

Introduction to statistical modelling

Gavin Band

gavin.band@well.ox.ac.uk



UNIVERSITY OF
OXFORD



Statistical modelling

- We are interested in studying natural processes occurring in natural populations.
- We observe some **data**.

Q. How to draw inferences about the underlying processes?

A. By modelling.

What is modelling anyway?

A “narrow” view of statistical modelling goes like this:

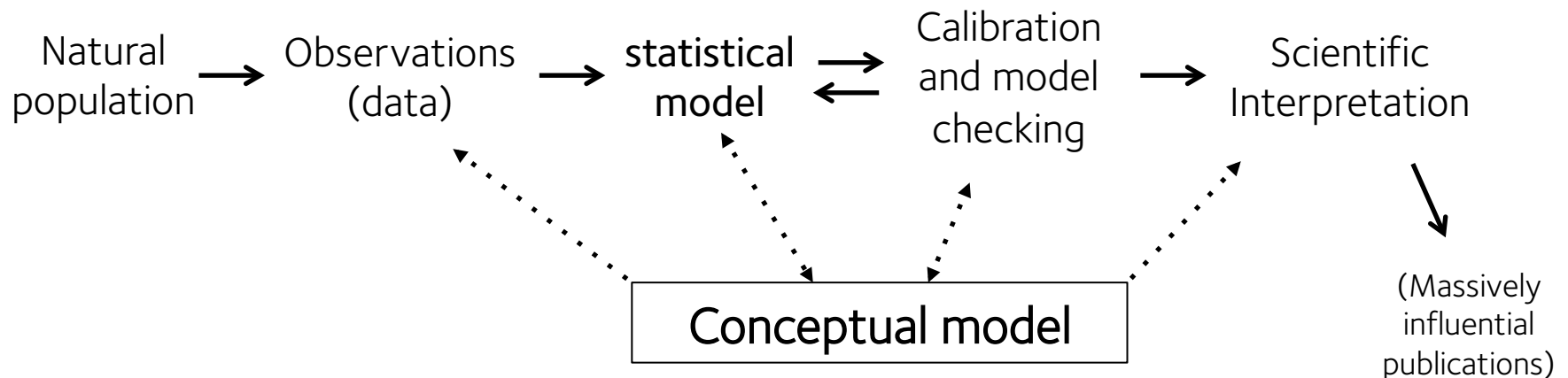


What is modelling?

A “narrow” view of statistical modelling goes like this:

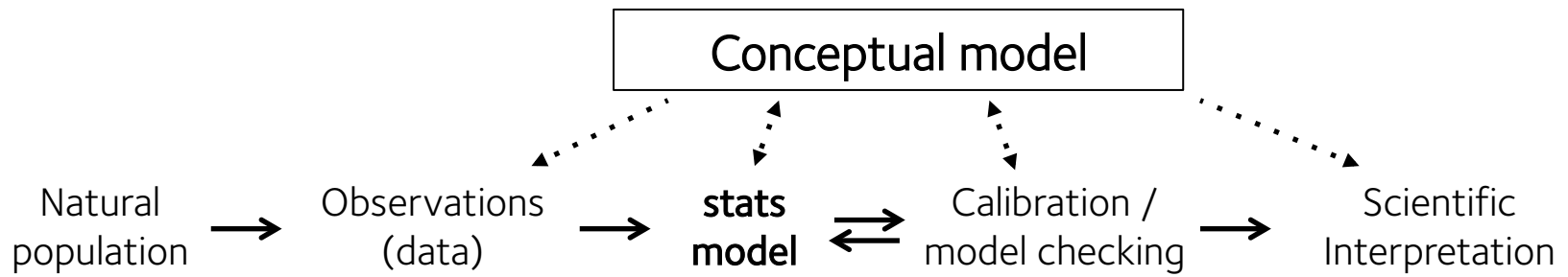


I’m going to argue it is instead this:

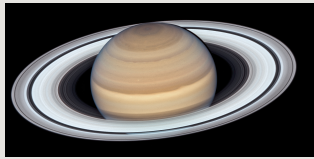
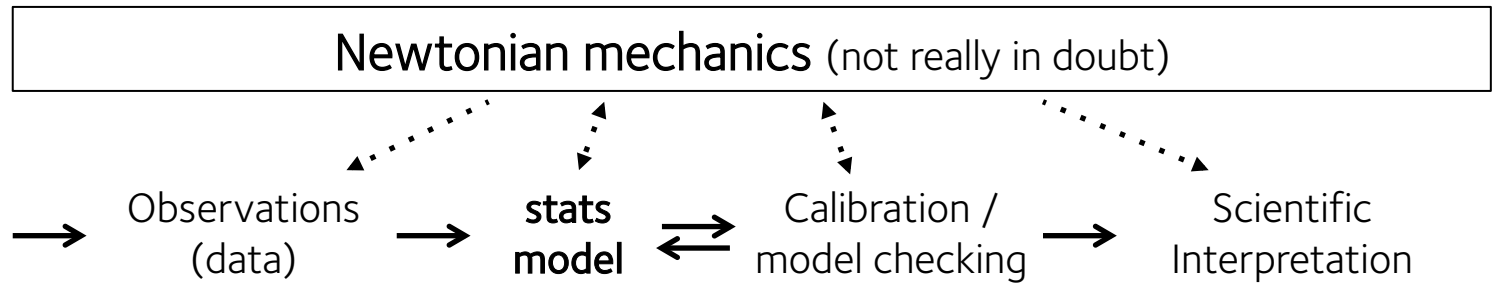


The conceptual model determines what is measured, how a formal statistical model is constructed, how we can tell whether the formal model has worked, and how we should interpret the results. If the conceptual model changes because of the data – then we have learnt something.

Examples

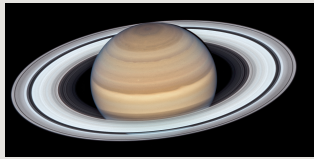
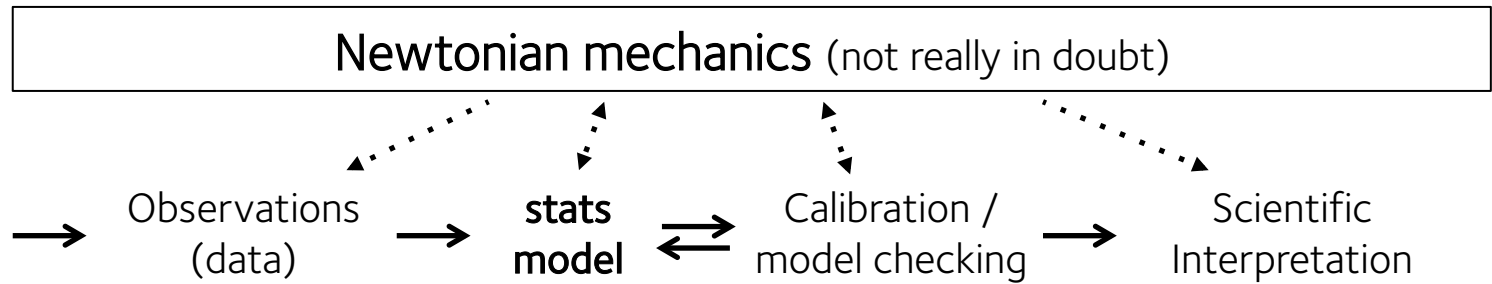


Example: the mass of Saturn



N=1

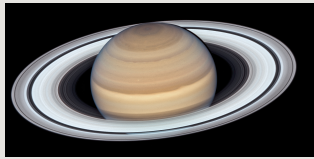
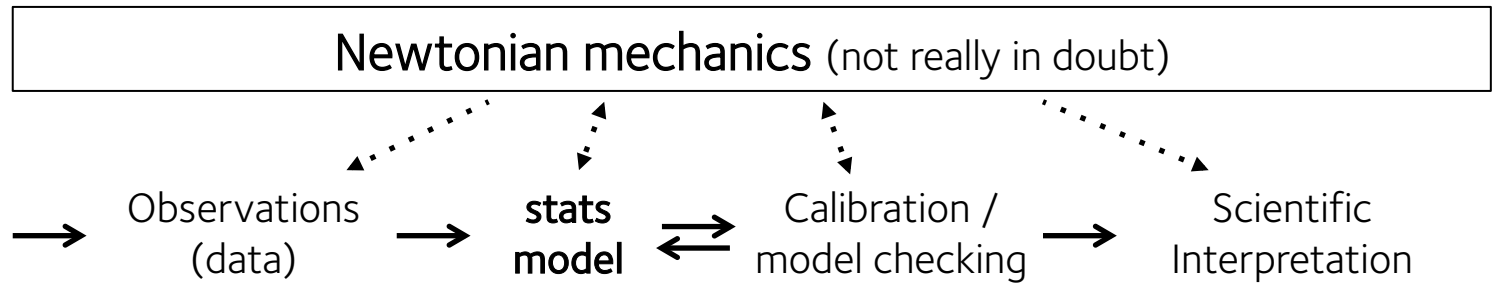
Example: the mass of Saturn



