

Introduction to Bayesian Statistics

Or: how I learned to stop worrying and love priors



Institute
of Zoology

Intro

- ❖ **Henry Häkkinen**

- ❖ Post-Doc studying the effects of climate change on seabirds and how to plan effective conservation.
Latterly branched out into impacts of renewable energy on biodiversity.



Content for today

- ❖ What is a “Bayesian” approach?
- ❖ A brief explanation of Bayes’ theorem
- ❖ Building a Bayesian model
 - ❖ Model design
 - ❖ Priors
 - ❖ Running a model
 - ❖ Model checks
 - ❖ Interpreting output
- ❖ Bayesian versus Frequentist statistics
- ❖ Practical: running example models in R!

What is a “Bayesian” approach?

- ❖ An alternative type of statistical inference than that of frequentist statistics.
- ❖ We use data and our prior knowledge of a system to inform our statistical model, which then calculates the probability of patterns and relationship between variables



Bayes' Theorem

- ❖ The Bayes theorem describes the probability of event A conditional on event B (the probability of A after B has already occurred).
- ❖ This is calculated using the probability of B conditional on A (what are the conditions of B that A occurs under) and the two separate probabilities of the events A and B.

$$P(A|B)$$

Bayes' Theorem

- ❖ The Bayes' theorem in practice:
- ❖ *"The serum test screens pregnant women for babies with Down's syndrome. The test is a very good one, but not perfect. Roughly 1% of babies have Down's syndrome. If the baby has Down's syndrome, there is a 90% chance that the result will be positive. If the baby is unaffected, there is still a 1% chance that the result will be positive. A pregnant woman has been tested and the result is positive."*
- ❖ What is the chance that her baby actually has Down's syndrome?
- ❖ 45%! This is non-intuitive, but the knowledge of the general probability of prevalence has revealed some interesting information!

Bayes' Theorem

- ❖ In the previous example: What is the probability that a patient has a disease (A) given that they tested positive for that disease (B)?
 - ❖ $P(A|B)$ is the probability that they have the disease, given that they tested positive. The *posterior*
 - ❖ $P(B)$ is the probability that someone tests positive, whether or not they have the disease. The *marginal probability*
 - ❖ $P(A)$ is the probability they actually have the disease (the overall incidence rate of a disease in the population). The *prior*.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Graphical Summary

- ❖ A Bayesian approach combines prior beliefs and evidence to form a posterior belief.

Source: <https://ourcodingclub.github.io/tutorials/brms/>

Wait, what are frequentist statistics?

- ❖ Probably most, if not all, statistical tests you have done up until this point:
 - ❖ T-tests, linear models, GLMs, GLMMs etc.
 - ❖ Quick test: if it has a p-value it's probably frequentist
- ❖ In a frequentist approach we assume there is an overall global distribution, from which our data are just samples
 - ❖ E.g. normal, binomial, Poisson etc.
- ❖ Frequentist statistics assess:
 - ❖ Whether the data fits our assumed distribution
 - ❖ Whether our data differs from a “null” distribution for hypothesis testing

Frequentist versus Bayesian

- ❖ Both are valid approaches for almost every statistical question, they just work differently
- ❖ You run a linear regression using a frequentist or Bayesian approach and they will likely give the same answer.
- ❖ Both of the following models are linear regressions, $y \sim x$, where alpha is the intercept and beta the slope estimate

```
variable   mean median    sd    mad      q5     q95 rhat
  lp__ -40.58 -40.27 1.20  0.98 -42.90 -39.26 1.00
alpha    8.65   8.65 0.25  0.24   8.25   9.06 1.00
beta     0.02   0.02 0.01  0.01   0.01   0.03 1.00
sigma   1.08   1.07 0.09  0.09   0.94   1.24 1.00
```

```
call:
lm(formula = data_lin$y ~ data_lin$x)

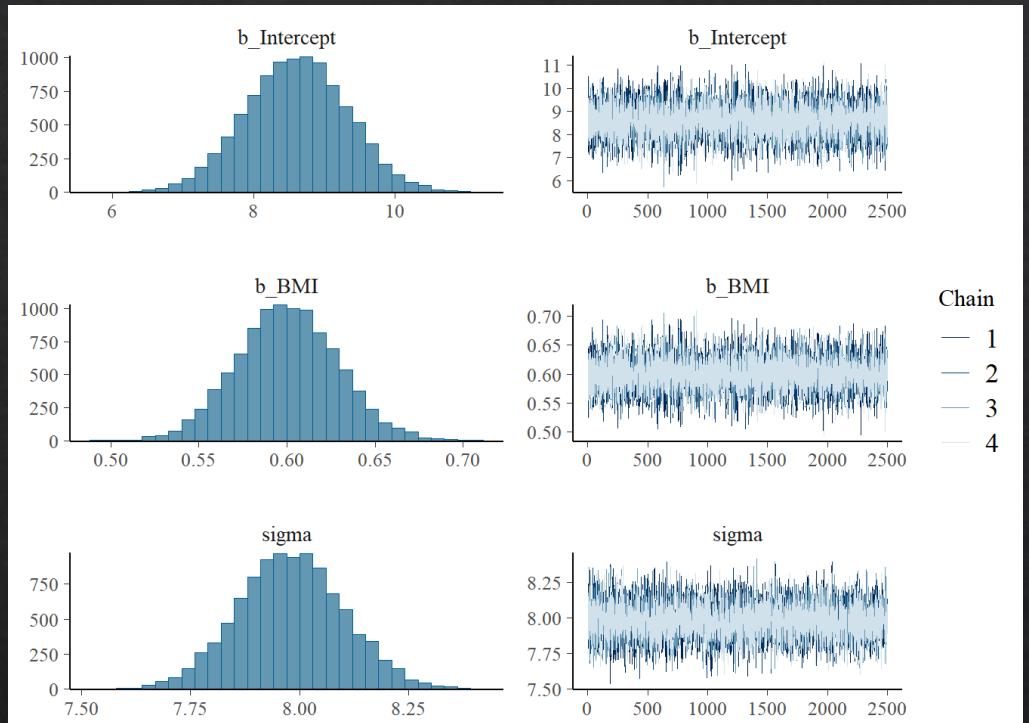
Residuals:
    Min      1Q  Median      3Q     Max 
-2.75887 -0.75660  0.04868  0.82874  2.22634 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 8.651608  0.249640 34.656 < 2e-16 ***
data_lin$x  0.022055  0.006155  3.583 0.000628 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.063 on 69 degrees of freedom
Multiple R-squared:  0.1569,    Adjusted R-squared:  0.1447 
F-statistic: 12.84 on 1 and 69 DF,  p-value: 0.000628
```

Frequentist versus Bayesian

- ❖ Bayesian approaches do not have:
 - ❖ Point estimates for your parameters
 - ❖ P-values (though they can be bootstrapped)
 - ❖ Residuals (though they can be approximated)
 - ❖ Degrees of freedom (sample size compared to model complexity is much less of an issue)
 - ❖ Less separation between fixed and random effects



variable	mean	median	sd	mad	q5	q95	rhat
1p_	-40.58	-40.27	1.20	0.98	-42.90	-39.26	1.00
alpha	8.65	8.65	0.25	0.24	8.25	9.06	1.00
beta	0.02	0.02	0.01	0.01	0.01	0.03	1.00
sigma	1.08	1.07	0.09	0.09	0.94	1.24	1.00

Part 1: Data and model design

Data and Model Design

- ❖ We are going to run a simple example to illustrate a simple Bayesian modelling process



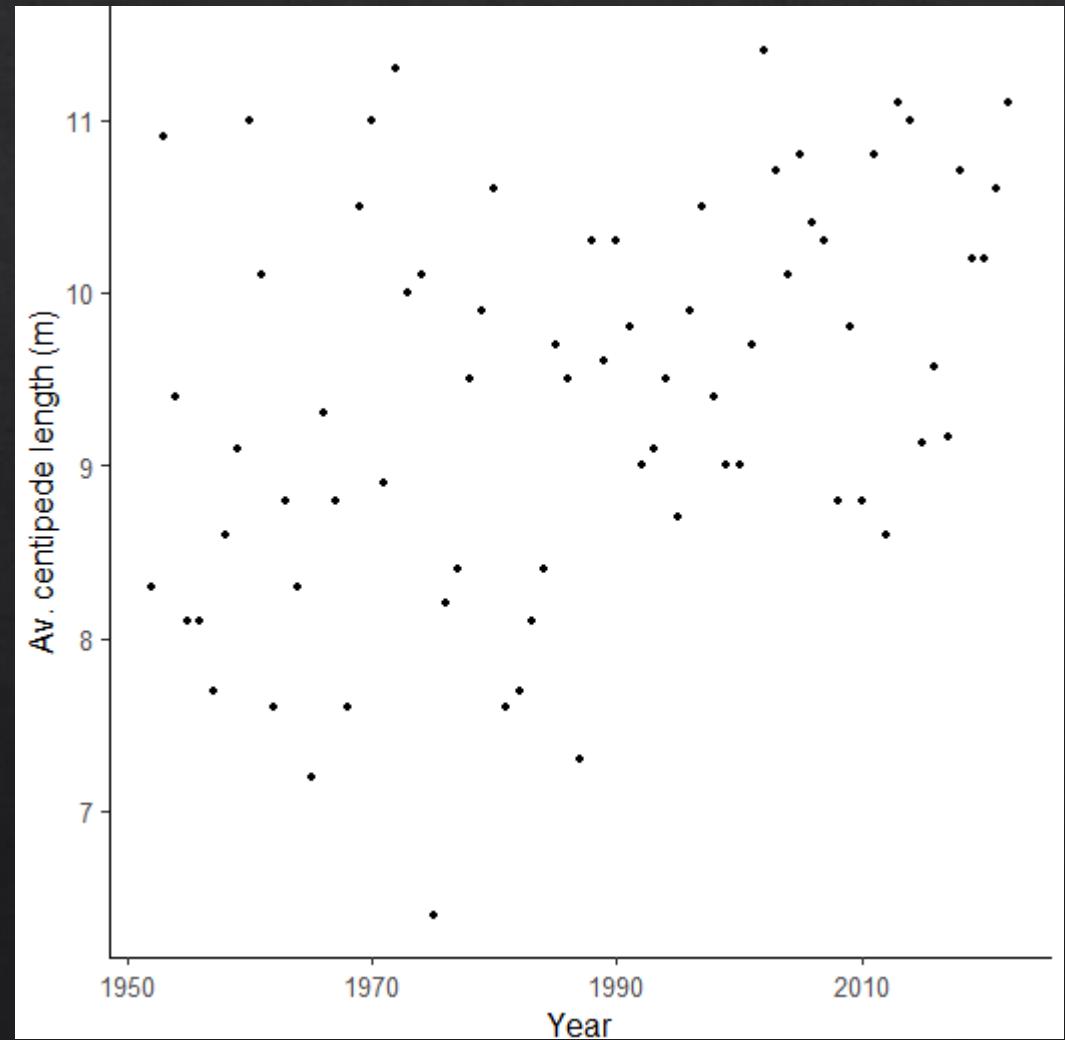
Example Data: Mutated Centipedes

- ❖ Noddy-over-sands had a mysterious chemical leak in the 1950s. The cleanup was long and difficult but was declared safe by the government. However, local ecologists have noticed that centipedes and millipedes in the area of the spill are much larger than expected. They want to analyse past data and see if centipedes nowadays are larger than those of the past.

