The bootstrap for sampling distributions

Assessing assumptions

- Our t-tests assume normality of variable being tested
- but, Central Limit Theorem says that normality matters less if sample is "large"
- in practice "approximate normality" is enough, but how do we assess whether what we have is normal enough?
- so far, use histogram/boxplot and make a call, allowing for sample size.

What actually has to be normal

- is: **sampling distribution of sample mean**
- the distribution of sample mean over *all possible samples*
- but we only have *one* sample!
- Idea: assume our sample is representative of the population, and draw samples from our sample (!), with replacement.
- This gives an idea of what different samples from the population might look like.
- Called *bootstrap*, after expression "to pull yourself up by your own bootstraps".

Blue Jays attendances

```
jays$attendance
```

```
##  [1] 48414 17264 15086 14433 21397 34743 44794 14184
##  [9] 15606 18581 19217 21519 21312 30430 42917 42419
## [17] 29306 15062 16402 19014 21195 33086 37929 15168
## [25] 17276
```

- A bootstrap sample:

```
s <- sample(jays$attendance, replace = TRUE)
s
```

```
##  [1] 21195 34743 21312 44794 16402 19014 34743 21195
##  [9] 17264 18581 19014 19217 34743 19217 14433 15062
## [17] 16402 15062 34743 15062 15086 15168 15086 48414
## [25] 30430
```

Getting mean of bootstrap sample

- A bootstrap sample is same size as original, but contains repeated values (eg. 15062) and missing ones (42917).
- We need the mean of our bootstrap sample:

```
mean(s)
```

```
## [1] 23055.28
```

- This is a little different from the mean of our actual sample:

```
mean(jays$attendance)
```

```
## [1] 25070.16
```

- Want a sense of how the sample mean might vary, if we were able to take repeated samples from our population.
- Idea: take lots of *bootstrap* samples, and see how *their* sample means vary.

Taking lots of bootstrap samples

- `rerun` does something as many times as you say. We just do 2 times to get the idea:

```
rerun(2, sample(jays$attendance, replace = TRUE))
```

```
## [[1]]
##  [1] 21195 34743 21312 44794 16402 19014 34743 21195
##  [9] 17264 18581 19014 19217 34743 19217 14433 15062
## [17] 16402 15062 34743 15062 15086 15168 15086 48414
## [25] 30430
##
## [[2]]
##  [1] 33086 44794 16402 30430 21195 21519 37929 21312
##  [9] 16402 19014 34743 15168 48414 15062 17264 14184
## [17] 18581 15606 33086 15606 17264 15168 34743 42917
## [25] 37929
```

Mean of each bootstrap sample

- Then take the mean of each of those:

```
rerun(2, sample(jays$attendance, replace = TRUE)) %>%
  map_dbl(~mean(.))
```

```
## [1] 23055.28 25512.72
```

- Last: make these into a dataframe:

```
rerun(2, sample(jays$attendance, replace = TRUE)) %>%
  map_dbl(~mean(.)) %>%
  enframe()
```

