



11th Meeting of the Hamburg R-User-Group, 7th Dec 2017

Bayesian Regression Models in R: Choosing informative priors in *rstanarm*



Dr. Daniel Lüdecke
d.luedecke@uke.de

Agenda

1. Introduction into the empirical example
2. A simple regression model (and its flaws)
3. Short introduction into Bayesian regression modelling
4. Short overview of *rstanarm*
5. Fitting and comparing Bayesian regression models
 - weakly informative priors
 - informative priors
 - how to choose informative priors
6. Conclusions

Predicting Fall Incidents in Hospitals for Patients with Dementia

1. Background: Increased risk of falling especially in patients with dementia (3 to 3.7 fold higher odds of fall incident)
2. Objective: Finding factors that explain the higher odds of fall incidents
3. Methods: Logistic regression
 1. Outcome: fall incident during hospital stay yes/no
 2. Predictors: age, gender, mobility, severity of dementia symptoms (mild, medium and severe), and others.

Focus of this talk:

Association between dementia (3-category) and fall risk



Fitting a simple logistic regression model

Data stem from a research project about a special care unit in internal medicine for patients with dementia.

Out of 526 cases, about 10% fall incidents (n=52).

From these 52 patients who did fall in hospital:

- 1 with mild dementia, 14 with medium dementia and 37 with severe dementia symptoms.

For this talk, we focus on the predictor „severe dementia“:

OR 8.65 (CI 1.62 – 161.19), p = .042

rather conspicuous values,
especially from what is known
from other research (OR 3.0 to 3.7)

```
# the model formula, also used in the other
# example models
mf <- formula(
  fall ~ age + dementia + multimorb +
  sex + mobility
)

m1 <- glm(
  mf,
  data = d,
  family = binomial("logit")
)
```

Bayesian Regression Models with Stan

The solution for this problem?

Bayesian Regression Models: Advantages

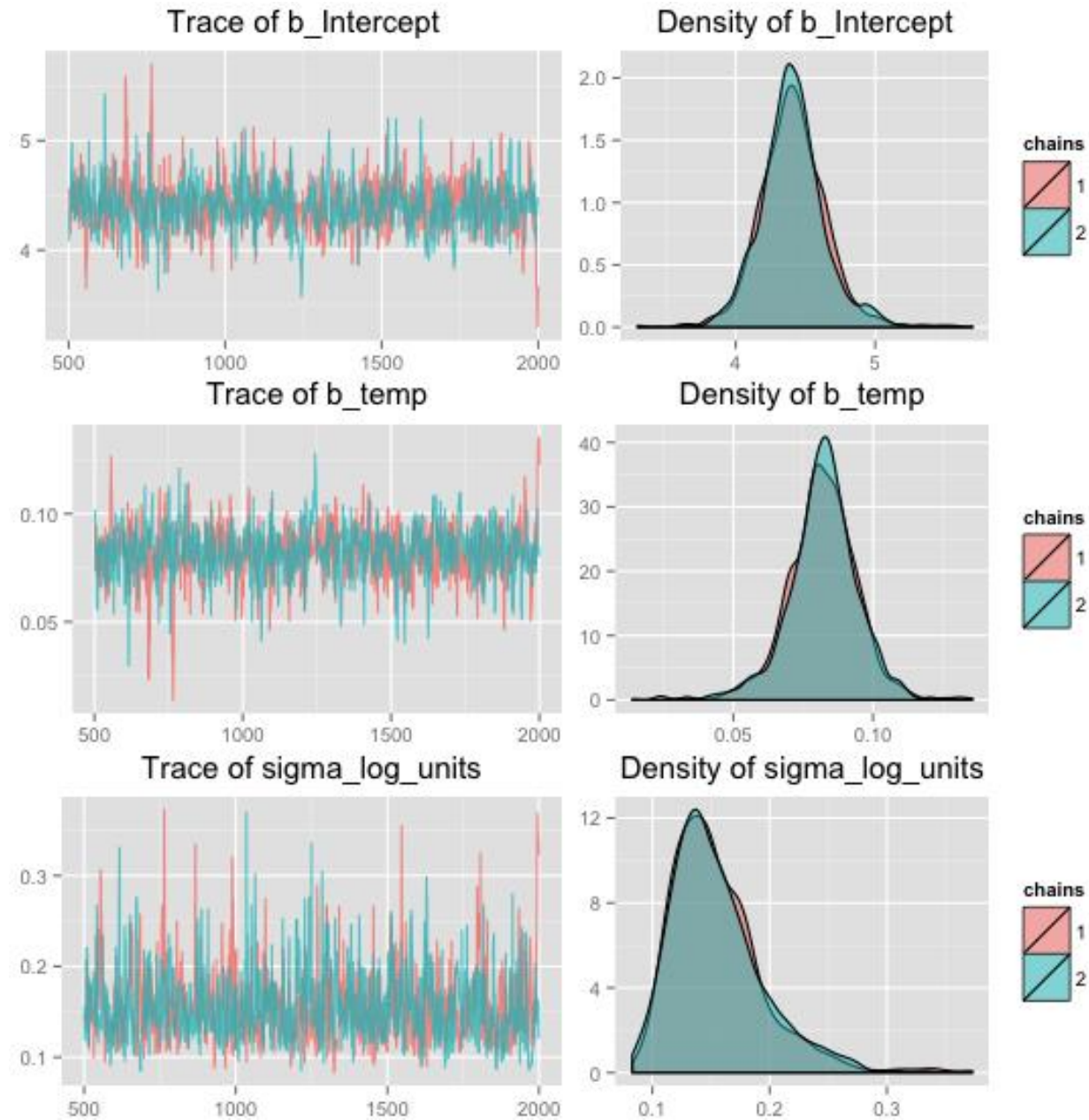
Some advantages of Bayesian regression models:

- better cope with small sample sizes
- penalize estimates towards a plausible parameter space
- incorporate prior knowledge
- don't link evidence to p-values

You can also shrink / penalize in frequentist framework (e.g. package *logistf*), but for the sake of demonstration, Bayesian modelling is shown here.

And what is Stan?

www.mc-stan.org



Basics of Bayesian Regression

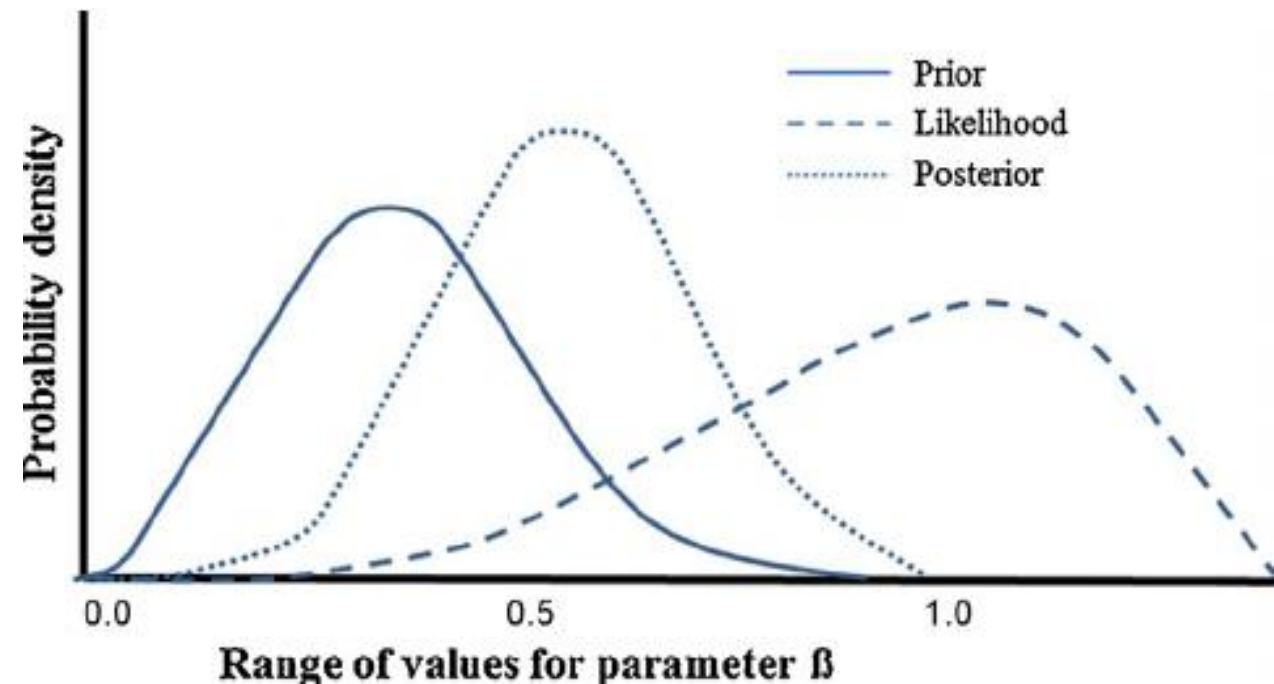
Markov-Chain Monte-Carlo Sampling from the Posterior Distribution