Source: Fabrizio Conte/ Shutterstock

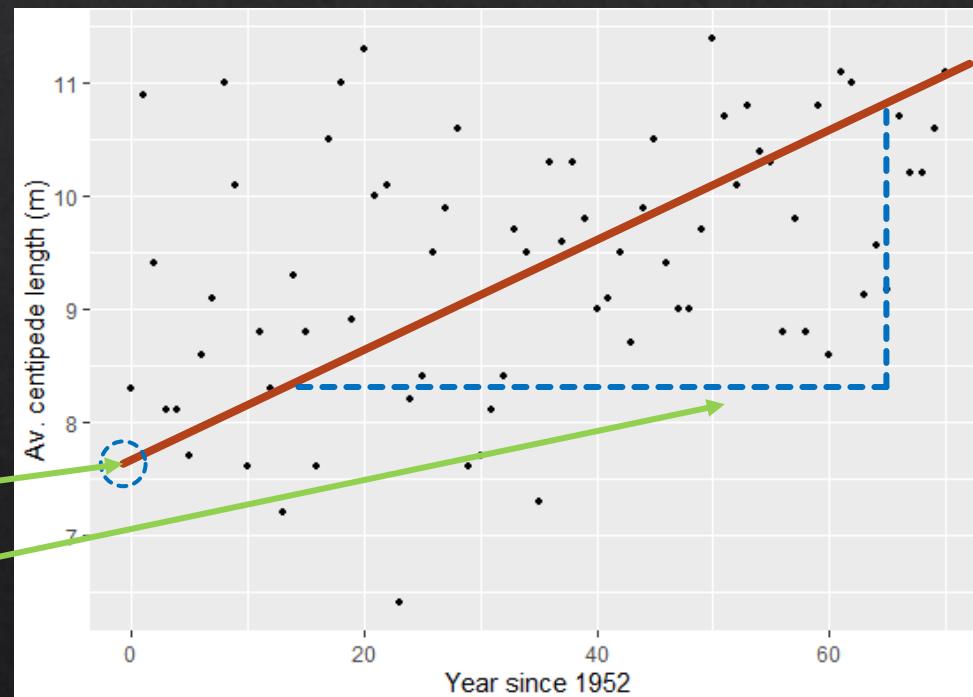
Example Model: Mutated Centipedes

- ◊ A frequentist model requires a distribution to use as its basis. So do Bayesian models
 - ◊ Common examples are: normal, binomial, Poisson, negative binomial, beta, gamma...
- ◊ We use this to model a relationship between predictors and response variables.
- ◊ Always plot your data!
- ◊ What kind of distribution does this look like?



Example Model: Mutated Centipedes

- ❖ This looks linear to me! I could use a normal linear regression with a frequentist approach, but I've decided to use a Bayesian approach.
- ❖ What does a linear model look like?
- ❖ $y \sim a(x) + b \rightarrow y = \alpha + \beta x,$
 - ❖ α is the intercept
 - ❖ β is the slope of the line

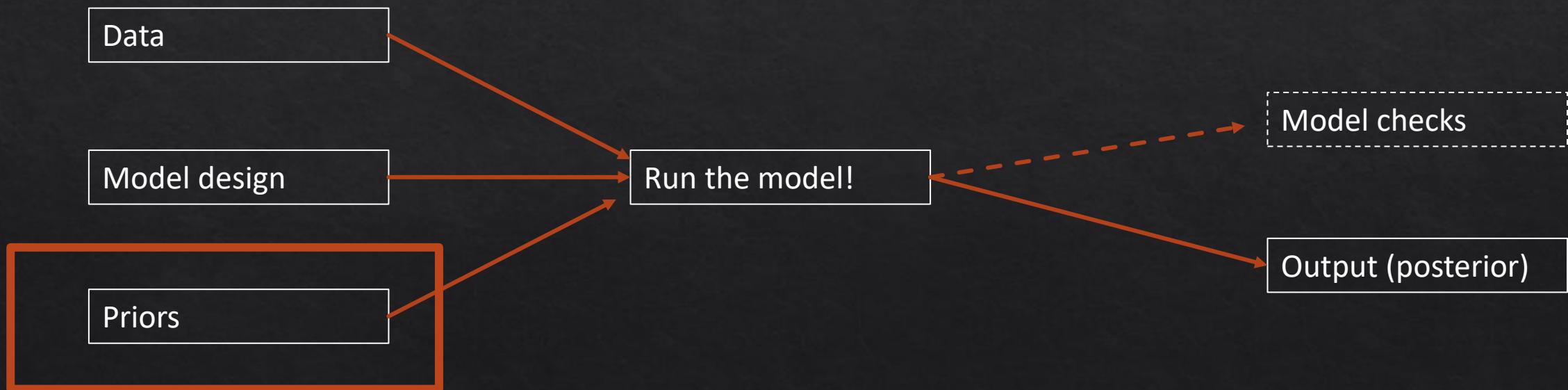


$$y_i = \alpha + \beta x_i + \epsilon_i$$

Part 1: Conclusion

- ❖ We are going to make a linear regression.
 - ❖ We assume the equations linking x (predictor) and y (response) is a linear equation
 - ❖ We have two parameters: alpha (intercept) and beta (slope)
- ❖ Using an explicit Bayesian approach for this is a little like hitting a moth with a sledgehammer, but simple examples are a good way to learn!

Part 2: Priors



What is a prior?

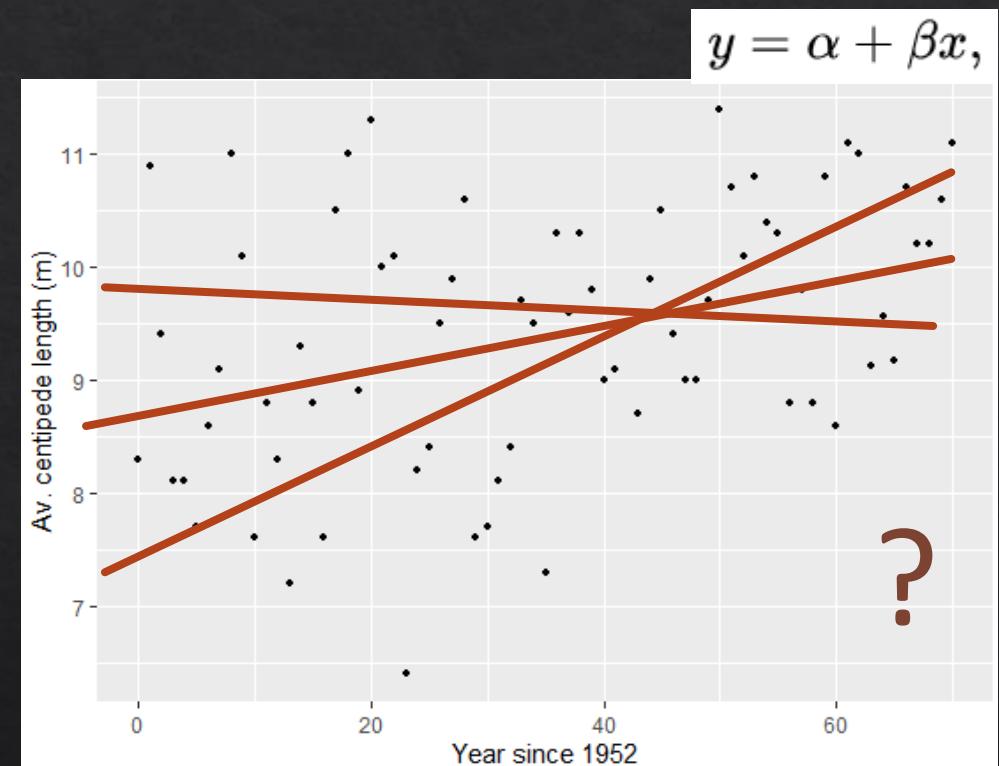
- ❖ Relevant information we should incorporate about this system.
It is our prior expectation.
- ❖ If this is not a new study system, there may be information
about what you expect to see
 - ❖ For example, mortality rate in a wildlife population can be
estimated from previous work
- ❖ Priors provide the model information to use, and inform it what
are the expected results and estimates for our parameter.

What is a prior?

- ❖ Priors can seem odd at first because they are not objective observations, but subjective decisions.
- ❖ You can make priors flexible so if they are wrong, then your model can adjust and update the priors.
- ❖ Think of priors as a good starting point for your model, not as prescriptive.
- ❖ They are more important when you have a small sample size, as they have a larger influence over the posterior distribution

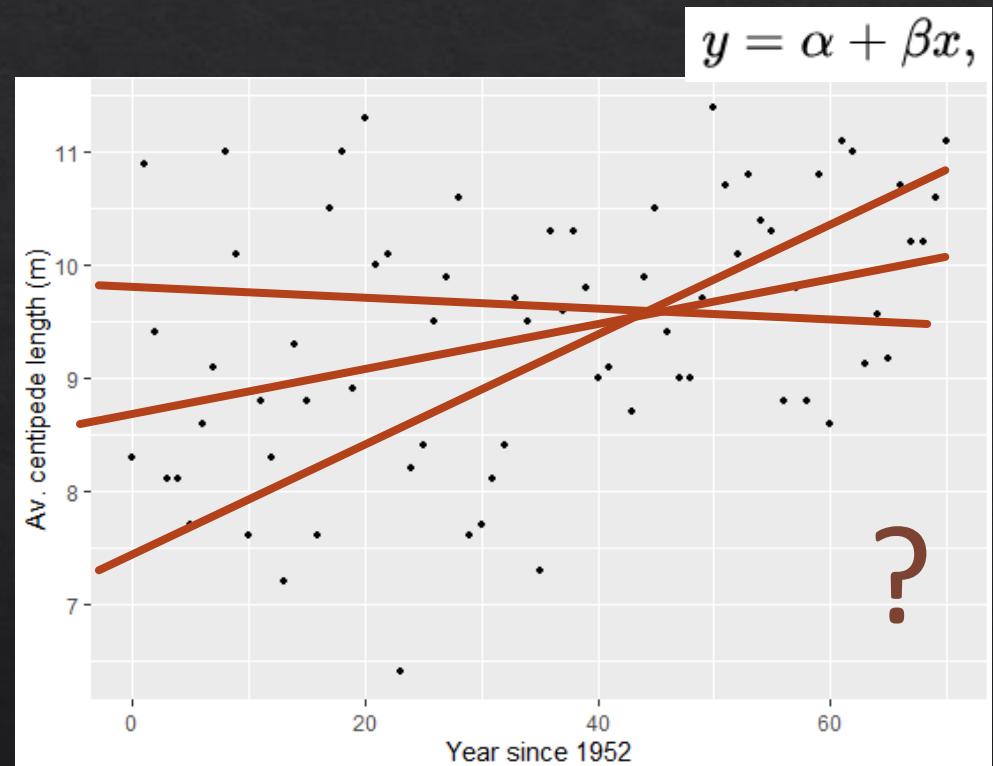
Example Priors: Mutated Centipedes

- ❖ We have two parameters, so we need two priors.
- ❖ What is a reasonable expectation for our intercept (alpha)?
 - ❖ It cannot be below 0, otherwise hard to tell. So we should set a positive prior.
 - ❖ Otherwise it could be anywhere between 6-12 (and probably in the middle)