name	value
1	23055.28
2	25512.72

Do it many times

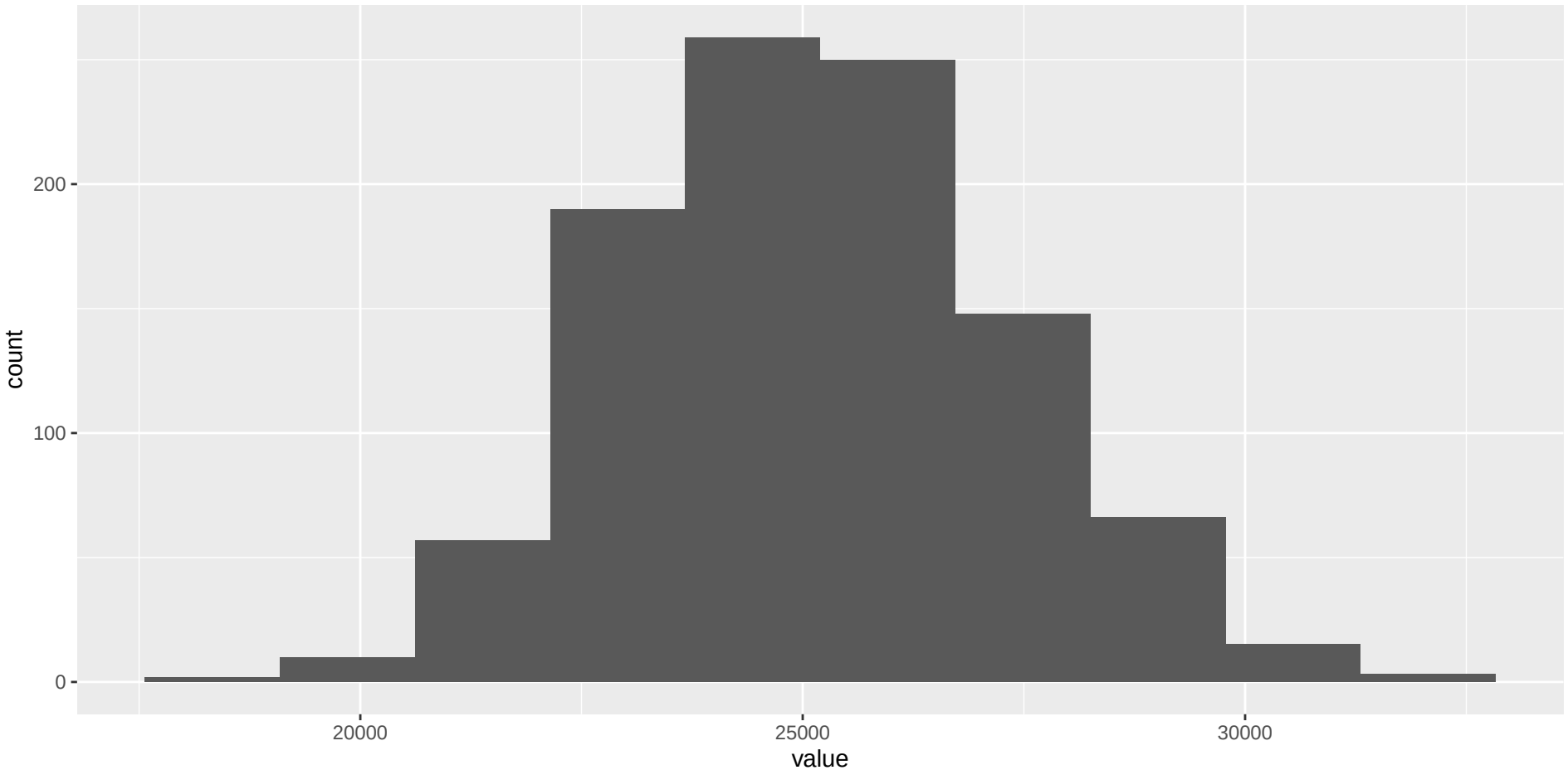
- Now that we know it works, replace 2 by 1000 (or larger) and save result:

```
rerun(1000, sample(jays$attendance, replace = TRUE)) %>%
  map_dbl(~mean(.)) %>%
  enframe() -> d
d
```

name	value
1	25562.52
2	29197.64
3	23614.68
4	28472.20
5	28647.52
6	23328.88
7	24808.20
8	24664.64
9	27186.36
10	25509.20
11	26449.16

