Validation results

This report contains the validation results of a small Bayesian hierarchical model. Here, we summarize the results computed in earlier targets of the pipeline. We reference our targets with `tar_load()` and `tar_read()`. This ensures

1. Because of the `tar_render()` function from the `tarchetypes` (https://wlandau.github.io/tarchetypes) package (see `_targets.R`) `targets` automatically detects the dependencies of this report and rebuilds it when its dependencies change.
2. We can run the report by itself if the targets are already in the `_targets/` data store.

Continuous covariate

```
library(targets)
tar_load(fit_continuous)
```

Our results are in a data frame with one row per simulated dataset and columns with information about our fitted models.

```
fit_continuous
```

```
## # A tibble: 1,000 x 13
##    rep   variable    mean  median      sd     mad      q5     q95  rhat ess_bulk
##    <chr> <chr>      <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <dbl>    <dbl>
##  1 rep1… beta     -0.366  -1.25   1.25e+0 1.69e-1 -1.40    1.44    1.91     5.60
##  2 rep1… beta     -0.815  -0.816  6.77e-2 6.69e-2 -0.926  -0.703   1.00  2198.
##  3 rep1… beta     -0.119  -0.118  9.41e-2 9.30e-2 -0.275   0.0313  1.00  1277.
##  4 rep1… beta     -0.0301 -0.0300 4.72e-2 4.63e-2 -0.110   0.0470  1.00  2131.
##  5 rep1… beta     -0.740  -0.741  7.85e-2 7.81e-2 -0.866  -0.613   1.00  1853.
##  6 rep1… beta     -0.0925 -0.0925 4.39e-3 4.41e-3 -0.0997 -0.0851  1.00  1733.
##  7 rep1… beta     -0.765  -0.767  8.53e-2 8.41e-2 -0.901  -0.622   1.00  1616.
##  8 rep1… beta      0.640   0.776  4.08e-1 2.58e-2 -0.545   0.814   1.30    12.4
##  9 rep1… beta      0.0191  0.0191 2.72e-4 2.74e-4  0.0187  0.0196  1.00  2716.
## 10 rep1… beta     -0.875  -0.875  6.50e-2 6.73e-2 -0.983  -0.769   1.00  1688.
## # … with 990 more rows, and 3 more variables: ess_tail <dbl>, beta_true <dbl>,
## #   cover_beta <lgl>
```

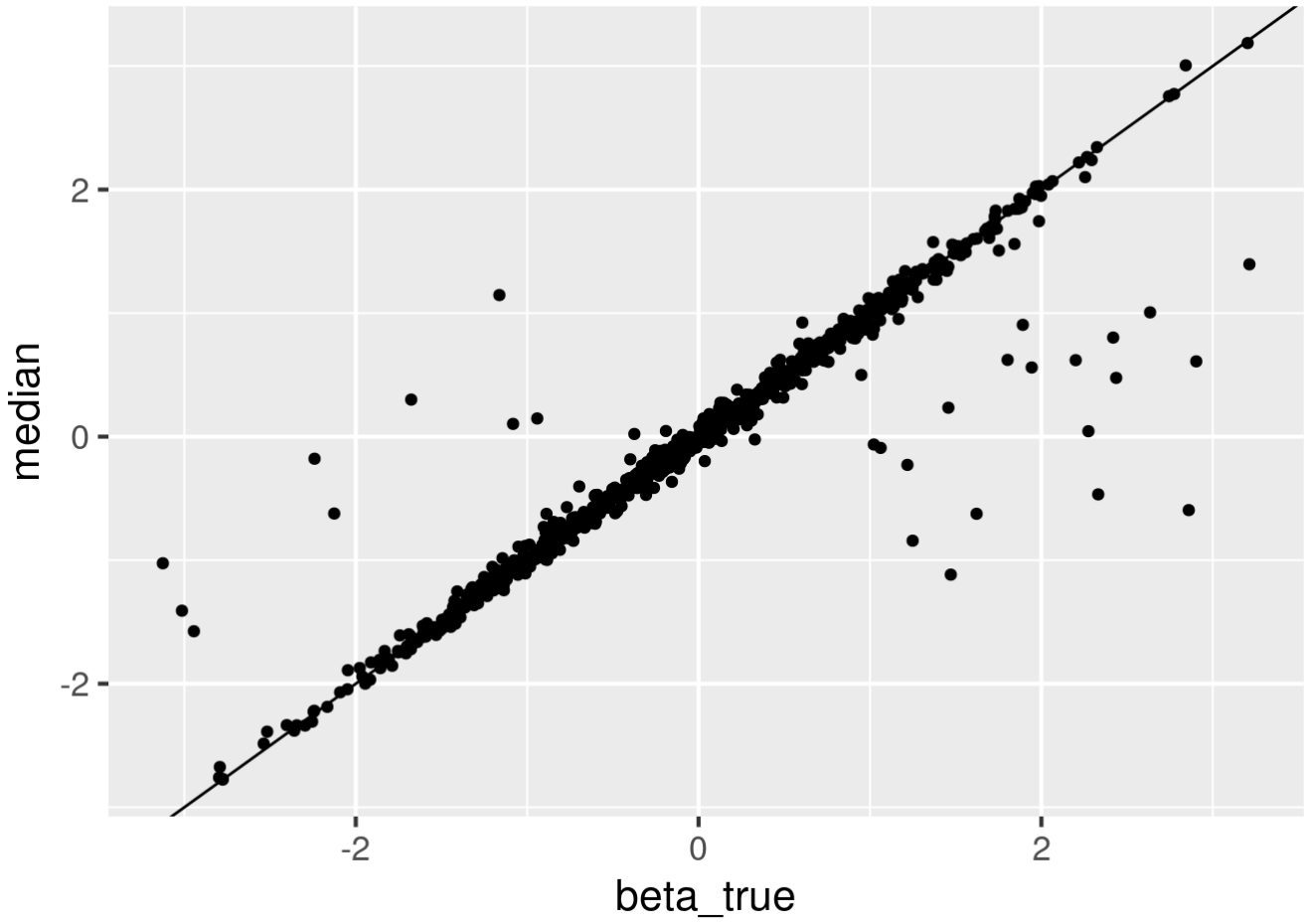
If we implemented the model in `stan/model.stan` correctly, then roughly 90% of model fits should cover the true `beta` parameter that generated the data in 90% credible intervals.

```
mean(fit_continuous$cover_beta)
```

```
## [1] 0.91
```