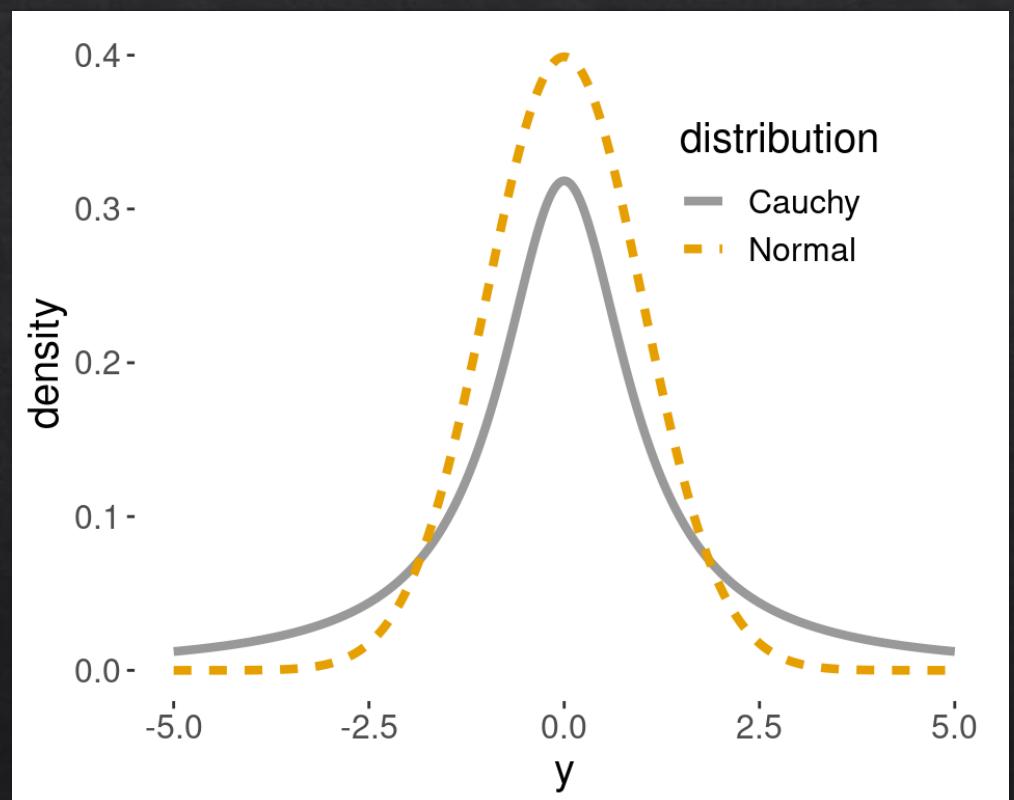
Example Priors: Mutated Centipedes

- ❖ We have two parameters, so we need two priors.
- ❖ What is a reasonable expectation for our slope (beta)?
 - ❖ We have no expectation on positive or negative slope (null hypothesis is slope is 0), but it could be either
 - ❖ So we set a prior with a centre point of 0, but with the flexibility to move to positive or negative.
 - ❖ But how much flexibility?

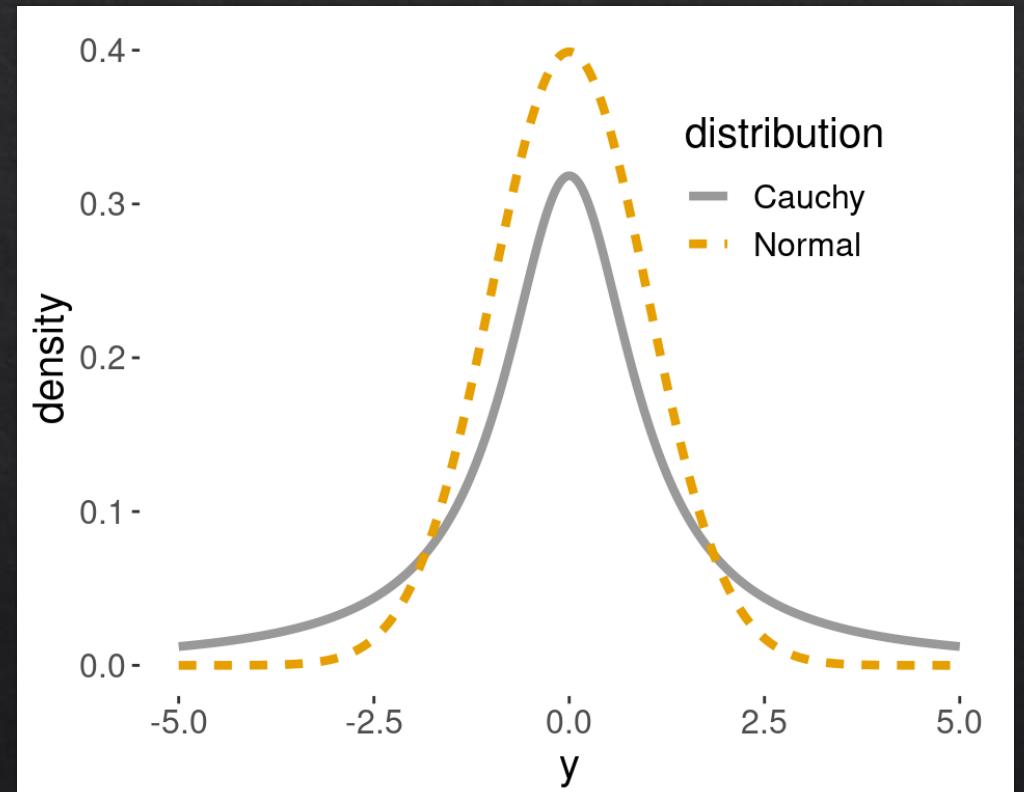
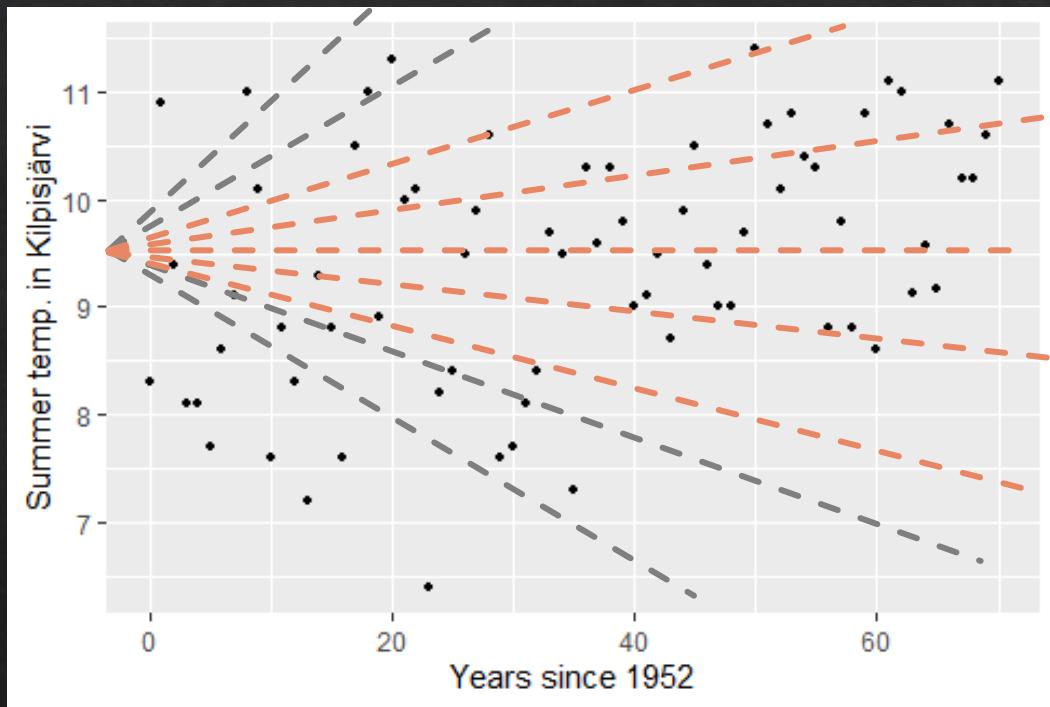


Defining a prior

- ❖ For linear regressions we often have a null hypothesis that there is no relationship between a predictor and the response variable (i.e. slope is 0).
- ❖ In which case we might assume our β prior is around 0, but might be positive and negative
- ❖ We can restrict this by setting a limit on how positive or negative it can be.
- ❖ A strict limit is referred to as “strongly informative” or “enthusiastic” prior
- ❖ A loose limit with a lot of flexibility is referred to as a “weakly informative” or “skeptical” prior

