All of Statistics: Probability

Giang Le

4/21/2022

Selected Exercises and My Solutions

Exercise 11

Suppose that A and B are independent events. Show that A^c and B^c are independent events.

Solution: In order to show that A^c and B^c are independent events, I need prove that $P(A^c \cap B^c) = P(A^c)P(B^c)$.

By the law of total probability, $P(A) = P(B \cap A) + P(B^c \cap A) = P(B)P(A) + P(B^c \cap A)$ because A and B are independent.
$\therefore P(B^c \cap A) = P(A) - P(B)P(A) = P(A)(1 - P(B)) = P(A)P(B^c)$

Similarly, $P(A^c \cap B) = P(A^c)P(B)$

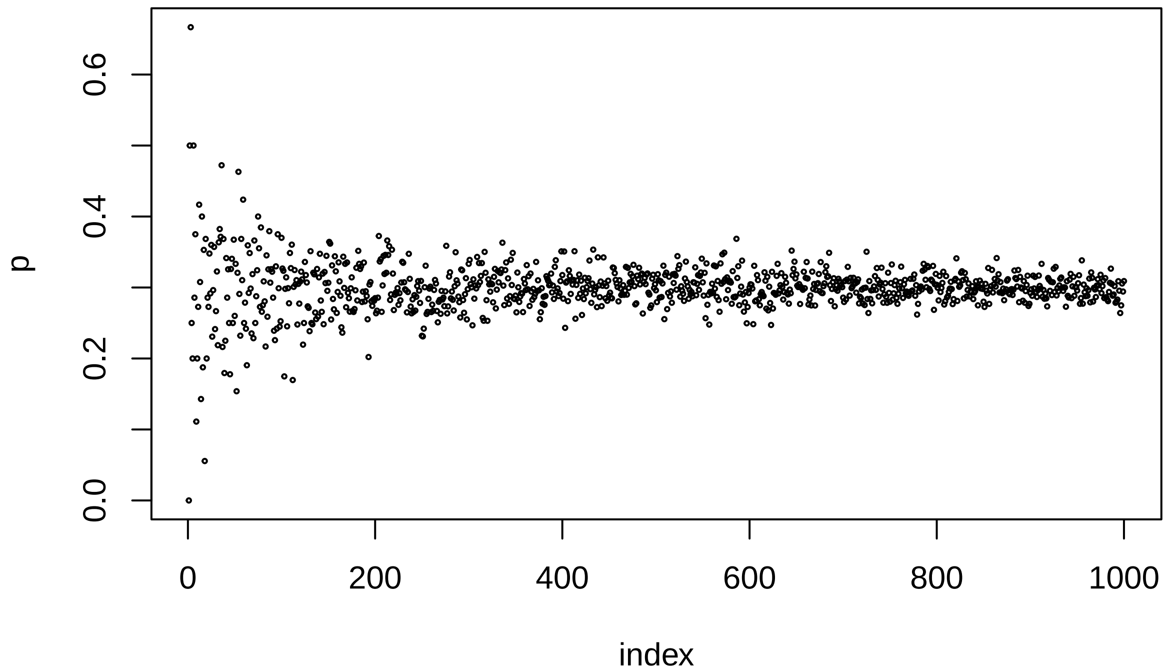
By the law of total probability, $P(A^c) = P(A^c \cap B) + P(A^c \cap B^c) = P(A^c)P(B) + P(A^c \cap B^c)$
$\therefore P(A^c \cap B^c) = P(A^c) - P(A^c)P(B) = P(A^c)(1 - P(B)) = P(A^c)P(B^c)$
So A^c and B^c are independent events (QED).

