All of statistics
(in a few slides)

Random variables and their distributions

Let *X* be a ‘random variable’ (i.e. *X* represents some kind of random quantity). E.g. *X* could be the number you get if you toss a dice, or the number of heads if you toss a coin 10 times, or the height of a person sampled at random from the phone book etc.)

X can be discrete (e.g. the dice or the coin example) or continuous (e.g. the height example)

Important things you need to know about random variables: Distribution

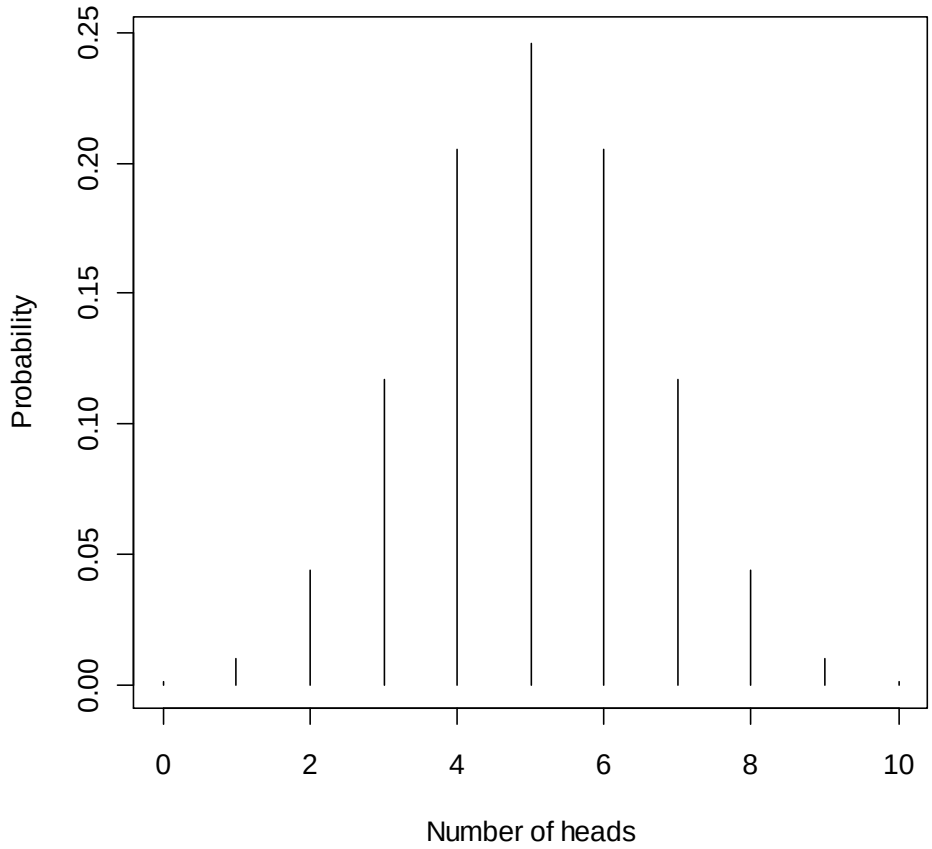
Distribution. For this we need to distinguish between discrete and continuous random variables

Discrete: the distribution tells you how likely each value is to come up (e.g. a fair dice – each outcome has probability 1/6)

Continuous: similar to a discrete distribution except that now you need to think about the probability the random variable lies within some interval (because theoretically, the probability that it takes on any exact value is always 0).

