Possible Protocol 1(PP1) : roll once; if get 6 then conclude the dice is not fair; if roll any other number then conclude it is fair. Analyze PP1: if the dice were fair, what is the probability it would be judged to be unfair? Oppositely, if the dice were unfair, what is the probability that it would be judged to be fair?

If the dice were fair, the probability of getting a 6 is P(6)=1/6

If the dice were unfair, P(1,2,3,4,5)=5/6

PP2: roll the dice 20 times. (Each person should have done this beforehand.) Group can specify a decision rule to judge that dice is fair or unfair. Consider the stats question: if fair dice are rolled 20 times, what is likely number of 6 resulting? How unusual is it, to get 1 more or less than that? How unusual is it, to get 2 more or less? 3? Analyze PP2 including the question: if the dice were fair, what is the chance it could be judged as unfair?

if fair dice are rolled 20 times, what is likely number of 6 resulting? P(20/6)=3.3333333…means, 3.3333 is likely number of 6 resulting. Which integer is 3 or 4

We make assumssion as follow:

1. Hypothesis A: the observed value matches the excepted value: p1=p2=p3=p4=p5=p6

Hypothesis B: don't match the expected value.

1. We set up 8 times of six in 20 trial as the level of confidence and claim that the dice is unfair.
2. Gather and interpret data.

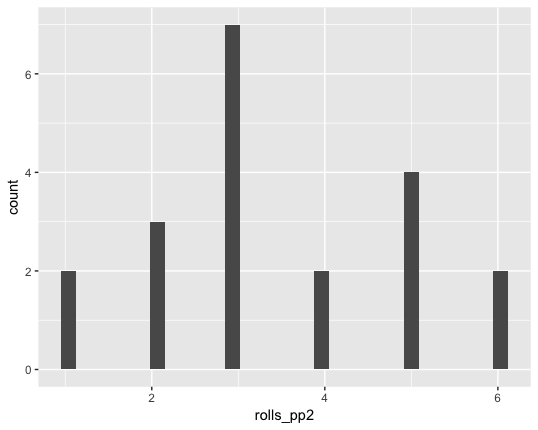
> rolls\_pp2<-sample(1:6,20,rep=T)

> print(rolls\_pp2)

[1] 3 4 3 5 2 6 6 3 1 5 5 3 3 1 3 2 3 5 2 4

> library(ggplot2)

> qplot(rolls\_pp2,binwith=1)



> chisq.test(table(rolls\_pp2), p = rep(1/6, 6))

Chi-squared test for given probabilities

data: table(rolls\_pp2)

X-squared = 5.8, df = 5, p-value = 0.3262

Thus, we fail to reject the null hypothesis. there is no sufficient evidence to support the claim that the expected value doesn’t match the expected value. The dice is fair.

the probability of getting 6 five time

x<-pbinom(5,20,1/6)

> print(x)

[1] 0.8981595

> x<-pbinom(4,20,1/6)

> print(x)

[1] 0.7687492

> x<-pbinom(3,20,1/6)

> print(x)

[1] 0.5665456

> x<-pbinom(2,20,1/6)

> print(x)

[1] 0.3286591

The probability of getting less and less six become smaller, vice versa.

PP3: roll 100 times and specify decision rules. Some cases are easy: if every roll comes to 6 then might quickly conclude. But what about the edge cases? Is it fair to say that every conclusion has some level of confidence attached? Where do you set boundaries for decisions? Analyze PP3. What is the chance that fair dice could be judged to be unfair?

> rolls\_dice\_pp3<-sample(1:6,100,rep=T)

> print(rolls\_dice\_pp3)

[1] 6 6 3 2 4 4 5 4 2 2 6 2 1 6 4 4 4 5 4 3 3 4 5 4 4 4 4 3 5 3 4 2 3 2 1 2 6 2

[39] 5 4 6 4 6 5 6 4 2 6 4 6 5 4 4 3 1 4 1 1 2 6 3 4 6 1 4 4 2 6 3 4 3 3 1 2 3 1

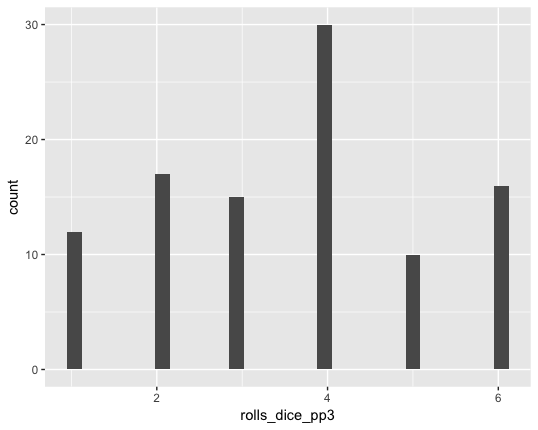
[77] 4 3 1 5 4 6 2 2 6 4 5 1 2 4 4 3 3 1 6 5 2 4 2 1