Are these normal?

```
ggplot(d, aes(x=value)) + geom_histogram(bins=10)
```



Comments

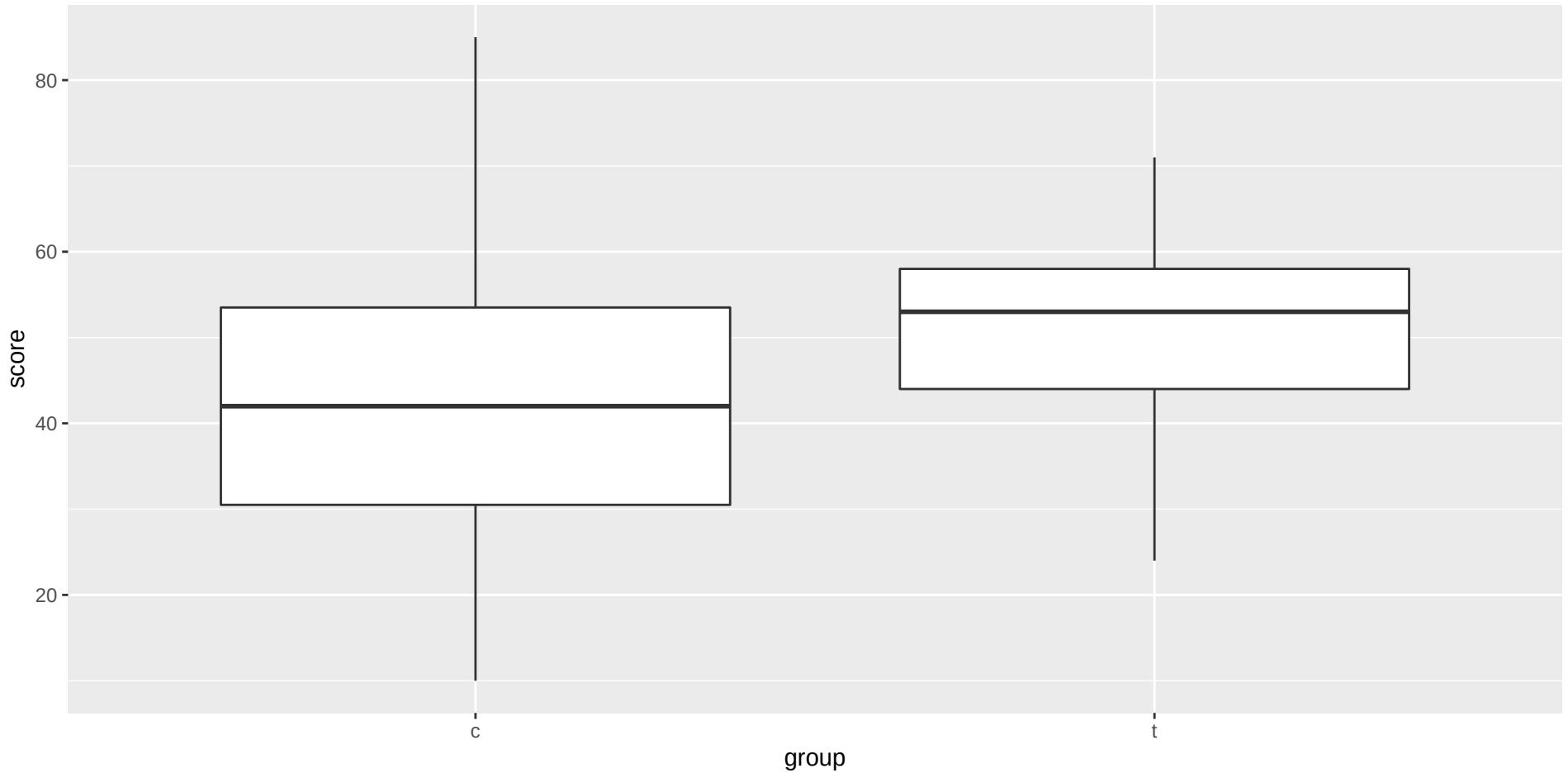
- This is very close to normal
- The bootstrap says that the sampling distribution of the sample mean is close to normal, even though the distribution of the data is not
- A sample size of 25 is big enough to overcome the skewness that we saw
- This is the Central Limit Theorem in practice
- It is surprisingly powerful.
- Thus, the t-test is actually perfectly good here.

Two samples

- Assumption: *both* samples are from a normal distribution.
- In practice, each sample is "normal enough" given its sample size, since Central Limit Theorem will help.
- Use bootstrap on each group independently, as above.

