Probability in-class demo

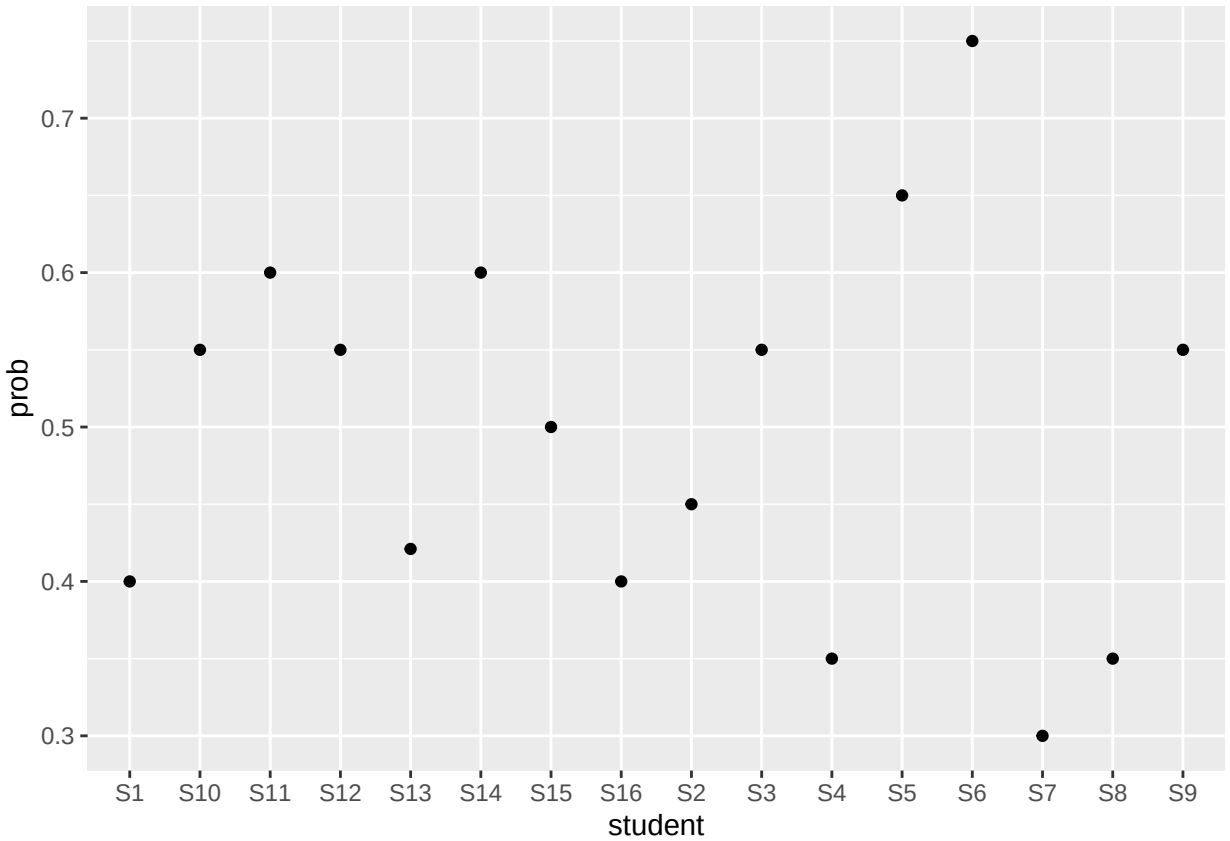
Grusha Prasad

2024-10-17

Visualizing parity dat

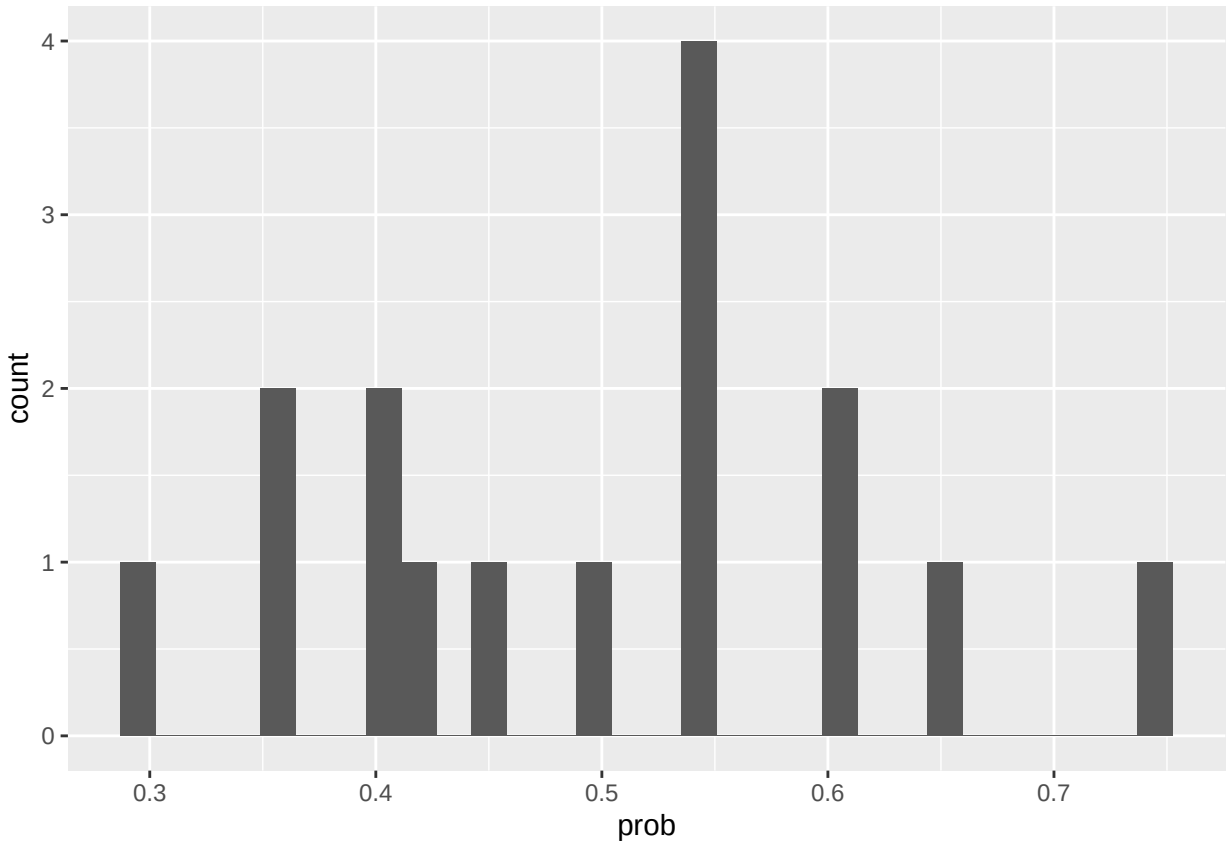
```
parity_dat = read.csv('parity.csv')%>%
  select(-Trial) %>%
  gather(key="student", value="acc") %>%
  group_by(student) %>%
  summarize(prob = mean(acc, na.rm=TRUE))
```

```
ggplot(parity_dat, aes(x=student, y=prob)) +