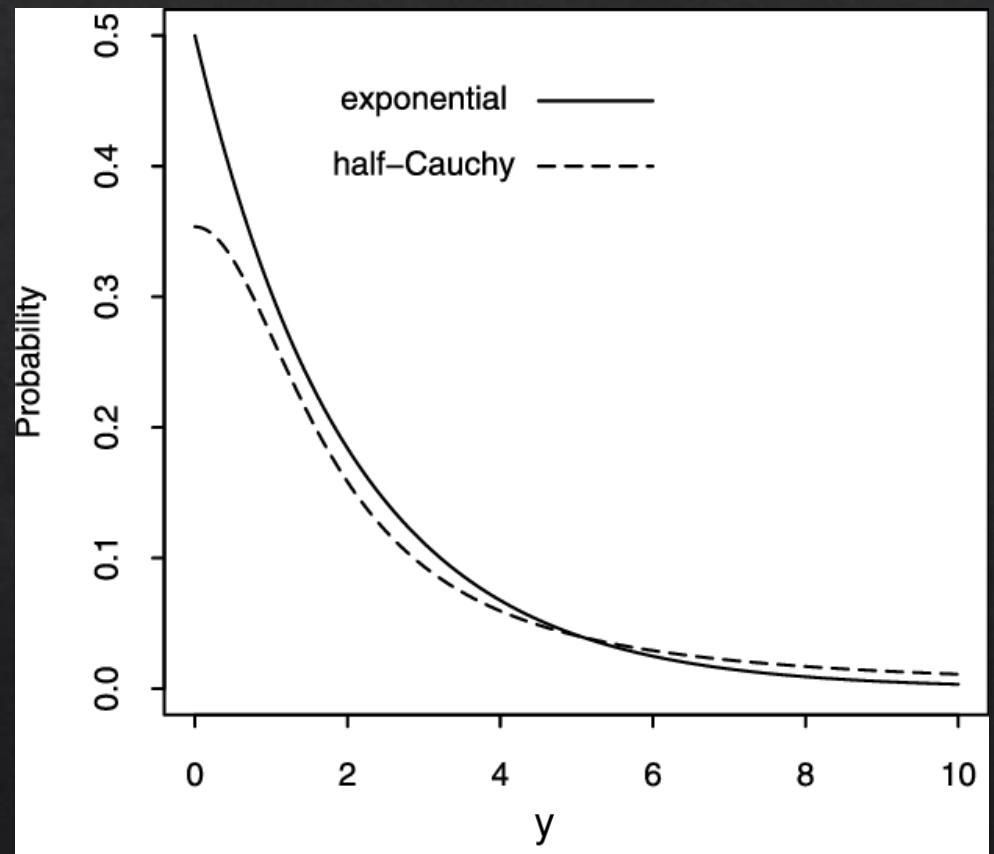


Defining a prior



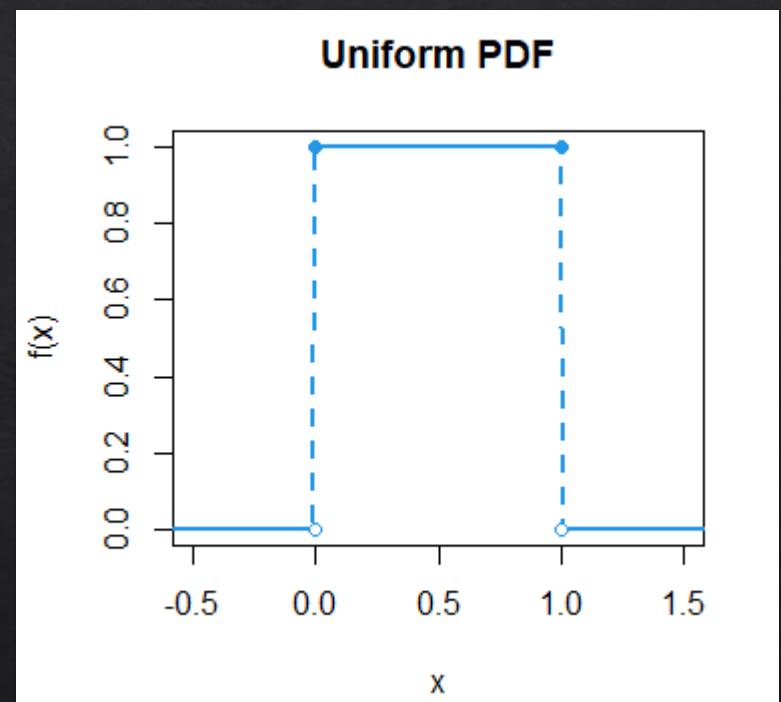
Defining a prior

- ❖ In other cases we may expect a parameter to always be positive, but likely to be close to 0 if the null hypothesis is true.
- ❖ And so on! There are many types of different distributions for different applications.

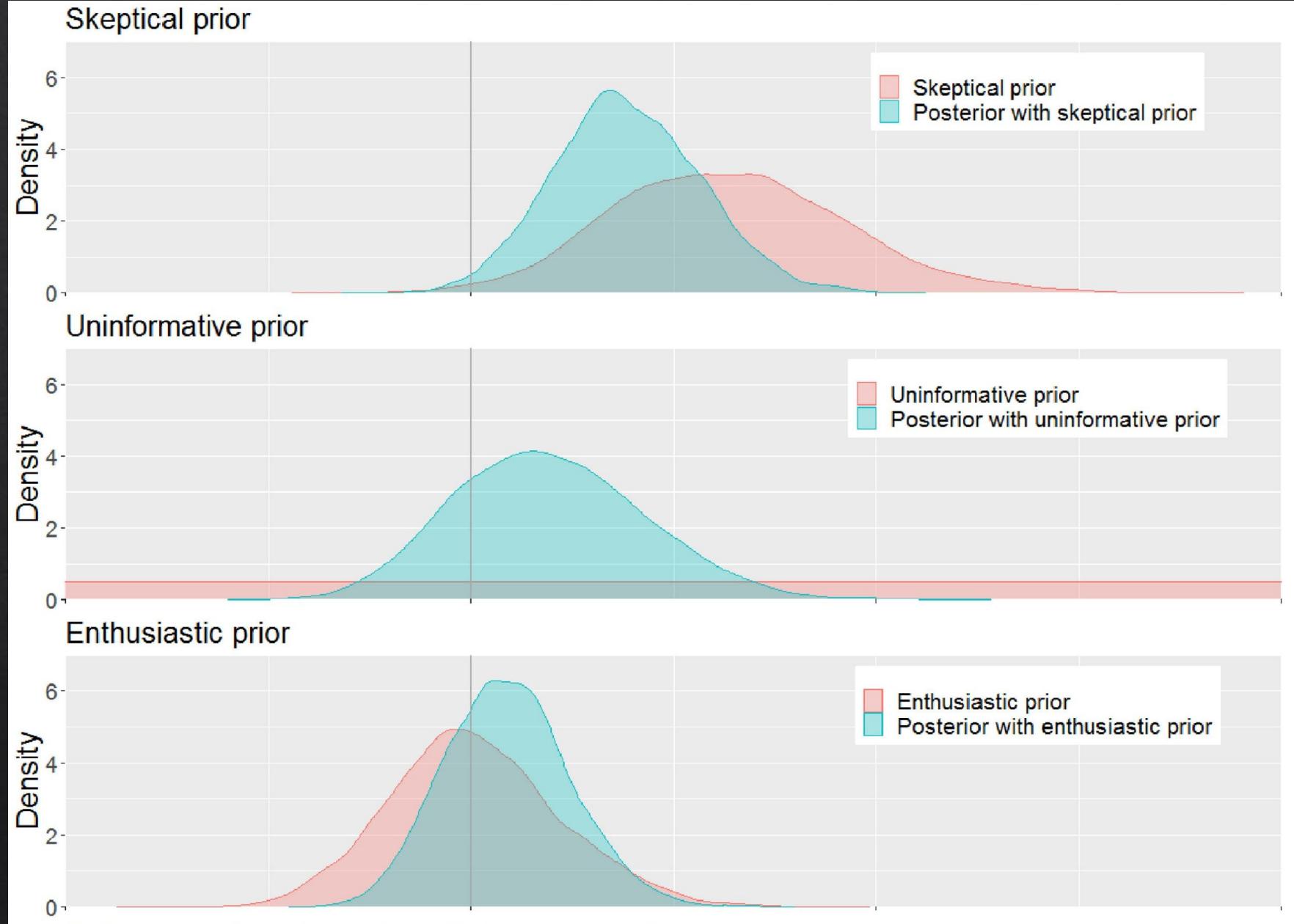


Uninformative priors

- ❖ If we have no prior information about the system we can set an “uninformative” prior
- ❖ Many people favour this as it appears to be “neutral”, but beware! It can bias results more than an informative prior!
- ❖ It can also cause models to be a lot slower as it has more parameter space to check
- ❖ Common sense will often tell us some results are impossible
 - ❖ E.g. cannot have a negative mortality or growth rate!



- ❖ Your choice of priors can affect your results.
Always make sure you can defend your priors.



The Bayesian Approach: Example

- ❖ It is up to us on how restrictive or flexible the priors are
- ❖ The smaller the sample size, the more important your priors are!
- ❖ Some packages choose priors for you, or hide them altogether. Be VERY careful when using these.

Part 3: Running the Model



Build and run the model

- ❖ There are a lot of ways to build models in R!
- ❖ They all use the same mathematical approach, but notation and technical details differ:
 - ❖ Rjags
 - ❖ Rstan (and cmdstanr)
 - ❖ rstanarm
 - ❖ brms
 - ❖ R2OpenBUGS
 - ❖ LaplacesDemon
 - ❖ And probably more....
- ❖ Some are very technical and require a lot of work to run. Some hide a lot of the Bayesian aspects to make it look more like a standard frequentist model. Which you use is up to you, but I really dislike packages that “pretend” they’re frequentist so they look more friendly, I would advise to avoid them

Build and run the model

- ❖ We are going to use rstan (with CmdStanR as a helper package)!
- ❖ It runs C++ in the background, rstan is just the R wrapper on the top
- ❖ It is extremely fast, powerful and flexible, BUT requires some knowledge and work to get it to do stuff
- ❖ Many (if not most) Bayesian packages are based on stan, but are simplified
- ❖ I will teach using stan, because
 - ❖ All the major principles are the same across packages
 - ❖ It forces you to be specific and to understand all the major components of a Bayesian model
 - ❖ If you understand rstan, then many other packages are very similar, but easier!