> library(ggplot2)

> qplot(rolls\_dice\_pp3,binwith=1)

Warning: Ignoring unknown parameters: binwith

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



> chisq.test(table(rolls\_dice),p=rep(1/6,6))

Chi-squared test for given probabilities

data: table(rolls\_dice)

X-squared = 7.76, df = 5, p-value = 0.17

Conclusion for PP3, 100 times are still not enough to reject the dice is fair.

OK now you can actually look at the data, maybe actually roll the dice. Only now will you actually perform the experiment. What conclusion do you draw from your experiment? How confident are you, in this conclusion? How would you revise your protocol if you were to devise EP2?

We propose to roll the dice for 1000 times, set as EP1:

> roll\_dice\_ep1<-sample(1:6,1000,rep=T)

> print(roll\_dice\_ep1)

[1] 4 1 1 5 5 5 4 1 3 3 1 3 6 3 3 1 6 2 4 6 4 6 6 1 3 3 3 6 3 4 3 3 3 3 3 4 2 2

[39] 5 6 1 2 6 4 6 6 2 6 1 4 6 3 3 5 1 6 2 3 5 6 6 1 6 2 4 3 5 3 5 4 2 2 5 4 5 5

[77] 1 5 5 1 3 1 3 1 2 5 6 4 4 3 4 5 3 4 3 2 1 4 4 6 2 5 6 3 6 4 1 2 6 5 1 3 5 4

[115] 2 1 6 5 1 1 1 5 3 4 1 1 4 4 2 2 4 5 6 6 3 2 3 5 2 3 1 6 5 4 2 3 2 6 3 3 1 2

[153] 2 2 2 6 4 2 4 5 5 3 2 1 1 2 1 2 5 1 4 2 4 5 1 1 2 6 2 5 2 1 3 2 3 1 3 5 2 1

[191] 5 5 1 2 2 3 5 4 6 3 6 6 3 3 2 1 1 2 1 5 5 1 6 1 4 6 6 5 2 3 4 1 1 2 4 1 3 3

[229] 2 1 6 4 1 3 1 3 5 3 3 1 2 6 3 3 4 1 5 5 6 2 1 4 2 2 2 1 1 3 2 2 1 4 3 2 6 4

[267] 6 4 2 3 4 3 2 2 5 2 3 5 2 5 2 2 5 2 5 4 6 6 2 4 5 5 3 4 2 2 1 5 1 4 5 6 6 4

[305] 3 6 1 1 1 1 2 2 2 6 2 2 6 4 6 2 6 5 5 3 5 6 4 6 2 2 4 5 3 1 4 6 2 1 3 5 4 6

[343] 6 4 2 6 3 1 1 6 6 3 6 3 2 3 5 3 1 5 3 3 5 5 3 1 3 3 3 1 1 5 3 3 6 1 6 1 4 1

[381] 6 3 5 2 4 3 4 4 6 3 2 2 6 2 1 5 4 6 3 6 6 6 2 1 3 3 3 4 4 4 6 1 6 1 6 5 4 6

[419] 2 4 2 2 4 3 3 6 2 5 1 2 5 6 1 6 3 3 1 6 3 5 1 4 5 3 4 1 5 6 1 1 2 5 6 1 1 4

[457] 2 1 6 2 4 5 3 3 5 4 5 2 6 2 3 4 6 1 1 4 1 5 2 1 5 2 6 5 2 2 4 5 1 3 1 5 1 1

[495] 1 6 3 4 1 2 5 2 6 2 6 6 6 6 6 4 5 6 4 5 4 4 4 4 4 6 6 4 2 3 1 4 2 4 6 1 1 4

[533] 2 2 2 6 5 4 4 3 6 6 4 2 2 6 6 6 5 6 3 4 2 6 2 3 6 6 5 5 6 5 3 5 4 6 6 3 2 3

[571] 1 2 5 1 4 5 3 5 4 4 5 1 6 3 2 6 1 1 2 4 1 5 3 1 6 3 1 2 3 4 6 3 5 5 5 6 2 2

[609] 4 2 4 6 5 2 5 4 5 3 2 1 5 3 6 4 5 6 2 4 4 2 6 5 4 5 2 3 1 1 2 6 6 3 5 2 3 5