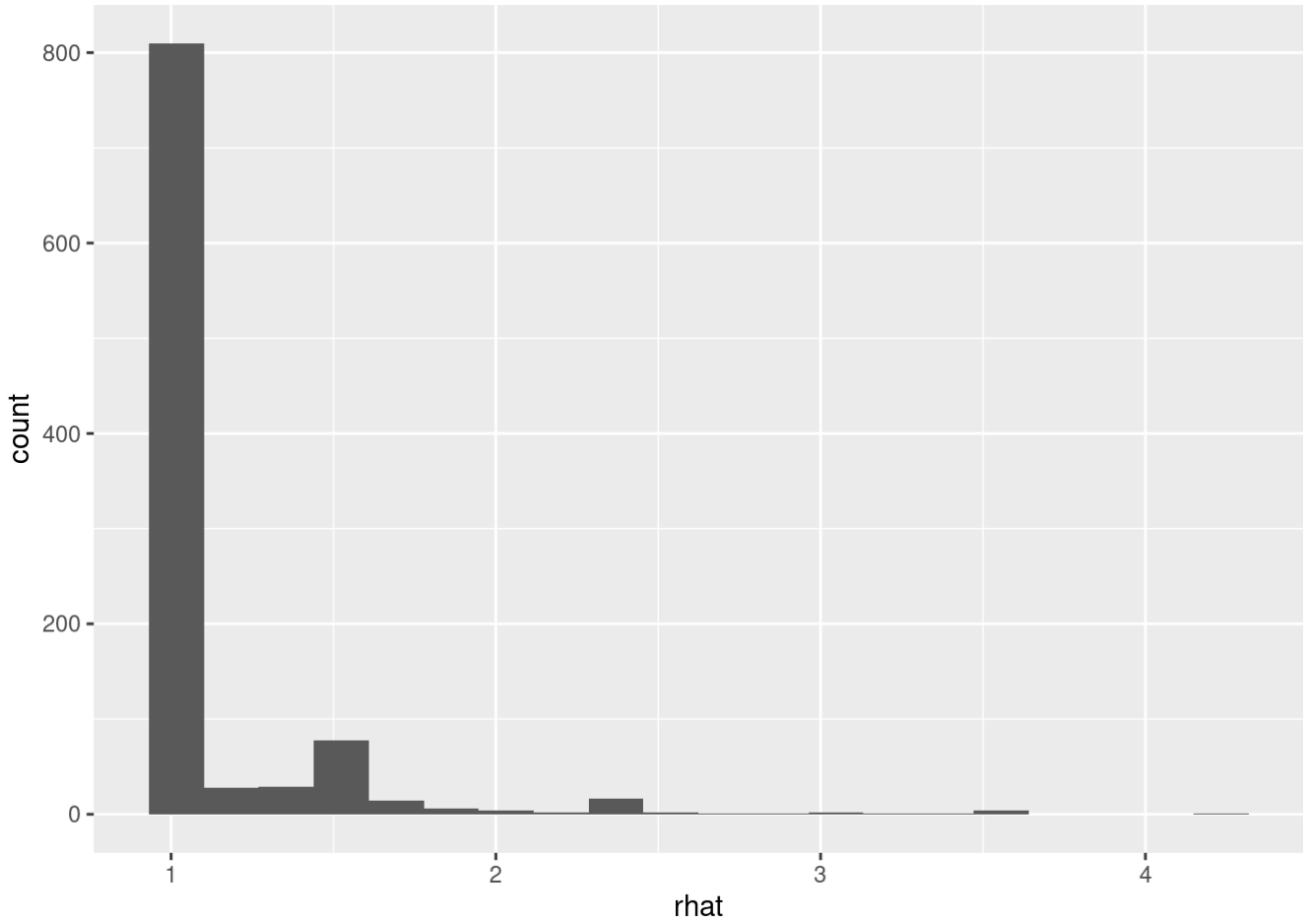
The posterior median of `beta` should be reasonably close to the true value.

```
ggplot(fit_continuous) +
  geom_point(aes(x = beta_true, y = median)) +
  geom_abline(intercept = 0, slope = 1) +
  theme_gray(16)
```



