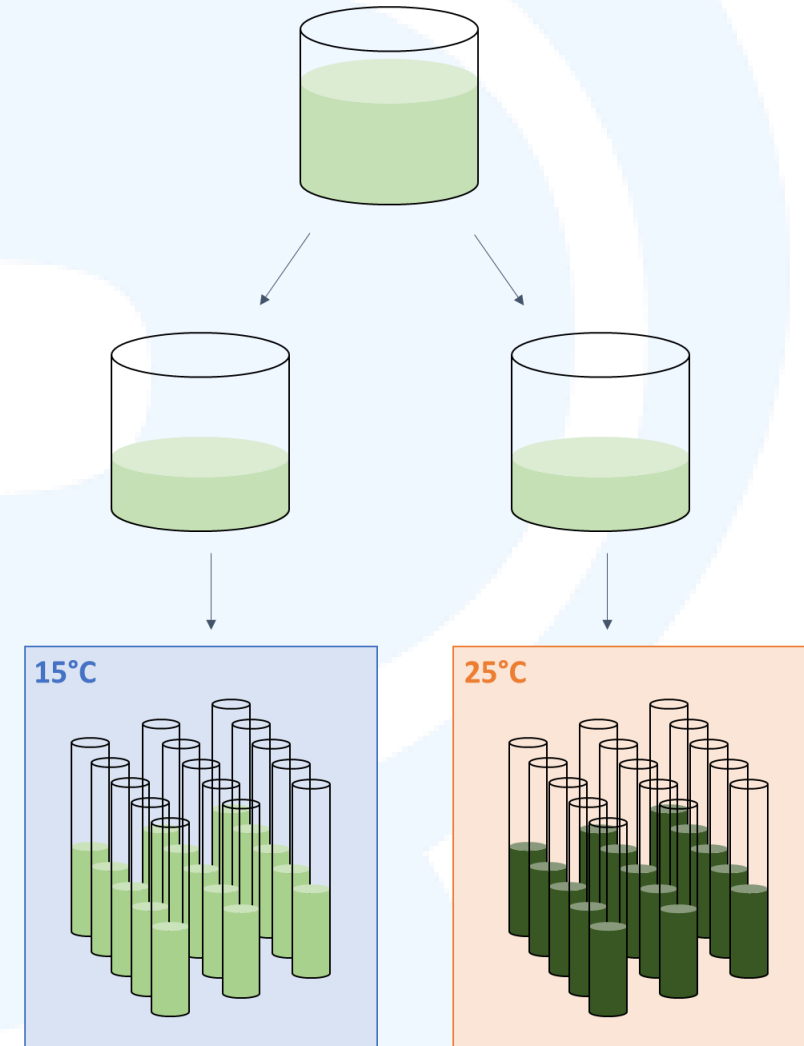
Example of a controlled study:

I want to know how temperature affects phytoplankton growth.

I split them into two groups of test tubes: (Group A) I grow at 15°C; (Group B) I grow at 25°C. I keep everything else the same (e.g. growth time, culture medium, light level, etc.).

I measure the turbidity (more turbid = more growth) of the cultures in the tubes.

I run a simple test and it tells me that the cultures grown at 25°C are more turbid (more growth).



# Experimental design: When NOT to use basic tests?

Not good for:

- Experiments with multiple explanatory variables
- Observational studies where conditions cannot be controlled – many “confounders”!

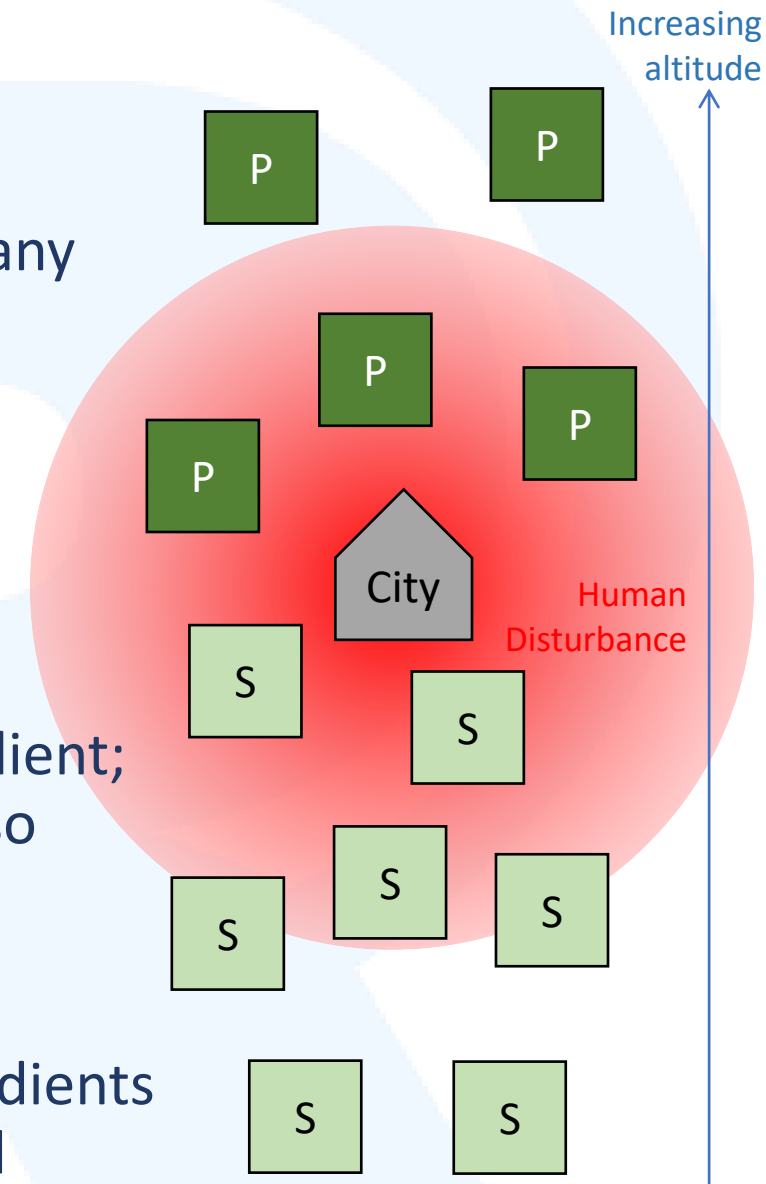
Example of an observational study:

I measure the abundance of a species in Primary (P) and Secondary forest (S) plots. I want to know if P supports more individuals.

