

# Frequentist foundations

## SMLP 2017, Potsdam, Germany

Shravan Vasishth

Universität Potsdam  
vasishth@uni-potsdam.de  
<http://www.ling.uni-potsdam.de/~vasishth>

August 9, 2017

# What this first day is about

- 1 In Monday's lectures, we aim to provide a quick review of the foundational ideas in frequentist statistical theory.
- 2 We will cover what you will need as a basis for the rest of the summer school.
- 3 We will mainly use simulation to understand the key concepts, and this requires some knowledge of the language R.
- 4 We assume you have downloaded and installed R from:  
<http://cran.r-project.org/>

# Prerequisites for this course

- 1 For this course, we only assume that you are willing to put in some work on your own.
- 2 This means doing the exercises we provide.
- 3 A certain amount of fearlessness is also assumed.
- 4 Only minimal math is used.

# Course lecture notes (optional during the summer school)

We have two sets of notes. Choose the one you like more.

- 1 A less technical, more intuitive presentation:  
<https://github.com/vasishth/Statistics-lecture-notes-Potsdam/tree/master/IntroductoryStatistics>
- 2 A more technical presentation assuming basic calculus and linear algebra:  
<https://github.com/vasishth/LM>
- 3 During this summer school, it is enough to just follow the slides. Read the notes later, after you go home!

We suggest reading the first set of notes and then the second set.

# Exercises

We will provide exercises during the summer school.  
Solutions will be provided, and discussed in class.

# What the frequentist stream is about

We will cover the following topics:

- 1 Random variables, including jointly distributed RVs, univariate probability distributions, Maximum Likelihood Estimation.
- 2 The sampling distribution of the mean, null hypothesis, t-tests, confidence intervals.
- 3 Type I error, Type II error, power, Type M and Type S errors.
- 4 An introduction to linear modeling.
- 5 An introduction to linear mixed modeling.

## for-loops

One construct we will use often is calculating some (varying) quantity repeatedly, and then storing the result of that calculation in a vector.

An example:

```
## number of iterations:
nsim<-10
## vector for storing results:
results<-rep(NA,10)
for(i in 1:nsim){
  results[i]<-1+2*i
}
results

##  [1]  3  5  7  9 11 13 15 17 19 21
```

## The definition of a random variable

A random variable  $X$  is a function  $X : S \rightarrow \mathbb{R}$  that associates to each outcome  $\omega \in S$  exactly one number  $X(\omega) = x$ .

$S_X$  is all the  $x$ 's (all the possible values of  $X$ , the support of  $X$ ).

i.e.,  $x \in S_X$ .

**Discrete example:** number of coin tosses till H

- $X : \omega \rightarrow x$
- $\omega$ : H, TH, TTH, ... (infinite)
- $x = 0, 1, 2, \dots; x \in S_X$

We will write  $X(\omega) = x$ :

$$H \rightarrow 1$$

$$TH \rightarrow 2$$

$$\vdots$$



## Probability mass/distribution function

Every discrete random variable  $X$  has associated with it a **probability mass function (PMF)**. Continuous RVs have **probability distribution functions** (PDFs). We will call both PDFs (for simplicity).

$$p_X : S_X \rightarrow [0, 1] \quad (1)$$

defined by

$$p_X(x) = P(X(\omega) = x), x \in S_X \quad (2)$$

This pmf tells us the probability of having getting a heads on 1, 2, ... tosses.

# The cumulative distribution function

The **cumulative distribution function** in the discrete case is

$$F(a) = \sum_{\text{all } x \leq a} p(x) \quad (3)$$

The cdf tells us the *cumulative* probability of getting a heads in 1 or less tosses; 2 or less tosses, . . . .

It will soon become clear why we need this.

## Discrete example: The binomial random variable

Suppose that we toss a coin  $n = 10$  times. There are two possible outcomes, success and failure, each with probability  $\theta$  and  $(1 - \theta)$  respectively.