Build and run the model

- ❖ Complexity varies between packages...
- ❖ Stan model:

```
data {  
    int n; //number of observations in the data  
    array [n] int<lower = 0, upper = 1> admit; //integer of length n for admission decision  
    vector[n] gre; //vector of length n for GRE scores  
    vector[n] gpa; //vector of length n for GPA  
    vector[n] ranking; //vector of length n for school ranking  
}  
  
parameters {  
    real alpha; //intercept parameter  
    vector[3] beta; //vector of coefficients, we want three beta parameters (gre/gpa/ranking)  
    them all manually  
}  
  
model {  
    //linear predictor  
    vector[n] p;  
  
    //linear equation  
    p = alpha + beta[1] * gre + beta[2] * gpa + beta[3] * ranking;  
  
    //prior expectations  
    // NOTE: can comment out for demonstration purposes  
    alpha ~ normal(-1, 1.5);  
    beta ~ normal(0.5, 1.0);  
  
    //likelihood and link function  
    admit ~ bernoulli_logit(p);  
}
```

```
# #run stan and prepare model  
logit.model <- cmdstan_model(stan_file = code_1in)  
  
# sample from our model  
logit.fit <- logit.model$sample(  
    data = model.data,  
    iter_warmup = 500,  
    iter_sampling = 1500,  
    chains = 4,  
    parallel_chains = 4,  
    save_warmup = F,  
    seed = 123,  
    refresh = 500 # print update every 500 iterations)
```

- ❖ Stanarm. Uses all the same structures, but hides them under the hood

```
#a minimal example is just:  
fit1 <- stan_glm(switch ~ dist100,  
                  data = wells,  
                  family = binomial(link = "logit"))
```

Example: Mutated Centipedes

- ❖ Stan has three parts to a model
- ❖ Data: we have to state all the relevant parameters, including sample size! We have to declare what type of data it is.
- ❖ Parameters: we have to declare all the parameters in the model, and an error term
- ❖ Model: we have to declare the formula for our model, and our model priors.
- ❖ For most common models, you can find many examples online.

$$y_i = \alpha + \beta x_i + \epsilon_i$$

```
data {  
    int < lower = 1 > N; // Sample size  
    vector[N] x; // Predictor  
    vector[N] y; // Outcome  
}
```

```
parameters {  
    real alpha; // Intercept  
    real beta; // Slope (regression coefficients)  
    real < lower = 0 > sigma; // Error SD  
}
```

```
model {  
    alpha ~ normal(1, 10); //prior intercept  
    beta ~ normal(1, 0.1); //prior slope  
    y ~ normal(alpha + x * beta , sigma);  
}
```

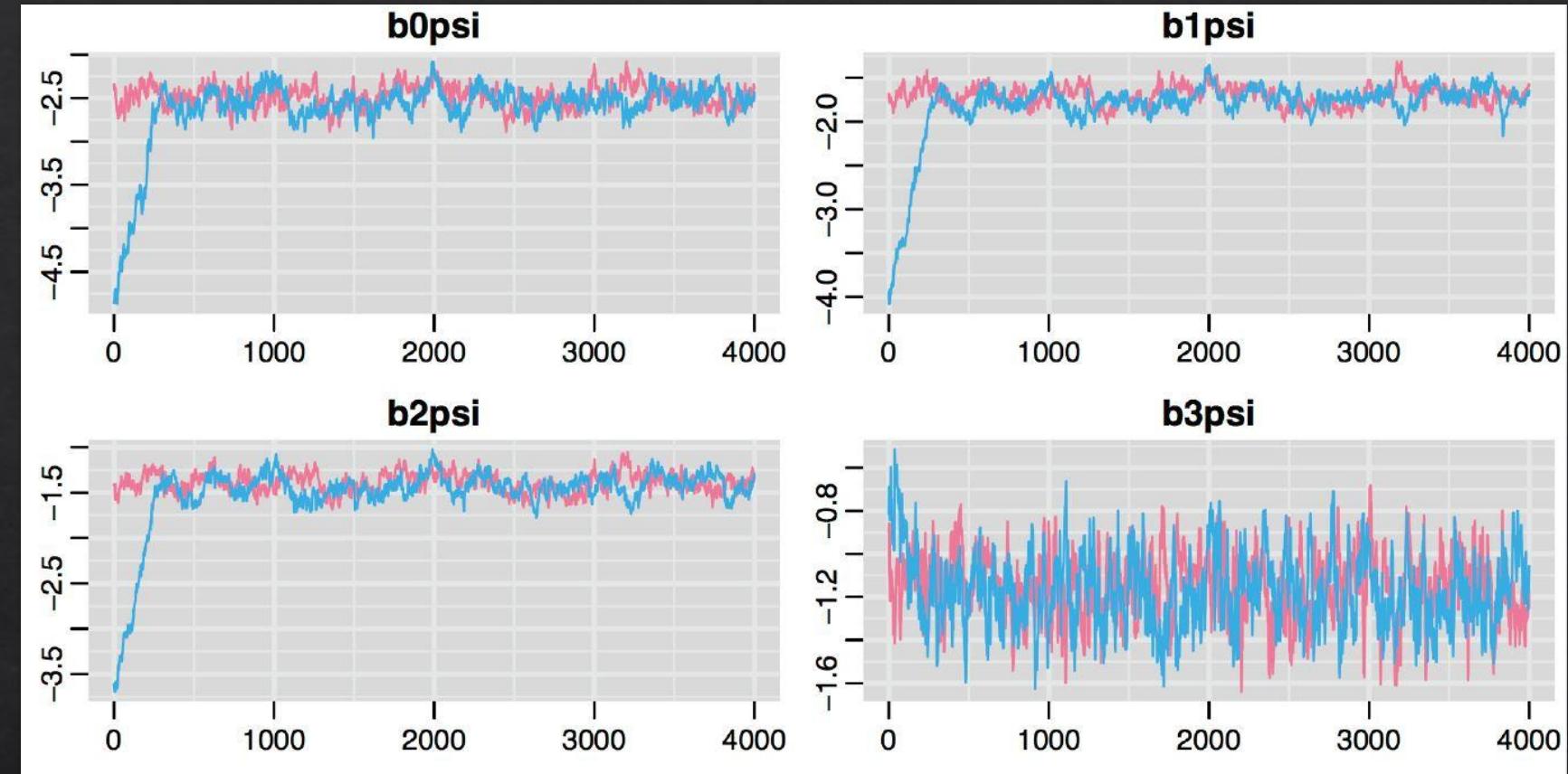
How does a Bayesian model actually work?

- ❖ Bayesian models will take the input data and model and try to estimate the best possible answers and present its results for parameter estimates

How does a Bayesian model actually work?

- ❖ How? Through iteration.
- ❖ It will try lots of different combinations of parameters, measure how well these fit the data (through log-likelihood fitting), and report back what it finds.
- ❖ It will use previous results to improve its guesswork going forward.
- ❖ This approach is collectively known as Monte Carlo sampling, of which there are various sub-types.

```
fit <- stan(file = stan_model11,  
            data = stan_data,  
            warmup = 500,  
            iter = 4000,  
            chains = 2,  
            cores = 2,  
            thin = 1)
```



- ❖ In total, a Bayesian model will often run thousands of iterations, and will “converge” towards a consensus answer. The answer will never be perfect, but will be very close to the correct answer.

Key options when running a model

```
fit <- stan(file = stan_model1,  
            data = stan_data,  
            warmup = 500,  
            iter = 4000,  
            chains = 2,  
            cores = 2,  
            thin = 1)
```

- ❖ **Iterations:** how many times should the model look for an answer? The more iterations, the more likely our posterior distribution will be representative of the true parameter estimate
- ❖ **Burn-in/warm-up:** how many times should the model train itself before we start logging the output? The longer the warm-up, the more likely the model will converge towards a solution before we start sampling the posterior
- ❖ **Chains:** how many times should the model run a warm-up and posterior sampling process? Each time will be independent and multiple chains will ensure the model is “stable”!
- ❖ **Thin:** estimates within a chain will correlate since they build on previous estimates. How many iterations should we skip between sampling?
- ❖ **Cores:** if you have a multi-core processor, how many cores should be used for parallel processing

Side note: iterations are not inherently Bayesian

- ❖ Until recently most statistics were not Bayesian
- ❖ Why? Because it was not possible to evaluate the integrals of most probability distributions directly, which is required to calculate the posterior distribution
- ❖ Computing power means we can now solve Bayesian probability equations numerically, i.e. through iteration!
- ❖ Several frequentist packages in R secretly use iterative processes but they tend to be hidden from view. However, they work very similarly!

Part 4: Model checks and diagnostics



Model checks and diagnostics

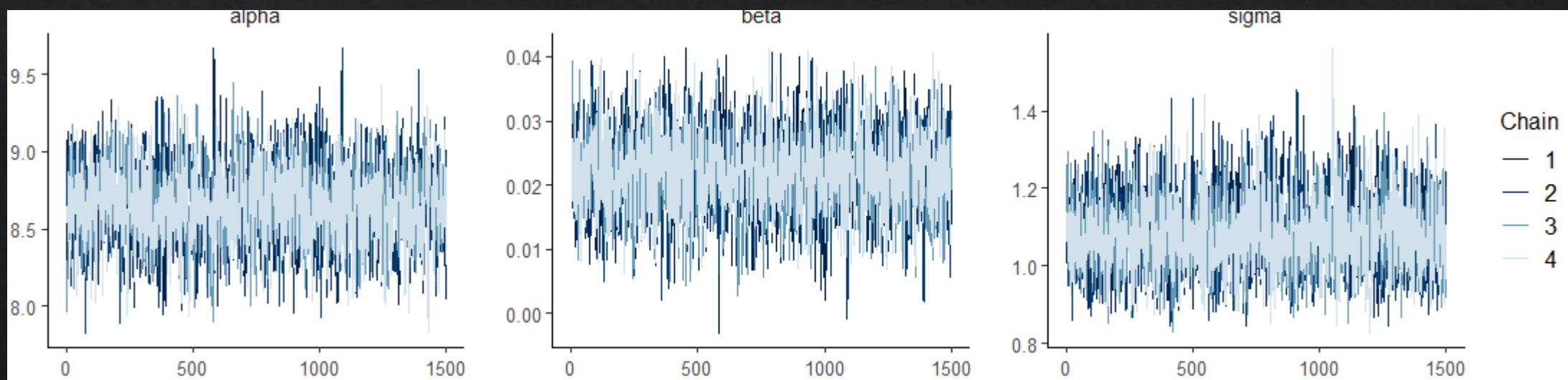
- ❖ Once the model finishes, we need to check it ran correctly.
- ❖ Did the model find a valid solution to adequately fit our data?

```
#now we run the model with our data
fit_lin <- mod_lin$sample(
  data = data_lin,
  iter_warmup = 500,
  iter_sampling = 1500,
  chains = 4,
  parallel_chains = 4,
  save_warmup = F,
  seed = 123,
  refresh = 500
)
```

Running MCMC with 4 parallel chains...

