hw 4 solution

Statistical Computing, Jieun Shin

Autumn 2021

문제 1.

1. 문제에서 지정한 승산합동법 알고리즘 함수를 정의한다.

```
RANDU = function(n, seed = 1){
  x = NULL
  a = 2^16 + 3
  m = 2^31

  for (i in 1:n){
    seed = (a * seed) %% m
    x[i] = seed / m
  }

  return (x)
}
```

2. RANDU로부터 난수 100개를 생성하였다.

```
N = 100
x = RANDU(N)
x
```

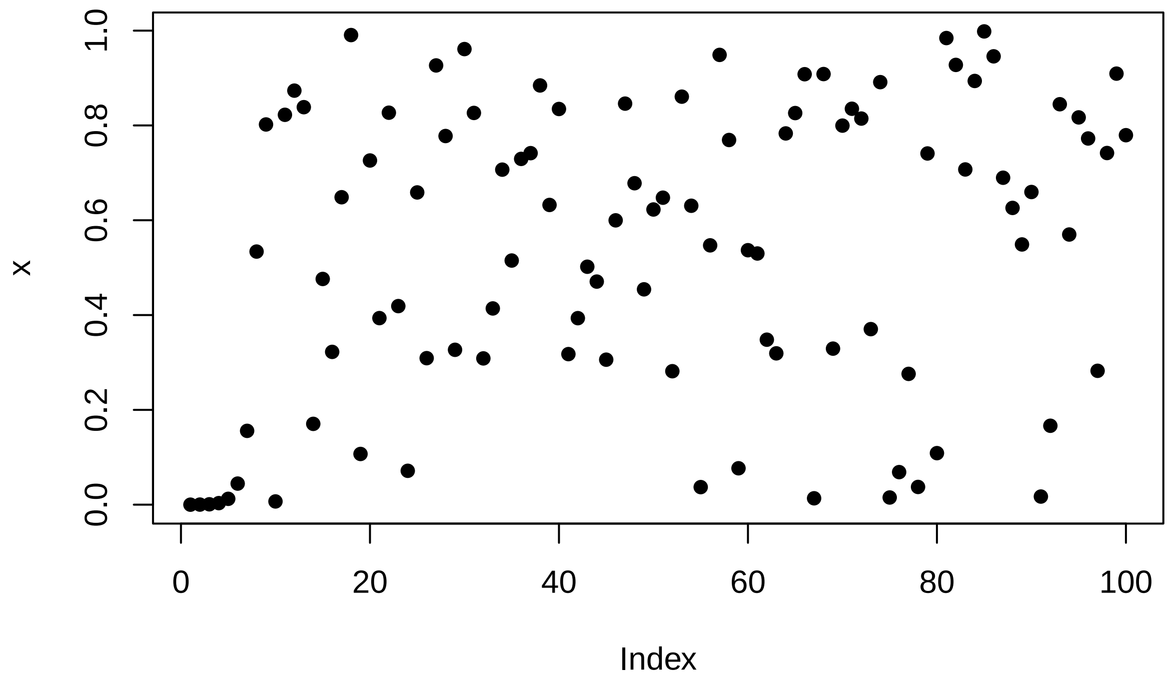
```
##   [1] 3.051898e-05 1.831097e-04 8.239872e-04 3.295936e-03 1.235973e-02
##   [6] 4.449497e-02 1.557322e-01 5.339386e-01 8.020416e-01 6.802399e-03
##  [11] 8.224397e-01 8.734164e-01 8.385415e-01 1.705012e-01 4.761336e-01
##  [16] 3.222913e-01 6.485450e-01 9.906484e-01 1.069856e-01 7.260775e-01
##  [21] 3.935951e-01 8.268728e-01 4.188815e-01 7.143317e-02 6.586660e-01
##  [26] 3.090973e-01 9.265899e-01 7.776641e-01 3.266751e-01 9.610740e-01
##  [31] 8.263676e-01 3.085402e-01 4.139323e-01 7.067321e-01 5.150022e-01
##  [36] 7.294243e-01 7.415262e-01 8.843381e-01 6.322928e-01 8.347140e-01
##  [41] 3.176489e-01 3.934680e-01 5.019678e-01 4.705947e-01 3.058576e-01
##  [46] 5.997932e-01 8.460410e-01 6.781070e-01 4.542734e-01 6.226773e-01
##  [51] 6.476034e-01 2.815249e-01 8.607185e-01 6.305869e-01 3.705499e-02
##  [56] 5.470475e-01 9.487903e-01 7.693141e-01 7.677164e-02 5.368030e-01
##  [61] 5.298730e-01 3.480114e-01 3.192112e-01 7.831646e-01 8.260869e-01
##  [66] 9.080397e-01 1.345624e-02 9.083802e-01 3.291750e-01 7.996282e-01
##  [71] 8.351945e-01 8.145130e-01 3.703275e-01 8.913474e-01 1.513707e-02
##  [76] 6.869613e-02 2.759431e-01 3.739363e-02 7.408735e-01 1.086984e-01
##  [81] 9.843288e-01 9.276874e-01 7.071651e-01 8.938037e-01 9.983360e-01
##  [86] 9.457833e-01 6.896752e-01 6.260021e-01 5.489351e-01 6.595923e-01
##  [91] 1.713738e-02 1.664939e-01 8.447271e-01 5.699174e-01 8.169605e-01
##  [96] 7.725062e-01 2.823928e-01 7.418012e-01 9.092720e-01 7.794213e-01
```