Then, the probability of  $x$  successes out of  $n$  is defined by the pmf:

$$p_X(x) = P(X = x) = \binom{n}{x} \theta^x (1 - \theta)^{n-x} \quad (4)$$

[assuming a binomial distribution]

## Discrete example: The binomial random variable

Example:  $n = 10$  coin tosses. Let the probability of success be  $\theta = 0.5$ .

We start by asking the question:

What's the probability of  $x$  or fewer successes, where  $x$  is some number between 0 and 10?

Let's compute this. We use the built-in CDF function `pbinom`.

## Discrete example: The binomial random variable

```
## sample size
n<-10
## prob of success
p<-0.5
probs<-rep(NA,11)
for(x in 0:10){
  ## Cumulative Distribution Function:
  probs[x+1]<-round(pbinom(x,size=n,prob=p),digits=2)
}
```

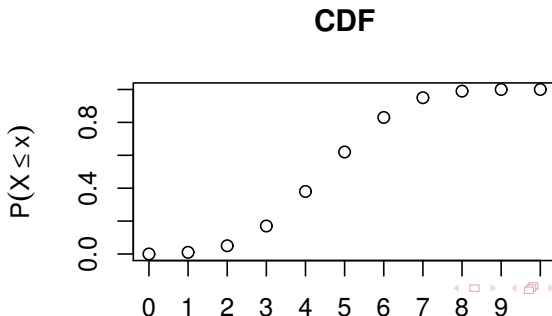
We have just computed the cdf of this random variable.

## Discrete example: The binomial random variable

	$P(X \leq x)$	cumulative probability
1	0	0.00
2	1	0.01
3	2	0.05
4	3	0.17
5	4	0.38
6	5	0.62
7	6	0.83
8	7	0.95
9	8	0.99
10	9	1.00
11	10	1.00

## Discrete example: The binomial random variable

```
## Plot the CDF:  
plot(1:11, probs, xaxt="n", xlab="x",  
      ylab=expression(P(X<=x)), main="CDF")  
axis(1, at=1:11, labels=0:10)
```



## Discrete example: The binomial random variable

Another question we can ask involves the pmf: What is the probability of getting exactly  $x$  successes? For example, if  $x=1$ , we want  $P(X=1)$ .

We can get the answer from (a) the cdf, or (b) the pmf:

```
## using cdf:
pbinom(1,size=10,prob=0.5)-pbinom(0,size=10,prob=0.5)

## [1] 0.009765625

## using pmf:
choose(10,1) * 0.5 * (1-0.5)^9

## [1] 0.009765625
```



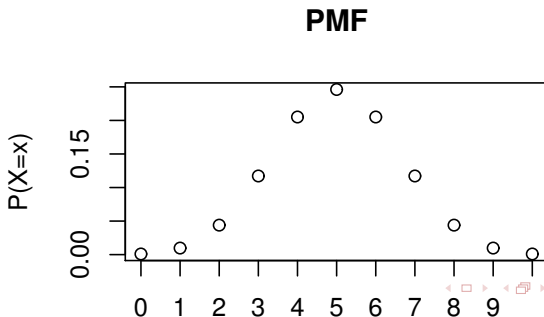
## Discrete example: The binomial random variable

The built-in function in R for the pmf is `dbinom`:

```
##  $P(X=1)$   
choose(10,1) * 0.5 * (1-0.5)^9  
  
## [1] 0.009765625  
  
## using the built-in function:  
dbinom(1,size=10,prob=0.5)  
  
## [1] 0.009765625
```

## Discrete example: The binomial random variable

```
## Plot the pmf:  
plot(1:11,dbinom(0:10,size=10,prob=0.5),main="PMF",  
     xaxt="n",ylab="P(X=x)",xlab="x")  
axis(1,at=1:11,labels=0:10)
```