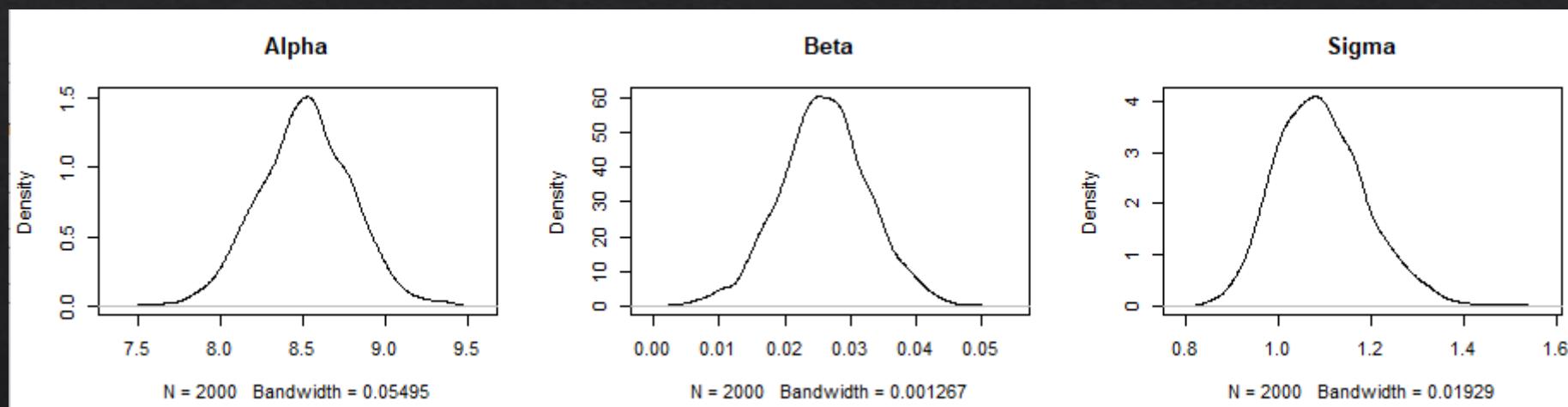
```
Chain 1 Iteration: 1 / 2000 [  0%] (warmup)
Chain 1 Iteration: 500 / 2000 [ 25%] (warmup)
Chain 1 Iteration: 501 / 2000 [ 25%] (sampling)
Chain 1 Iteration: 1000 / 2000 [ 50%] (sampling)
Chain 1 Iteration: 1500 / 2000 [ 75%] (sampling)
Chain 1 Iteration: 2000 / 2000 [100%] (sampling)
Chain 2 Iteration: 1 / 2000 [  0%] (warmup)
Chain 2 Iteration: 500 / 2000 [ 25%] (warmup)
Chain 2 Iteration: 501 / 2000 [ 25%] (sampling)
Chain 2 Iteration: 1000 / 2000 [ 50%] (sampling)
Chain 2 Iteration: 1500 / 2000 [ 75%] (sampling)
Chain 2 Iteration: 2000 / 2000 [100%] (sampling)
```

variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
lp__	-40.58	-40.27	1.20	0.98	-42.90	-39.26	1.00	2406	3275
alpha	8.65	8.65	0.25	0.24	8.25	9.06	1.00	2485	2715
beta	0.02	0.02	0.01	0.01	0.01	0.03	1.00	2669	2877
sigma	1.08	1.07	0.09	0.09	0.94	1.24	1.00	3372	2977



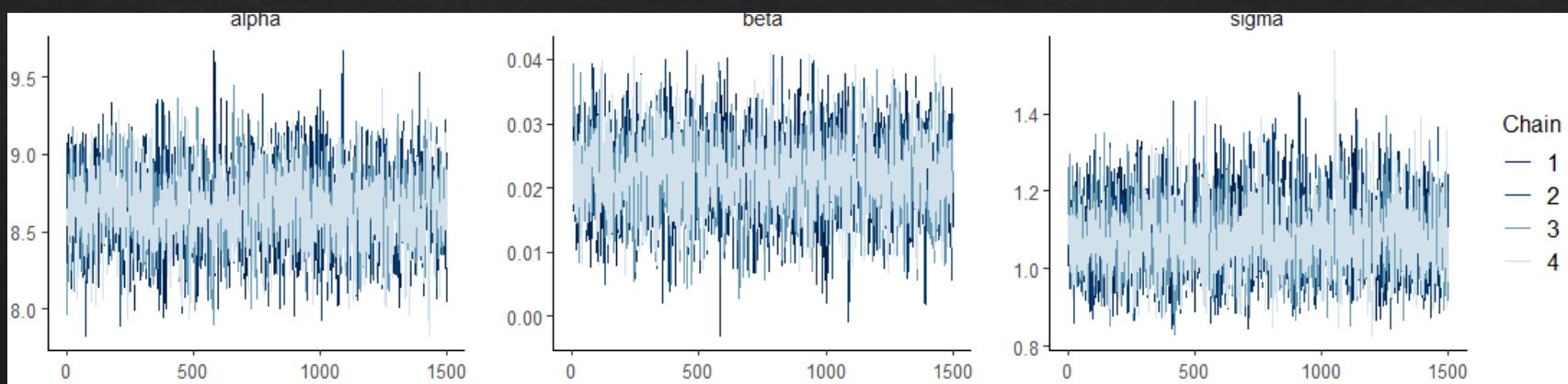
Checking the posterior distribution

- ❖ Parameters should usually look normally distributed.
- ❖ Multiple peaks means model cannot find a single, valid solution
- ❖ Strong skew sometimes mean your priors are constraining the model



Key checks for model convergence

- ◆ Model converges to stable space, all chains show similar behaviour
- ◆ “fuzzy caterpillar”
- ◆ \hat{R} should be close to 1.0
- ◆ Do posterior distributions look sensible?



variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
1p_	-40.58	-40.27	1.20	0.98	-42.90	-39.26	1.00	2406	3275
alpha	8.65	8.65	0.25	0.24	8.25	9.06	1.00	2485	2715
beta	0.02	0.02	0.01	0.01	0.01	0.03	1.00	2669	2877
sigma	1.08	1.07	0.09	0.09	0.94	1.24	1.00	3372	2977

Part 5: Understanding the Output



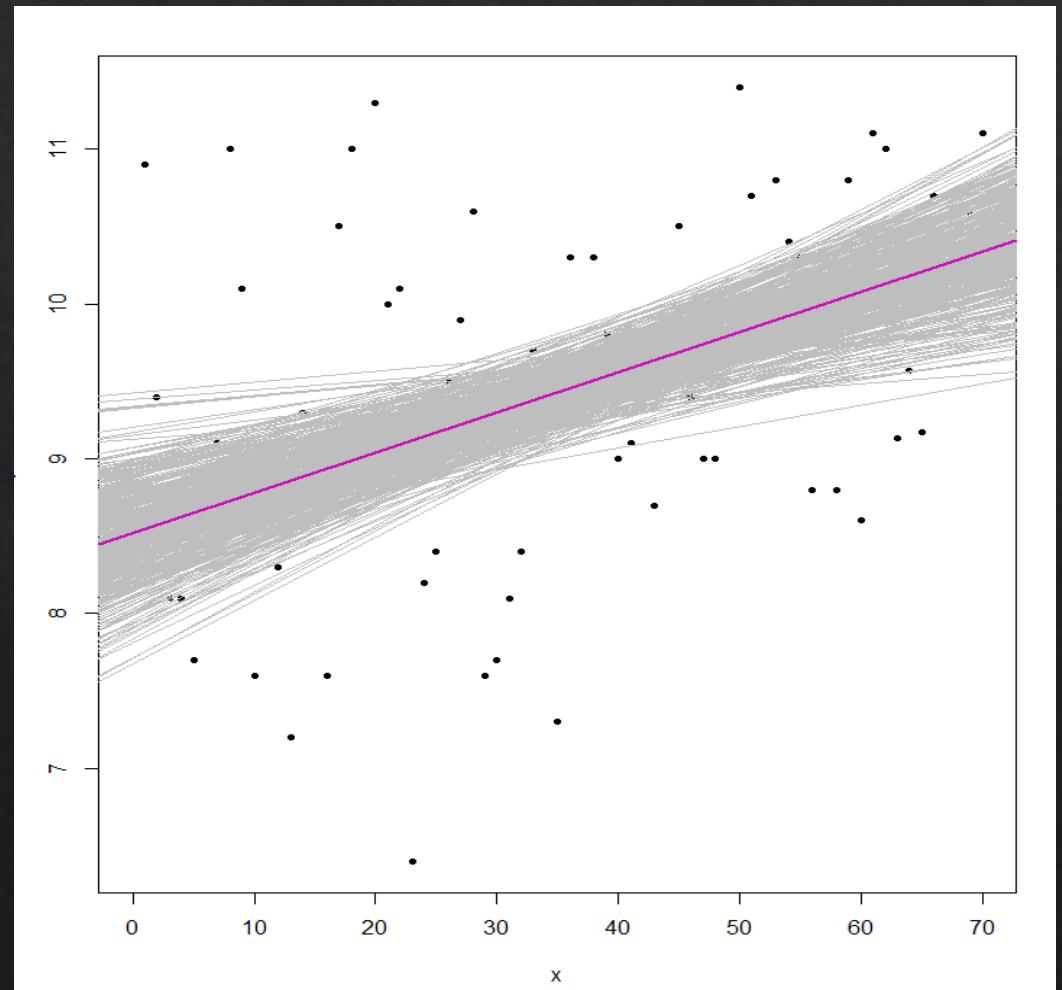
Part 5: Understanding the Output

- ◆ Bayesian models do not give point estimates, standard errors, or p-values!
- ◆ Instead, they have an estimate per parameter per iteration, these are then formed into a posterior distribution

```
# A draws_array: 1500 iterations, 4 chains, and 1 variables
, , variable = beta

      chain
iteration   1    2    3    4
  1 0.025 0.023 0.039 0.017
  2 0.024 0.020 0.030 0.024
  3 0.034 0.021 0.031 0.023
  4 0.029 0.017 0.035 0.022
  5 0.020 0.025 0.030 0.027

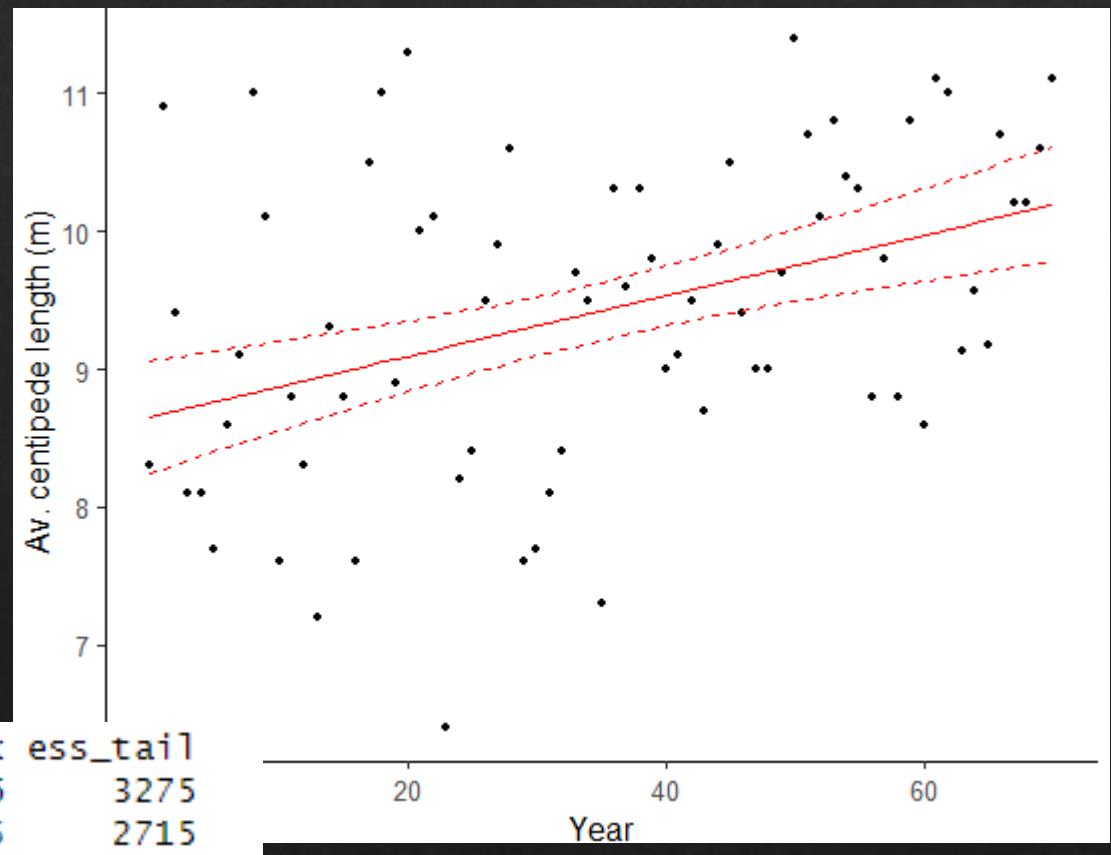
# ... with 1495 more iterations
> |
```