3. 생성한 난수들이 $U(0,1)$에서의 확률 난수인지 확인하기 위해 산점도를 그리고 적합도검정, 독립성 검정을

수행해 보았다.

(1) 먼저 산점도를 그렸을 때는 난수들이 0~1 사이에 비교적 고르게 분포해 보인다.

```
plot(x, pch = 16)   # 산점도
```



(2) 카이제곱 적합도 검정은 가설 (H_0: 생성된 난수들은 $[0,1]$사이의 균일분포를 따른다. vs H_1: not H_0)에 따라 검정하였다. 카이제곱 적합도 검정의 결과는 $[0,1]$의 구간을 어떻게 나누느냐에 따라 결과가 달라지는데, 8개의 구간을 기준으로 7개 이하의 구간으로 나누면 H_0를 기각하지 않고 (난수들이 균일분포를 따름), 7개보다 많이 나누면 H_0을 기각한다 (난수들이 균일분포를 따르지 않음).

구간을 많이 나눌수록 구간에 속하는 난수의 갯수가 고르지 않아 발생하는 문제이므로 적당히 7개의 구간을 준 결과를 출력하였고, p-value가 0.05보다 크므로 생성한 난수들이 균일분포를 따른다고 할 수 있다.

```
k = 8
range = seq(0, 1, length = k)

# [0, 1]을 k등분
n = as.numeric(table(cut(x, range)))
n  # 구간별 난수의 갯수
```

```
## [1] 17  6 14 11 13 23 16
```

```
# 구간별로 난수의 갯수 세기
W = ((k - 1) / N) * sum( (n - ( N/(k - 1) ) )^2)
pchisq(W, df = k-2, lower.tail = FALSE)   # p-value
```

```
## [1] 0.06851482
```

(3) 또 다른 적합도 검정으로 콜모고로프-스미르노프 적합도 검정을 할 수 있는데(`ks.test` 함수 사용), 그

결과 생성한 난수들이 균일분포를 따르지 않는다고 할 수 있다.

```
u = runif(N)
ks.test(x, u)
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  x and u
## D = 0.09, p-value = 0.8127
## alternative hypothesis: two-sided
```

(4) 마지막으로 독립성 검정인 런 검정 (run test)를 가설 (H_0: 난수들이 서로 독립이다. vs H_1: not H_0) 하에서 시행하였고, 그 결과 생성한 난수들이 서로 독립이라고 할 수 있다.

```
library(snpar)
runs.test(x)
```

```
##
##  Approximate runs rest
##
## data:  x
## Runs = 48, p-value = 0.5465
## alternative hypothesis: two.sided
```

문제 2.

1. 확률 모의실험을 위해 먼저 각 10,000개의 균일난수를 아래와 같이 생성하였다.

```
set.seed(123)
u1 = runif(10000)
u2 = runif(10000)
```

2. (a) 독립인 두 확률변수의 분산의 합과 합의 분산은 이론적으로 같으며 $(2 \times \frac{1}{12})$, 경험적 추정치 역시 참값과 비슷한 값을 출력하였다.

```
# (a)
2 * (1 / 12)  # 참값
```

```
## [1] 0.1666667
```

```
var(u1 + u2)
```