## Summary: Random variables

To summarize, the discrete binomial random variable  $X$  will be defined by

- 1 the function  $X : S \rightarrow \mathbb{R}$ , where  $S$  is the set of outcomes (i.e., outcomes are  $\omega \in S$ ).
- 2  $X(\omega) = x$ , and  $S_X$  is the **support** of  $X$  (i.e.,  $x \in S_X$ ).
- 3 A PMF is defined for  $X$ :

$$p_X : S_X \rightarrow [0, 1]$$

$$p_X(x) = \binom{n}{x} \theta^x (1 - \theta)^{n-x} \quad (5)$$

- 4 A CDF is defined for  $X$ :

$$F(a) = \sum_{\text{all } x \leq a} p(x)$$

## Generating random binomial data

We can use the **rbinom** function to generate binomial data. So, 10 coin tosses can be simulated as follows:

```
rbinom(1,n=10,prob=0.5)
```

```
##      [1] 0 0 0 1 0 1 1 0 1 1
```

## Exercise

Do Exercise 1 now.

## Continuous example: The normal random variable

The pdf of the normal distribution is:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{(x-\mu)^2}{\sigma^2}}, \quad -\infty < x < \infty \quad (6)$$

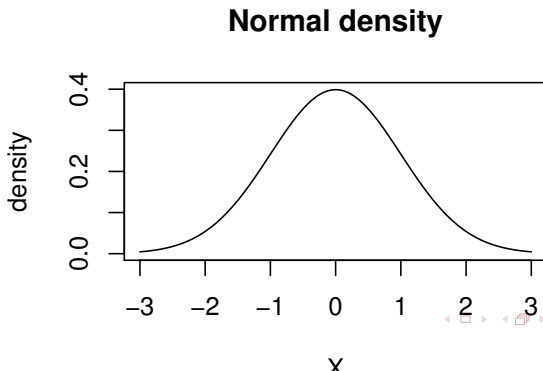
We write  $X \sim \text{norm}(\text{mean} = \mu, \text{sd} = \sigma)$ .

The associated R function for the pdf is `dnorm(x, mean = 0, sd = 1)`, and the one for cdf is `pnorm`.

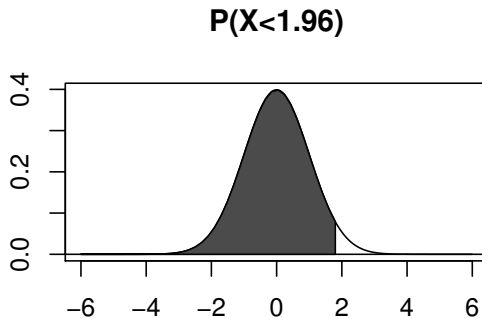
Note the default values for  $\mu$  and  $\sigma$  are 0 and 1 respectively. Note also that R defines the PDF in terms of  $\mu$  and  $\sigma$ , not  $\mu$  and  $\sigma^2$  ( $\sigma^2$  is the norm in statistics textbooks).

## Continuous example: The normal RV

```
plot(function(x) dnorm(x), -3, 3,  
      main = "Normal density", ylim=c(0, .4),  
      ylab="density", xlab="X")
```



## Probability in continuous RVs: The area under the curve





## Continuous example: The normal RV with $\mu = 0, \sigma = 1$

Computing probabilities using the CDF:

```
## The area under curve between +infy and -infy:
```

```
pnorm(Inf)-pnorm(-Inf)
```

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

```
## The area under curve between 2 and -2:
```

```
pnorm(2)-pnorm(-2)
```

```
## [1] 0.9544997
```

```
## The area under curve between 1 and -1:
```

```
pnorm(1)-pnorm(-1)
```

```
## [1] 0.6826895
```

## Finding the quantile given the probability

We can also go in the other direction: given a probability  $p$ , we can find the quantile  $x$  of a  $Normal(\mu, \sigma)$  such that  $P(X < x) = p$ .

For example:

The quantile  $x$  given  $X \sim N(\mu = 500, \sigma = 100)$  such that  $P(X < x) = 0.975$  is

```
qnorm(0.975, mean=500, sd=100)
```

```
## [1] 695.9964
```

This will turn out to be very useful in statistical inference.