Part 5: Understanding the Output

- ◆ Bayesian models do not give point estimates, standard errors, or p-values!
- ◆ Instead, they have an estimate per parameter per iteration, these are then formed into a posterior distribution
- ◆ We can use this to make a mean estimate, and a 95% credible interval

variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
lp__	-40.58	-40.27	1.20	0.98	-42.90	-39.26	1.00	2406	3275
alpha	8.65	8.65	0.25	0.24	8.25	9.06	1.00	2485	2715
beta	0.02	0.02	0.01	0.01	0.01	0.03	1.00	2669	2877
sigma	1.08	1.07	0.09	0.09	0.94	1.24	1.00	3372	2977

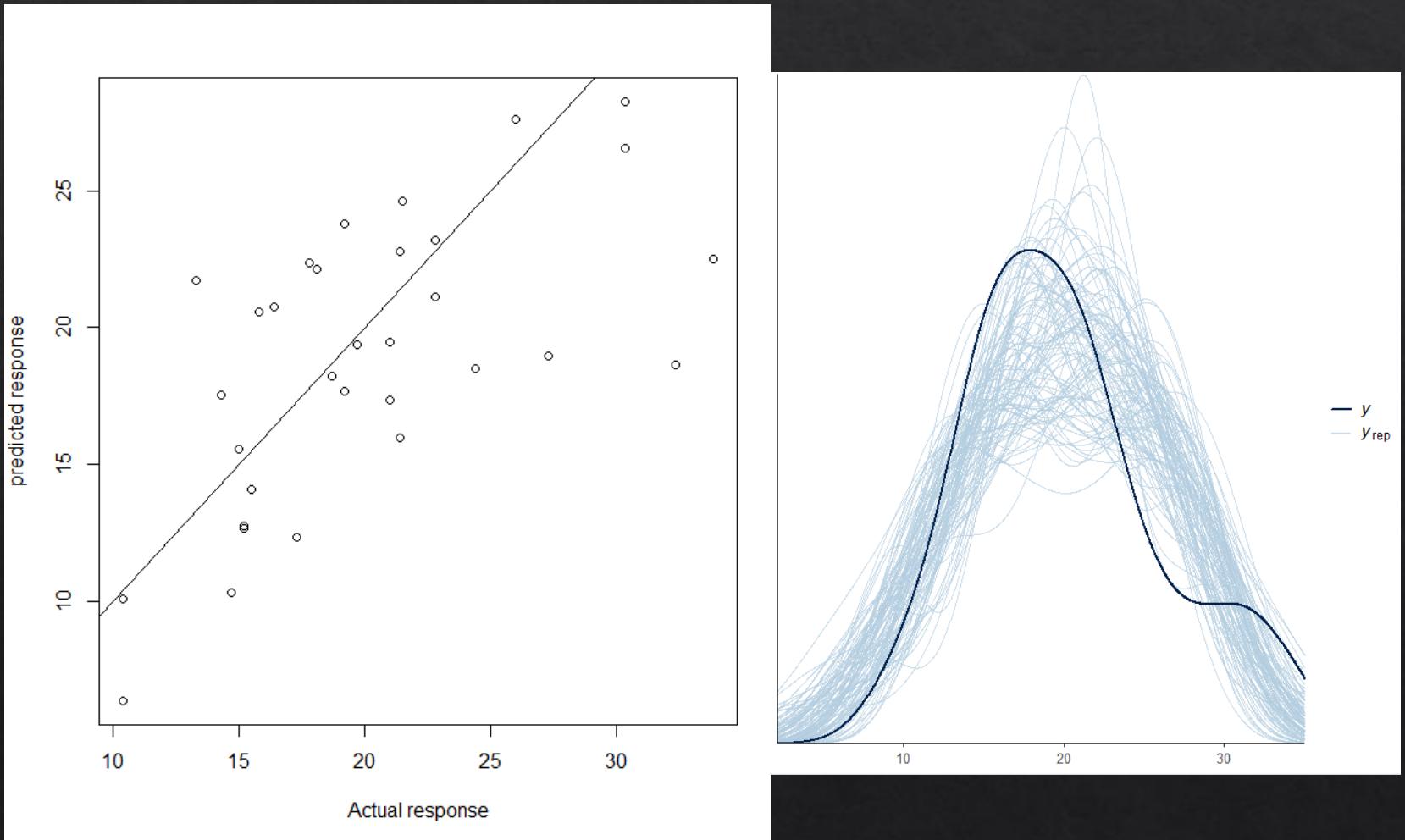


Checking model fit

- ❖ You cannot calculate classical residuals in a Bayesian approach
- ❖ But that doesn't mean we cannot check model performance!
- ❖ Key ways to check are:
 - ❖ Posterior predictive checks = can our model predict our data well? Does it have any obvious flaws?
 - ❖ Validation = build the model with a part of the data-set, can it successfully predict the left-out part
- ❖ Note: some packages provide pseudo-residuals, but they are technically different to ones you're used to. Read up on them if interested!

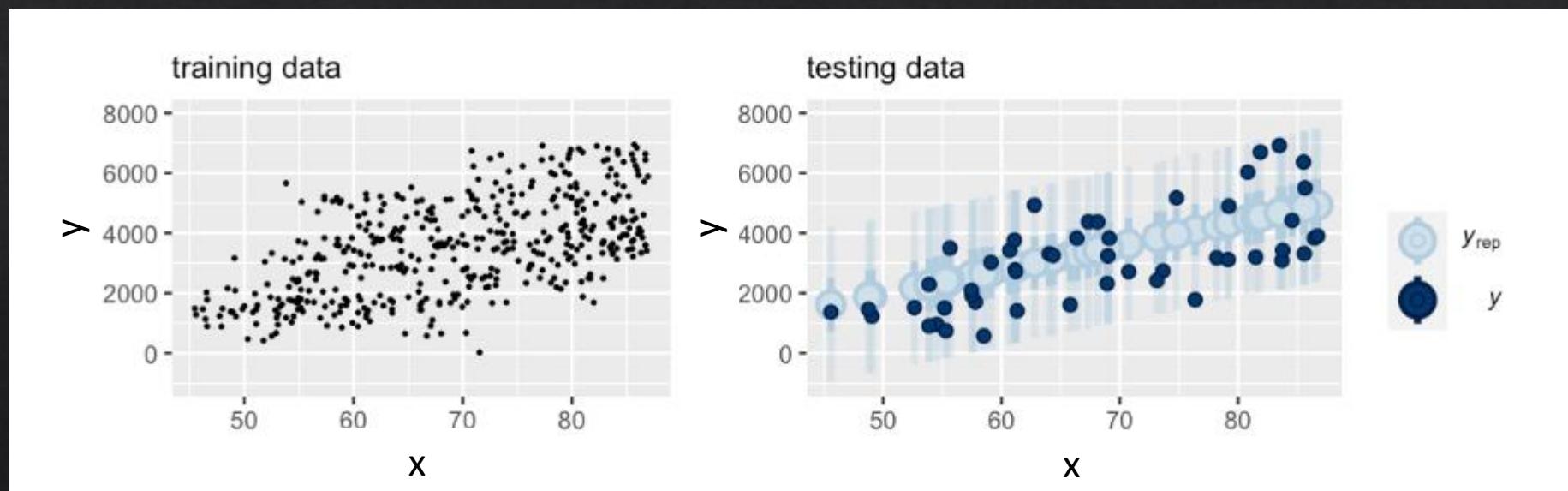
The Bayesian Approach: Example

- ❖ Posterior predictive checks = can our model predict our data well?
Does it have any obvious flaws?



The Bayesian Approach: Example

- ◊ Validation = build the model with a part of the data-set, can it successfully predict the left-out part?



Part 6: Model Comparisons

Source: Raf Schoenmaekers

Model comparisons

- ❖ Sometimes we might want to build different models with different priors, model structures and parameters. In which case we need to compare them and find which performs better
- ❖ In a frequentist framework, we might use AIC, etc
- ❖ Bayesian frameworks have equivalents, that essentially work the same way!
 - ❖ Common ones include: BIC, WAIC, etc
 - ❖ The most commonly used (and recommended) option is LOO (leave-one-out cross-validation)
 - ❖ In theory it would rebuild a model without a single data point and then see if it can predict it accurately. This would normally take ages, so we approximate this process through PSIS
 - ❖ LOO is an excellent estimate of model fit, and can spot outliers in your data that don't fit your distribution