N=1

Measured orbits
of Saturn and
other bodies

Example: the mass of Saturn

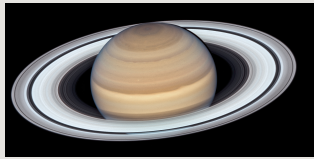
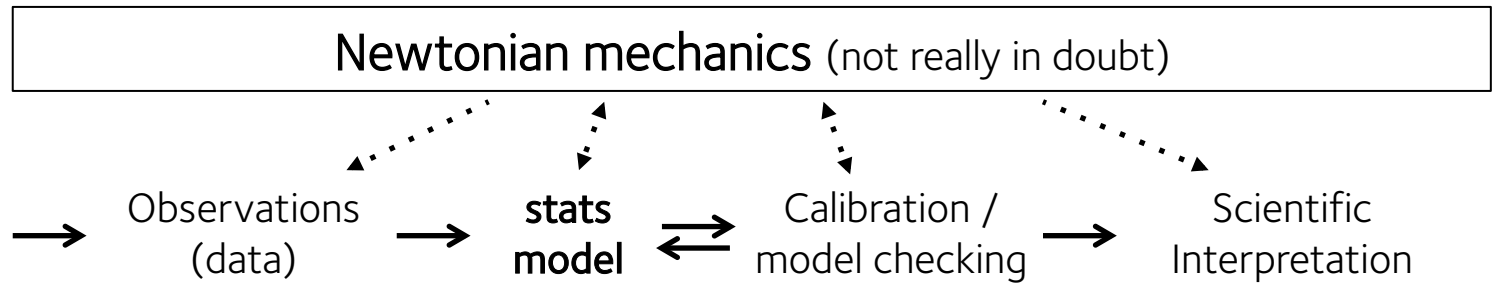


N=1

Measured orbits
of Saturn and
other bodies

$P(\text{data}|\text{mass of Saturn})$ - **Likelihood function**
(probability of data given parameters). Accounts for all the
uncertainties in the observations.

Example: the mass of Saturn cf. Laplace



N=1

Measured orbits
of Saturn and
other bodies

$P(\text{data}|\text{mass of Saturn})$ - **Likelihood function**
(probability of data given parameters). Accounts for all the
uncertainties in the observations.

Laplace used 'inverse probability' or 'Bayes theorem':

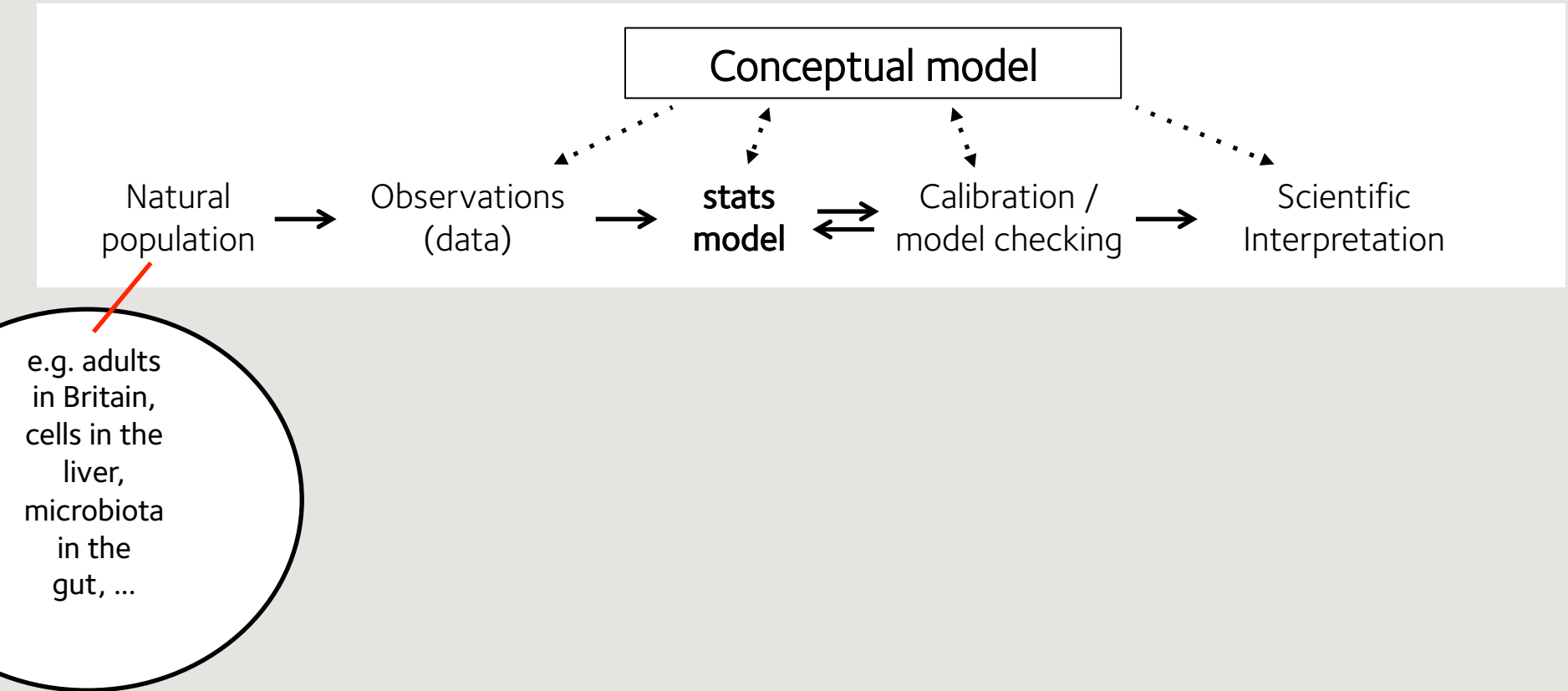
$$P(\text{mass}|\text{data}) \propto P(\text{data}|\text{mass}) \times P(\text{mass})$$