But there's (i) an altitude gradient which means a temperature gradient; and (ii) a city with a disturbance gradient. These 2 variables may also affect abundance.

I run a test and find that the species is more abundant in S plots.

Is abundance explained by forest type (S vs. P) or the other two gradients or all of them? The simple test cannot account for temperature and disturbance and is therefore not appropriate.



# Designing a controlled experiment

- 1) Create many replicates (to reduce bias) and keep replicates independent
  - How many? Do a **power analysis** (end of this lecture)
  
- 2) Between your treatment and your control, change one and only one variable
  - What you change is your research question
  - If you are interested in 2 variables, then you will need more complicated analyses (future lectures)
  
- 3) Keep all other variables constant
  - Previous studies/experience will give you information on what is important to control
  - If you cannot keep other variables constant, you will need more complicated analyses (future lectures)



## Important assumptions about your data!

### 1) Datapoints are **I**ndependent and **I**dentically **D**istributed

- Independent: individual specimens are not related

Examples of non-independence: same specimen at different times, related specimen

- Identically Distributed: groups to be compared have the same probability distribution and **equal variance**

### 2) Distribution of datapoints...

- Follow a **normal distribution**: use parametric tests
- Is not normally distributed: use non-parametric tests

Main difference: parametric tests use exact values, non-parametric tests use ranks (more robust but less powerful)



# Test assumptions

## Testing for equal variance

To test whether the variance of two groups of (normally distributed) datapoints are equal.

Fisher's F test,  $H_0$ : variances are equal,  $H_1$ : variances are not equal.

## #Create 3 datasets

```
d1=rnorm(100,mean=0,sd=1)
```

```
d2=rnorm(100,mean=1,sd=1)
```

```
d3=rnorm(100,mean=0,sd=2)
```

```
#Test their variances
```

```
var.test(d1, d2)
```

```
var.test(d1, d3)
```

```
var.test(d2, d3)
```

```
> var.test(d1,d2)

      F test to compare two variances

data:  d1 and d2
F = 1.1467, num df = 99, denom df = 99,
p-value = 0.4971
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.7715568 1.7042848
sample estimates:
ratio of variances
 1.146714
```

```
> var.test(d1,d3)

      F test to compare two variances
data:  d1 and d3
F = 0.27813, num df = 99, den df = 99, p-value = 7.65e-10
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.1871372 0.4133656
sample estimates:
ratio of variances
 0.2781296
```

```
> var.test(d2,d3)
```

F test to compare two variances

data: d2 and d3  
F = 0.24254, num df = 99, denom df = 99, p-value = 1.328e-11  
alternative hypothesis: true ratio of variances is not equal to 1  
95 percent confidence interval:  
0.1631943 0.3604784  
sample estimates:  
ratio of variances  
0.2425449

What does this all mean

# Hypothesis testing using p-values

The tests we run look at your data and **tell you the probability that a certain hypothesis (called the “null hypothesis”) is true given your data: the p-value.**

$H_0$ : null hypothesis.

- This is your starting “default” assumption
- If your evidence is not strong enough ( $p\text{-value} > 0.05$ ), you cannot reject this belief
- E.g. “The data suggest that there is no significant difference between X and Y.”

$H_1$ : alternative hypothesis.

- This is what you’re trying to see whether your data support
- If your evidence is strong enough ( $p\text{-value} \leq 0.05$ ), you can claim that this is supported
- E.g. “The data suggest that there is a significant difference between X and Y.”

# Hypothesis testing using p-values

Every test has its own null and alternative hypothesis.

Test	Null hypothesis	Alternative hypothesis
Fisher's F-test	The variances of the two datasets are not different	The variances of the two datasets are different
Shapiro-Wilk test	The data are normally distributed	The data are not normally distributed
One sample t-test	The mean of the data is not different from the specified value	The mean of the data is different from the specified value
Pearson's chi-squared test	The two categorical variables are independent	The two categorical variables are not independent
Two-proportions test	The two proportions are not different from each other	The two proportions are different from each other

The output here very thoughtfully tells you what the alternative hypothesis is (not every test does)

```
> var.test(d1,d2)

      F test to compare two variances

data:  d1 and d2
F = 1.1467, num df = 99, denom df = 99,
p-value = 0.4971
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.7715568 1.7042848
sample estimates:
ratio of variances
      1.146714
```

This is our p-value: 0.4971, which means we should  our null hypothesis, i.e. the evidence suggests that the variances of d1 and d2  significantly different

# But why 0.05?

Historically decided on by Ronald Fisher (1925).

Fisher, Ronald (1925). Statistical Methods for Research Workers. Edinburgh, Scotland: Oliver & Boyd. ISBN 978-0-05-002170-5.

What does a p-value of 0.05 mean?

If X and Y were the same, there is only a 5% chance of us observing the data that we collected (i.e. a 5% chance that the null hypothesis is true).

But it's not totally arbitrary; there's a reason based on human intuition...



“Why learn statistics if you can’t use it to win some money?” Chan I.Z.W. (2021)

# Testing for equal variance (reprise)

To test whether the variance of two groups of (normally distributed) datapoints are equal.