  geom_point()
```




```
ggplot(parity_dat, aes(x=prob)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Binomial distribution

Lets assume that all students are at chance with guessing the parity (i.e., no mind readers amongst us)

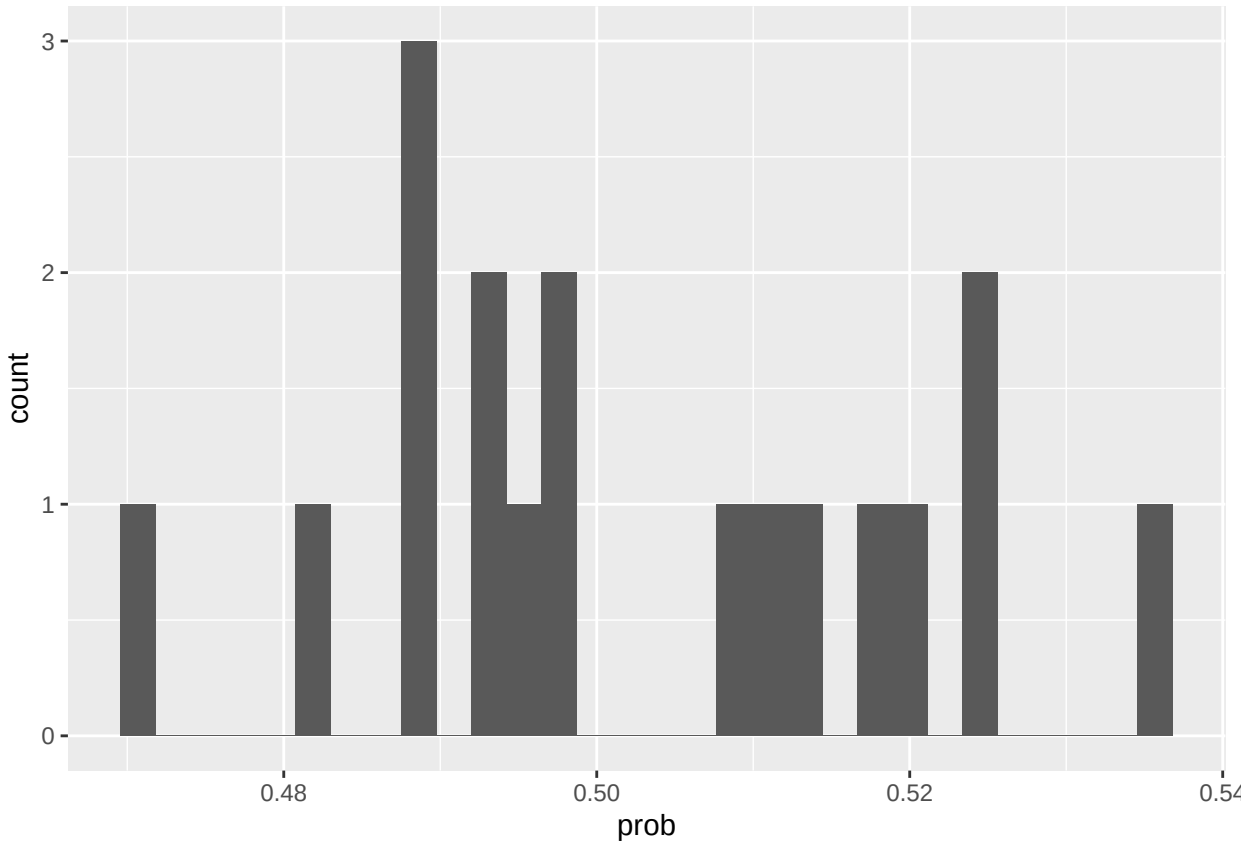
Ths code below simulates data for 18 students and gives the proportion of correct guesses for 1000 guesses.