Posterior

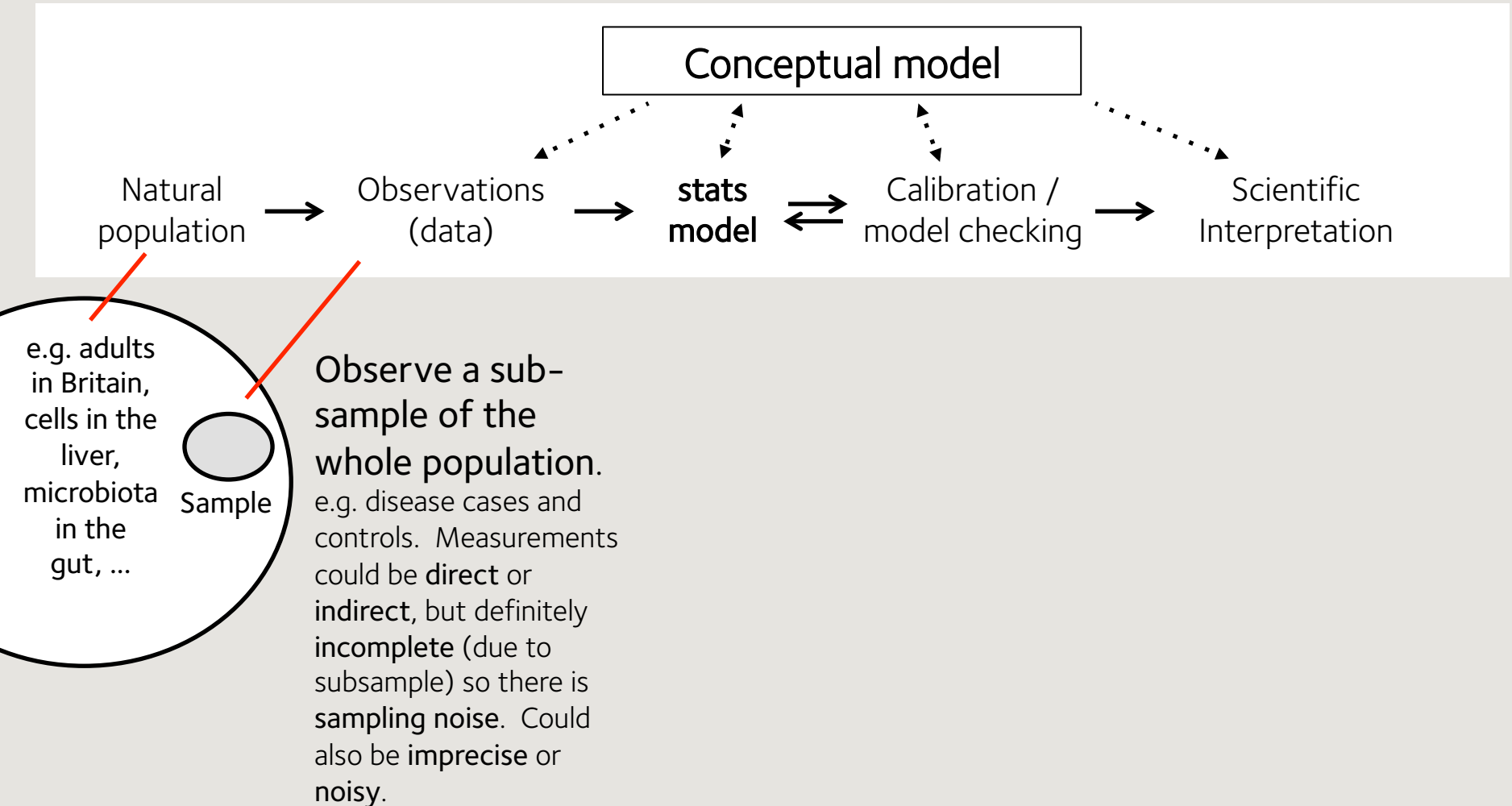
Likelihood

Prior

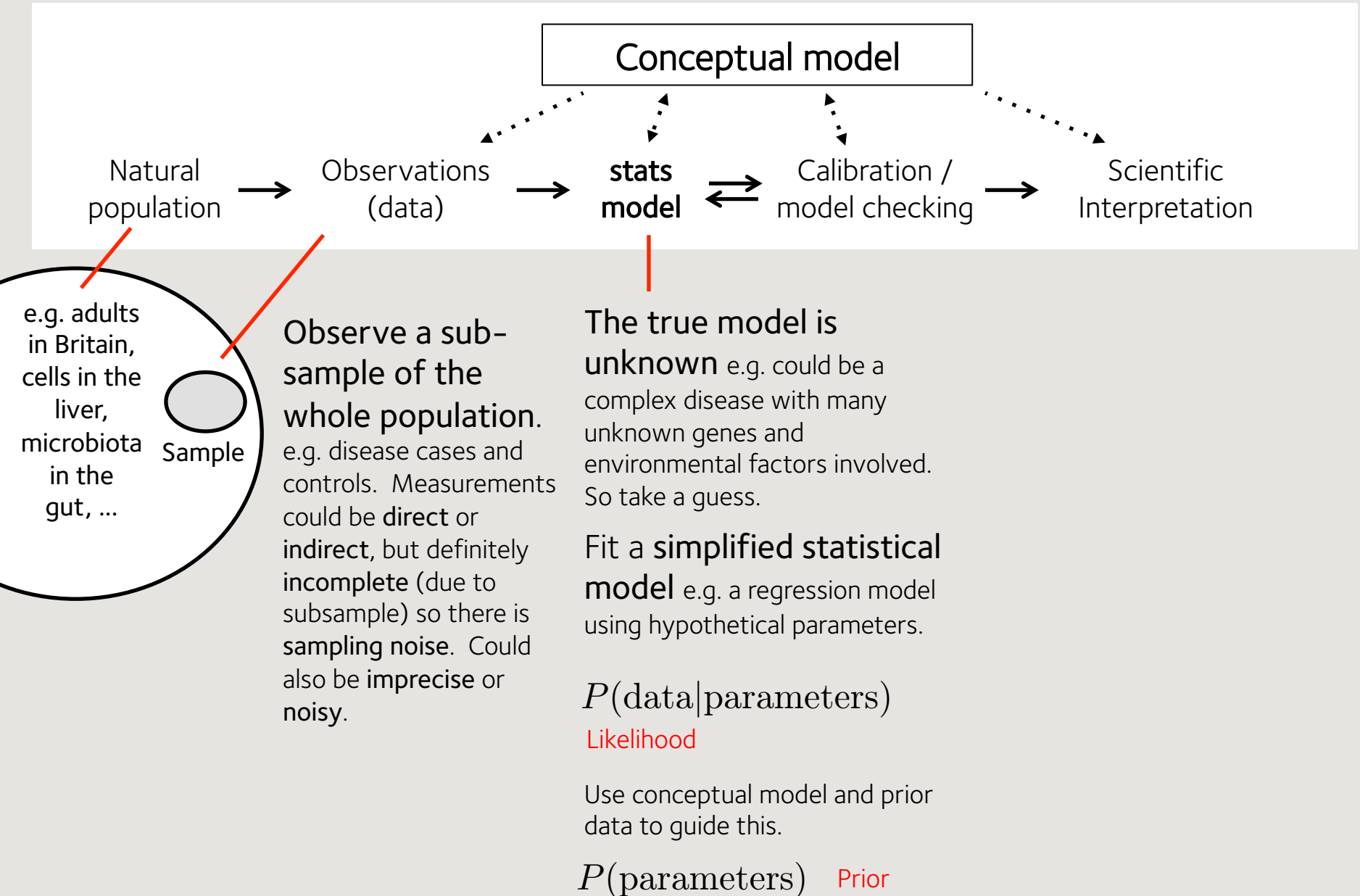
An example closer to home: genetic predisposition to disease



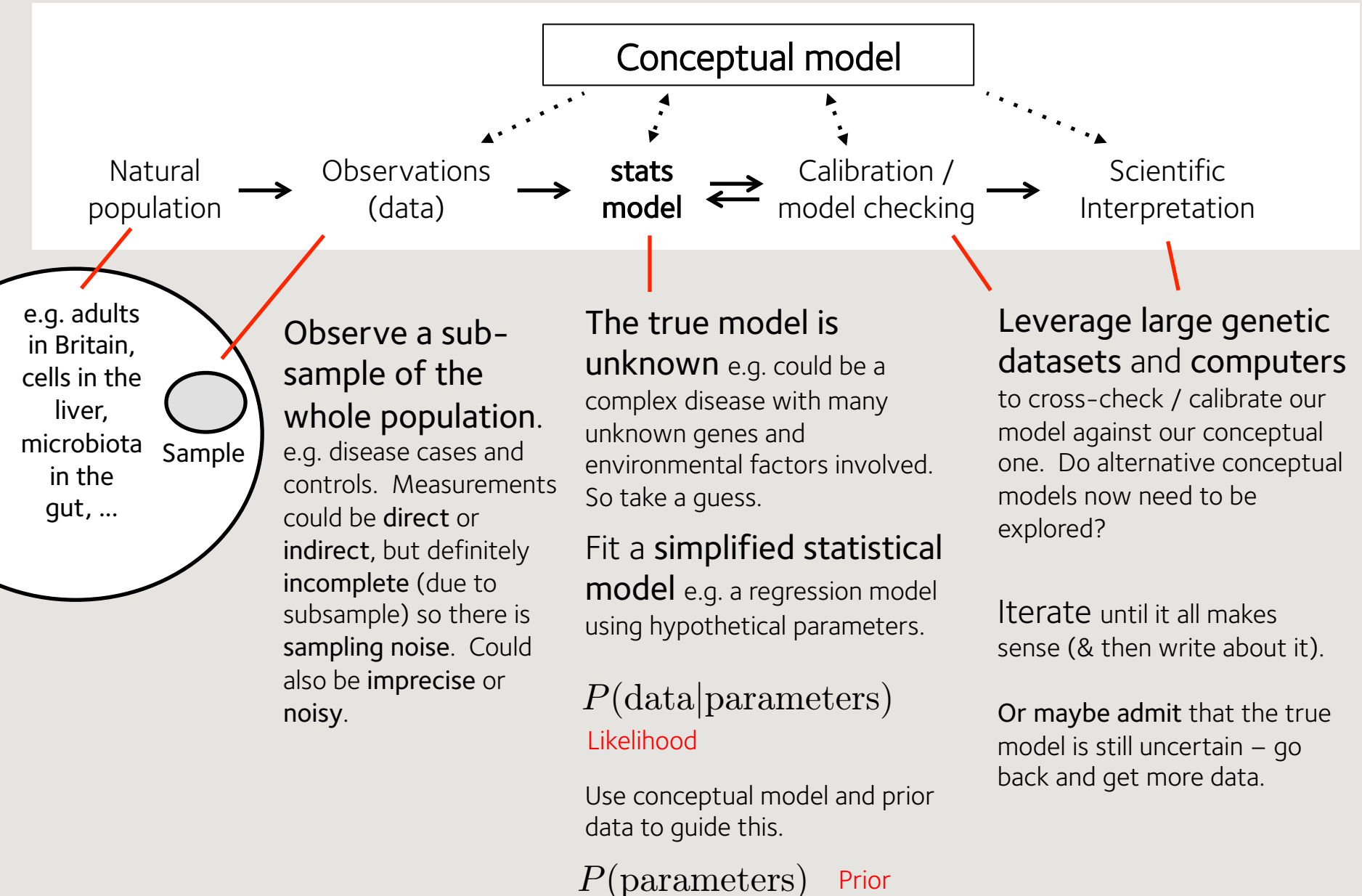
An example closer to home: genetic predisposition to disease