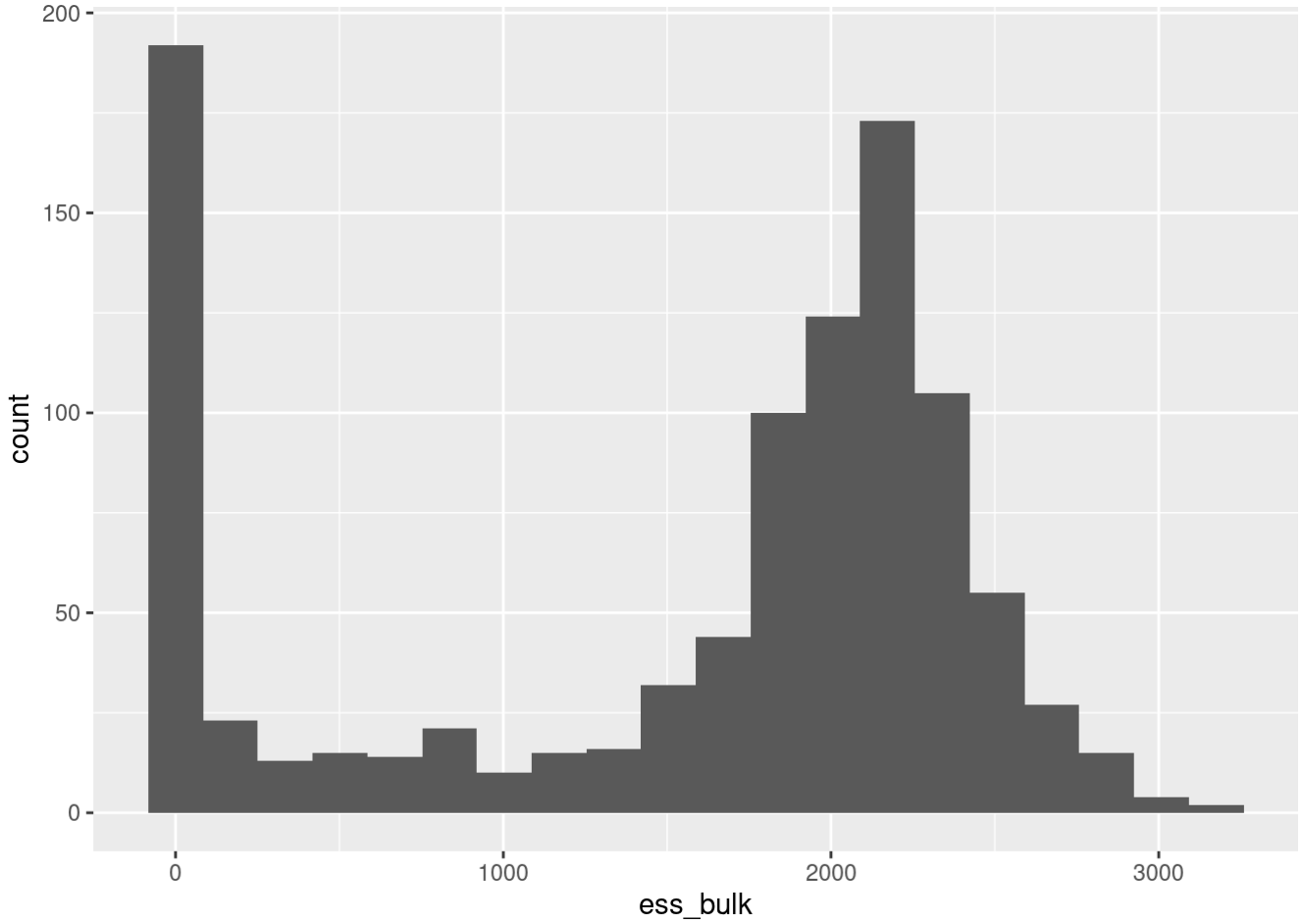
We should also check convergence diagnostics. `rhat` should ideally be close to 1.

```
ggplot(fit_continuous) +
  geom_histogram(aes(x = rhat), bins = 20)
```