```
num_guesses = 1000

parity_sim = data.frame(student = c(1:18),
                        success = rbinom(n=18, size=num_guesses, prob=0.5)) %>%
  mutate(prob = success/num_guesses)
```

```
ggplot(parity_sim, aes(x=prob)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

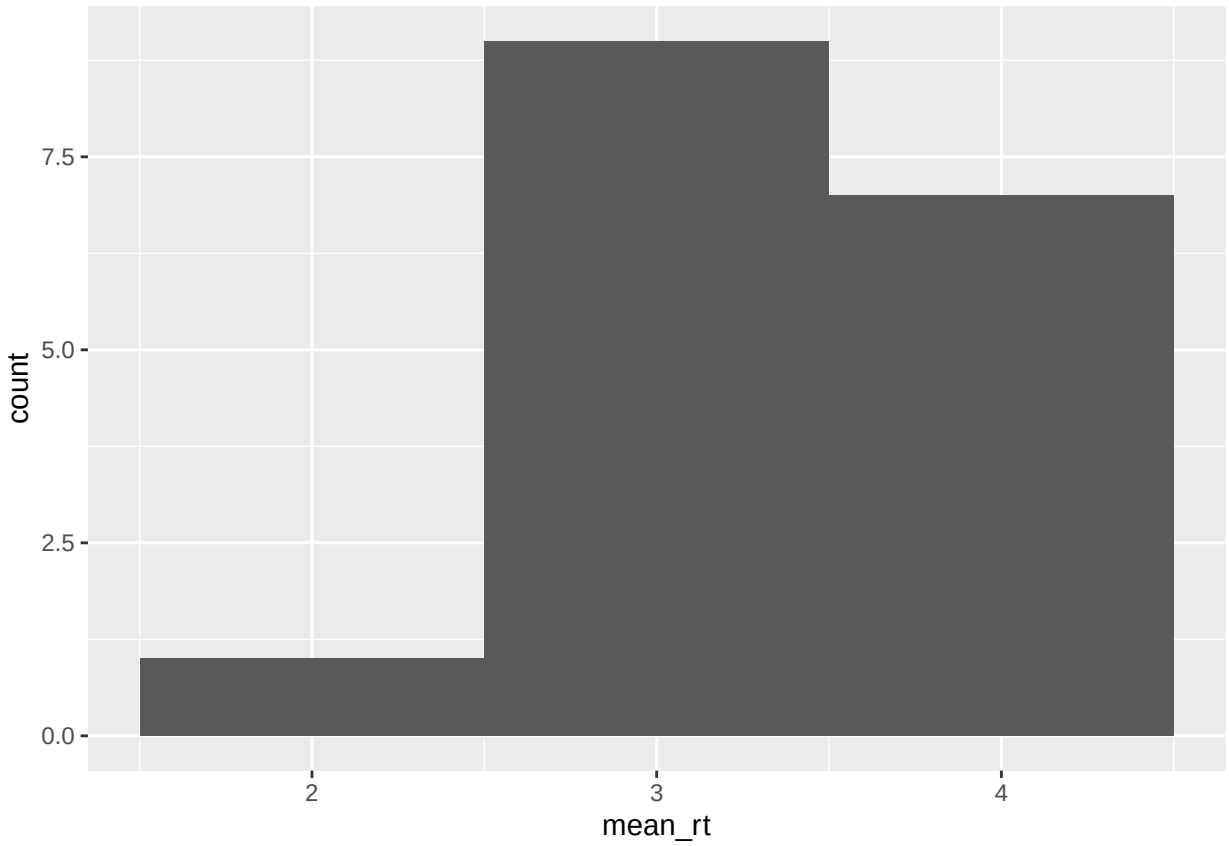


Visualizing recounting time data

```
rt_dat = read.csv('rt.csv')%>%
  select(-Trial) %>%
  gather(key="student", value="rt") %>%
  group_by(student) %>%
  summarize(mean_rt = mean(rt, na.rm=TRUE))
```

```
ggplot(rt_dat, aes(x=mean_rt)) +
  geom_histogram(binwidth = 1)
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_bin()`).
```