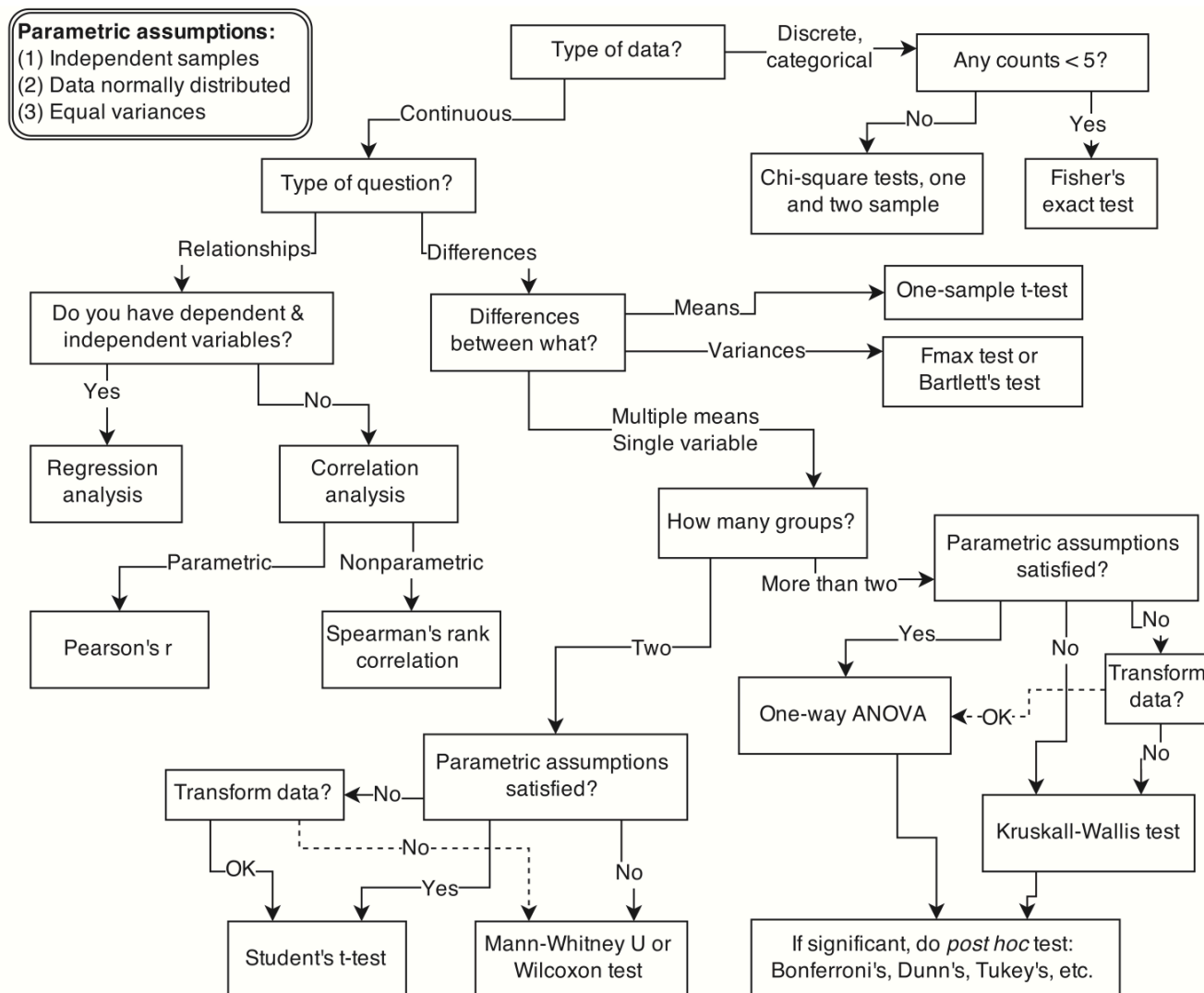
An example closer to home: genetic predisposition to disease



An example closer to home: genetic predisposition to disease



What we're not going to cover



Why probability?

To keep track of uncertain quantities and we use *probability*:

$$P(A|I) = \text{probability of } A, \text{ given information } I$$

Why? Turns out $P()$ is the only sane way to measure how plausible things are*:

1. $P()$ is a single real number, with 0=definitely false and 1=definitely true
2. $P()$ qualitatively corresponds to common sense (e.g. increasing plausibility of one thing increases plausibility of dependent things.)
3. $P()$ always gives consistent results (e.g. working out the same computation two different ways always gives the same answer)
4. There aren't any other functions $P()$ which do this.

*Jaynes, "Probability Theory: The Logic of Science" Chapters 1 & 2. (See READING LIST.md on github site.)

Why probability?


To keep track of uncertain quantities and we use *probability*:

$$P(A|I) = \text{probability of } A, \text{ given information } I$$

All probabilities are conditional on background information, though this often taken as implicit in the notation. They are ‘small world’ values*. All our P-values, Bayes factors, posterior probabilities are only valid under the assumed statistical model, which as we’ve argued is not the same as either reality, or even our conceptual model.

Also useful to write probabilities in another way – via the names of distributions.
For example:

$$X \sim N(\mu, \sigma^2) \quad X \text{ has a Gaussian distribution} \quad P(X = x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$


or “normal”

$$Y \sim \text{binomial}(n, p) \quad Y \text{ has a binomial distribution} \quad P(Y = y|n, p) = \binom{n}{y} p^y (1 - p)^{n-y}$$

*McElreath, “Statistical Rethinking”, <https://xcelab.net/rm/statistical-rethinking/>

Challenge #1

Implement a binomial likelihood in R

$$Y \sim \text{binomial}(n, p) \quad Y \text{ has a binomial distribution} \quad P(Y = y | n, p) = \binom{n}{y} p^y (1 - p)^{n-y}$$

(meaning: sample n events each independently occurring with probability p – this is the probability of getting exactly y of them. Think coin tosses, black or white socks in a drawer, genotypes at a variant with frequency p , ...)

Should look something like this:

```
binomial_likelihood <- function( y, n, p ) {  
  ...computation goes here...  
  return( result )  
}
```

Note: we are most interested in the parameter p , so the first term is not very important for us. See `?choose` in R for a way to compute it.

Pick a value of p (e.g. 0.25) and plot the distribution for $n = 5..13$

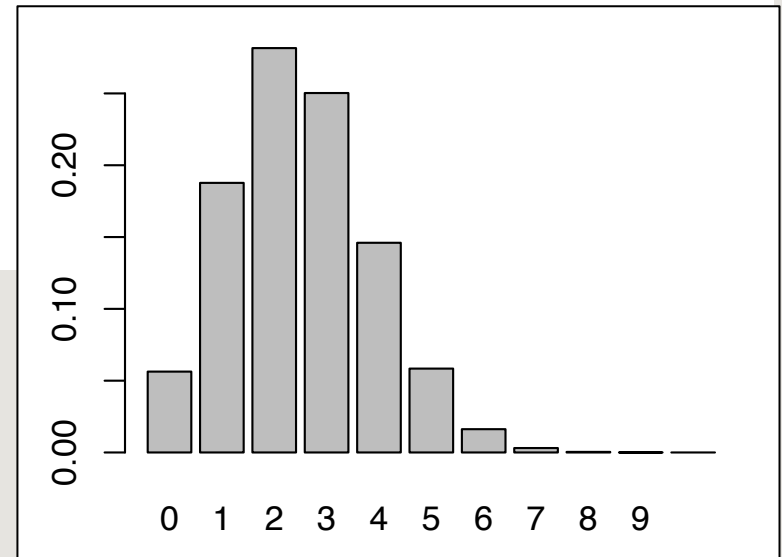
Note: you can use e.g. `par(mfrow = c(3, 3))` before plotting to put multiple plots on one page. Use `barplot()` to make bars.