Kids learning to read

```
ggplot(kids, aes(x=group, y=score)) + geom_boxplot()
```



Getting just the control group

```
kids %>% filter(group=="c") -> controls
controls
```

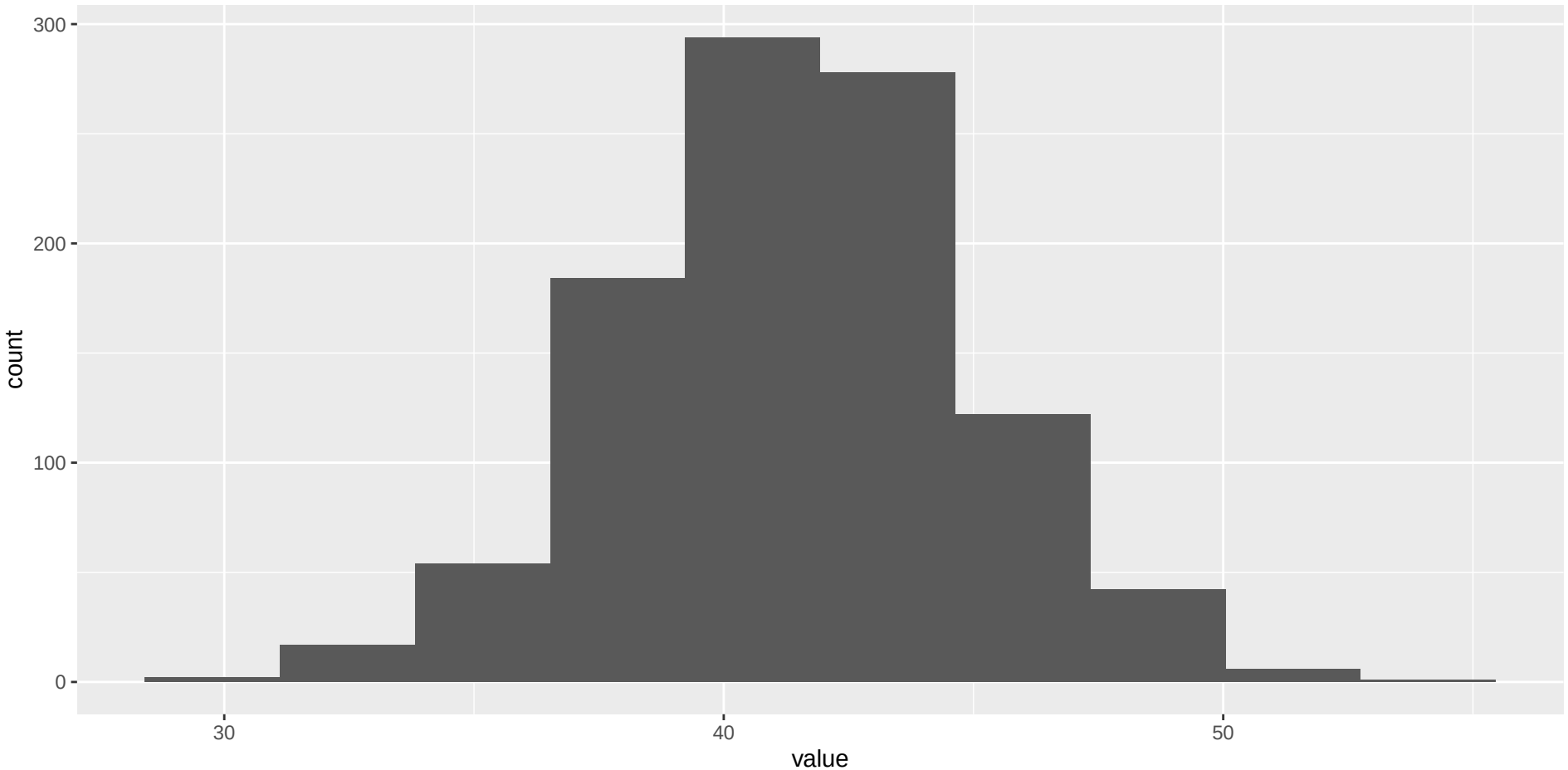
group	score
c	42
c	33
c	46
c	37
c	43
c	41
c	10
c	42
c	55
c	19
c	17
c	55

Bootstrap these

```
rerun(1000, sample(controls$score, replace = TRUE)) %>%
  map_dbl(~mean(.)) %>%
  enframe() -> d
```