Normal distribution

Lets assume that when asked to recite the English alphabet rapidly people take on average 3.5 seconds to with a standard deviation of 1 second. The code below simulates mean RTs for 18 students under these assumptions.

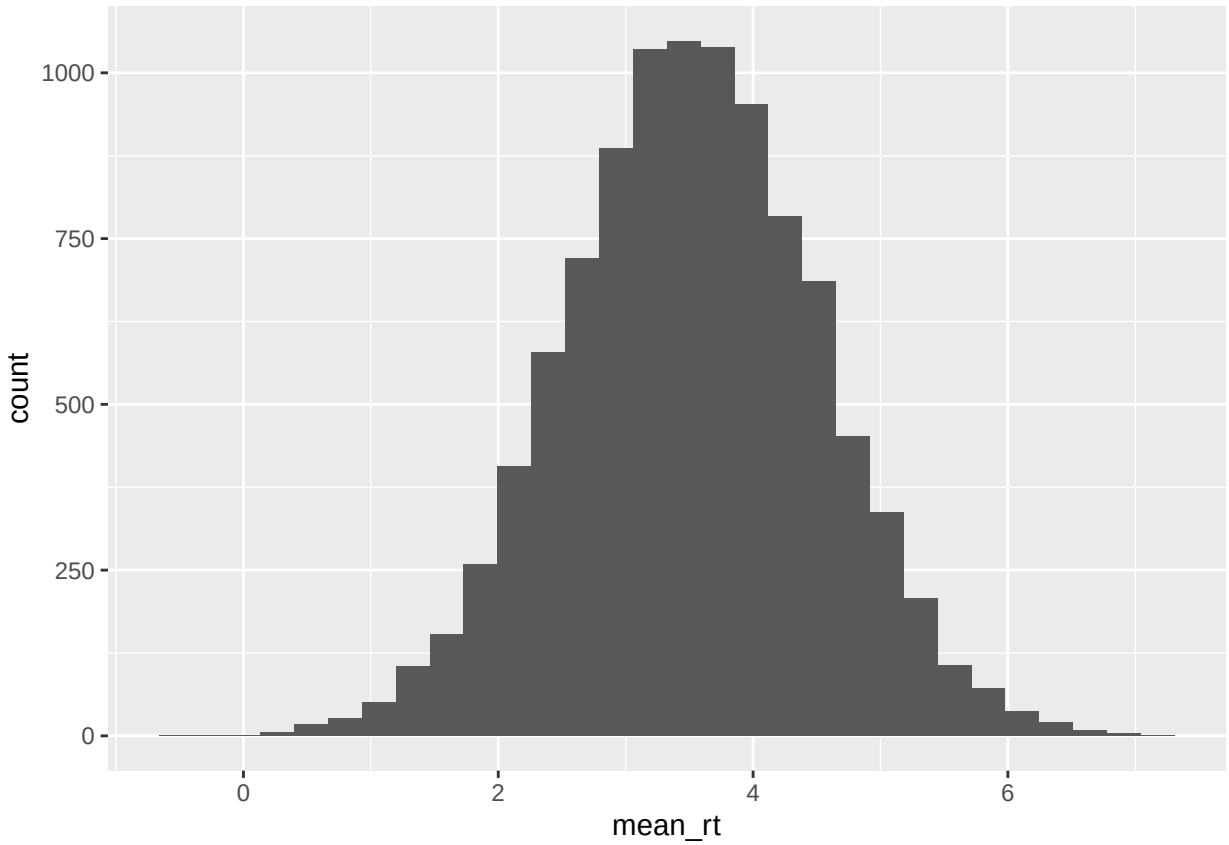
Note, unlike with the binomial distribution we don't need to specify the exact number of trials that this average was computed over or the proportion of success. Instead, for a normal distribution we need to specify the mean and standard deviation.

```
num = 10000

rt_sim = data.frame(student = c(1:num),
                    mean_rt = rnorm(n=num, mean=3.5, sd=1))
```

```
ggplot(rt_sim, aes(x=mean_rt)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


1000
750
500
250
0
count
0
2
4
6
mean_rt