Fisher's F test,  $H_0$ : variances are equal,  $H_1$ : variances are not equal.

```
#Create 3 datasets
```

```
d1=rnorm(100,mean=0,sd=1)
```

```
d2=rnorm(100,mean=1,sd=1)
```

```
d3=rnorm(100,mean=0,sd=2)
```

```
#Test their variances
```

```
var.test(d1,d2)
```

```
var.test(d1,d3)
```

```
var.test(d2,d3)
```

```
#Can also test directly from a dataframe
```

```
var.test(yvar~xvar,data=dataset)
```

Variable to test variance for

Grouping variable

```
> var.test(d1,d2)
```

F test to compare two variances

data: d1 and d2  
F = 1.1467, num df = 99, denom df = 99,  
p-value = 0.4971  
alternative hypothesis: true ratio of variances is not equal to 1  
95 percent confidence interval:  
0.7715568 1.7042848  
sample estimates:  
ratio of variances  
1.146714

```
> var.test(d1,d3)
```

F test to compare two variances

data: d1 and d3  
F = 0.27813, num df = 99, denom df = 99, p-value = 7.65e-10  
alternative hypothesis: true ratio of variances is not equal to 1  
95 percent confidence interval:  
0.1871372 0.4133656  
sample estimates:  
ratio of variances  
0.2781296

```
> var.test(d2,d3)
```

F test to compare two variances

data: d2 and d3  
F = 0.24254, num df = 99, denom df = 99, p-value = 1.328e-11  
alternative hypothesis: true ratio of variances is not equal to 1  
95 percent confidence interval:  
0.1631943 0.3604784  
sample estimates:  
ratio of variances  
0.2425449

d1 and d2 are similar

d1 and d3 are different

d2 and d3 are different

# Testing for Normality

To test whether a group of datapoints are normally distributed.

## Option 1: Graphical test

Plot a Q-Q plot, i.e. theoretical quantiles from a normal distribution against the actual quantiles in your sample.

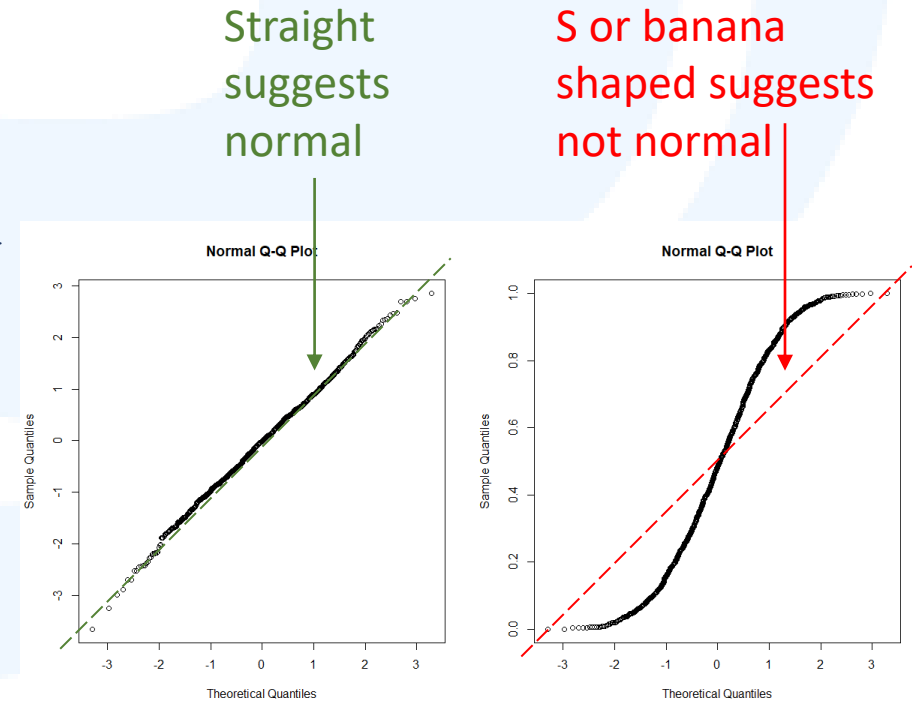
```
#Create a vector of normally distributed data  
vecNorm=rnorm(n=1000, mean = 0, sd = 1)
```

```
#Create a vector of non-normally distributed data  
vecUnif=runif(n=1000, min=0, max=1)
```

```
#Basic R code to plot two plots side by side
```

```
par(mfrow=c(1,2)) ← This function tells R to plot the next 2  
qqnorm(vecNorm)    plots side by side. Note: you must  
                    click the (x) to exit this mode
```

```
qqnorm(vecUnif) ← This function plots a Q-Q plot
```





# Testing for Normality

There's also the Kolmogorov-Smirnov test but I prefer Shapiro-Wilk. See: <https://stats.stackexchange.com/questions/362/what-is-the-difference-between-the-shapiro-wilk-test-of-normality-and-the-kolmog>.

**Option 2: Shapiro-Wilk test**,  $H_0$ : normally distributed,  $H_1$ : not normally distributed ( $P < 0.05$ ).

```
#Shapiro-Wilk test
```

```
shapiro.test(vecNorm) #P = 0.16, cannot reject  $H_0$  (i.e. normal)
```

```
shapiro.test(vecUnif) #P < 0.001, confidently reject  $H_0$  (i.e. not normal)
```

```
> shapiro.test(vecNorm)
```

Shapiro-Wilk normality test

```
data:  vecNorm  
W = 0.99764, p-value = 0.1629
```

```
> shapiro.test(vecUnif)
```

Shapiro-Wilk normality test

```
data:  vecUnif  
W = 0.95169, p-value < 2.2e-16
```

But this test is sometimes over-sensitive:

- Be careful and use both methods

```
> shapiro.test(vecNorm)
```

Shapiro-Wilk normality test

```
data:  vecNorm  
W = 0.99709, p-value = 0.06615
```

# What to do if you have non-normally distributed data?

If you have a dataset that is not normally distributed, you can either:

a) **Transform the data** (e.g. sqrt, log or take the reciprocal of all the values) and try again (very hit or miss);

```
#Log transforming a vector  
exDat=c(53,46,77,82)  
exDat2=log(exDat)
```

OR

b) **Use a non-parametric test** (does not assume that the data is normally distributed). We will be using this option by default.