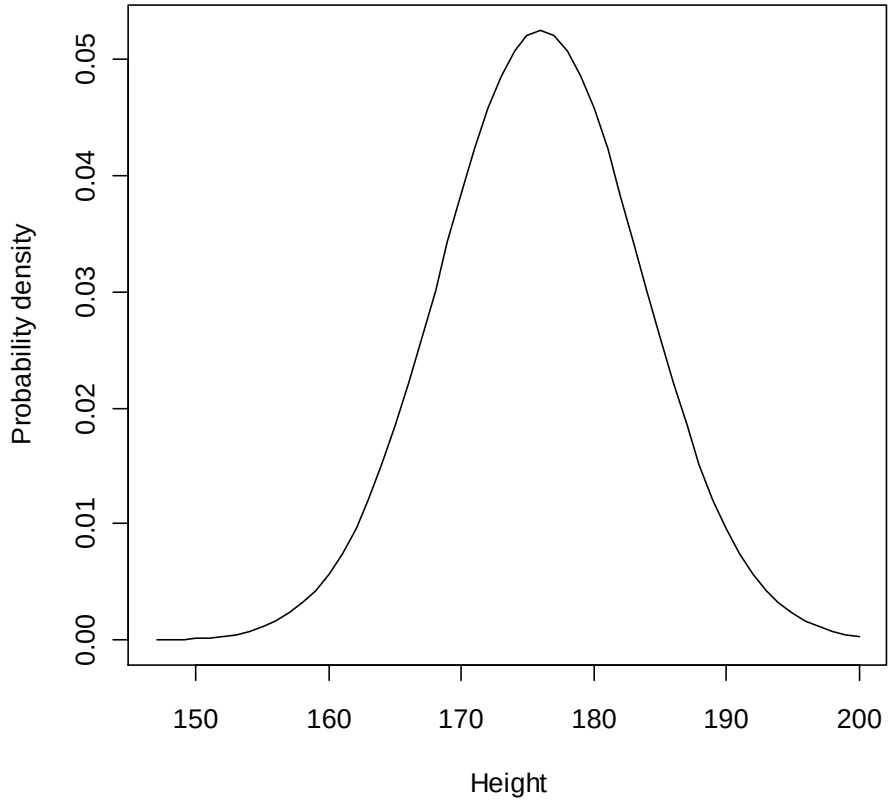
Example of a discrete random variable (binomial random variable)

The number of heads you get if you toss a coin 10 times



Example of a continuous random variable (normal random variable)

Describes many quantities in nature (e.g. male/female height, approximately)



R functions for distributions

In the previous slide, you need the area under a curve. This can be calculated by integration. Fortunately, you don't need to do this because R has functions for all of the standard random variables.

pnorm(1,3,1.5) – the probability that a normal RV with mean 3 and standard deviation 1.5 lies below 1 (i.e. the area under the curve of this normal RV to the left of 1).

rnorm(20,3,1.5) – generates 20 random instances of a normal RV with mean 3 and standard deviation 1.5

dbinom(5,20,0.5) – this is the probability of getting 5 heads when you toss a fair coin 20 times (why?)

dnorm(1.76,1.74,0.5) – this is not the probability that a randomly sampled man from a population with normally distributed heights (mean 1.74m, sd 0.5m) has height 1.76meters. Why not?

Important things you need to know about random variables: Expected value

Discrete case: The ‘Expected Value’ of a random variable is a kind of average value. You calculate it by summing up all the values the distribution can take on, weighted by their probabilities of coming up (so it’s a kind of weighted average of the random variable – with each value weighted by it’s probability).

Continuous case: The same basic idea applies, but now to calculate the weighted average you need to use integration.

Important things you need to know about random variables: Variance

The variance tells you how much uncertainty there is in the random variable. It's designed to give an idea of how far, on average, the values of the random variable are from the expected value.

Technically: the variance is the expected value of $(X - E(X))^2$

The standard deviation (σ) is just the square root of the variance.

Central Limit Theorem

Consider a random variable, X, with mean, μ, and variance, σ^2. You take a 'random sample' of size n from X, and calculate the mean of the sample. The sample mean is then an (approximately) normal random variable with mean, μ, and variance (approximately) σ^2/n (less approximate for larger n!)

Homework assignment: demonstrate the CLT using R

Elements of a statistical hypothesis test

Typically…

Two hypotheses, conventionally called H_0 and H_1

Hypotheses designed so that H_1 is true if H_0 is false

A ‘test statistic’ (quantity whose distribution is known under H_0)

p-value: the probability of a value of the test statistic as extreme or more extreme than the observed value (assuming H_0 is true)

=> p-value ‘controls the false positive rate’

Example of a statistical hypothesis test I

Simplest case:

Suppose the heights of Irish males are normally distributed with mean, 176cm, and standard deviation 7.6cm. A man of unknown nationality presents with height 200cm. Perform a statistical test to accept or reject the null hypothesis that the man is Irish.