Challenge #1: solution

$Y \sim \text{binomial}(n, p)$ Y has a binomial distribution $P(Y = y|n, p) = \binom{n}{y} p^y (1 - p)^{n-y}$

```
binomial.likelihood <- function( y, n, p ) {  
  return(  
    choose( n, y ) * p^y * (1-p)^(n-y)  
  ) ;  
}
```

```
n = 10  
p = 0.25  
barplot(  
  binomial.likelihood( 0:n, n, p ),  
  names.arg = 0:n  
)
```



Challenge #1b

Actually it is usually better to work in log space:

$$\log P(Y = y|n, p) = \log \binom{n}{y} + y \log(p) + (n - y) \log(1 - p)$$

Implement it:

```
binomial.ll <- function( y, n, p ) {  
  ...computation goes here...  
  return( result )  
}
```

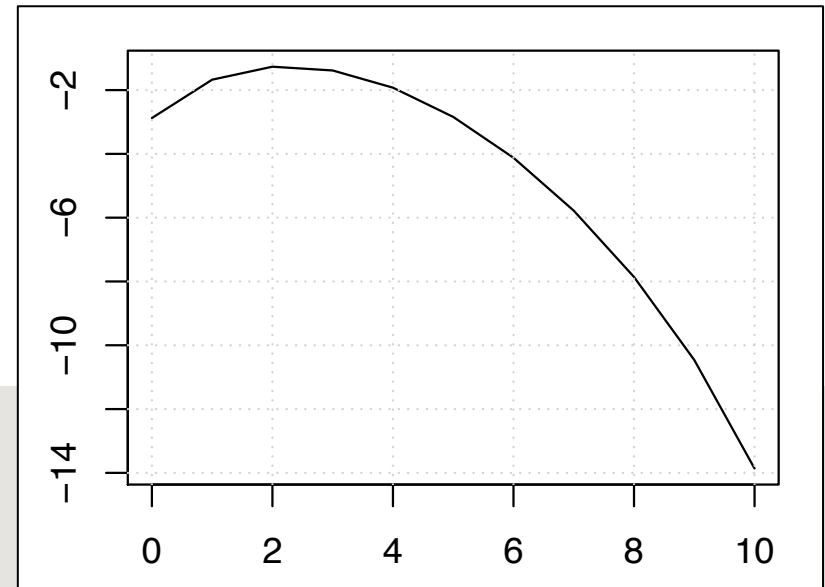
Note: See `?lchoose` in R for computing logarithm of the n choose k term.

Check your function against **`binomial.likelihood`**!

Challenge #1b: solution

$Y \sim \text{binomial}(n, p)$ Y has a binomial distribution $P(Y = y|n, p) = \binom{n}{y} p^y (1 - p)^{n-y}$

```
binomial.ll <- function( y, n, p ) {  
  return(  
    lchoose( n, y ) + y * log(p) + (n-y) * log(1-p)  
  ) ;  
}  
  
n = 10  
p = 0.25  
plot(  
  0:n,  
  binomial.ll( 0:n, n, p ),  
  type = "l"  
)  
grid()
```



Real genetic examples – two blood groups

Is O blood group protective
against severe malaria?

(Band et al 2019 says it is).

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738

Data from MalariaGEN, “Insights into malaria susceptibility from the genomes of 17,000 individuals”, <https://doi.org/10.1101/535898>

Real genetic examples – two blood groups

Is O blood group protective against severe malaria?

(Band et al 2019 says it is)

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738

Data from MalariaGEN, “Insights into malaria susceptibility from the genomes of 17,000 individuals”, <https://doi.org/10.1101/535898>.
(Data from Gambia, Burkina Faso, Ghana,, Malawi, Tanzania, Kenya)

Is Cromer 1 blood group under +ve selection due to malaria?

(Egan et al 2015 implied it might be).

	G allele	C allele
non-malaria-endemic populations* e.g. Europeans	1965	1
malaria-endemic populations* e.g. Africans	707	17

1000 Genomes Project data at rs60822373. c.f. Egan et al Science (2015), <https://doi.org/10.1126/science.aaa3526>

Challenge: interpret these tables.

Real genetic examples – two blood groups

Is O blood group protective against severe malaria?

(Band et al 2019 says it is)

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738

Data from MalariaGEN, “Insights into malaria susceptibility from the genomes of 17,000 individuals”, <https://doi.org/10.1101/535898>.

(Data from Gambia, Burkina Faso, Ghana,, Malawi, Tanzania, Kenya)

$$OR=0.74 \text{ (95\% CI = 0.69 – 0.79)}^*$$
$$P = 3.5 \times 10^{-18}$$

Is Cromer 1 blood group under +ve selection due to malaria?

(Egan et al 2015 implied it might be).

	G allele	C allele
non-malaria-endemic populations* e.g. Europeans	1965	1
malaria-endemic populations* e.g. Africans	707	17

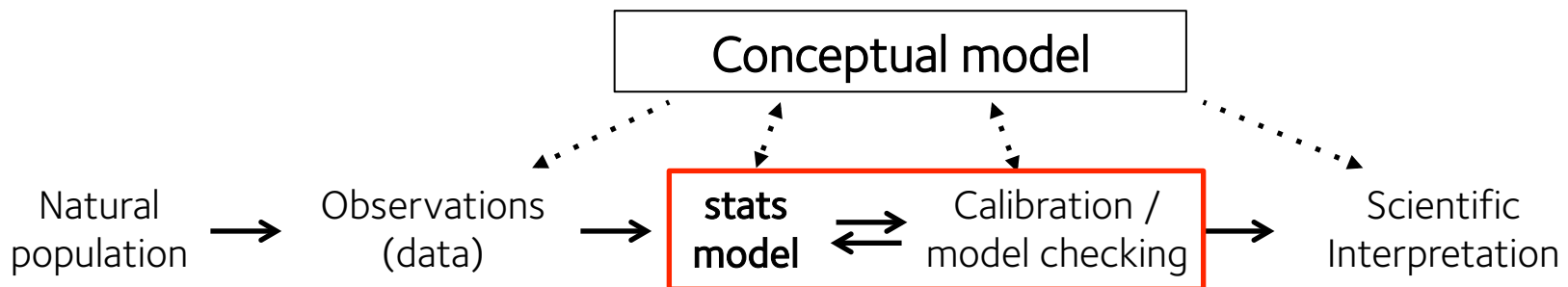
1000 Genomes Project data at rs60822373. c.f. Egan et al Science (2015), <https://doi.org/10.1126/science.aaa3526>

$$OR=47 \text{ (95\% CI = 6 – 356)}^*$$
$$P = 2 \times 10^{-18}$$

* Computed by looking up the formula for standard error of the log odds ratio

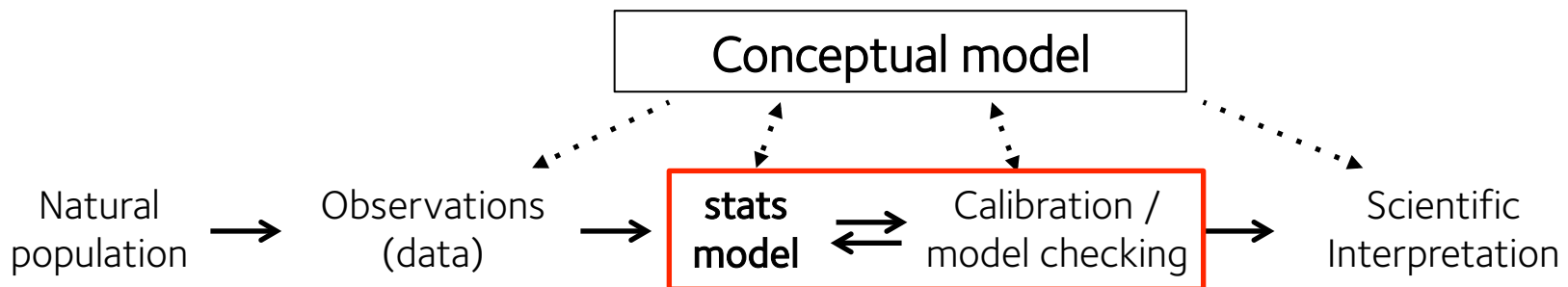
Plan

- Two parts to practical:
- 1: investigate the statistical model for the 2x2 table by implementing it and plotting it.
- 2: leverage genome-wide data to investigate whether the assumed model holds up – does our P-value really reflect our conceptual understanding?