## Choosing a transformation

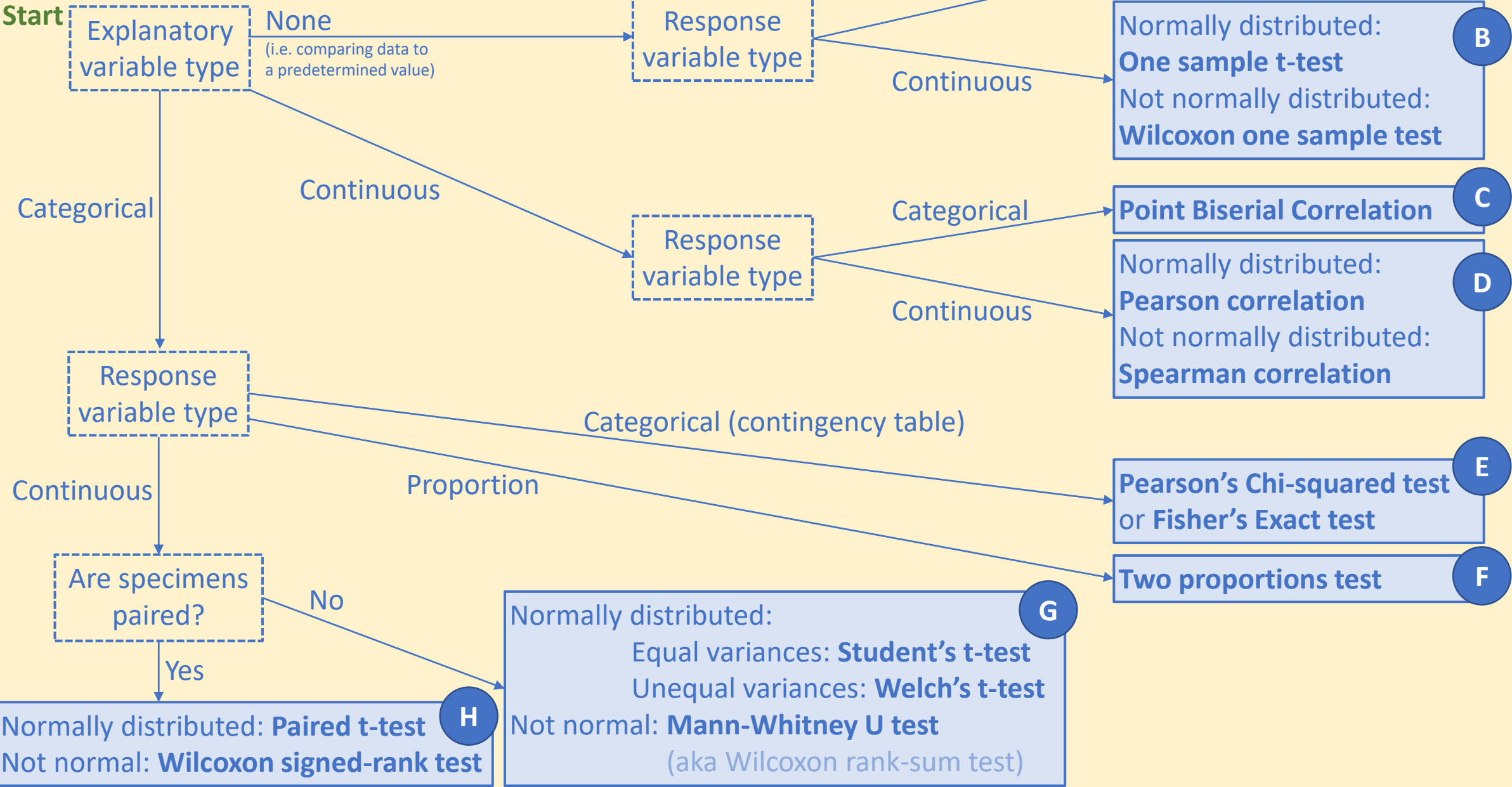
- 1) **Values in your dataset:** Sqrt for values  $\geq 0$ .  
Log and Reciprocal for values  $> 0$ .
- 2) **Normalisation “power”**, i.e. strength of effect to reduce skew: Sqrt < Log < Reciprocal
- 3) Sqrt usually works better for **count data**.

Bottom line: easiest to just try and plot/test.



# Basic statistical tests

# Basic tests – Analysis decision tree



## Chi-squared test on a 1-Way Contingency table

To test whether a categorical variable has an effect on the data

$H_0$ : no evidence for an effect,  $H_1$ : there is an effect.

Example: after an animal is exposed to a pollutant, its clutch has 31 males and 49 females. You want to test whether this is different from random (i.e. 50% chance of males or females).

```
#Create the contingency table  
count=c(31,49)  
#Perform the Chi-squared test  
chisq.test(count,p=c(0.5,0.5)) #p=0.044
```

	Count
Males	31
Females	49

This is the null hypothesis you're testing against: in this case random so 0.5 and 0.5. The values should sum to 1.

Note: Can also test for another probability (e.g. 25% chance of Males, 75% of Females) or for three different outcomes (“c(1/3,1/3,1/3)” instead of “c(0.5,0.5)”).

# One sample t-test / Wilcoxon one sample test

To test if the mean of one group of data is equal to a predetermined value

$H_0$ : mean is not different from the value,  $H_1$ : mean is different.