Plot

```
ggplot(d, aes(x=value)) + geom_histogram(bins=10)
```

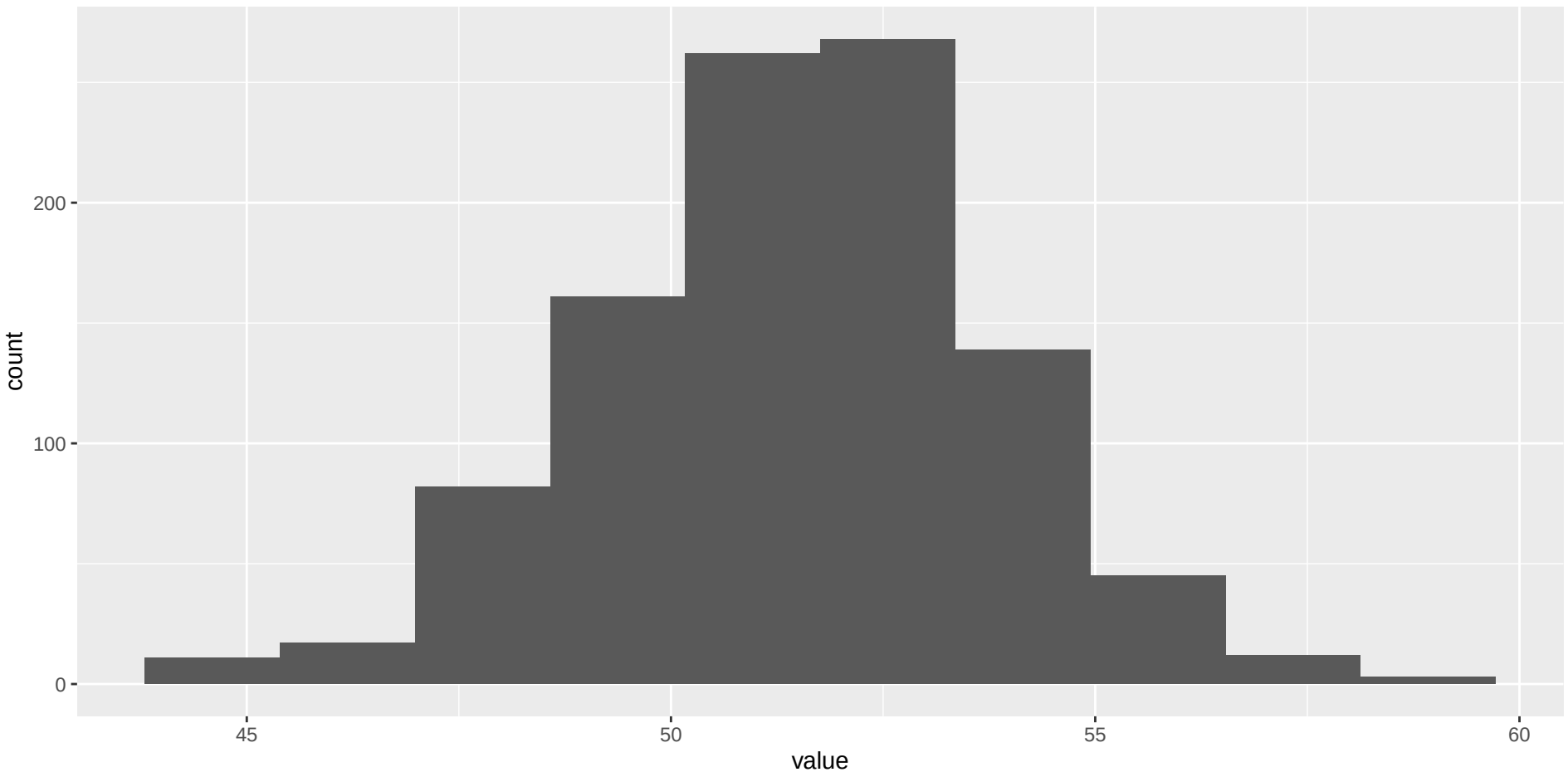


... and the treatment group:

```
kids %>% filter(group=="t") -> treats
rerun(1000, sample(treats$score, replace = TRUE)) %>%
  map_dbl(~mean(.)) %>%
  enframe() %>%
  ggplot(aes(x=value)) + geom_histogram(bins=10) -> g
```

Histogram

```
g
```



Comments

- sampling distributions of sample means both look pretty normal
- as we thought, no problems with our two-sample t at all.