Plan

- Two parts to practical:
 - 1: investigate the statistical model for the 2x2 table by implementing it and plotting it.
 - 2: leverage genome-wide data to investigate whether the assumed model holds up – does our P-value really reflect our conceptual understanding?



Challenge #3: write the loglikelihood for the table

Basic model: binomial sampling in rows.

	A	B
controls	$1-\theta_1$	θ_1
cases	$1-\theta_2$	θ_2

The basic model is binomial sampling in rows. The basic parameterisation is shown on the left (one frequency for controls and a possibly different one for cases.)

```
table.ll <- function( data, params ) {  
  # data is a 2x2 matrix  
  # params[1] =  $\theta_1$  and params[2] =  $\theta_2$   
  ...  
  return( result )  
}
```

Hint: you have already written `binomial.ll()`.

Sanity check for the 1st table: `table.ll(data, c(0.5, 0.5)) = -118.1644`

Sanity check for the 2nd table: `table.ll(data, c(0.5, 0.5)) = -1778.735`

Challenge #4: Solution

	A	B
controls	$1-\theta_1$	θ_1
cases	$1-\theta_2$	θ_2

Would be better to parameterise in terms of an *effect size parameter* that directly lets us reason about effect. We'll use the *logistic function* to rewrite in terms of the *log odds ratio*.

logistic() converts log-odds to probabilities. Its inverse is the **logit** function that computes log odds.

$$\text{logistic}(x) = \frac{e^x}{1 + e^x}$$

$$\text{logit}(p) = \log \frac{p}{1 - p}$$

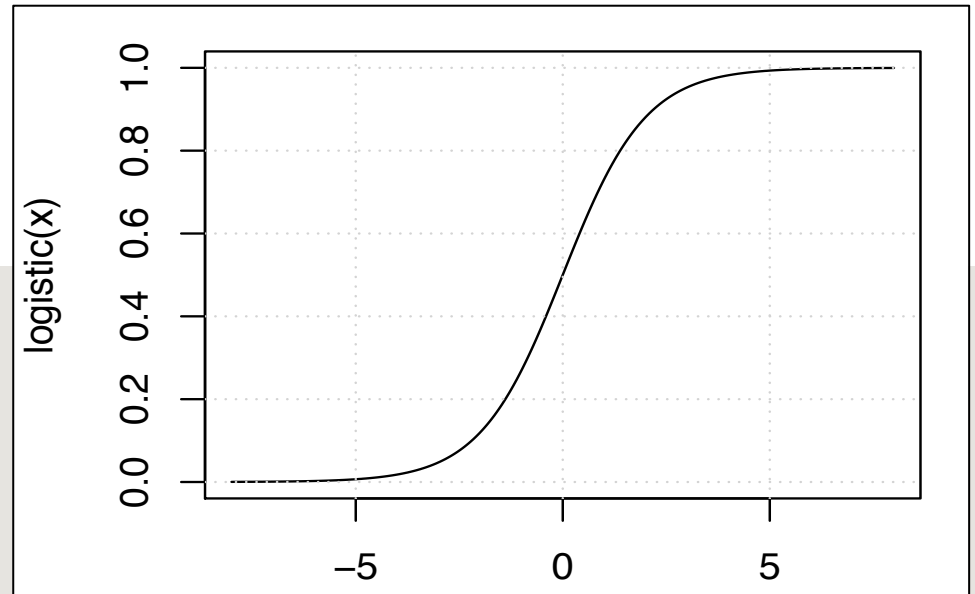
```
logistic <- function( x ) {  
  ...  
}
```

Plot the logistic function.

$$\text{logistic}(x) = \frac{e^x}{1 + e^x}$$

```
logistic <- function( x ) {  
  exp(x) / ( 1 + exp(x))  
}
```

```
x = seq( from = -8, to = 8, by = 0.01 )  
plot(  
  x,  
  logistic(x),  
  type = 'l'  
)  
grid()
```



Challenge #5

	A	B
controls	$1-\theta_1$	θ_1
cases	$1-\theta_2$	θ_2

To reparameterise, set:

$$\theta_1 = \text{logistic}(\mu)$$

$$\theta_2 = \text{logistic}(\mu + \beta)$$

So that:

μ is the log-odds of outcome B in controls

β is the log of the odds ratio (as you can calculate)

```
table.ll.reparameterised <- function( data, params ) {  
  theta1 = logistic( ... )  
  theta2 = logistic( ... )  
  ...  
  return( ... )  
}
```

Challenge #5 solution

$$\theta_1 = \text{logistic}(\mu)$$

$$\theta_2 = \text{logistic}(\mu + \beta)$$

```
reparameterised.table.ll <- function( data, params ) {  
  thetas = c(  
    logistic( params[1] ),  
    logistic( params[1] + params[2] )  
  )  
  return( table.ll( data, thetas ) )  
}
```

Fitting the model for our examples

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738

	G allele	C allele
non-malaria- endemic popns	1965	1
malaria- endemic popns*	707	17

```
data = matrix( c( ... ), byrow = T, ncol = 2 )
fit = optim(
  par = c( 0, 0 ),
  fn = function( params ) {
    reparameterised.table.ll( data, params )
  },
  # tell optim() to maximise, not minimise.
  control = list(
    fnscale = -1,
    trace = TRUE
  )
)

maximum.likelihood.estimate = fit$par
```

All code is reusable! (Put it in a function)

```
find.mle <- function( data ) {  
  fit = optim(  
    par = c( 0, 0 ),  
    fn = function( params ) {  
      reparameterised.table.ll( data, params )  
    },  
    # tell optim() to maximise, not minimise.  
    control = list(  
      fnscale = -1,  
      trace = TRUE  
    )  
  )  
  return( fit$par )  
}  
  
maximum.likelihood.estimate = find.mle( data )
```

Sanity check: it should be that $\exp()$ of the 2nd parameter is the original odds ratio, i.e. 0.74 for the O blood group table or 47 for the Cromer blood group table .

Challenge #6: plot the likelihood function

Plot the likelihood function:

1: Simple version: pretend the baseline log odds μ (i.e. the first parameter) is fixed – we'll focus on β . Plot the log-likelihood in a range of β values around the estimated maximum likelihood. Also plot the (non-logged) likelihood for comparison.

2: Complex version: plot the full bivariate loglikelihood – this can be either a heatmap, by contours, or as multiple “slices”.)

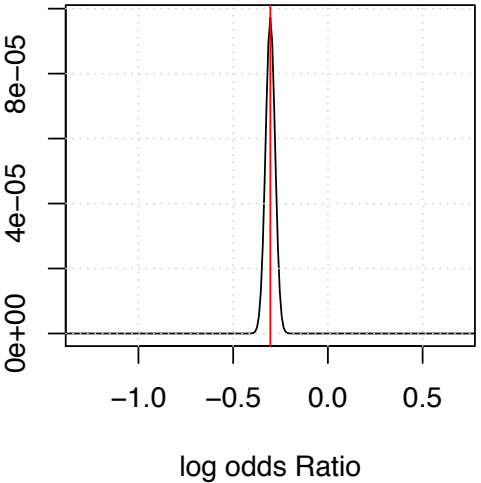
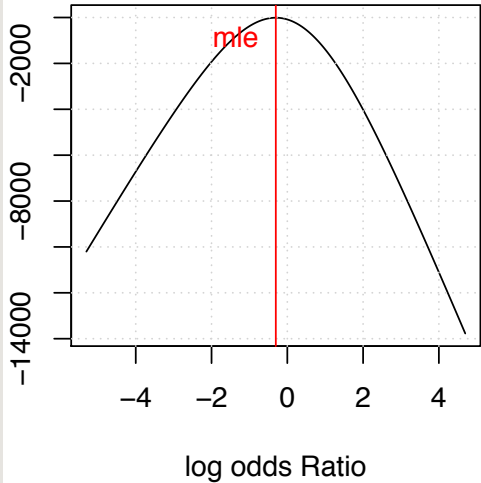
```
plot.ll <- function( data, maximum.likelihood.estimate ) {  
  mu_hat = maximum.likelihood.estimate[1]  
  beta_hat = maximum.likelihood.estimate[2]  
  range = beta_hat + c( -10, 10 ) # may need to adjust  
  at = seq( from = range[1], to = range[2], by = 0.01 )  
  ll.evaluations = sapply(  
    at,  
    function(x) {  
      reparameterised.table.ll(  
        data,  
        c( mu_hat, x )  
      ) }  
  )  
  ...  
}
```