First test whether data are normally distributed:

```
shapiro.test(mtcars$mpg) #p=0.12
```

```
shapiro.test(mtcars$hp) #p=0.049
```

If normally distributed, use a one sample t-test:

#Test whether the mean is different from 20

```
t.test(mtcars$mpg, mu=20) #p=0.93
```

If not normally distributed, use a Wilcoxon one sample test:

#Test whether the mean is different from 100

```
wilcox.test(mtcars$hp, mu=100) #p=0.001
```

```
wilcox.test(mtcars$hp, mu=100, Answer) #p=0.001
```

```
> wilcox.test(mtcars$hp, mu=100)
```

Wilcoxon signed rank test with continuity correction

data: mtcars\$hp

V = 440, p-value = 0.001027

alternative hypothesis: true location is not equal to 100

Warning message:

In wilcox.test.default(mtcars\$hp, mu = 100) :  
cannot compute exact p-value with ties

Warning message. How  
to solve? Try Googling...

# Point Biserial Correlation

To test whether an explanatory continuous variable and a categorical response variable are independent (no effect on each other) or correlated (vary together)

$H_0$ : independent,  $H_1$ : correlated.

#Create the continuous x and categorical y variables

```
x=c(1,11,2,14,3,9,1,12,3,10)
```

```
y=c("a","b","a","b","a","b","a","b","a","b")
```

#Testing Assumption 1

```
shapiro.test(x) #p=0.084
```

```
boxplot(x) #look for outliers: none
```

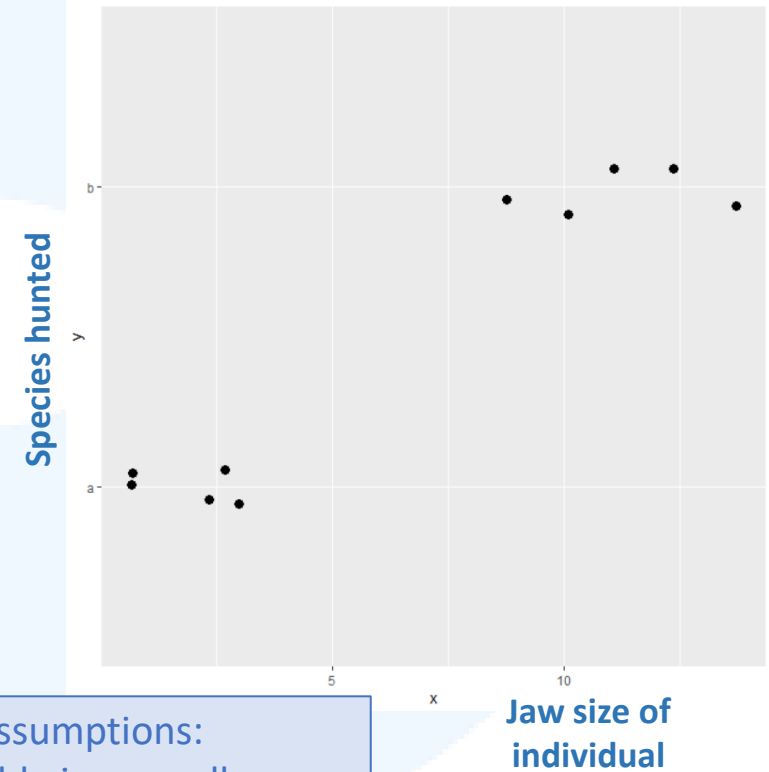
#Testing Assumption 3

#Splitting "a" and "b" into 2 vectors

```
xA=x[which(y=="a")]
```

```
xB=x[which(y=="b")]
```

```
var.test(xA,xB) #p=0.23
```

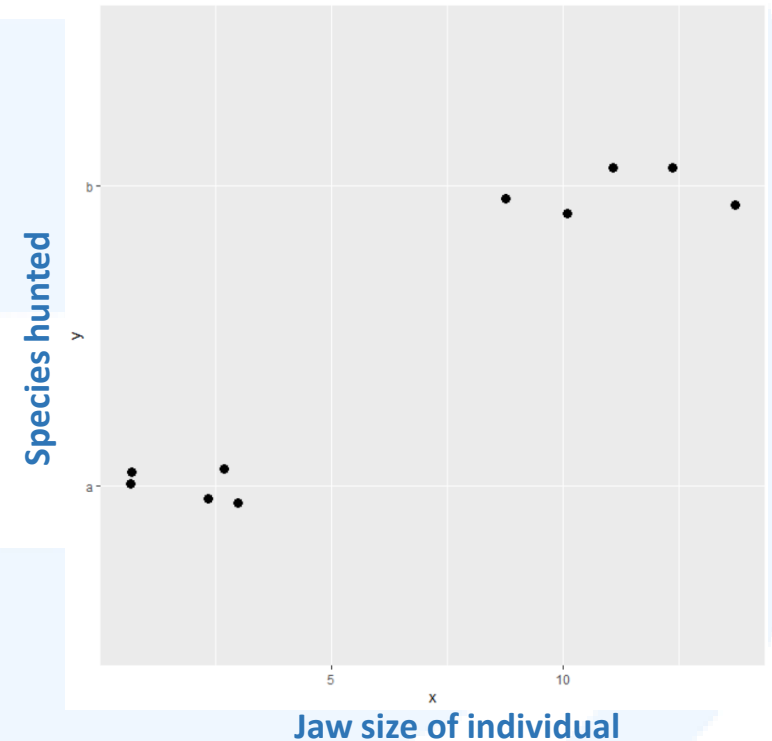


Point Biserial Test assumptions:

- 1) Continuous variable is normally distributed and has no outliers
- 2) Categorical variable only has 2 levels
- 3) Equal variance in the two groups of the categorical variable
- 4) The relationship is linear

# Point Biserial Correlation

```
#Performing Point Biserial Correlation
cor.test(x,y) #Error: y must be numeric
#Convert y to numeric with 2 values first
yNum=y
yNum[which(yNum=="a")] =1
yNum[which(yNum=="b")] =2
yNum=as.numeric(yNum)
#Calculating Point Biserial Correlation
with new yNum
cor.test(x,yNum) #p<0.001
```



The result makes sense, it looks like individuals with larger jaw size (x-axis) tend to prefer preying on species b (upper value on y-axis)?