After fitting the model, you don't have an exact point estimate, but a „distribution“ of plausible estimated values (the approx. posterior distribution).

And you don't have confidence intervals, but so called „uncertainty“ (or credible, or also high density) intervals, which are quantiles of draws from the posterior distribution (e.g. 2.5% and 97.5% quantiles of the posterior as 95% „CI“).

However: Typically, 90% intervals are reported, because these are more stable than the 95% intervals, or a 89% interval (because 89 is the closest prime number below the conventional (but arbitrary) 95; see McElreath 2015).

95% usually require about at least 10.000 samples / draws from the posterior (Kruschke 2015).



Refitting the model in Stan using *rstanarm*

Using *rstanarm* to fit Bayesian regression models in R

rstanarm makes it very easy to start with Bayesian regression

- You can take your „normal“ function call and simply prefix the regression command with „stan_“ (e.g. `stan_lm`, `stan_glm`, `stan_lmer`, `stan_glm.nb`, `stan_betareg`, `stan_polr`)
- You have the typical „S3“ available (`summary`, `print`, `coef`, `ranef`, `vcov`...)
- Additionally, you can call „`as.data.frame()`“ on a stanreg-object to extract the posterior sample and return it as data frame (each column represents a regression coefficient, each row one of the 4000 samples).

```

# to fit a model in Stan with rstanarm,
# simply prefix your regression call
# with "stan_"
library(rstanarm)
m2 <-
  stan_glm(
    mf,
    data = d,
    family = binomial("logit")
  )
# 4000(!) observations of 6 variables
m2_df <- as.data.frame(m2)

```


Refitting the model in Stan using *rstanarm*

Comparing the two models (coefficient: severe dementia)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with weakly informative priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

estimated values are on a much more plausible parameter space – yet, against the background of what is known, they are still a bit conspicuous.

```
# obtain "point estimate" (posterior median)
coef(m2)
# same as
purrr::map_dbl(m2_df, median)

# obtain uncertainty interval
posterior_interval(m2)
# same as
purrr::map(
  m2_df,
  ~ quantile(.x, probs = c(.05, .95))
)
# or for High Density Intervals
sjstats::hdi(m2)
```

Refitting the model in Stan using *rstanarm*

Comparing the two models (coefficient: severe dementia)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with weakly informative priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

To be precise: There is no unique Bayesian “point estimate”. The posterior mean minimizes expected *squared* error, whereas the posterior median minimizes expected *absolute* error (i.e. the difference of estimates from true values over samples).

```
# obtain "point estimate" (posterior median)
coef(m2)
# same as
purrr::map_dbl(m2_df, median)

# obtain uncertainty interval
posterior_interval(m2)
# same as
purrr::map(
  m2_df,
  ~ quantile(.x, probs = c(.05, .95))
)
# or for High Density Intervals
sjstats::hdi(m2)
```

What are „weakly informative“ priors? And what are priors at all?

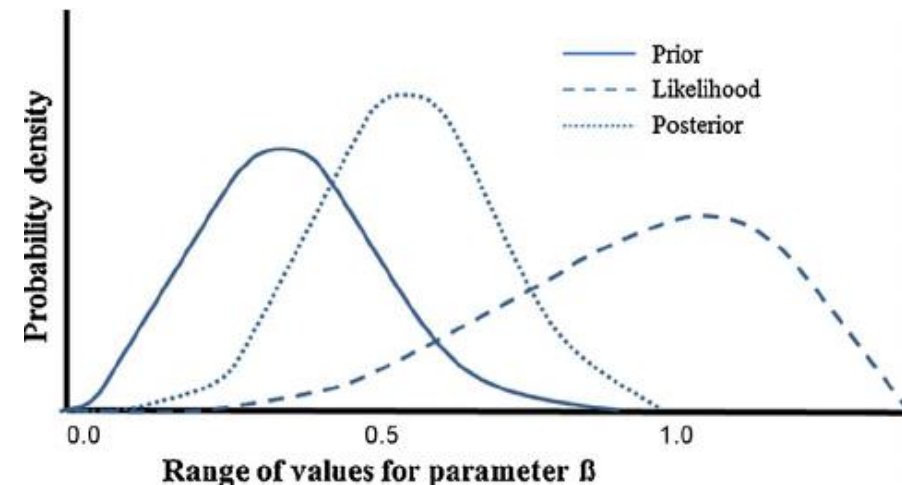
Bayes Theorem

$$\text{posterior} \sim \text{prior} * \text{likelihood}$$

- Strong evidence of data (large sample size)
= stronger impact of likelihood
- Weak evidence of data (small sample size)
= stronger impact of prior knowledge

prior, *likelihood* and *posterior* are probability distributions that make values at their tails less likely

(thus, they „regularize“ or „penalize“ parameter estimates at the boundaries of plausible parameter space)



What are „weakly informative“ priors? And what are priors at all?

Flat prior

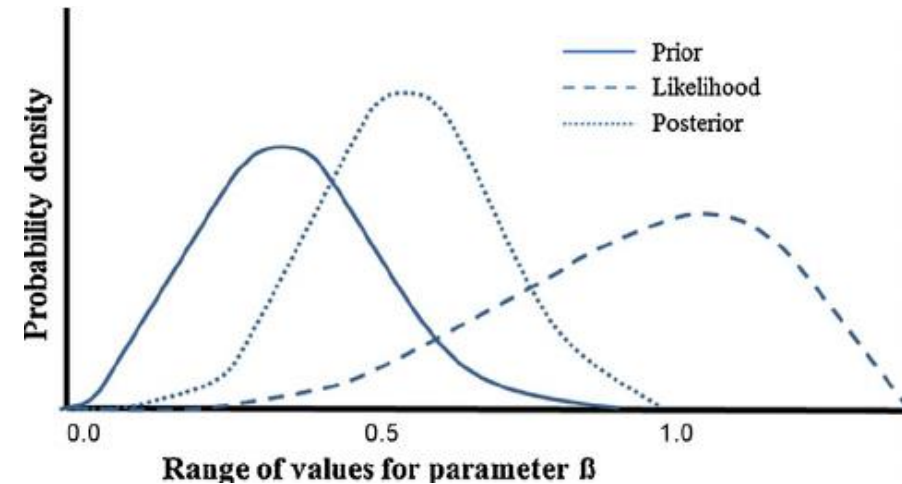
The worst prior, with (almost) no information and regulation, is the flat (*uniform*) prior. You should (almost) never use such priors!