Null hypothesis: Man is Irish

Test statistic: The man's height

Distribution of the test stat under null hypothesis: $N(176,7^2)$

2*(1 - pnorm(200,176,7)) will tell you the probability of getting a height as extreme as this under the null hypothesis.

(procedure outlined above is called the Z test)

Example of a statistical hypothesis test II

Student's T test (named after William Sealy Gosset, aka Student, a Guinness brewer)

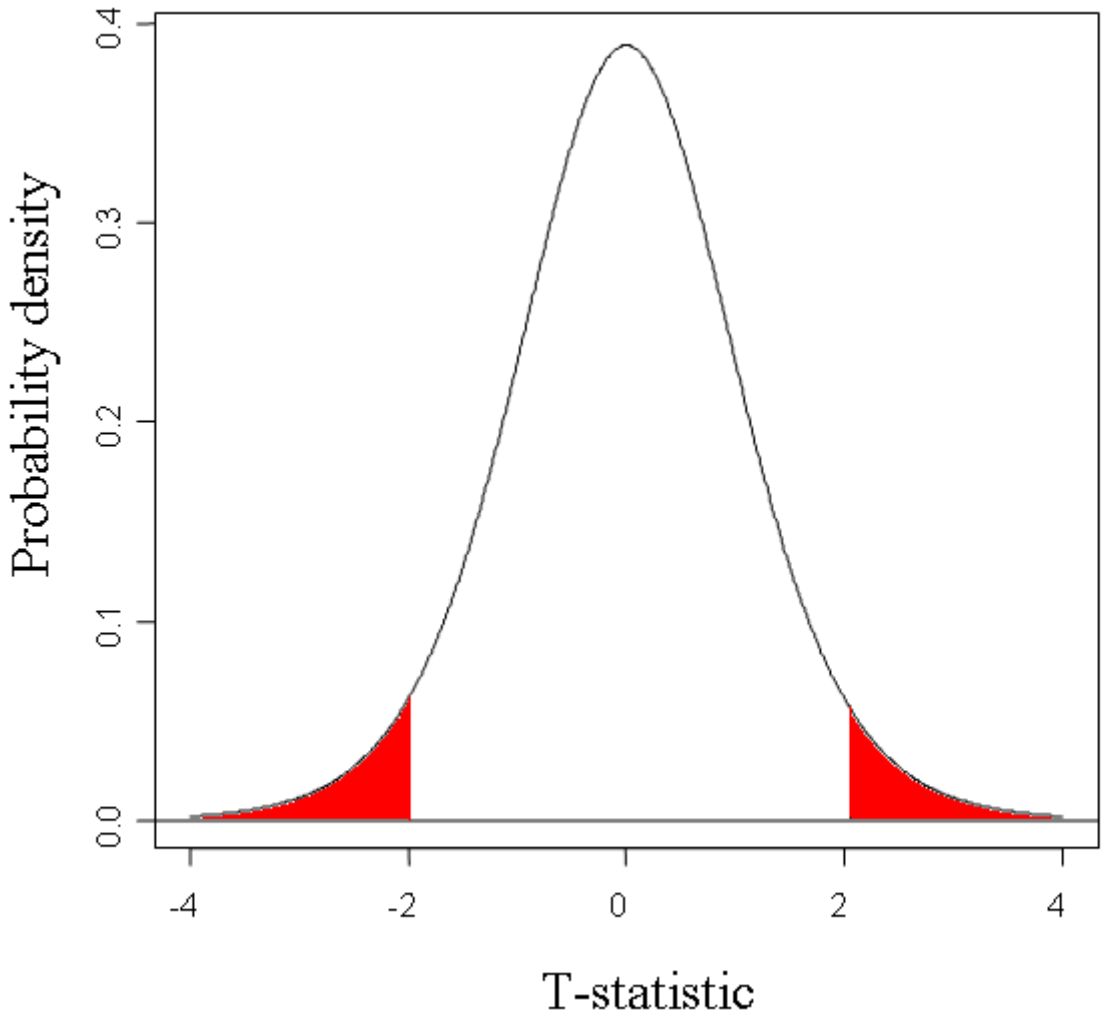
E.g. given a 'random' sample of the heights of Irish males and UK males, decide whether the mean heights of the two populations are different (without being told the population mean and variance of the heights).