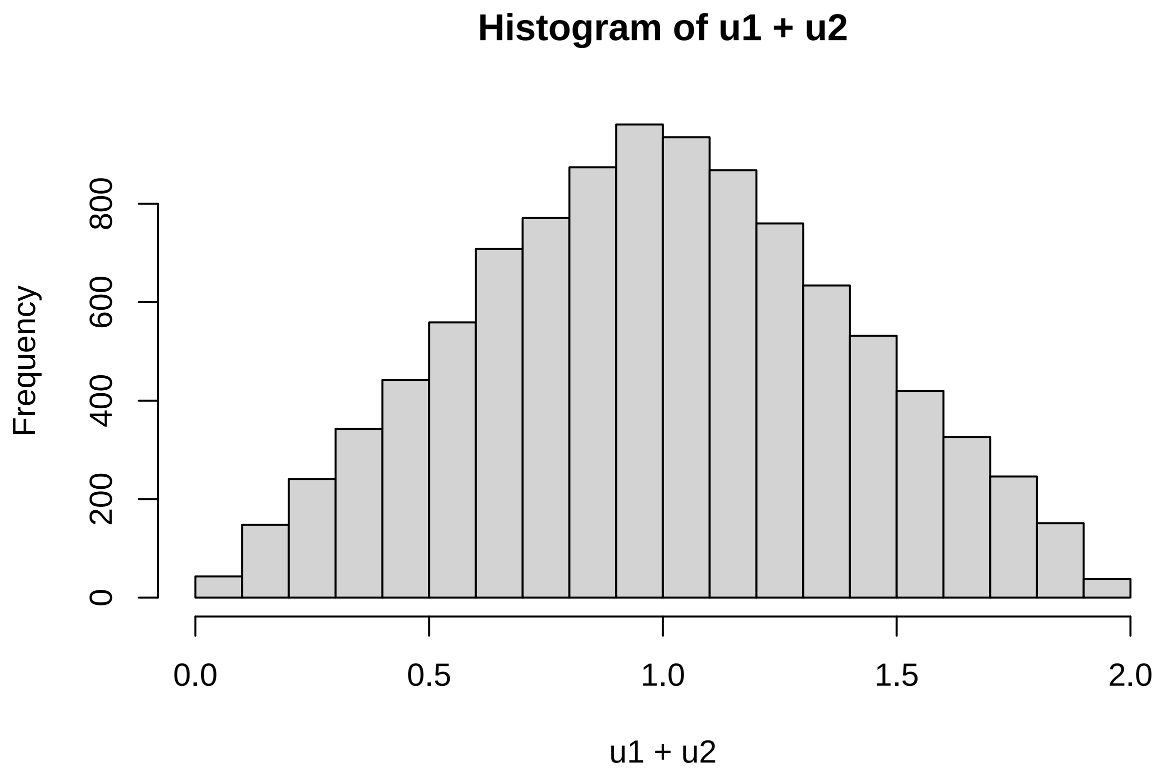
```
## [1] 0.1623921
```

```
var(u1) + var(u2)
```

```
## [1] 0.1653023
```

(b) $X = U_1 + U_2$라 하면 X의 분포는 $f(x) = xI(0 \le x \le 1) + (2-x)I(1 \le x \le 2)$를 따른다. 따라서 참값은 $P(X \le 1.5) = \int_0^{1.5} f(x)dx = 0.875$ 이며, 경험적 추정치는 100개 발생한 난수에 대해 $\frac{1}{100}\sum_{i=1}^{100} I(u_1 + u_2 \le 1.5)$로 계산할 수 있다. 결과에 따라 참값과 경험적 추정치가 비슷하게 나타남을 볼 수 있다.

```
# (b)
hist(u1+u2)  # distribution of U1 + U2
```




```
f = function(x){ x * (0 <= x) * (x <= 1) + (2 - x) * (1 <= x) * (x <= 2) }

integrate(f, 0, 1.5)      # true value
```

```
## 0.875 with absolute error < 9.7e-16
```

```
mean( (u1 + u2) <= 1.5 ) # empirical estimator
```

```
## [1] 0.8819
```

문제 3.

1. 총 100명의 학생들이 20문제를 풀었다고 하자. 각 문제를 맞출 확률은 독립적으로 0.5이기 때문에 베르누이 시행으로 문제를 맞춘 경우를 1, 틀린 경우를 0의 값을 생성하였다.

```
set.seed(123)
n = 100   # 100명의 학생
m = 20    # 20문제

M = matrix(sample( c(0, 1), n * m, replace = TRUE, prob = c(1/2, 1/2)),
           nrow = n, ncol = m)  # 행: 학생, 열: 문제의 수
```

2. 각 문제에 대한 점수는 5점이므로 학생별로 맞춘 문제 수에 5를 곱해주었다. 이렇게 구한 점수로부터 전체 평균 점수는 약 50.5점이며, 표준편차는 약 11.9이다. 그리고 30점 이상 받은 학생의 비율 ($\frac{1}{100}\sum_{i=1}^{100} I(x \leq 30)$)은 약 0.94이다.

```
score = rowSums(M) * 5    # 각 학생들의 점수

mean(score)     # 학생들 점수의 평균
```

```
## [1] 50.5
```

```r
sd(score)       # 학생들 점수의 표준편차
```

```
## [1] 11.90238
```

```r
mean( score >= 30 )    # 30점 이상 받은 학생의 비율
```

```
## [1] 0.94
```