Exercise 21

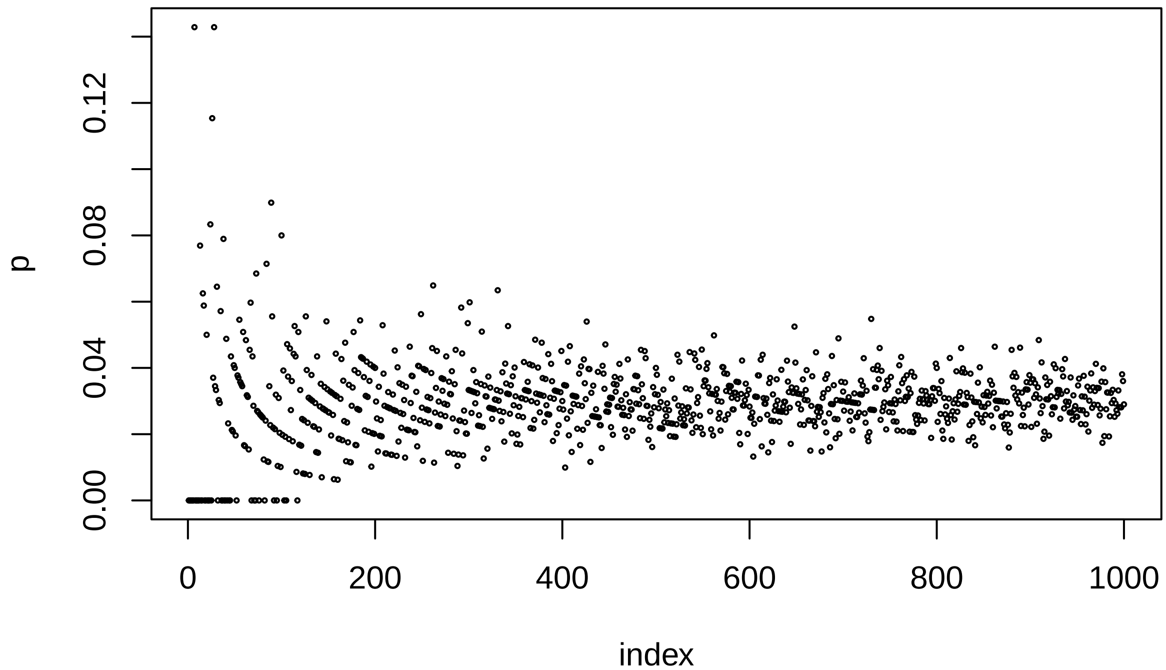
Suppose a coin has probability p of falling heads up. If we flip the coin many times, we would expect the proportion of heads to be near p. Take $p = .3$ and $n = 1000$ and simulate n coin flips. Plot the proportion of heads as a function of n. Repeat for $p = .03$

Simulate the coin flip by using a Bernoulli distribution.

```
library(purrr)
set.seed(123)

index=seq(1:1000)
p = vector(mode="numeric", length=length(index))

for (i in index) {
  sim = rbernoulli(n=i, p=0.3)
  p[i] = sum(sim)/length(sim)
}
plot(index, p, cex=0.3)
```




```
library(purrr)
set.seed(123)

index=seq(1:1000)
p = vector(mode="numeric", length=length(index))

for (i in index) {
  sim = rbernoulli(n=i, p=0.03)
  p[i] = sum(sim)/length(sim)
}
plot(index, p, cex=0.3)
```



Exercise 22

Suppose we flip a coin n times and let p denote the probability of heads. Let X be the number of heads. We call X a binomial random variable. Intuition suggests that X will be close to np. To see if this is true, we can repeat this experiment many times and average the X values. Carry out a simulation and compare the average of the X's to np. Try this for $p = .3$ and $n = 10$, $n = 100$, and $n = 1000$.

```
set.seed(345)

simulation <- function(n) {
  flips = rbernoulli(n = n, p = .3)
  num_heads = sum(flips)
}

mean(replicate(n = 1000, simulation(n=10)))
```

```
## [1] 2.958
```

```
mean(replicate(n = 1000, simulation(n=100)))
```

```
## [1] 30.071
```

```
mean(replicate(n = 1000, simulation(n=1000)))
```

```
## [1] 300.938
```

The averages of X's are close to np (3, 30, 300 respectively).