Null hypothesis: Difference in the means is 0

Test statistic: Student's T (related to the difference in the means; takes account also of the standard deviation of this difference – which is known because of the central limit theorem)

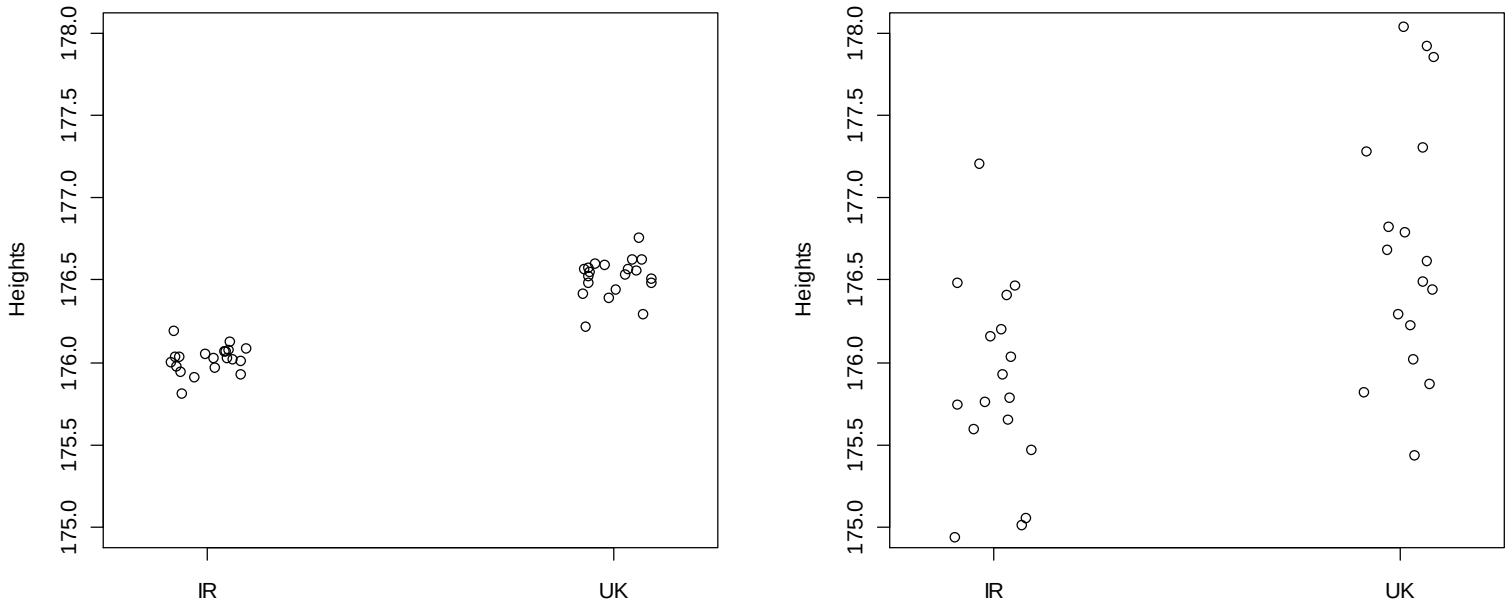
R function: t.test(x,y)

Suppose the observed value of the t-statistic is 2. The fraction of the area under the curve which is more extreme – i.e. more towards the tails) than 2 is the p-value



Aside: why is it necessary to take into account the standard deviation

In both cases the IR and UK means differ by 0.5cm. On the left the standard deviation is 0.1cm, on the right it is 0.9cm



Example: t-test

```
h = rnorm(200,m=1.7,sd=0.2)
g = c(rep("IR",100),rep("GB",100))

t.test(h[g == "IR"],h[g == "GB"]) # two-sided t-test

t.test(h[g == "IR"],h[g == "GB"],alt="less") # one-sided t-test

t.test(h[g == "IR"],h[g == "GB"],alt="greater") # the other one-sided t-test
```