# Pearson and Spearman Correlation

To test whether two continuous variables are independent or correlated.

$H_0$ : Independent,  $H_1$ : correlated.

If both normally distributed, use Pearson Correlation:

```
shapiro.test(mtcars$mpg) #p=0.1229  
shapiro.test(mtcars$drat) #p=0.1101  
cor.test(mtcars$drat,mtcars$mpg) #p<0.001
```

If at least one non-normal, use Spearman Correlation:

```
shapiro.test(mtcars$hp) #p=0.049  
cor.test(mtcars$drat,mtcars$hp,method="spearman") #p=0.002
```

# Chi-squared test or Fisher's Exact test on a 2-way Contingency table

To test whether two categorical variables are independent (do not affect each other) or associated (affect each other)

$H_0$ : independent,  $H_1$ : associated.

Example: in this contingency table we have counts of all the times that a combination of two categorical variables happened, i.e. (i) Colour A or B in crabs and (ii) whether a Fight occurred. Question: Are Colour and Fight associated? We use a Chi-squared test to compare the Observed counts to the Expected counts if the variables were independent (i.e. equal proportions of Fight/No fight in Colour A and B).

	Fight	No fight
Colour A	53	426
Colour B	231	298

The values in our data are the **Observed values**. In most cases, **Expected values** are the values if all categories had an equal number.

Using our example, the expected value for each category would be:  
 $(53+231+426+298)/4 = 252$  (this is  $> 5$ )

## From a contingency table of counts

```
count=matrix(c(53,231,426,298),nrow=2)
```

```
#Perform the Chi-squared test
```

```
chisq.test(count) #p<0.001
```

```
#If any of the Expected values are < 5, perform Fisher's Exact test instead
```

```
fisher.test(count) #p<0.001
```

# Chi-squared test or Fisher's Exact test on a 2-way Contingency table

## Directly from a dataframe

You can also run the test directly from categorical variables within a dataframe (i.e. no need to create a matrix), the `chisq.test()` command is smart enough to recognise what you're keying in

#Create a dataframe with two categorical variables

```
cDat=data.frame(  
  type=c(1,2,1,1,2,2,2,1,1,1,2),  
  cat=c(T,F,T,T,F,T,F,F,F,T,T)  
)  
#Perform the Chi-squared test  
chisq.test(x=cDat$type,y=cDat$cat) #p=0.78
```

	type	cat
1	1	TRUE
2	2	FALSE
3	1	TRUE
4	1	TRUE
5	2	FALSE
6	2	TRUE
7	2	FALSE
8	1	FALSE
9	1	FALSE
10	1	TRUE
11	2	TRUE

## Two proportions test

To test whether two proportions are significantly different from each other

$H_0$ : no evidence of a difference,  $H_1$ : evidence for a significant difference

Example: in a herbicide experiment...

- Control plot (without herbicide): 10 out of 876 seedlings die;
- Herbicide-treated plot: 7 seedlings out of 233 die.

Is this by chance alone? Is there any evidence the herbicide was effective?

```
#Run the two proportions test
```

```
prop.test(c(10,7),c(876,233)) #p=0.079
```

```
#Note: the command takes 2 vectors, the first containing number of  
"successes", the second taking the total number of trials; in order.
```

## Student's t-test / Welch's t-test

To test whether the means of two groups with **normally distributed data** are significantly different,  $H_0$ : no evidence for a difference,  $H_1$ : the means are significantly different.

Example: dataset on <age> and pollutant <levels> (response variable) in two <group>s of animals (explanatory variable)

Test whether the response variable is normally distributed:

```
d1=read.csv("ageData.csv")
```

```
shapiro.test(d1$age) #p=0.1229: normal
```

```
shapiro.test(d1$levels) #p=0.049: non-normal(?)
```

```
> str(d1)
'data.frame':  32 obs. of  3 variables:
 $ age    : num  21 21 22.8 21.4 18.7 18.1 14.3 2
 $ levels: num  110 110 93 110 175 105 245 62 95
 $ group  : num   0 0 1 1 0 1 0 1 1 1 ...
```

If normally distributed, test for equal variances:

```
var.test(age~group,data=d1) #p=0.1997
```

# Student's t-test / Welch's t-test

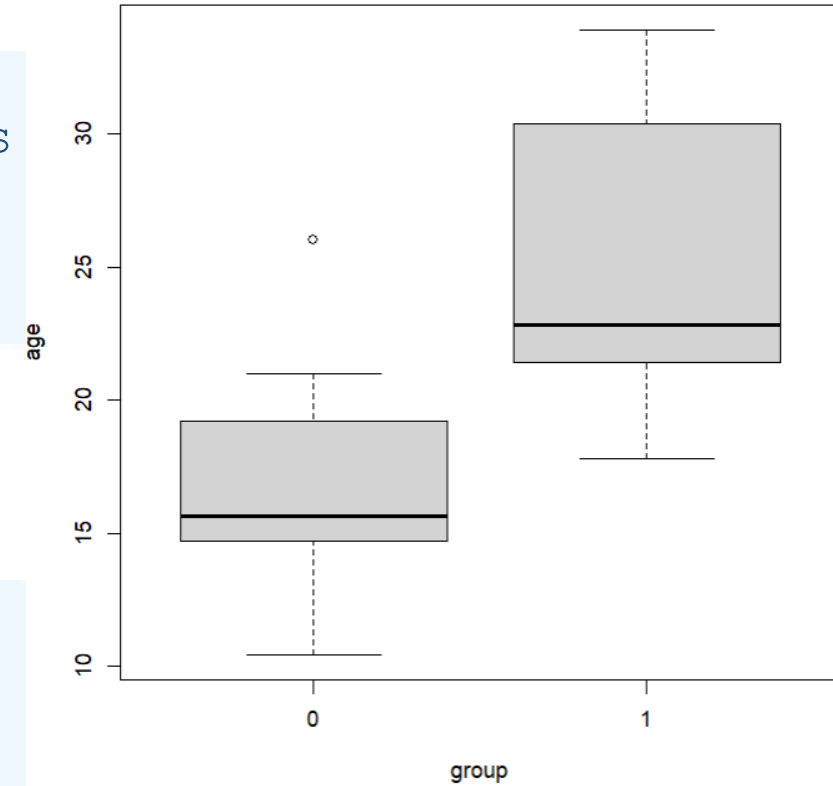
If variances are equal, do Student's t-test:

```
t.test(age~group,data=d1) #two-tailed: are the means
different?; p<0.001
```

```
t.test(age~group,data=d1,alternative="less")
#one-tailed, is "0" < "1"? (alphabetical); p<0.001
```

```
t.test(age~group,data=d1,alternative="greater")
#one-tailed, is "0" > "1"?; p=1
```

```
boxplot(age~group,data=d1)
#see if the results make sense
```