Challenge #6: plot the likelihood function

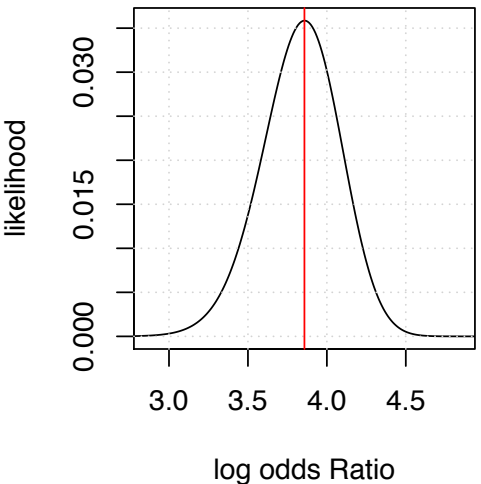
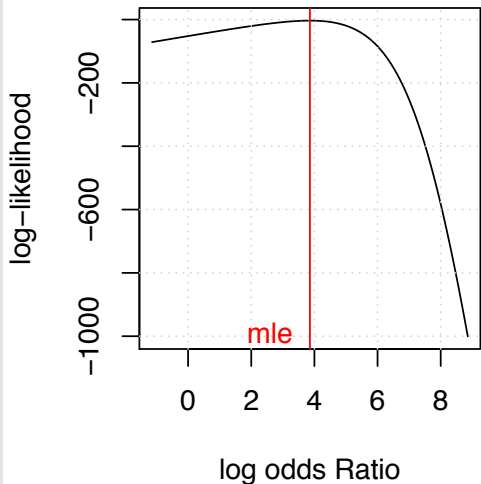
```
plot.ll <- function( data, maximum.likelihood.estimate ) {  
  mu_hat = maximum.likelihood.estimate[1]  
  beta_hat = maximum.likelihood.estimate[2]  
  range = beta_hat + c( -10, 10 ) # may need to adjust  
  at = seq( from = range[1], to = range[2], by = 0.01 )  
  ll.evaluations = sapply(  
    at,  
    function(x) {  
      reparameterised.table.ll(  
        data,  
        c( mu_hat, x )  
      )  
    }  
  )  
  plot(  
    at,  
    ll.evaluations,  
    type = "l",  
    xlab = "log odds Ratio", ylab = "log-likelihood"  
  )  
  grid()  
  
  abline( v = beta_hat, col = "red" )  
  text( beta_hat, -1000, "mle", col = "red", pos = 2 )  
}
```

Solutions

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738



	G allele	C allele
n.e popns	1965	1
endemic popns*	707	17



Challenge #7

The behaviour we have just seen is typical for statistical likelihoods. The functions you've implemented above and some additional functions to simulate data are in the file `solutions.R` on the github page. You need:

```
find.mle()  
plot.ll()  
simulate.tables()  
plot.simulated.tables()
```

Create some simulated data with

```
tables = simulated.tables()
```

(What does this simulate? Can you read the code?)

Now make similar plots to the above with:

```
plot.simulated.tables()
```

Asymptotic normality of the likelihood

“Less data”

controls	100	0
cases	100	1

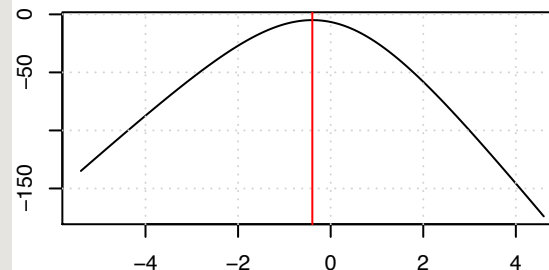
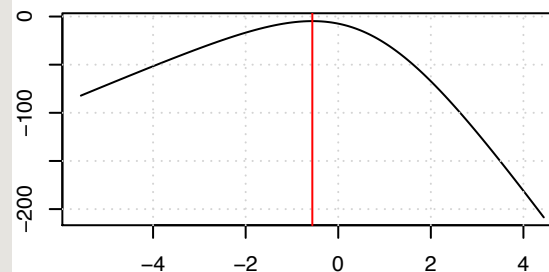
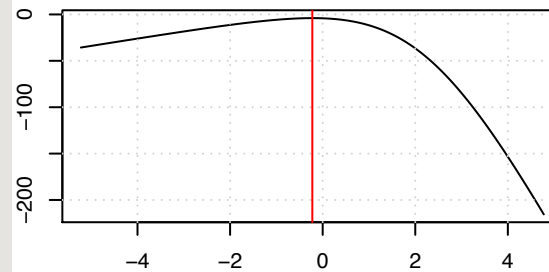
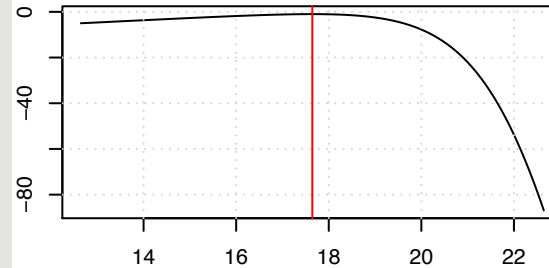
controls	90	10
cases	90	8

controls	65	35
cases	65	20

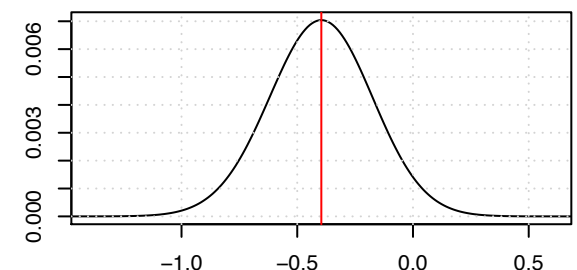
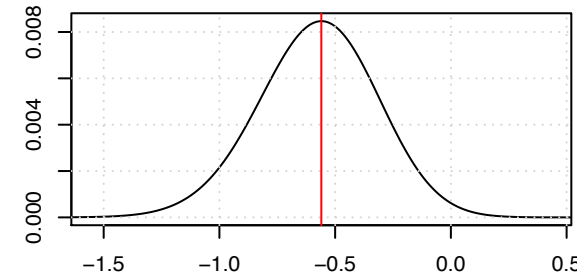
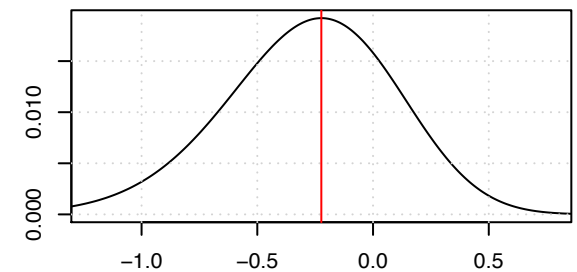
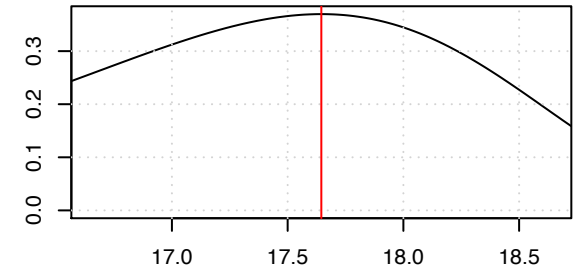
controls	48	52
cases	48	35

“More data”

log-likelihood



likelihood



Asymptotic normality of the likelihood

In short (more next week):

As the amount of data grows:

1. The likelihood function itself becomes approximated by a Gaussian distribution in a neighborhood of the maximum likelihood estimate.
2. The maximum likelihood estimate itself becomes approximately distributed as a Gaussian around the true parameter value.