## Standard or unit normal random variable

If  $X$  is normally distributed with parameters  $\mu$  and  $\sigma$ , then  $Z = (X - \mu)/\sigma$  is normally distributed with parameters  $\mu = 0, \sigma = 1$ .

We conventionally write  $\Phi(x)$  for the CDF of  $N(0,1)$ :

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{\frac{-y^2}{2}} dy \quad \text{where } y = (x - \mu)/\sigma \quad (7)$$

## Standard or unit normal random variable

For example:  $\Phi(2)$ :

```
pnorm(2)
```

```
## [1] 0.9772499
```

For negative  $x$  we write:

$$\Phi(-x) = 1 - \Phi(x), \quad -\infty < x < \infty \quad (8)$$

## Standard or unit normal random variable

In R:

```
1-pnorm(2)

## [1] 0.02275013

## alternatively:
pnorm(2,lower.tail=F)

## [1] 0.02275013
```

## Standard or unit normal random variable

If  $Z$  is a standard normal random variable (SNRV) then

$$p\{Z \leq -x\} = P\{Z > x\}, \quad -\infty < x < \infty \quad (9)$$

Since  $Z = ((X - \mu)/\sigma)$  is an SNRV whenever  $X$  is normally distributed with parameters  $\mu$  and  $\sigma$ , then the CDF of  $X$  can be expressed as:

$$F_X(a) = P\{X \leq a\} = P\left(\frac{X - \mu}{\sigma} \leq \frac{a - \mu}{\sigma}\right) = \Phi\left(\frac{a - \mu}{\sigma}\right) \quad (10)$$

The standardized version of a normal random variable  $X$  is used to compute specific probabilities relating to  $X$ .

We will soon see the relevance of the SNRV in hypothesis testing.

## dnorm, pnorm, qnorm

1 For the normal distribution we have built in functions:

1 dnorm: the pdf

2 pnorm: the cdf

3 qnorm: the inverse of the cdf

2 Other distributions also have analogous functions:

1 Binomial: dbinom, pbinom, qbinom

2 t-distribution: dt, pt, qt

We will be using the t-distribution's dt, pt, and qt functions a lot in statistical inference.

# Exercise

Do Exercise 2 now.



# Maximum Likelihood Estimation

We now turn to an important topic: maximum likelihood estimation.

## MLE: The binomial distribution

Suppose we toss a fair coin 10 times, and count the number of heads each time; we repeat this experiment 5 times in all. The observed sample values are  $x_1, x_2, \dots, x_5$ .

```
(x<-rbinom(5,size=10,prob=0.5))
```

```
## [1] 6 6 3 5 4
```

The joint probability of getting all these values (assuming independence) depends on the parameter we set for the probability  $\theta$ :

$$\begin{aligned} &P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ &= P(X_1 = x_1) \times P(X_2 = x_2), \times \dots P(X_n = x_n) \\ &= f(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n; \theta) \end{aligned}$$

## MLE: The binomial distribution

$$\begin{aligned} &P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) \\ &= P(X_1 = x_1) \times P(X_2 = x_2) \times \dots \times P(X_n = x_n) \\ &= f(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n; \theta) \end{aligned}$$

The above joint probability is a function of  $\theta$ . When the probability is expressed as a function of  $\theta$ , we call it the **likelihood function**.

Note that the likelihood function itself is not a pdf.