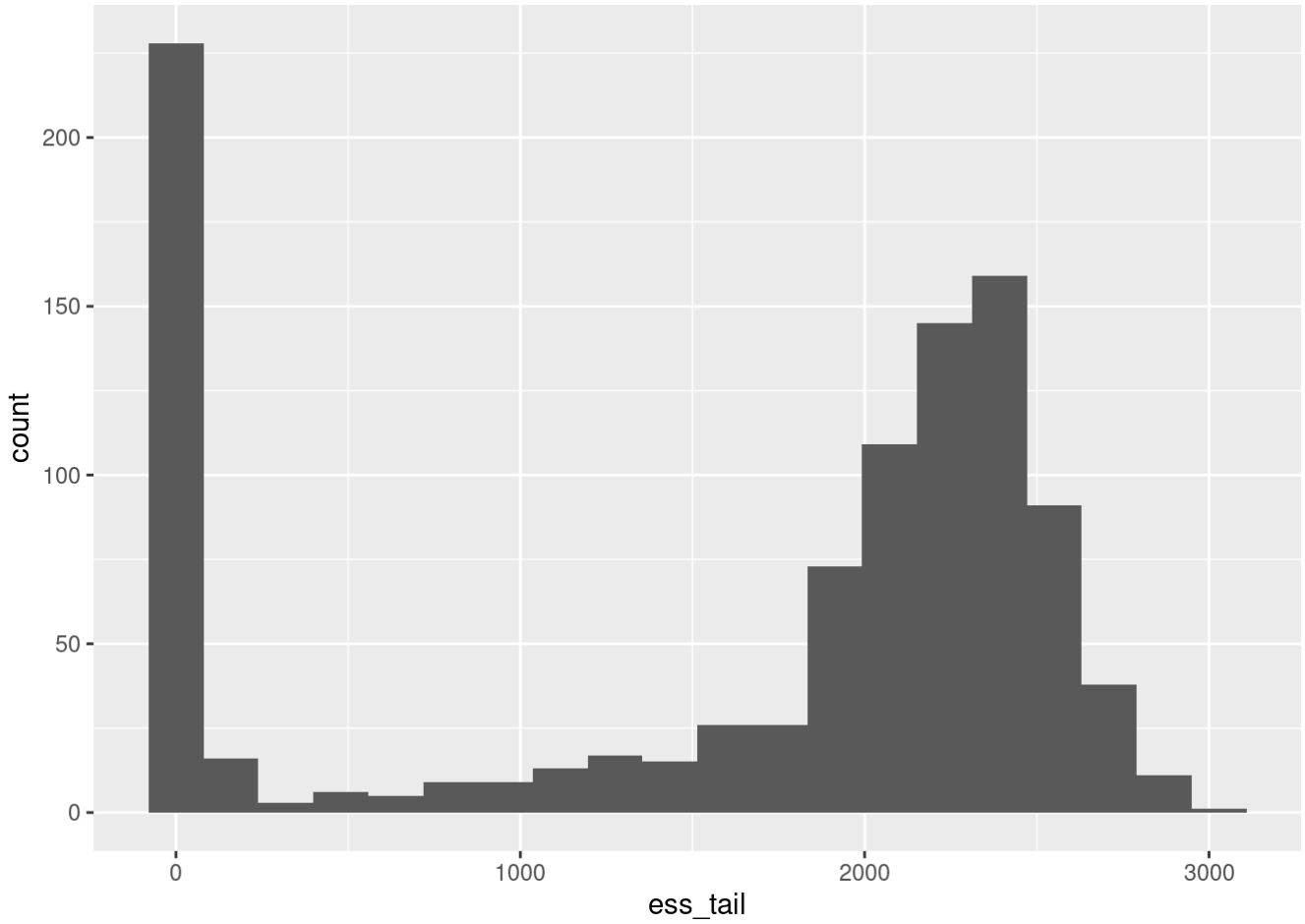


Effective sample size should ideally be high.

```
ggplot(fit_continuous) +
  geom_histogram(aes(x = ess_bulk), bins = 20)
```




```
ggplot(fit_continuous) +
  geom_histogram(aes(x = ess_tail), bins = 20)
```