Homework challenge (if keen): update the plots above to overlay a Gaussian distribution. (Use `dnorm()` or implement it yourself, and `points()` to overlay on the plots). For the standard deviation, use the 'standard' formula:

$$se = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$$

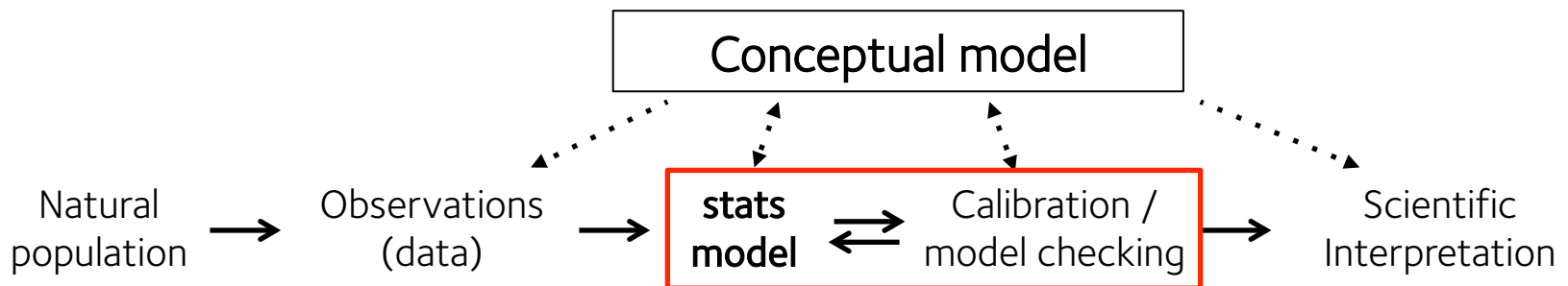
where a, b, c, d are the 4 table entries.

Summary

- When there is lots of data, and all the categories are at reasonably high frequency, the likelihood becomes approximately Gaussian. (The log-likelihood becomes approximately quadratic)
- In this case we can use the standard tools – data estimates of odds ratios, ‘standard’ formulae for standard errors, chi-squared tests etc. to get approximately right results.
- When there is less data (fewer samples, lower frequency predictors) then this starts to break down and we need to be more cautious to make convincing analysis. Things like adding prior information, using numerical integration, or assessing distributions via simulation can help.
- If in doubt plot the likelihoods.

Second part of practical – calibration against genome-wide data

- Two parts to practical:
- 1: investigate the statistical model for the 2x2 table by implementing it and plotting it.
- 2: leverage genome-wide data to investigate whether the assumed model holds up – does our P-value really reflect our conceptual understanding?



Second part of practical – calibration against genome-wide data

Is O blood group protective against severe malaria?

(Band et al 2019 says it is)

	non o bld. grp.	o bld.grp
controls	3420	3233
severe malaria cases	3925	2738

Data from MalariaGEN, “Insights into malaria susceptibility from the genomes of 17,000 individuals”, <https://doi.org/10.1101/535898>.

(Data from Gambia, Burkina Faso, Ghana,, Malawi, Tanzania, Kenya)

Is Cromer 1 blood group under +ve selection due to malaria?

(Egan et al 2015 implied it might be).

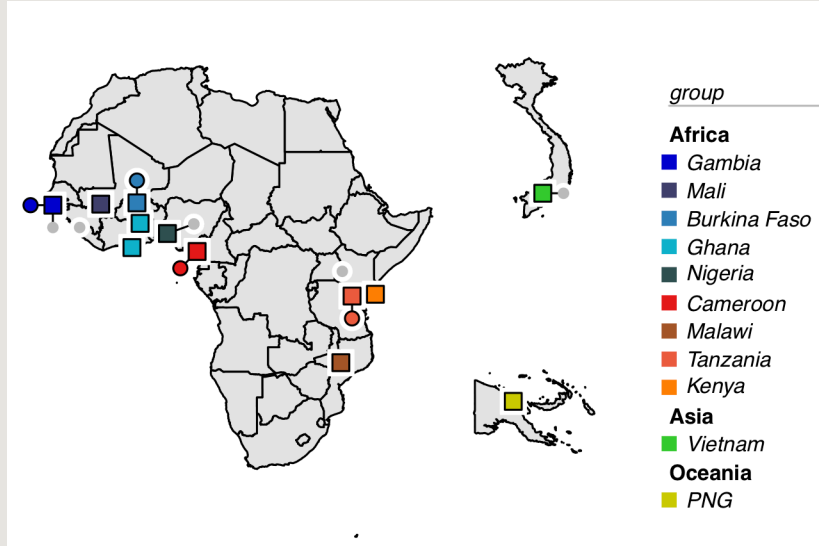
	G allele	C allele
non-malaria-endemic populations* e.g. Europeans	1965	1
malaria-endemic populations* e.g. Africans	707	17

1000 Genomes Project data at rs60822373. c.f. Egan et al Science (2015), <https://doi.org/10.1126/science.aaa3526>

 o_bld_group_genome_wide_comparison.md

 1000Genomes_genome_wide_comparison.md

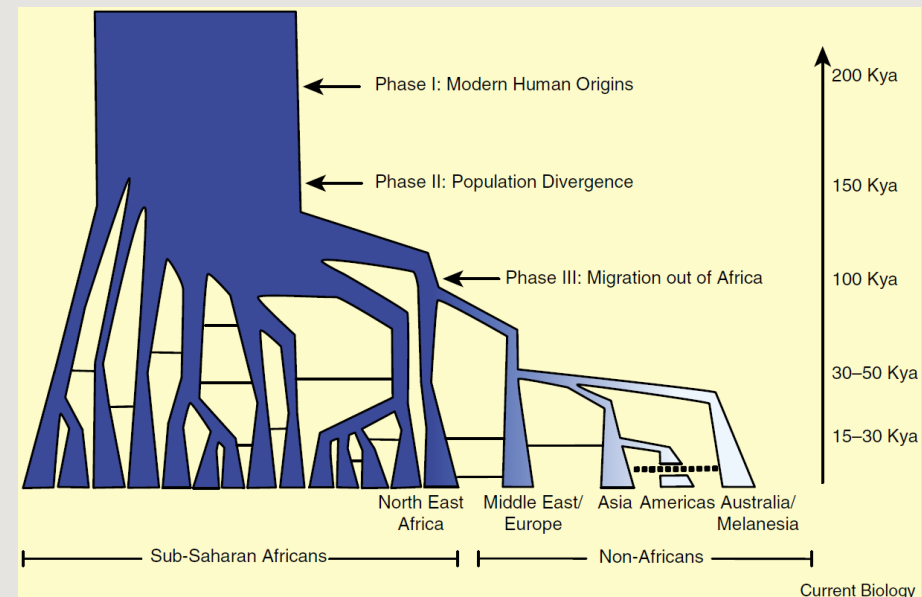
Second part of practical – calibration against genome-wide data



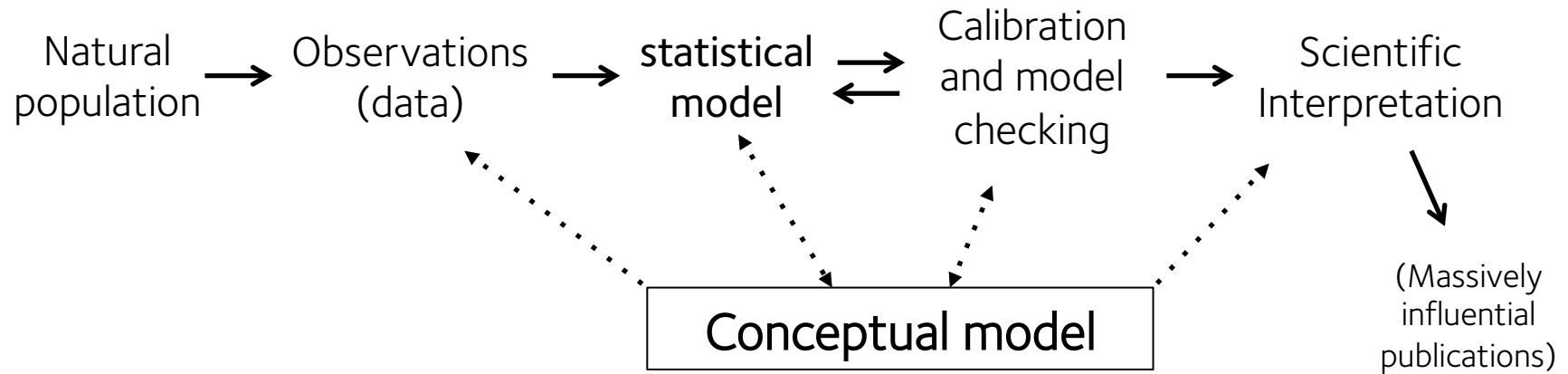
Conceptual model seems broadly appropriate for O blood group data. Most SNPs with similar frequency have smaller odds ratios and are more or less consistent with no true effect (with a 'fat tail' that we should investigate further).

Conceptual model seems completely inappropriate for the Cromer blood group data – it would suggest half the genome has been under selection! What's going on?

Many alleles were lost during out-of-Africa bottleneck!



Conclusion



Next week: logistic regression, summary statistics, meta-analysis, bayesian statistics and preparation for GWAS.

+ Bayesian analysis taught by Andre Python.