## MLE: The binomial distribution

The value of  $\theta$  for which this function has the maximum value is the **maximum likelihood estimate**.

```
## probability parameter fixed at 0.5
```

```
theta<-0.5
```

```
prod(dbinom(x,size=10,prob=theta))
```

```
## [1] 0.0002487367
```

```
## probability parameter fixed at 0.1
```

```
theta<-0.1
```

```
prod(dbinom(x,size=10,prob=theta))
```

```
## [1] 1.809443e-14
```

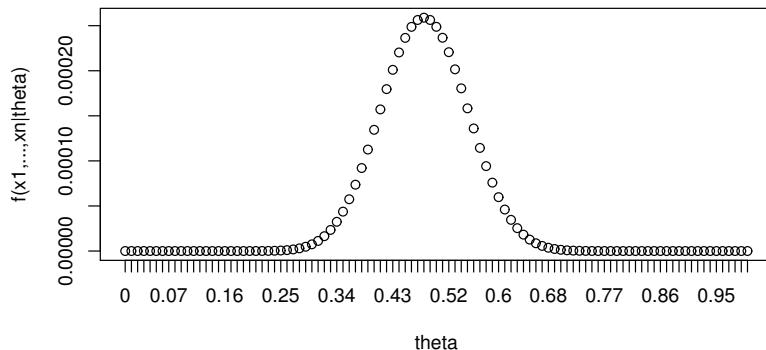
## MLE: The binomial distribution

Let's compute the product for a range of probabilities:

```
theta<-seq(0,1,by=0.01)
store<-rep(NA,length(theta))
for(i in 1:length(theta)){
  store[i]<-prod(dbinom(x,size=10,prob=theta[i]))
}
head(store)

## [1] 0.000000e+00 2.156526e-37 2.778674e-30 3.582737e-26
## [6] 4.398793e-21
```

# MLE: The binomial distribution



# MLE: The binomial distribution

Detailed derivations: see lecture notes

We can obtain this estimate of  $\theta$  that maximizes likelihood by computing:

$$\hat{\theta} = \frac{x}{n} \quad (11)$$

where  $n$  is sample size, and  $x$  is the number of successes.

For the analytical derivation, see the Linear Modeling lecture notes, section 4: <https://github.com/vasisht/LM>

# MLE: The normal distribution

Detailed derivations: see lecture notes

For the normal distribution, where  $X \sim N(\mu, \sigma)$ , we can get MLEs of  $\mu$  and  $\sigma$  by computing:

$$\hat{\mu} = \frac{1}{n} \sum x_i = \bar{x} \quad (12)$$

and

$$\hat{\sigma}^2 = \frac{1}{n} \sum (x_i - \bar{x})^2 \quad (13)$$

you will sometimes see the “unbiased” estimate (and this is what R computes) but for large sample sizes the difference is not important (see p 38 of Linear Modeling notes):

$$\hat{\sigma}^2 = \frac{1}{n-1} \sum (x_i - \bar{x})^2 \quad (14)$$



## The significance of the MLE

The significance of these MLEs is that, having assumed a particular underlying pdf, we can estimate the (unknown) parameters (the mean and variance) of the distribution that generated our particular data.

This leads us to the distributional properties of the mean **under repeated sampling**.

- └ The sampling distribution of the mean
  - └ Sampling from the normal distribution

## The sampling distribution of the mean

When we have a **single sample**, we know how to compute MLEs of the sample mean and standard deviation,  $\hat{\mu}$  and  $\hat{\sigma}$ .

Suppose now that you had many repeated samples; from each sample, you can compute the mean each time. We can simulate this situation:

```
x<-rnorm(100,mean=500,sd=50)
mean(x)
```

```
## [1] 493.3942
```

```
x<-rnorm(100,mean=500,sd=50)
mean(x)
```

```
## [1] 505.7302
```

- └ The sampling distribution of the mean
  - └ Sampling from the normal distribution