Discrete covariate

```
tar_load(fit_discrete)
```

Here the analogous results for the discrete covariate simulations.

```
fit_discrete
```

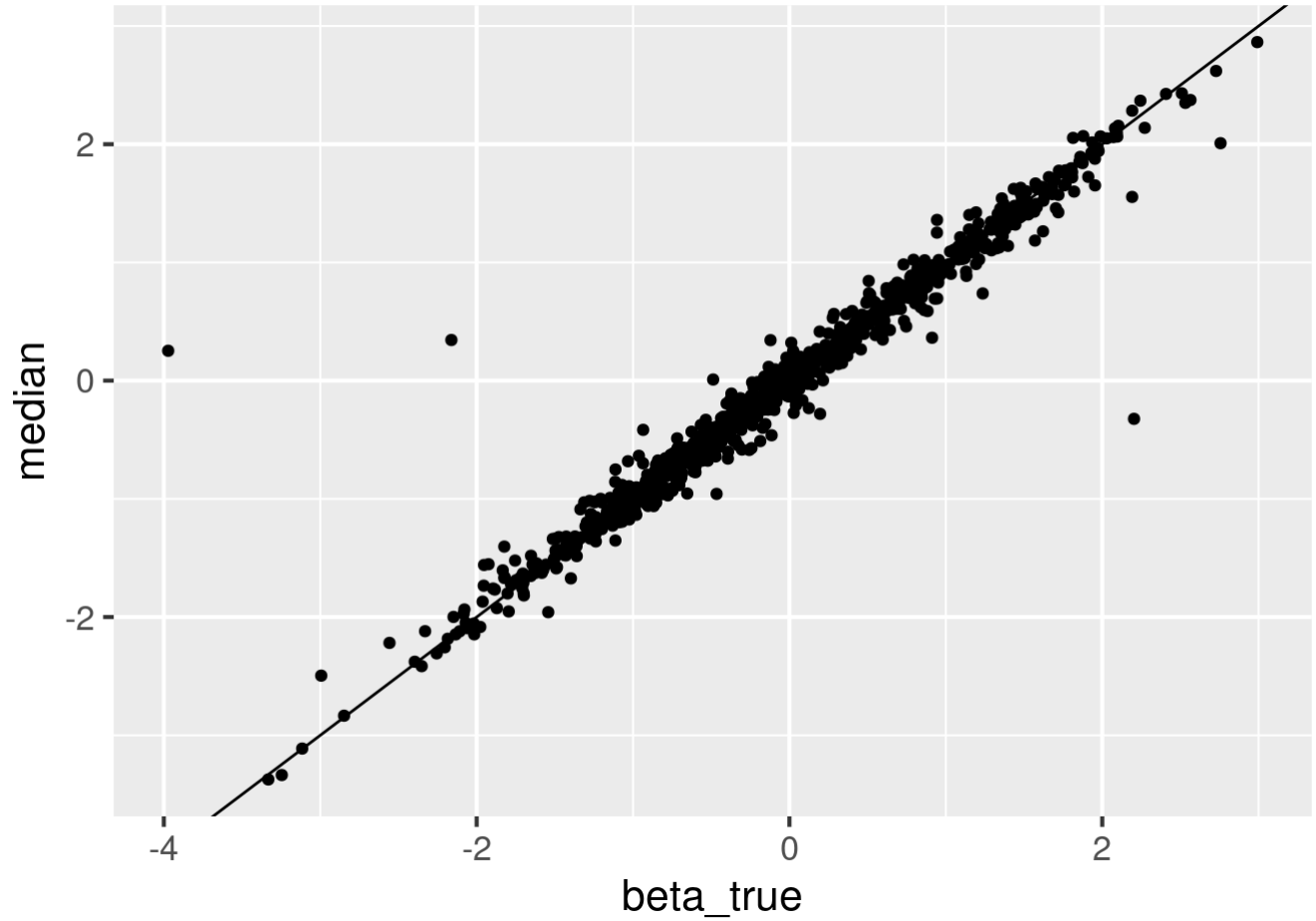
```
## # A tibble: 1,000 x 13
##    rep   variable    mean  median      sd     mad     q5    q95  rhat ess_bulk
##    <chr> <chr>      <dbl>   <dbl>   <dbl>   <dbl>  <dbl>  <dbl> <dbl>    <dbl>
##  1 repb… beta       0.108   0.139  0.259   0.138   -0.110  0.364  1.01     234.
##  2 repb… beta       0.598   0.598  0.0553  0.0540   0.505  0.690  1.00    2309.
##  3 repb… beta      -0.564  -0.565  0.176   0.179   -0.848 -0.276  1.00    1234.
##  4 repb… beta       0.355   0.355  0.0667  0.0656   0.245  0.461  1.00    2356.
##  5 repb… beta       0.0315  0.0339 0.157   0.153   -0.226  0.291  1.00    2331.
##  6 repb… beta       0.160   0.159  0.00981 0.00979  0.144  0.175  1.00    2010.
##  7 repb… beta      -0.299  -0.301  0.0651  0.0655  -0.408 -0.195  1.00    1914.
##  8 repb… beta      -2.38   -2.38   0.0256  0.0245  -2.42  -2.33   1.00    2537.
##  9 repb… beta      -0.399  -0.399  0.127   0.125   -0.609 -0.189  1.00    2000.
## 10 repb… beta       1.60    1.60   0.0432  0.0434   1.53   1.67   1.00    2059.
## # … with 990 more rows, and 3 more variables: ess_tail <dbl>, beta_true <dbl>,
## #   cover_beta <lgl>
```

```
mean(fit_discrete$cover_beta)
```

```
## [1] 0.906
```