Weakly informative priors

A well working prior for many situations and models is the *weakly informative prior*. Use this if you have no reliable knowledge about a parameter.

The default weakly informative priors in *rstanarm* are normal distributed with location 0 and a feasible scale.

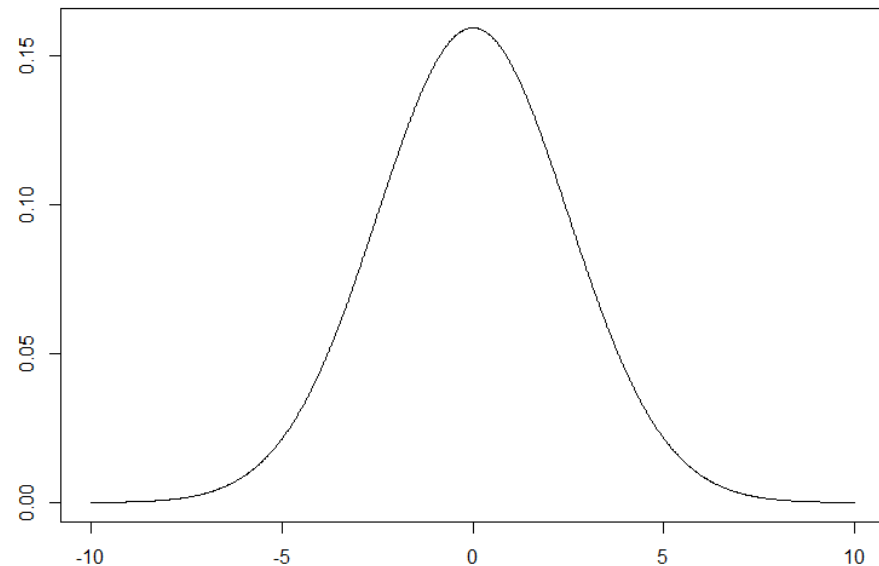
(the scale is adjusted internally, depending on the data type, i.e. continuous or dichotome etc. – however, the deviation is usually large enough to allow enough variance in the data)



`stan_lm()` is an exception here: the prior is placed on the location of R^2

Weakly informative priors in *rstanarm*

rstanarm does not adjust predictors with one value
the prior assumes a parameter estimate normally
distributed around zero, with standard deviation 2.5 for our
estimate „severe dementia“.



```
x <- seq(-5, 5, length = 1000)
y <- dnorm(x, mean = 0, sd = 2.5)
plot(x, y, type="l", lwd=1)
```

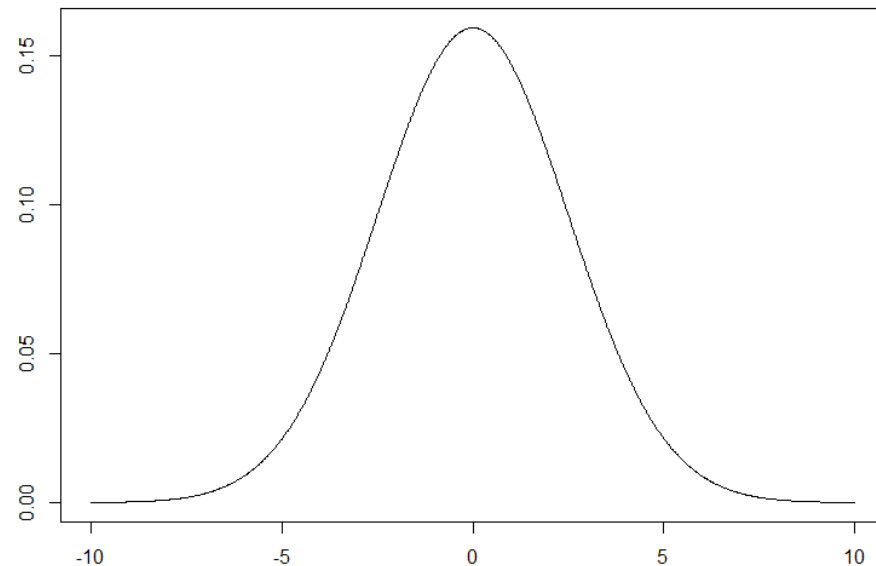
```
ps2 <- prior_summary(m2)
ps2$prior
#> $dist
#> [1] "normal"
#>
#> $location
#> [1] 0 0 0 0 0 0
#>
#> $scale
#> [1] 2.5 2.5 2.5 2.5 2.5 2.5
#>
#> $adjusted_scale
#> [1] 0.3939634 2.5000000 2.5000000 1.7935885
#> [5] 2.5000000 0.08357976
```

highlighted in red: location and
(adjusted) scale for coefficient
“severe dementia”

Weakly informative priors in *rstanarm*

rstanarm does not adjust predictors with one value

the prior assumes a parameter estimate normally distributed around zero, with standard deviation 2.5 for our estimate „severe dementia“.



```
x <- seq(-5, 5, length = 1000)
y <- dnorm(x, mean = 0, sd = 2.5)
plot(x, y, type="l", lwd=1)
```

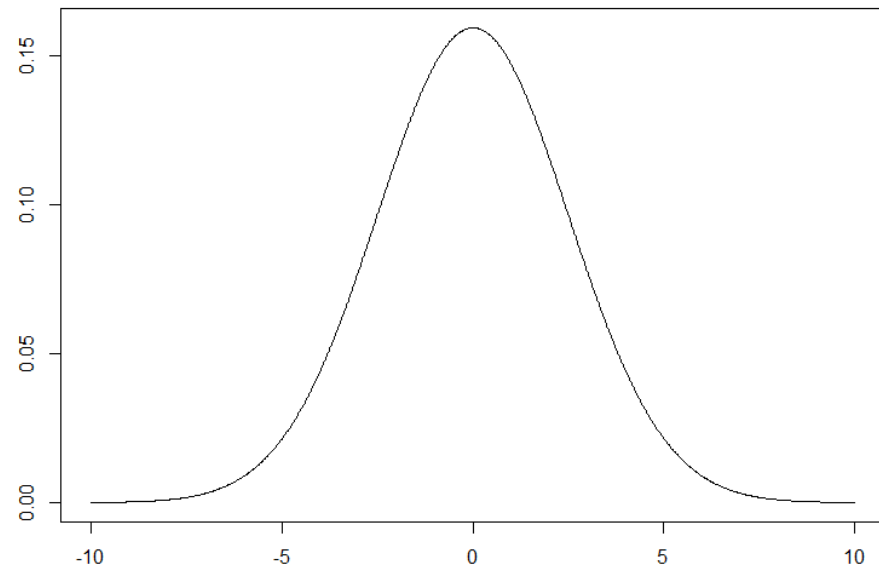
A note on default weakly informative priors

The default prior “normal(0, 2.5)” rules out very large effects, that’s why it’s called *weakly informative*.

The centering at zero means that negative and positive values are equally likely, so it’s still very conservative (e.g. when you expect a positive odds ratio).