If variances are NOT equal, do Welch's t-test:

```
t.test(age~group,data=d1,var.equal=F)
#p<0.001 (can also run other alternative hypotheses)
```

# Mann-Whitney U test

If the response data is **non-normally distributed**:

```
shapiro.test(d1$levels) #p=0.049
```

Use the Mann-Whitney U test:

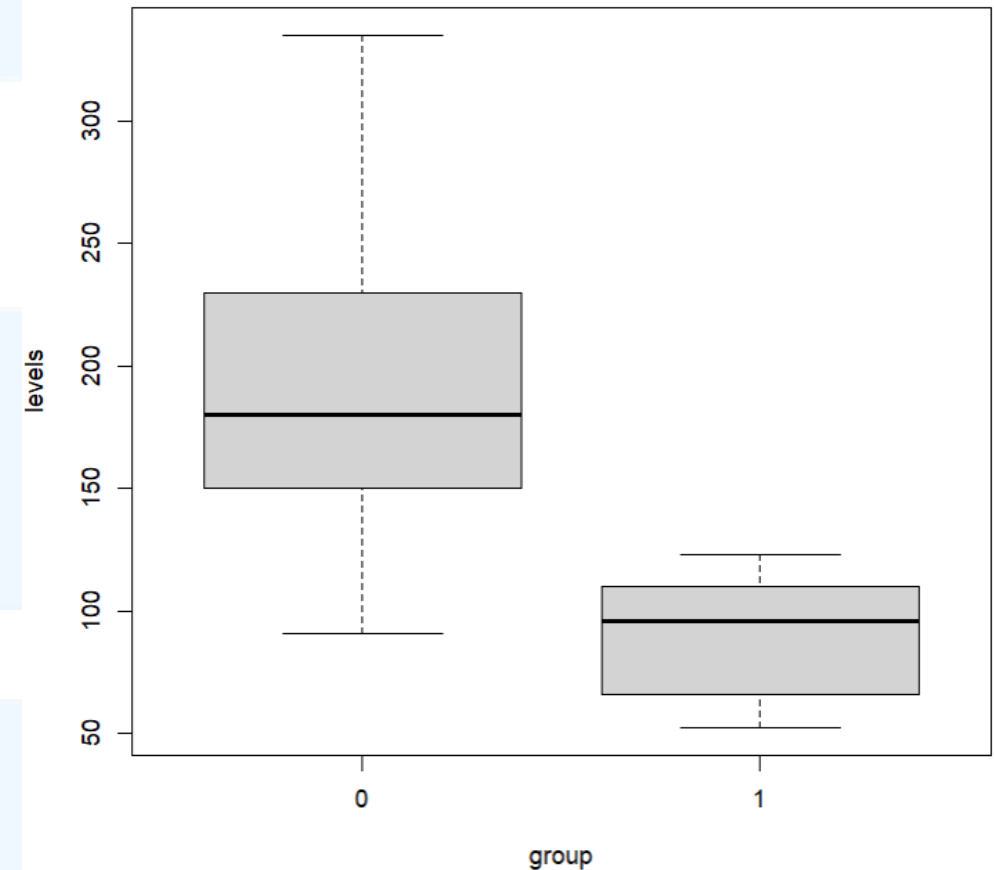
(same as before,  $H_0$ : no evidence for a difference,  $H_1$ : means are significantly different)

```
wilcox.test(levels~group, data=d1) #p<0.001
```

```
wilcox.test(levels~group, data=d1, alternative="less") #p=1
```

```
wilcox.test(levels~group, data=d1, alternative="greater") #p<0.001
```

```
boxplot(levels~group, data=d1) #see if the results make sense
```



## Student's t-test / Welch's t-test / Mann-Whitney U test **with separate vectors**

In the previous slides, we used a grouping variable in a dataframe `<d1$group>` to tell R what our 2 groups are. If your data are in 2 separate vectors (one vector for each group), we can also compare them directly.

Subset the `<level>` variable into 2 vectors based on `<group>`:

```
levels0=d1$levels[d1$group==0]
```

```
levels1=d1$levels[d1$group==1]
```

Run a Mann-Whitney U test:

```
wilcox.test(levels0, levels1)
```

Subset the `<age>` variable into 2 vectors based on `<group>`:

```
age0=d1$age[d1$group==0]
```

```
age1=d1$age[d1$group==1]
```

Run a t-test

```
t.test(age0, age1, var.equal=F) #two-tailed; can also run both one-tailed tests
```



## Paired t-test / Wilcoxon signed-rank test

To test whether the means of **two groups of related samples** are different

$H_0$ : no evidence of a difference,  $H_1$ : the means are different.