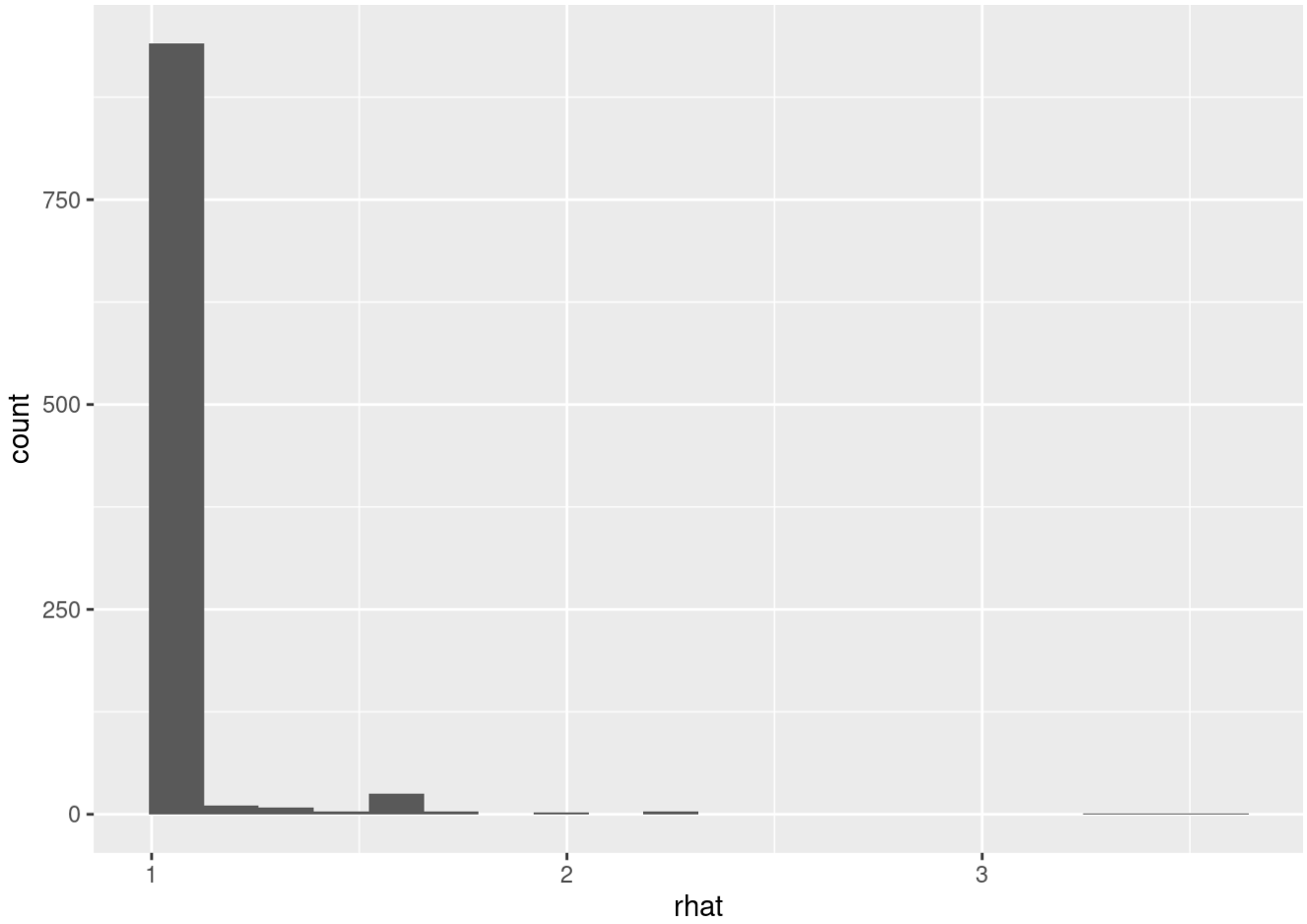
The posterior median of `beta` should be reasonably close to the true value.

```
ggplot(fit_discrete) +
  geom_point(aes(x = beta_true, y = median)) +
  geom_abline(intercept = 0, slope = 1) +
  theme_gray(16)
```