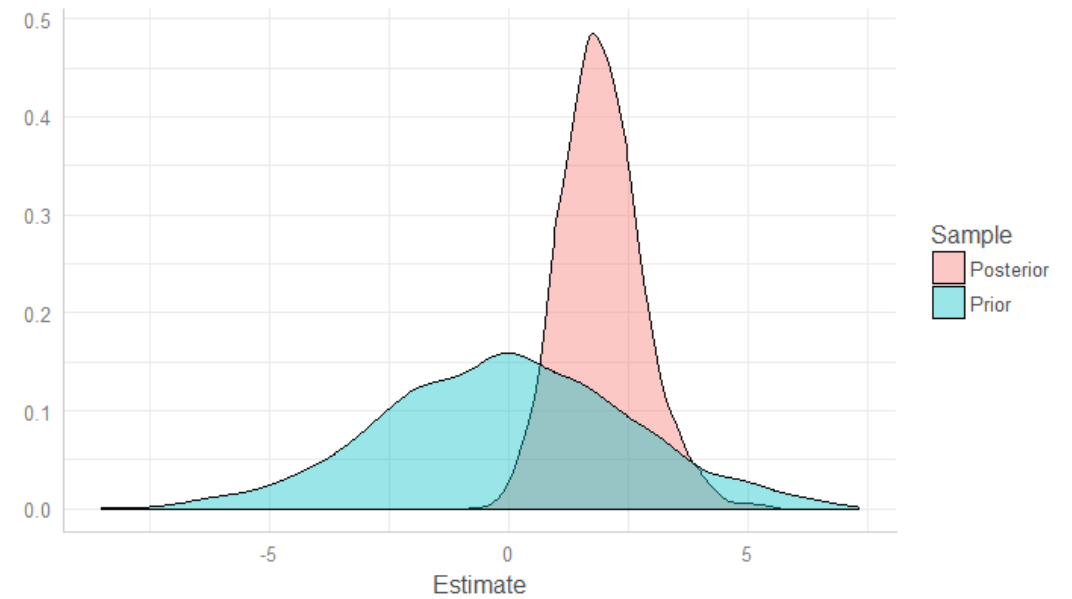
Weakly informative priors in *rstanarm*

rstanarm does not adjust predictors with one value
the prior assumes a parameter estimate normally distributed around zero, with standard deviation 2.5 for our estimate „severe dementia“.



```
x <- seq(-5, 5, length = 1000)
y <- dnorm(x, mean = 0, sd = 2.5)
plot(x, y, type="l", lwd=1)
```

After seeing the data, the posterior distribution (i.e. distribution of plausible estimates for our coefficient *severe dementia*) looks like this:



Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-0.6832	1.3458	1.8820	1.9308	2.4594	5.4322

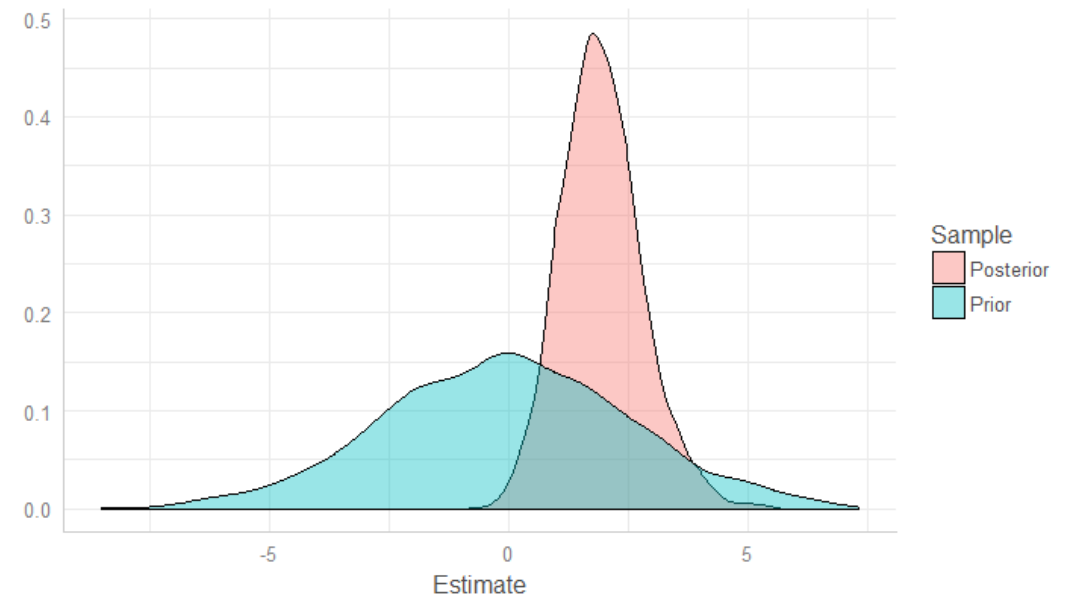
Weakly informative priors in *rstanarm*

Prior and posterior distributions on the linear scale

How does this distribution match our OR 6.57 with CI 1.72 – 24.84?

- The location and scale parameters of the prior distributions are always defined on the linear scale.
- The posterior distribution is also on the linear scale; in case of logistic regression, the posterior represents samples of estimates on the *logit (log-odds)* scale.
- $\exp(1.882) \sim 6.57$

After seeing the data, the posterior distribution (i.e. distribution of plausible estimates for our coefficient *severe dementia*) looks like this:



Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-0.6832	1.3458	1.8820	1.9308	2.4594	5.4322

Informative priors in *rstanarm*

Including prior knowledge about the parameters

Even better than weakly informative are *informative priors*.

- Informative priors describe your knowledge about parameters of interest.
- The knowledge may be based on former research, systematic reviews, ... but should *not* stem from the data you currently use to fit the model!

From literature, we know that

- medium dementia symptoms are associated with an approximate 2-fold higher odds in falling
- severe dementia is associated with an approximate 3.3 to 3.7-fold higher odds (so we take the odds ratio of 3.5)

```

p_dem_mid <- log(2)
p_dem_hi <- log(3.5)

m3 <- stan_glm(
  mf, data = d,
  family = binomial("logit"),
  prior = normal(
    location = c(
      0, p_dem_mid, p_dem_hi, 0, 0, 0),
    scale = NULL
  )
)
  
```

Informative priors in *rstanarm*

Including prior knowledge about the parameters

Since priors are defined on the linear scale, we simply take the log of our odds ratios (= prior knowledge) as location parameter for the prior distribution.

- medium dementia = $\log(2)$
- severe dementia = $\log(3.5)$
- default location parameter (= zero) for remaining predictors
- no assumptions on scale parameters (standard deviation), so we leave it `NULL`.

```
p_dem_mid <- log(2)
p_dem_hi <- log(3.5)

m3 <- stan_glm(
  mf, data = d,
  family = binomial("logit"),
  prior = normal(
    location = c(
      0, p_dem_mid, p_dem_hi, 0, 0, 0),
    scale = NULL
  )
)
```

we don't make assumptions about the standard deviation of our parameter yet...

Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

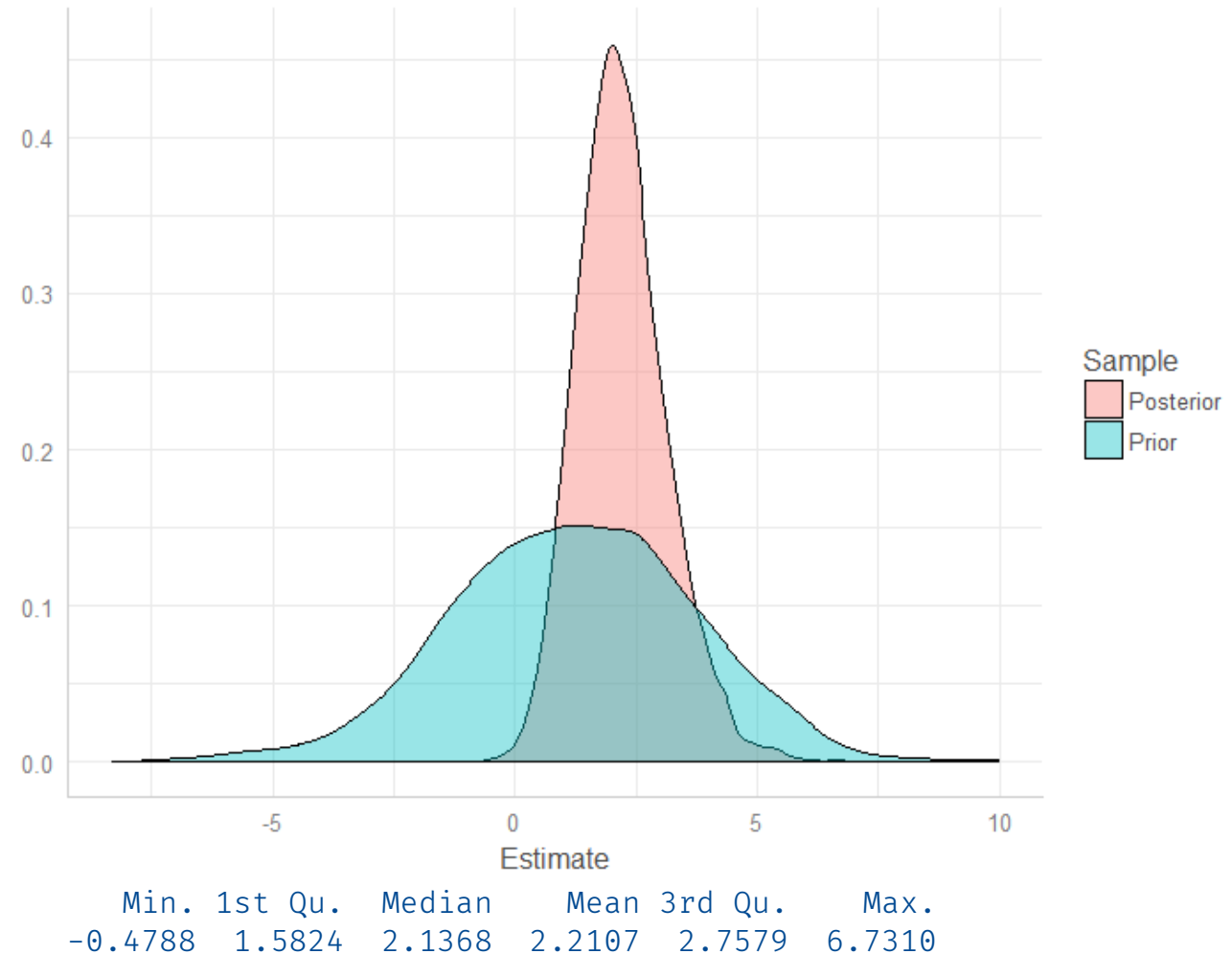
- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

- OR 8.47 (CI 1.50 – 53.00), no p-value



Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

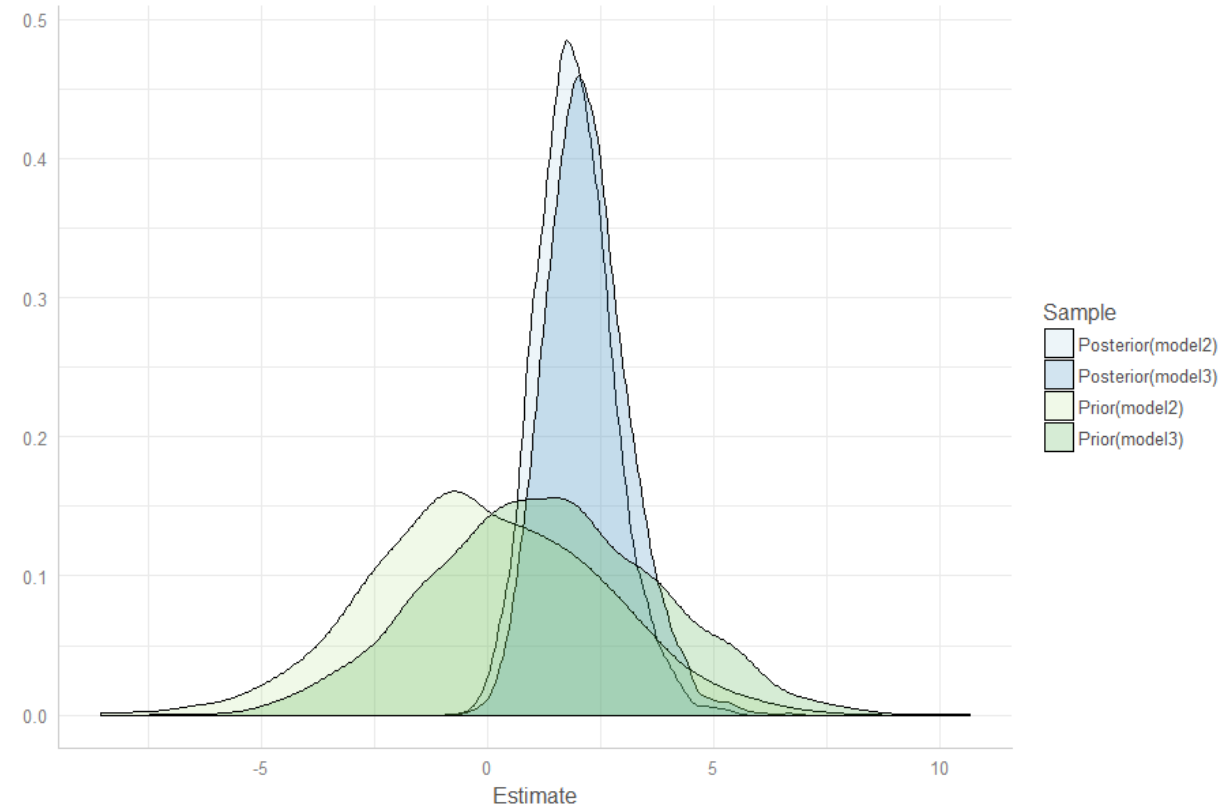
- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

- OR 8.47 (CI 1.50 – 53.00), no p-value



Informative priors in *rstanarm*

Including prior knowledge about the outcome

Informative priors can also be applied to the outcome variable.

However, for logistic regression, we need transformation to linear scale again. From literature, we know that

- Fall incidents among dementia patients varies between 30% to 60%.
- We assume a fall incident rate (probability of falls) of about 40%. `qlogis(.4)` transforms a probability of 40% on the linear scale.
- `scale=.5` (on linear scale) allows a variation of about 12%, i.e. the assumed range of fall incidents is $\sim 28\%$ to 52% (`plogis(qlogis(.4) +/- .5)`).

```
p_fall <- qlogis(.4)

m4 <- stan_glm(
  mf, data = d,
  family = binomial("logit"),
  prior = normal(location=c(0, p_dem_mid,
    p_dem_hi, 0, 0), scale=NULL),
  prior_intercept = normal(
    location = p_fall,
    scale = 0.5,
    autoscale = F
  )
)
```

Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

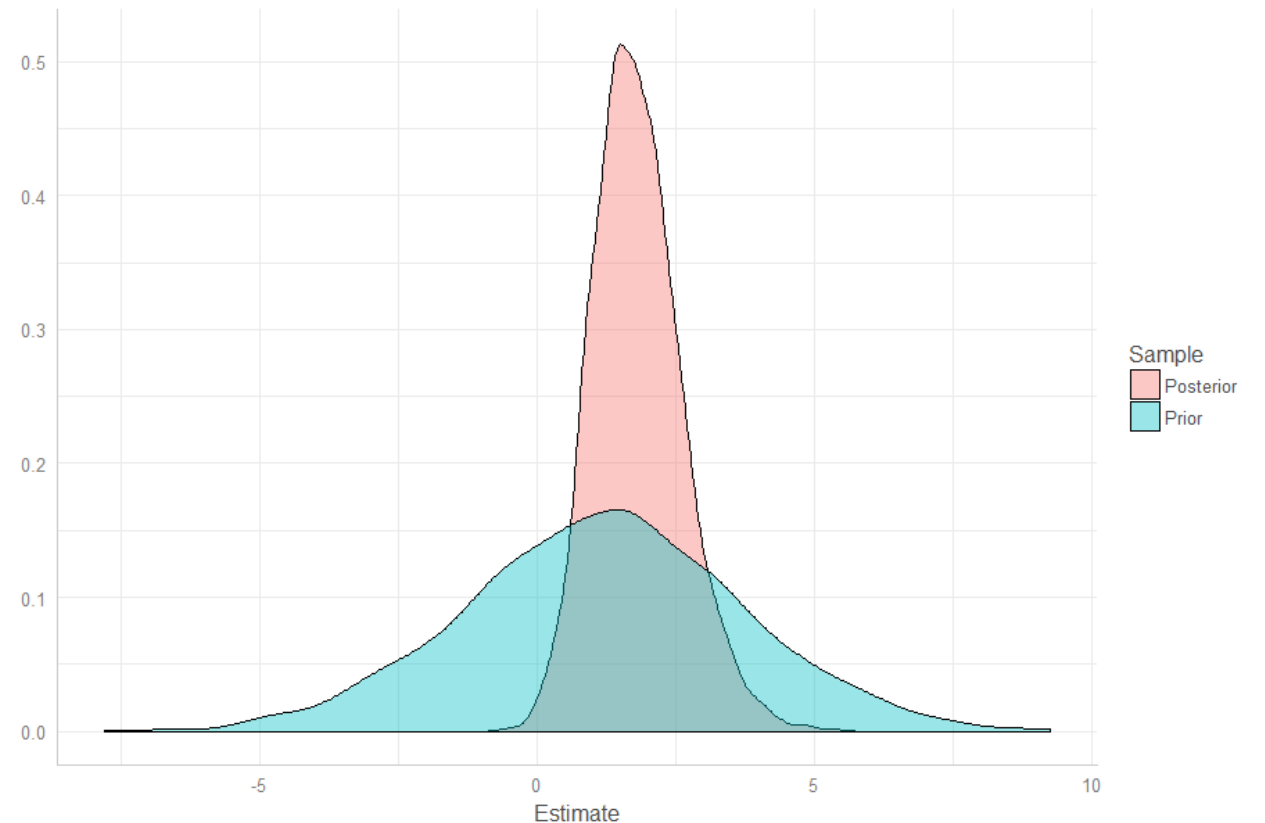
- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

- OR 8.47 (CI 1.50 – 53.00), no p-value

Model 4: *informative* priors for predictors *and* intercept

- OR 5.72 (CI 1.43 – 31.7), no p-value



Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-0.6163	1.2547	1.7438	1.8019	2.2877	5.4550

Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

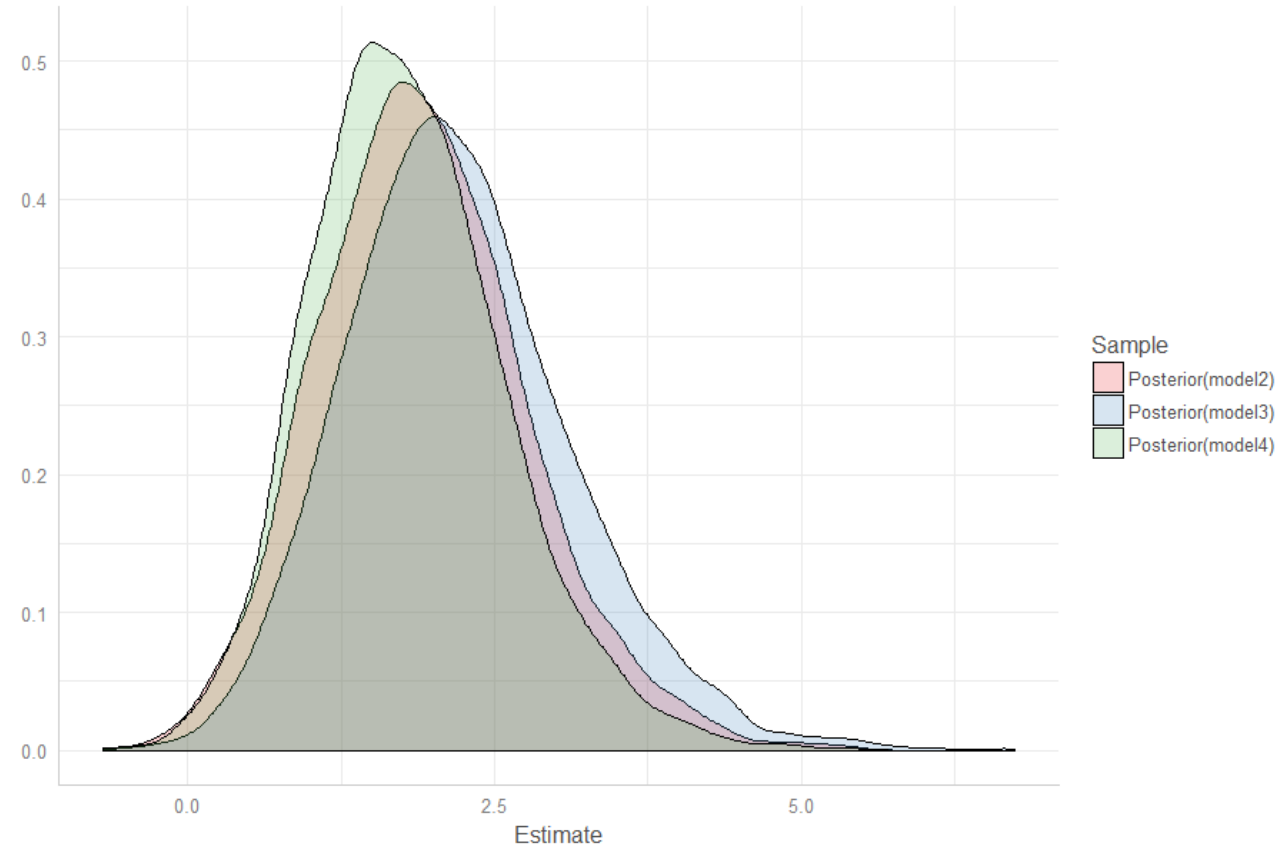
- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

- OR 8.47 (CI 1.50 – 53.00), no p-value

Model 4: *informative* priors for predictors *and* intercept

- OR 5.72 (CI 1.43 – 31.7), no p-value



Informative priors in *rstanarm*

Including prior knowledge about the variance of predictors

Now we want to regulate the parameter space by defining the scale parameters for our prior distribution.

From literature, we know that

- severe dementia is associated with an approximate 3.3 to 3.7-fold higher odds (so we take the odds ratio of 3.5)
- and a confidence interval about 2 to 7

Translated into scale parameter

- we assume a standard deviation of about $\log(2.5)$ for these parameters (just a very rough guess)
- And keep default scale parameters for remaining predictors.

```

m5 <- stan_glm(
  mf, data = d,
  family = binomial("logit"),
  prior = normal(
    location = c(<...>),
    scale = c(2.5, log(2.5), log(2.5),
              2.5, 2.5, 2.5)
  ),
  prior_intercept = normal(<...>)
)
  
```


Informative priors in *rstanarm*

Including prior knowledge about the variance of predictors

Now we want to regulate the parameter space by defining the scale parameters for our prior distribution.