[647] 2 2 1 1 3 5 4 4 6 2 3 2 6 1 4 3 5 3 3 1 4 4 6 2 2 1 6 2 6 3 6 1 1 6 3 2 1 6

[685] 1 5 6 3 3 3 3 6 1 4 2 6 1 4 4 4 2 2 2 5 2 3 3 4 6 5 6 5 4 2 2 5 2 3 3 3 4 5

[723] 2 4 2 4 1 3 6 3 3 3 6 3 1 5 4 5 4 1 2 2 1 5 1 6 5 4 1 5 1 4 1 2 4 6 5 2 6 5

[761] 5 1 3 3 3 2 1 6 2 3 3 6 4 3 4 5 1 5 4 4 6 3 6 3 2 4 2 3 6 1 1 1 4 5 1 3 5 3

[799] 4 2 1 1 5 3 6 1 1 3 3 6 2 5 6 3 3 3 5 6 6 1 2 2 6 6 2 4 1 5 5 3 6 4 6 2 2 2

[837] 1 1 5 3 1 2 3 2 6 3 2 3 5 1 4 6 5 1 3 6 5 5 3 6 1 6 6 2 6 1 5 3 6 5 2 4 1 2

[875] 1 4 6 6 5 1 5 4 4 3 5 1 6 3 3 2 4 5 1 3 3 1 4 1 5 1 6 4 6 6 2 1 3 1 2 1 2 5

[913] 2 5 4 2 2 3 3 4 2 1 2 4 2 6 5 6 3 6 6 4 6 1 6 6 2 2 5 4 2 6 1 5 5 1 2 5 2 4

[951] 3 4 6 3 2 2 1 6 1 6 1 6 5 6 6 2 6 3 4 2 5 1 1 5 2 5 1 1 4 5 5 5 5 4 3 2 4 3

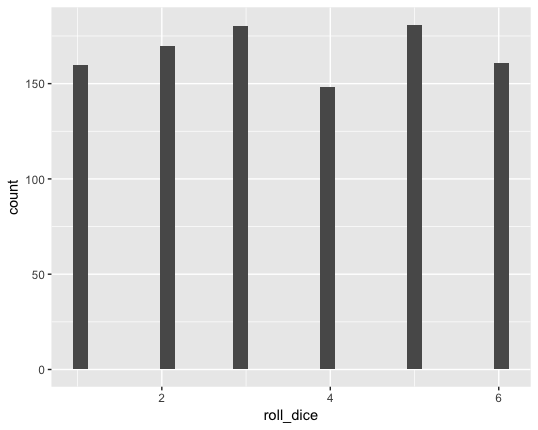
[989] 1 5 4 1 1 3 4 6 5 6 6 6

> library(ggplot2)

> qplot(roll\_dice,binwith=1)

Warning: Ignoring unknown parameters: binwith

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



We can see with the population becomes larger, the distribution of each side become equally smooth bit by bit.

Our assumption is if the dice is a fair dice than the mean value should be equal to 3.5, and we set the mean value diff should less than 0.025 compare to 3.5(more or less):

**PP2**

> t.test(rolls\_pp2,mu=3,alternative="greater",conf.level = 0.95)

One Sample t-test

data: rolls\_pp2

t = 1.3385, df = 19, p-value = 0.09826

alternative hypothesis: true mean is greater than 3

95 percent confidence interval:

2.868676 Inf

sample estimates:

mean of x

3.45 -----DIFF 3.5-3.45=0.05

**PP3**

> t.test(rolls\_dice\_pp3,alternative="greater",conf.level = 0.95)

One Sample t-test

data: rolls\_dice\_pp3

t = 22.624, df = 99, p-value < 2.2e-16

alternative hypothesis: true mean is greater than 0

95 percent confidence interval:

3.307995 Inf

sample estimates:

mean of x

3.57 -----DIFF 3.5-3.57=-0.07

**EP1**

> t.test(roll\_dice\_ep1,mu=3,alternative="greater",conf.level = 0.95)

One Sample t-test

data: roll\_dice\_ep1

t = 8.6401, df = 999, p-value < 2.2e-16

alternative hypothesis: true mean is greater than 3

95 percent confidence interval:

3.384488 Inf

sample estimates:

mean of x

3.475-----DIFF 3.5-3.475=0.025

So our final conclusion is: as the sample quantity getting larger enough, in our case 1000 times is a reasonable quantity trial. Which get the mean value as 3.475 which only has 0.025 difference with our assumption of 3.5.