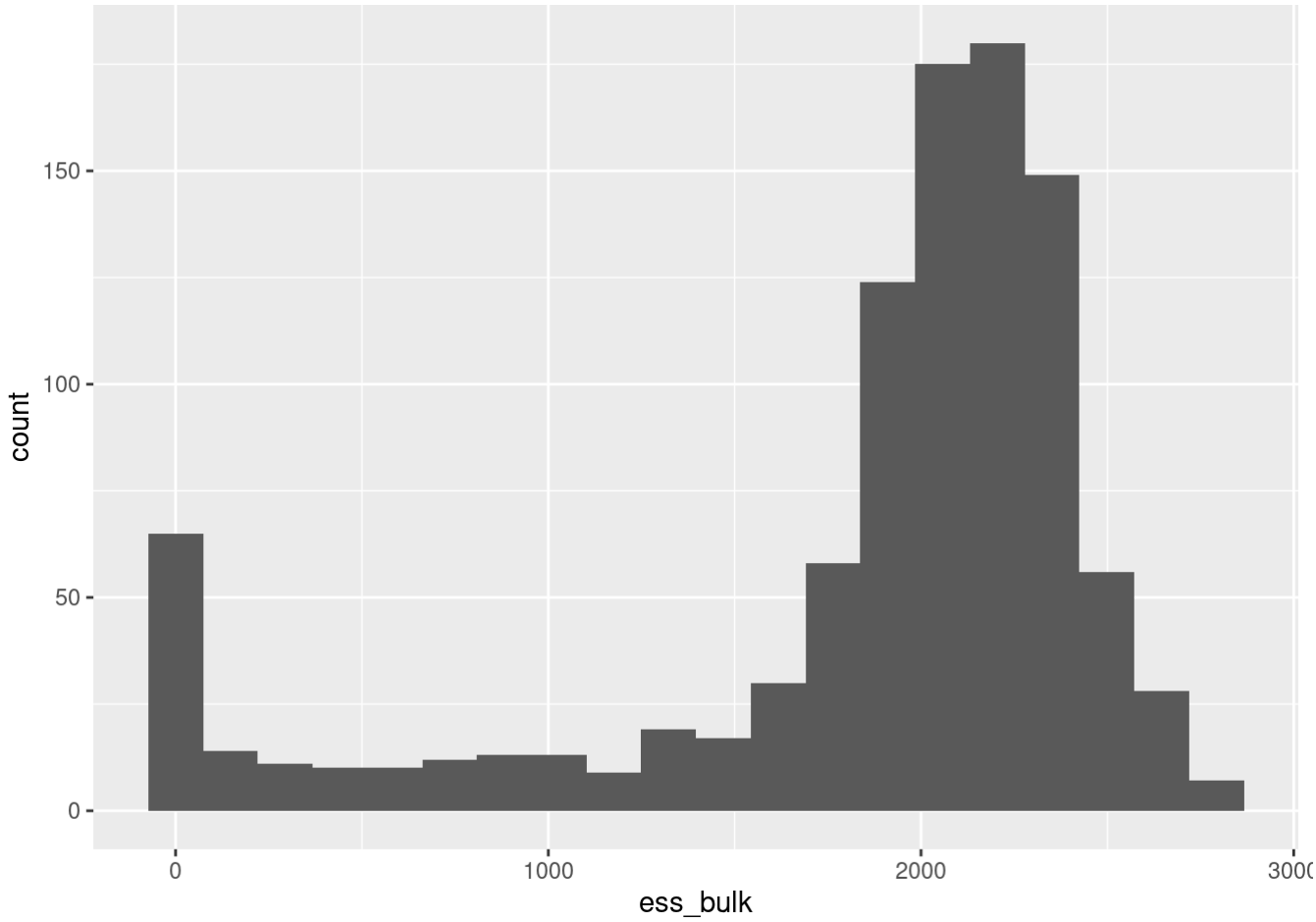
We should also check convergence diagnostics. `rhat` should ideally be close to 1.

```
ggplot(fit_discrete) +
  geom_histogram(aes(x = rhat), bins = 20)
```