From literature, we know that

- severe dementia is associated with an approximate 3.3 to 3.7-fold higher odds (so we take the odds ratio of 3.5)
- and a confidence interval about 2 to 7

Translated into scale parameter

- we assume a standard deviation of about $\log(2.5)$ for these parameters (just a very rough guess)
- And keep default scale parameters for remaining predictors.

```
m5 <- stan_glm(
  mf, data = d,
  family = binomial("logit"),
  prior = normal(
    location = c(<...>),
    scale = c(2.5, log(2.5), log(2.5),
              2.5, 2.5, 2.5)
  ),
  prior_intercept = normal(<...>)
)
```

Check the adjusted scale! As said, *rstanarm* does not adjust predictors with one value (like our predictors for mid and severe dementia), but usually all other predictors by default. *This affects your scale parameter!* Either set `autoscale = FALSE`, or multiply the scale-value by the standard deviation of your predictor, so the *adjusted* scale matches your intended scale parameter value!

Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

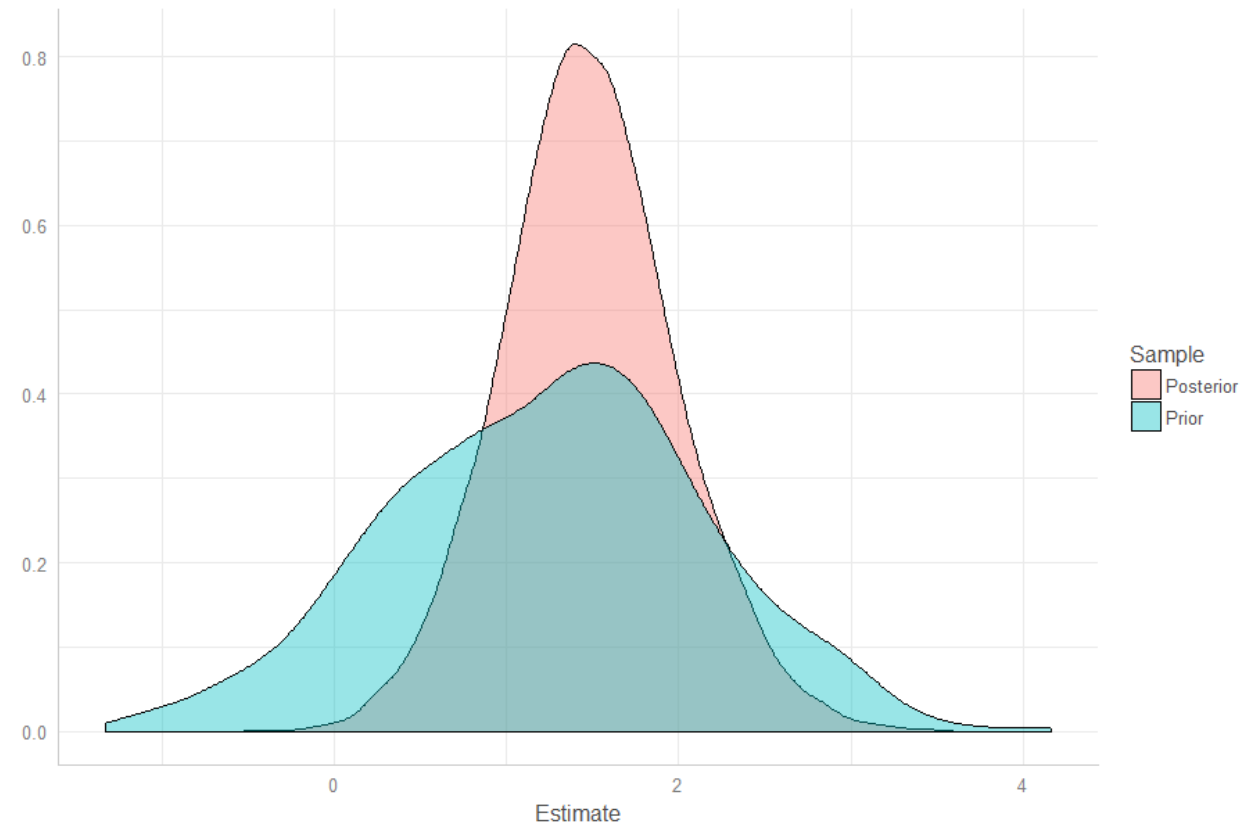
- OR 8.47 (CI 1.50 – 53.00), no p-value

Model 4: *informative* priors for predictors *and* intercept

- OR 5.72 (CI 1.43 – 31.7), no p-value

Model 5: *informative* priors including user defined scales

- OR 4.34 (CI 1.58 – 11.90), no p-value



Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-0.3635	1.1455	1.4671	1.4816	1.8055	3.4201

Informative priors in *rstanarm*

Comparing the three models (coefficient: **severe dementia**)

Model 1: simple logistic regression model

- OR 8.65 (CI 1.62 – 161.19), $p = .042$

Model 2: bayesian model with *weakly informative* priors

- OR 6.57 (CI 1.72 – 24.84), no p-value

Model 3: bayesian model with *informative* priors

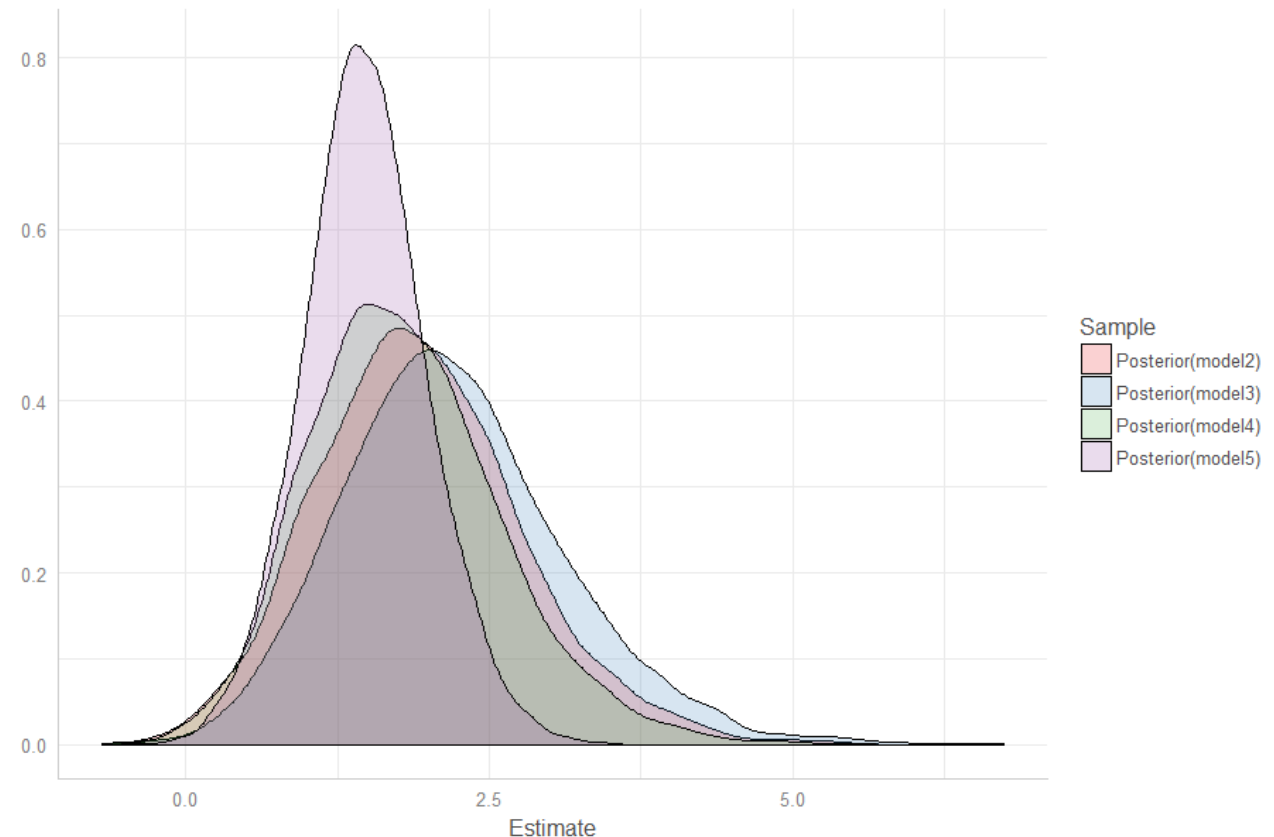
- OR 8.47 (CI 1.50 – 53.00), no p-value

Model 4: *informative* priors for predictors *and* intercept

- OR 5.72 (CI 1.43 – 31.7), no p-value

Model 5: *informative* priors including user defined scales

- OR 4.34 (CI 1.58 – 11.90), no p-value



Conclusions

Bayesian models have many advantages

Important advantages are:

- Sampling technique (MCMC) helps if data is skewed or sample size is low
- Prior knowledge ensures the estimates / parameters are within plausible boundaries

Weakly informative priors work well

Comparing the different models, weakly informative priors may outperform informative priors *in case* you have prior knowledge only for some of the predictors, and probably no information about the variance of the parameters

Informative priors work very well

Informative priors help reducing „bias“ in parameter estimation, being (very) conservative.