## The sampling distribution of the mean

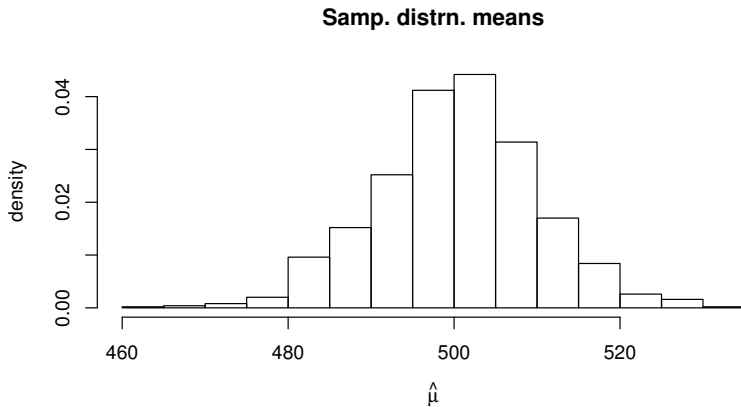
Let's repeatedly simulate sampling 1000 times:

```
nsim<-1000
n<-100
mu<-500
sigma<-100
samp_distrn_means<-rep(NA,nsim)
samp_distrn_sd<-rep(NA,nsim)
for(i in 1:nsim){
  x<-rnorm(n,mean=mu,sd=sigma)
  samp_distrn_means[i]<-mean(x)
  samp_distrn_sd[i]<-sd(x)
}
```

- └ The sampling distribution of the mean
- └ Sampling from the normal distribution

# The sampling distribution of the mean

Plot the distribution of the means under repeated sampling:



- └ The sampling distribution of the mean
  - └ Sampling from the exponential distribution

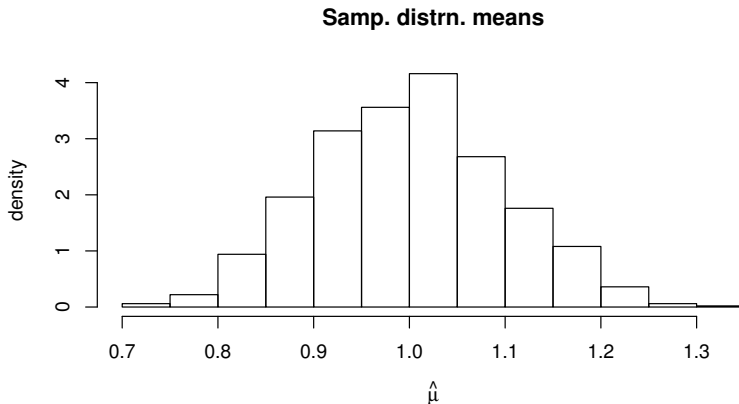
## The sampling distribution of the mean

Interestingly, it is not necessary that the distribution that we are sampling from be the normal distribution.

```
for(i in 1:nsim){  
  x<-rexp(n)  
  samp_distrn_means[i]<-mean(x)  
  samp_distrn_sd[i]<-sd(x)  
}
```

- └ The sampling distribution of the mean
  - └ Sampling from the exponential distribution

# The sampling distribution of the mean



# The central limit theorem

- 1 For large enough sample sizes, the sampling distribution of the means will be approximately normal, regardless of the underlying distribution (as long as this distribution has a mean and variance defined for it).
- 2 This will be the basis for statistical inference.

# The sampling distribution of the mean

We can compute the standard deviation of the sampling distribution of means:

```
## estimate from simulation:
```

```
sd(samp_distrn_means)
```

```
## [1] 0.09896015
```



## The sampling distribution of the mean

A further interesting fact is that we can compute this standard deviation of the sampling distribution **from a single sample** of size  $n$ :

$$\frac{\hat{\sigma}}{\sqrt{n}}$$

```
n<-100
mu<-500
sigma<-100
x<-rnorm(n,mean=mu,sd=sigma)
hat_sigma<-sd(x)
hat_sigma/sqrt(n)

## [1] 10.16731
```

See linear modeling notes, section 5.1.5, for an analytical proof.

# The sampling distribution of the mean

- 1 So, from a sample of size  $n$ , and sd  $\sigma$  or an MLE  $\hat{\sigma}$ , we can compute the standard deviation of the sampling distribution of the means.
- 2 We will call this standard deviation the estimated **standard error**.

$$SE = \frac{\hat{\sigma}}{\sqrt{n}}$$

I say *estimated* because we are estimating SE using an estimate of  $\sigma$ .