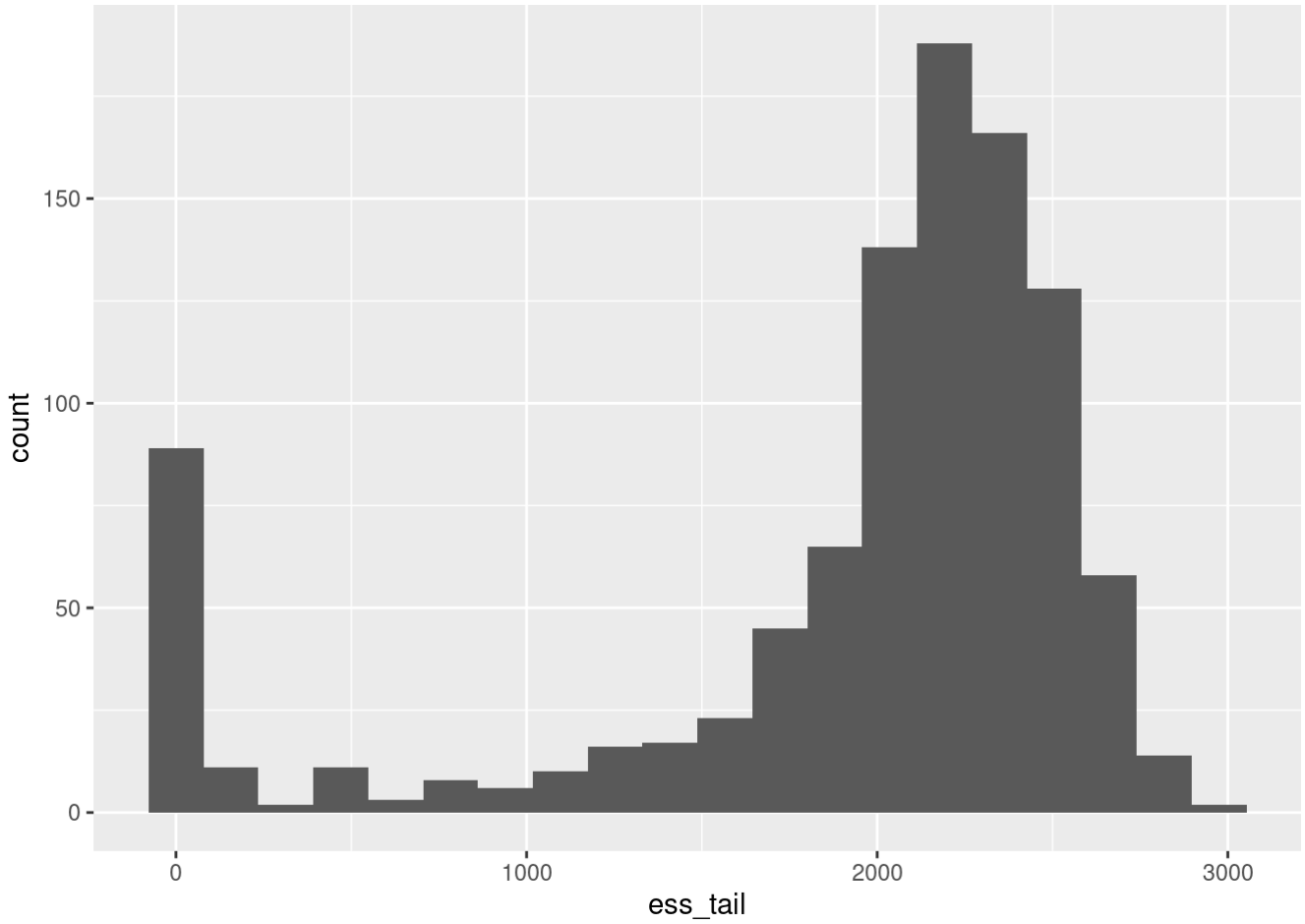


Effective sample size should ideally be high.

```
ggplot(fit_discrete) +
  geom_histogram(aes(x = ess_bulk), bins = 20)
```




```
ggplot(fit_discrete) +
  geom_histogram(aes(x = ess_tail), bins = 20)
```


150
100
50
0
count
0
1000
2000
3000
ess_tail