### Examples:

- Paired over time: comparing biodiversity in 10 sites at  $T=1$  vs.  $T=2$
- Paired by relationship: comparing weight of male vs. female offspring from 10 pairs of parents

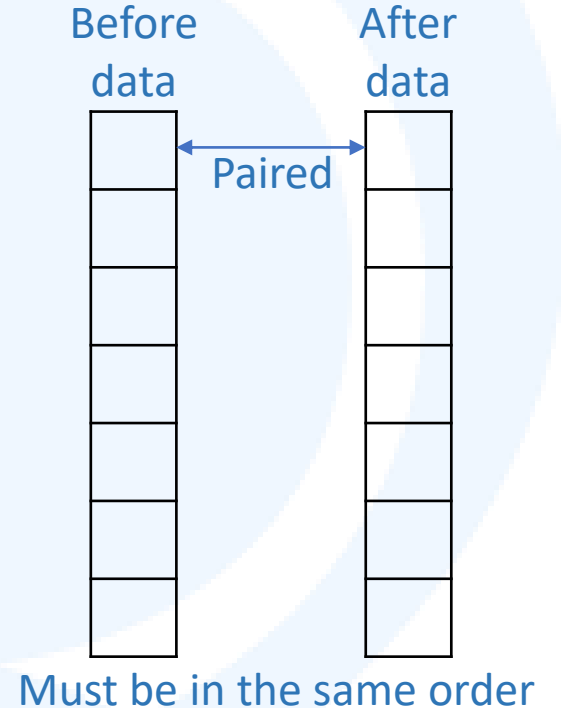
Assuming that our previous data was actually paired...

If normally distributed, use a Paired t-test:

```
t.test(age0, age1, paired=TRUE) #Can also add "var.equal=F" if needed
```

If not normally distributed, use a Wilcoxon signed-rank test:

```
wilcox.test(levels0, levels1, paired=TRUE)
```





# Other considerations

## Correcting for multiple comparisons

Be very careful if you perform multiple tests in your study!

Example:

- If you have 3 groups (A, B and C) and you want to compare them all, you may decide to perform 3 tests: A vs. B; A vs. C; and B vs. C.
- This increases the chance of getting a **significant result due to random chance**, especially if you make 10s or 100s of comparisons! (recall that  $p=0.05$  is a 1 in 20 chance)

There are a multiple ways to correct for this, I introduce 2...

- 1) **Bonferroni correction**: multiply your p-values (or divide your cut-off value, usually 0.05) by the number of comparisons conducted. The most strict but least powerful (i.e. most likely to incorrectly reject a significant result).
- 2) **Benjamini & Hochberg (BH) correction** (aka False Discovery Rate, FDR): calculates a critical value using the ranks of your p-values and rejects results based on that. A good balance between strictness and power.

## Correcting for multiple comparisons

Example: You did 20 comparisons and got these p-values

```
pVals=c(0.56,0.0001,0.032,0.045,0.12,0.44,0.22,0.013,0.54,0.72,0.11,0.35,0.003, 0.53,0.17,0.05,0.051,0.23,0.59,0.67)
```

Apply your chosen correction to the p-values:

```
#Correct using Bonferroni
```

```
pBf=p.adjust(p=pVals,method="bonferroni")
```

```
which(pBf<0.05) #2 only (more stringent)
```

```
#Correct using BH
```

```
pBH=p.adjust(p=pVals,method="BH")
```

```
which(pBH<0.05) #2 and 13 (less strict)
```

# Power analysis

Imagine you spend months doing an experiment and end up having no significant result. Is it because there is really no effect or your sample size was too small?

- You can't tell! Unless...

Power analysis is good to do before an experiment to estimate how big your sample size needs to be in order to detect that something has an effect and get a significant result.

For t-test: to detect an effect of 0.5 in data with SD=1 (power is usually set at 0.8 by default):

```
power.t.test(delta=0.5,sd=1,power=0.8,type="one.sample")  
power.t.test(delta=0.5,sd=1,power=0.8,type="two.sample")  
power.t.test(delta=0.5,sd=1,power=0.8,type="paired")
```

You need at least n=34

```
> power.t.test(delta=0.5,sd=1,power=0.8,type="one.sample")  
  
One-sample t test power calculation  
  
      n = 33.3672  
delta = 0.5  
      sd = 1  
sig.level = 0.05  
power = 0.8  
alternative = two.sided
```

# Power analysis

For Two proportions test: to detect a difference between 2 groups with actual success rates of 0.7 and 0.3 (power also 0.8 by default):

```
power.prop.test(p1=0.7,p2=0.3,power=0.8)
```

```
> power.prop.test(p1=0.7,p2=0.3,power=0.8)
```

Two-sample comparison of proportions p

```
      n = 23.31288
      p1 = 0.7
      p2 = 0.3
sig.level = 0.05
power = 0.8
alternative = two.sided
```

NOTE: n is number in \*each\* group

For Mann-Whitney U test: to detect a difference between 2 time-till-event datasets (exponential distribution):

```
install.packages("wmwpow")
```

```
library(wmwpow)
```

```
shiehpow(n=20,m=20,p=0.75,dist="exp")
```

#p is the effect size, proportion of the time that you would expect a random value from n to be smaller than a random value from m

#Manually change n and m to try to get Shieh Power > 0.8

```
> shiehpow(n=20,m=20,p=0.75
```

```
Distribution: exp
Sample sizes: 20 and 20
p: 0.75
WMW odds: 3
sides: Two-sided
alpha: 0.05
```

```
Shieh Power: 0.805
```

# Summary (Learning Objectives)

## Basic statistical concepts

- Variable types:

  - Properties: Continuous vs. Categorical vs. Discrete

  - Function: Explanatory vs. Response

- Describing data: Mean, Variance, df, SD, SE, Confidence Intervals

- Modelling data with Distributions: continuous and discrete

## Basic statistical tests

- Experimental design

- Test assumptions: **Independent and Identically Distributed**, normality, equal variances; testing for normality and equal variances

- Tests: based on number and type of explanatory and response variables

- Other considerations: Corrections for multiple comparisons, Power analysis