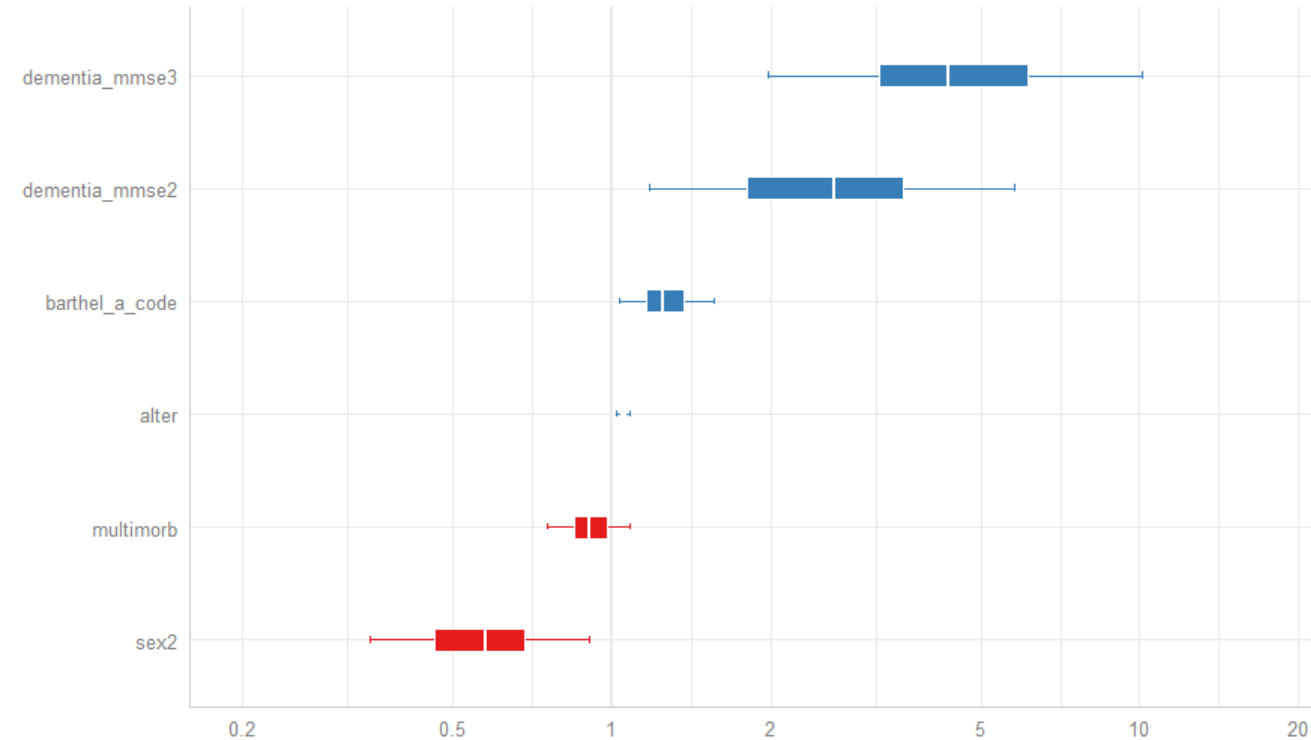
Prior information about the outcome (intercept) can be very helpful in getting „realistic“ posterior distributions – *especially when probability of events in data differs noticeably from prior knowledge!*

Caveat

Choosing to narrow *scales* for the priors may mislead the inference regarding the sign of the effect (i.e. you may think you have a purely positive or negative association, although both negative and positive values might be likely).

Recommendations

- In the examples, a 95% uncertainty interval was applied – better use a 90% range.
- The „Bayesian point estimate“ is just the value that divides the posterior distribution into two samples of values, which are equally likely.
- The „true“ value can be any value of the posterior distribution, with values around the median being more likely than at the tails of the distribution.
- Hence, it’s helpful to report „outer“ and „inner“ uncertainty intervals, e.g. the 50% and the 90% interval, plus the Bayesian point estimate.
- Packages like [sjPlot](#) help visualizing Bayesian models, [sjstats](#) provides functions for glancing at summaries/stats





Institute for Medical Sociology

HAMBURG

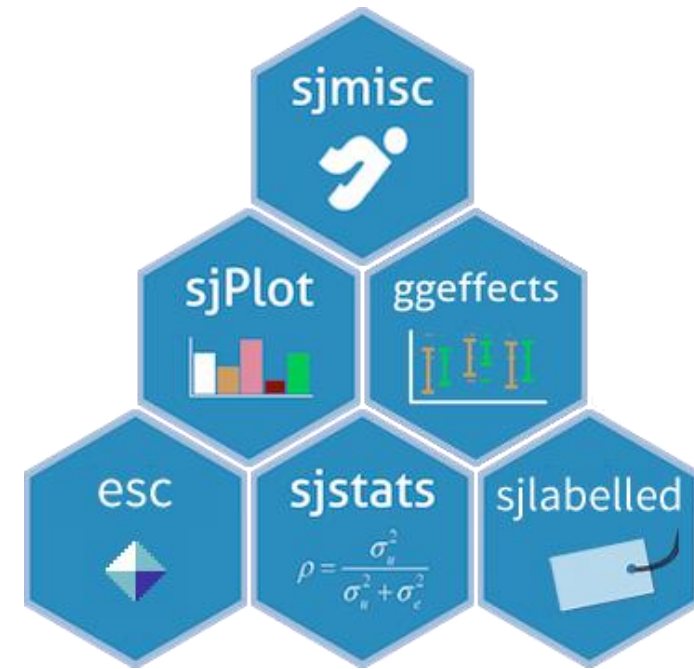
Dr. Daniel Lüdecke

Research Associate

d.luedecke@uke.de

<https://github.com/strengejacke>

twitter: @strengejacke



Recommended packages

Model fitting

- rstanarm (<https://cran.r-project.org/package=rstanarm>)
- brms (<https://cran.r-project.org/package=brms>)

Visualization

- bayesplot (<https://cran.r-project.org/package=bayesplot>)
- sjPlot (<https://cran.r-project.org/package=sjPlot>)

Other (summaries, statistics)

- sjstats (<https://cran.r-project.org/package=sjstats>)

Thanks for help and / or providing useful resources:

- Tristan Mahr (<https://tjmahr.github.io>)
- The users @ Stan discussion forums (<http://discourse.mc-stan.org>)
- Rasmus Bååth (<http://www.sumsar.net>)

Further reading

- McElreath R: Statistical Rethinking. A Bayesian Course with Examples in R and Stan. 2015, CRC Press
- rstanarm package vignettes (<https://cran.r-project.org/package=rstanarm>)