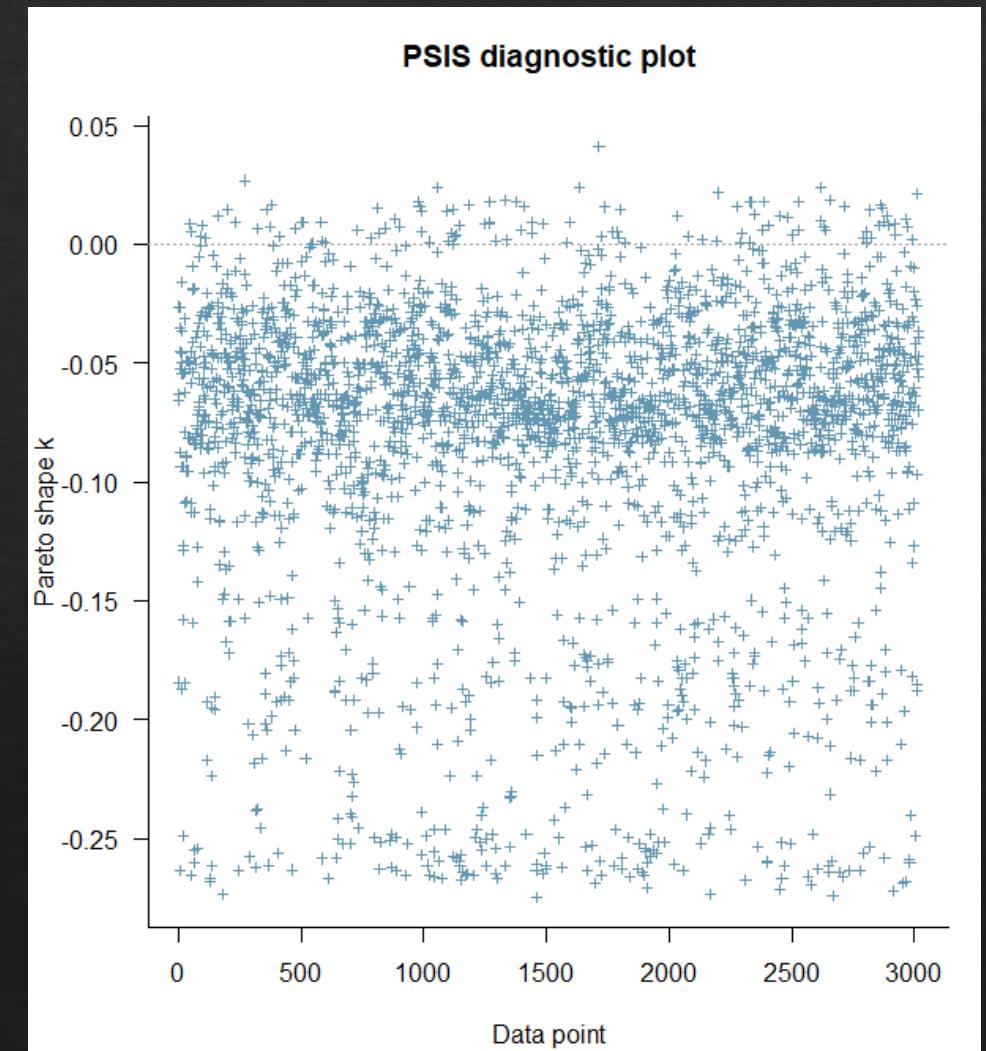
LOO

```
Computed from 3000 by 3020 log-likelihood matrix.

      Estimate    SE
elpd_loo -1968.4 15.6
p_loo       3.2  0.1
looic     3936.8 31.2
-----
MCSE of elpd_loo is 0.0.
MCSE and ESS estimates assume independent draws (r_eff=1).

All Pareto k estimates are good (k < 0.7).
See help('pareto-k-diagnostic') for details.
```

	elpd_diff	se_diff
fit2	0.0	0.0
fit1	-71.7	12.1



Part 7: Frequentist vs Bayesian (again)

<https://xkcd.com/1132/>

Frequentist versus Bayesian

- ❖ Frequentist models:
 - ❖ Have an intrinsic null hypothesis
 - ❖ Estimate probability of seeing the data given the hypothesis
 - ❖ Parameters are fixed, models return point estimates with associated errors
 - ❖ Uses confidence intervals
- ❖ Bayesian models:
 - ❖ Have prior beliefs, but no formal null hypothesis testing
 - ❖ Estimate probability of a hypothesis given the data by using prior beliefs
 - ❖ Parameters are estimated as a probability distribution
 - ❖ Uses credibility intervals directly representing probability

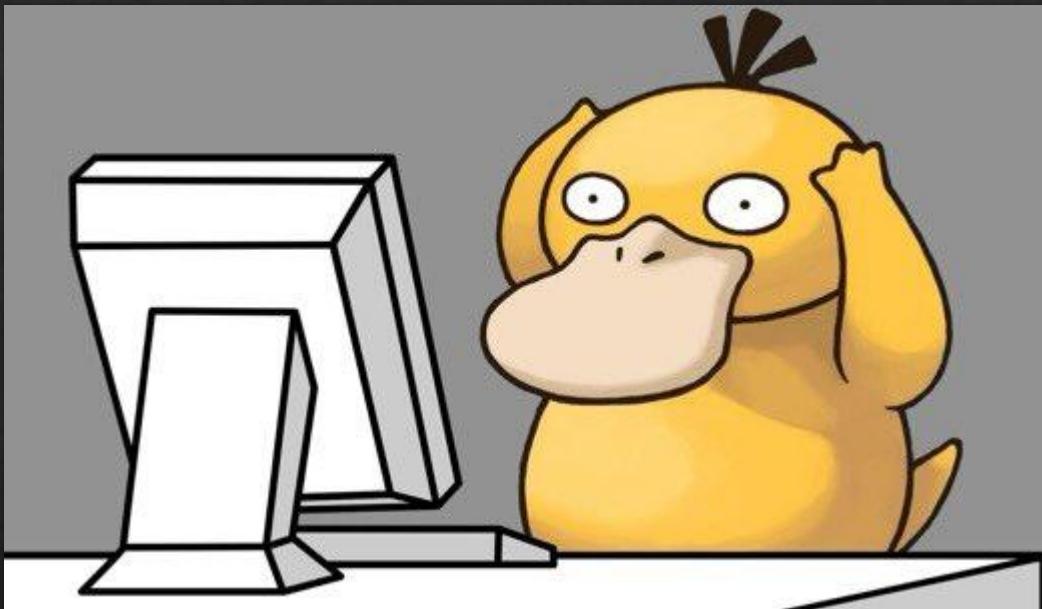
Frequentist versus Bayesian

- ❖ Advantages of Frequentist approach:
 - ❖ Generally faster, especially for simpler models
 - ❖ Still standard approach in ecology so better tutorials and support
 - ❖ Larger array of tools and packages to solve standard problems, little customisation needed
 - ❖ Results are (often) more interpretable, with single estimates
- ❖ Advantages of Bayesian approach:
 - ❖ Generally better at modelling complex systems, especially with small sample sizes
 - ❖ Much more powerful and flexible, can solve statistical problems that frequentist approaches cannot
 - ❖ Credible intervals are usually more robust, generally give a more intuitive sense of uncertainty

The Bayesian Approach: When do I use it?

- ❖ If it makes sense for your data (heavy conditional probabilities)
- ❖ If standard frequentist methods don't work
 - ❖ Small sample size?
 - ❖ Lots of random effects?
 - ❖ Highly hierarchical data?
 - ❖ No standard distributions work?
 - ❖ Other problems that can't be modelled easily in a frequentist approach (e.g. heteroskedasticity)?
- ❖ Bayesian approaches rely heavily on iterative problem solving, they can be used to solve many difficult statistical problems.

Questions?

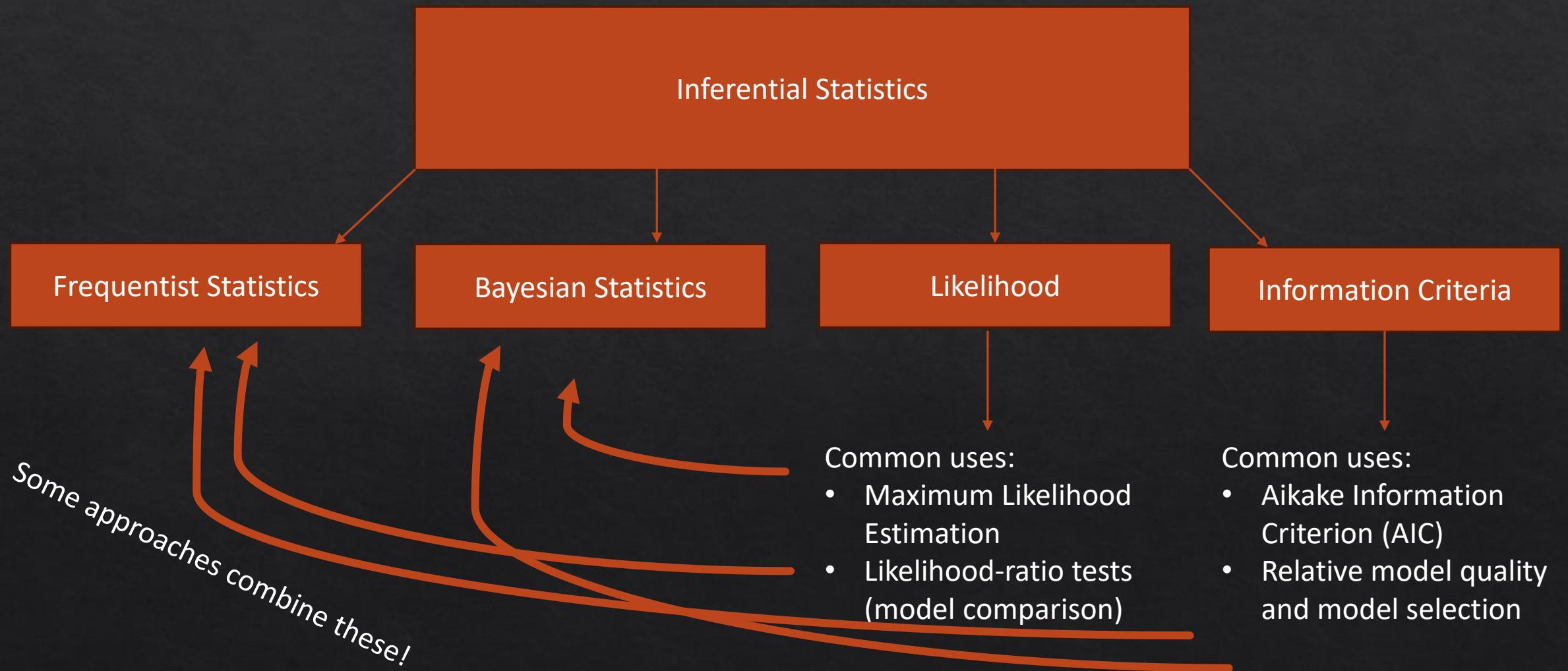


Exercises

- ❖ <https://github.com/HakkinenH/IoZ-BayesianCourse2025>
- ❖ Download the repository, unzip it
- ❖ Run “Code/0Setup.R” if you haven’t already
- ❖ Read the README file for description of code files
- ❖ Work through examples as you like!

Side note:

- ❖ Broadly speaking there are four branches of statistical inference you are likely to come across



Cut?: What are statistics?

- ❖ The collection, organization, analysis, interpretation, and presentation of data
- ❖ In science, we often use “statistics” as short-hand for “inferential statistics” which examines properties of data in relation to underlying probability distributions
 - ❖ This includes hypothesis testing! And most forms of classical models!
- ❖ Alternatives include “descriptive statistics” which we use all the time (mean, median, standard deviation etc.)
- ❖ In inferential statistics we attempt to quantify any patterns in our data, and just as importantly, the uncertainty around those conclusions.