## Exercise

Do Exercise 3 now.

## Confidence intervals

The standard error allows us to define an (approximate) **95% confidence interval**:

$$\hat{\mu} \pm 2SE \quad (15)$$

So, for the mean, we define a 95% confidence interval as follows:

$$\hat{\mu} \pm 2 \frac{\hat{\sigma}}{\sqrt{n}} \quad (16)$$

## Confidence intervals

In our example:

```
## lower bound:  
mu-(2*hat_sigma/sqrt(n))  
  
## [1] 479.6654  
  
## upper bound:  
mu+(2*hat_sigma/sqrt(n))  
  
## [1] 520.3346
```

## The meaning of the 95% CI

If you take repeated samples and compute the CI each time, 95% of those CIs will contain the true population mean.

```
nsim<-100
lower<-rep(NA,nsim)
upper<-rep(NA,nsim)
for(i in 1:nsim){
  x<-rnorm(n,mean=mu,sd=sigma)
  lower[i]<-mean(x) - 2 * sd(x)/sqrt(n)
  upper[i]<-mean(x) + 2 * sd(x)/sqrt(n)
}
```

# The meaning of the 95% CI

```
## check how many CIs contain mu:
CIs<-ifelse(lower<mu & upper>mu,1,0)
table(CIs)

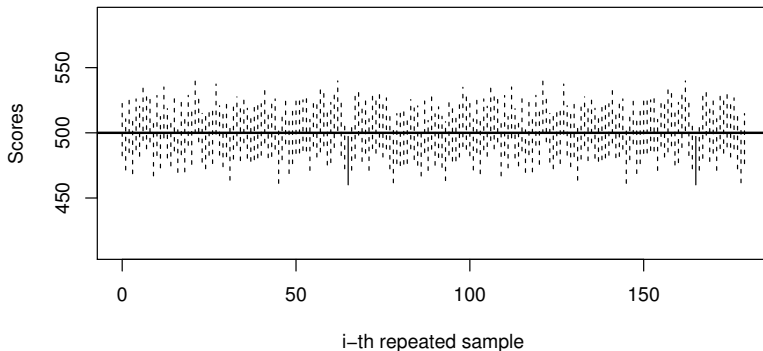
## CIs
##    0    1
##    2  98

## approx. 95% of the CIs contain true mean:
table(CIs)[2]/sum(table(CIs))

##      1
## 0.98
```

# The meaning of the 95% CI

**95% CIs in 100 repeated samples**





## The meaning of the 95% CI

- 1 The 95% CI from a **particular** sample does **not** mean that the true value of the mean (a point value) lies inside that particular CI with probability 95%.
- 2 The “95%” refers to the coverage properties of the CI under repeated sampling.
- 3 Thus, the CI has a very confusing and (not very useful!) interpretation.
- 4 In Bayesian statistics we use the credible interval, which has the above interpretation (more on this later).

However, in our examples, for large sample sizes, the credible and confidence intervals tend to be essentially identical.

For this reason, the CI is often treated (this is technically incorrect!) as a way to characterize uncertainty about our estimate of the mean.

## The meaning of the 95% CI

**Exercise:** Generate 95% confidence intervals from a normal distribution with mean 40 and sd 10, with 10,000 simulations instead of 100.

Verify that the proportion of intervals that do not contain the true mean is about 5%.

## Main points from this lecture

- 1 We compute maximum likelihood estimates of the mean  $\bar{x} = \hat{\mu}$  and standard deviation  $\hat{\sigma}$  to get estimates of the true but unknown parameters.

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

- 2 For a given sample, having estimated  $\hat{\sigma}$ , we estimate the standard error:

$$SE = \hat{\sigma} / \sqrt{n}$$

- 3 This allows us to define a 95% CI about the estimated mean:

$$\hat{\mu} \pm 2 \times SE$$

From here, we move on to statistical inference and null hypothesis significance testing (NHST).