Now let's use R to compare the samples using 'label permutation' and see how the results compare to a t-test…

```
random = c()

for(i in 1:1000) {
s = sample(h)
random = c(random,mean(s[g == "IR"]) - mean(s[g == "GB"]))
}

hist(random)
```

```
p_perm = length(random[random < mean(h[g == "IR"]) - mean(h[g
== "GB"])])/length(random)

t.test(h[g == "IR"],h[g == "GB"],alt="less")
```

When in doubt (and when it’s possible) label permutation can be a useful sanity check

t-test: some more details

t-test involves comparing values of the t-statistic to values that would be obtained under H_0

The t-statistic is calculated from the difference in the sample means, divided by an estimate of the standard deviation (sd) of this difference

Estimation of the sd of the difference in the means is the tricky part

There are different variants of the t-statistic exist, depending on the assumptions made in calculating this sd (e.g. assume the variance is the same in the two populations or not).

The distribution of the t-statistic depends on the sample size. Effectively, there are many different t-distributions (it's a 'family' of distributions, parameterized by the sample size/'degrees of freedom').

Correcting for multiple statistical hypothesis tests

Recall: significance threshold on a p-value ‘controls the false positive rate’

BUT, if we perform large numbers of hypotheses tests we expect large numbers of false positives, given a traditional cut-off such as 0.05.

Critical problem for which several responses are possible:

- Control the ‘family-wise error rate’ (i.e. use a much lower p-value threshold so that the probability of getting any false positives is low even with large number of tests – e.g. Bonferroni correction)

- Control the ‘false discovery rate’ (control the false positives as a proportion of the times you reject the null hypothesis – e.g. Benjamini & Hochberg method).

The p-value has a normal distribution under H_0

```
h = rnorm(200,m=1.7,sd=0.2)
g = c(rep("IR",100),rep("GB",100))

p_value = c()

for(i in 1:1000) {
s = sample(h)
p_value = c(p_value,t.test(s[g == "IR"],s[g == "GB"])$p.value)
}

hist(p_value)
```

Correcting for multiple statistical hypothesis tests in R

p.adjust( ) is an R function that corrects for multiple hypothesis testing

Takes as input a vector of p-values

Select from several correction methods

e.g. holm (a family-wise method)

Benjamini & Hochberg (an fdr method)

The linear model

Recall the equation of a line

$$Y = bX + a$$

In this case the slope of the line is *b* and the intercept (place where the line crosses the y-axis) is *a*

The linear model

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

I.e. the ‘dependent variable’ *Y* is related by a line with slope β_1 and intercept β_0 to the ‘independent variable’ *X*. The relationship is not perfect. There’s some additive ‘noise’ or error, given by ε. This ‘error’ component of the model is often taken to be normally distributed.

A hypothesis test for the linear model

Typically we are interested in whether β is non-zero. This would mean that there is evidence of a linear relationship between *Y* and *X*.

Null hypothesis: no relationship between *Y* and *X*

Under the null hypothesis β is zero.

Let’s use R to simulate a linear model

```
> x = runif(100)  # take 100 sample points from the uniform distribution
> y = 4 * x + rnorm(100,0,0.5)  # our linear model relating x and y
> plot(x,y)                     # scatter plot of y against x

> my_lm_fit = lm(y~x)     # fit a linear model to y as a function of x
> summary(my_lm_fit)  # summarize my linear model fit
```

Bioconductor

Open source and open development suite of packages, mostly in R, for the analysis of genomic data sets

Goals (see bioconductor.org)
- provide access to a broad range of powerful statistical and graphical methods for the analysis of genomic data.
- facilitate inclusion of biological metadata in the analysis of genomic data, e.g. literature data from PubMed, annotation data from Entrez genes.
- provide a common software platform that enables the rapid development and deployment of extensible, scalable, and interoperable software.
- further scientific understanding by producing high-quality documentation and reproducible research.
- train researchers on computational and statistical methods for the analysis of genomic data.

Bioconductor

Packages have extensive documentation

Include *Vignettes* – task oriented descriptions of the functionality of the package

Aims to provide convenient and seamless access to ‘annotation’ information (e.g. advanced analysis of microarray data requires annotation information that includes details of the array design, the genes that probes map to etc.; see next section of the course)

R and Bioconductor aim to support *reproducible* research