Homework 1

Due 1/30 12pm

---

1. (a) For case 1.1 the marginal probabilty is

$$P(x = 0) = 0.2, \quad P(x = 1) = 0.8$$
$$P(y = 0) = 0.6, \quad P(y = 1) = 0.4$$

For 1.2

$$P(x = 0) = 0.4, \quad P(x = 1) = 0.6$$
$$P(y = 0) = 0.6, \quad P(y = 1) = 0.4$$

(b) 1.1

$$P(x = 0|y = 0) = 1/5, \quad P(x = 1|y = 0) = 4/5$$
$$P(x = 0|y = 1) = 1/5, \quad P(x = 1|y = 1) = 4/5$$

1.2

$$P(x = 0|y = 0) = 1/3, \quad P(x = 1|y = 0) = 2/3$$
$$P(x = 0|y = 1) = 3/7, \quad P(x = 1|y = 1) = 4/7$$

(c) Variables x and y are independent iff $P(x, y) = P(x)P(y)$. For 1.1

$$P(x = 0, y = 0) = 0.12 \quad \text{and} \quad P(x = 0)P(y = 0) = 0.2(0.6) = 0.12$$
$$P(x = 0, y = 1) = P(x = 0)P(y = 1) = 0.2(0.4) = 0.08 \ldots$$
$$P(x, y) = P(x)P(y)$$

So x and y are independent for 1.1. You can also see that the condition of y does not change the marginal probability of x. For 1.2 there is a simple counterexample

$$P(x = 0, y = 0) = 0.1 \quad \text{and} \quad P(x = 0)P(y = 0) = 0.4(0.3) = 0.12$$
$$P(x = 0, y = 0) \neq P(x)P(y)$$

So x and y are not independent (dependent) for 1.2. You can also see that the conditional probability is not the same as the marginal probability for both cases.

2. For two random variables x and y to be independent, it must be true that

$$P(x, y) = P(x)P(y) \tag{1}$$

and from the definition of conditional probability

$$P(x|y = y_o) = \frac{P(x, y = y_o)}{P(y = y_o)}$$

substituting (1) into the joint probability

$$P(x|y = y_o) = \frac{P(x)P(y = y_o)}{P(y = y_o)}$$
$$P(x|y = y_o) = P(x)$$

3. (a) Since the two thrown dice are independent, the fair dice has 36 possible outcomes $A_{xy} = \{(1,1),(1,2),\ldots,(6,6)\}$ with equal probability

$$P(x,y) = P(x)P(y) = \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}$$

The probability distribution of the sum of the two dice $P(S)$ is

$$P(S) = \begin{cases} 1/36 & S = 2,12 \\ 2/36 & S = 3,11 \\ 3/36 & S = 4,10 \\ 4/36 & S = 5,9 \\ 5/36 & S = 6,8 \\ 6/36 & S = 7 \end{cases}$$

where $S = x + y$. For the absolute difference of the two dice $D = |x - y|$

$$P(D) = \begin{cases} 2/36 & D = 5 \\ 4/36 & D = 4 \\ 6/36 & D = 3 \\ 8/36 & D = 2 \\ 10/36 & D = 1 \\ 6/36 & D = 0 \end{cases}$$

for the difference $D = 0$ there are 6 possible outcomes $(1,1),(2,2),\ldots,(6,6)$, for $D = 1$ there are 10 possible outcomes $(1,2),(2,1),(2,3),(3,2),\ldots,(5,6),(6,5)$, and so on.

(b) For 100 dice, the probability distribution of the sum of the dice $P(S)$ would be roughly

$$P(S) = \begin{cases} 1/6^{100} & S = 100,600 \\ 100/6^{100} & S = 101,599 \\ 5050/6^{100} & S = 102,598 \\ \vdots & \vdots \\ 1.52 \times 10^{76}/6^{100} & S = 350 \end{cases}$$

first we find the mean of 1 independent dice roll (μ_1)':

$$\mu_1 = \sum_x P(x)x = \frac{1}{6}\sum_{x=1}^{6} x = \frac{21}{6} = 3.5$$

because the mean of N independent dice rolls is the sum of the means of each dice roll

$$\mu_N = \sum_{i=1}^{N} \mu_i$$

thus the mean of 100 dice rolls is

$$\mu_{100} = 100 \cdot \mu_1 = \boxed{350}$$

To find the Standard Deviation we first find the variance of 1 independent dice roll (σ_1^2):

$$\mathrm{Var}[x] = \mathrm{E}[(x - \mathrm{E}[x])^2] = \mathrm{E}[x^2 - 2x\,\mathrm{E}[x] + \mathrm{E}[x]^2]$$
$$= \mathrm{E}[x^2] - 2\,\mathrm{E}[x]^2 + \mathrm{E}[x]^2\,\mathrm{E}[1] = \mathrm{E}[x^2] - \mathrm{E}[x]^2$$

or in summation notation

$$\sigma_1^2 = \sum_x P(x)(x - \mu_1)^2 = \frac{1}{6}\sum_{x=1}^{6}(x - 3.5)^2 = \frac{17.5}{6} = 2.9167$$

2

for 2 independent variables x and y

$$
\begin{aligned}
\operatorname{Var}[x+y] &= \operatorname{E}[((x+y) - \operatorname{E}[x+y])^2] \\
&= \operatorname{E}[(x - \operatorname{E}[x]) + (y - \operatorname{E}[y])^2] \\
&= \operatorname{E}[(x - \operatorname{E}[x])^2 + (y - \operatorname{E}[y])^2 + 2(x - \operatorname{E}[x])(y - \operatorname{E}[y])] \\
&= \operatorname{E}[(x - \operatorname{E}[x])^2] + \operatorname{E}[(y - \operatorname{E}[y])^2] + 2\operatorname{E}[(x - \operatorname{E}[x])(y - \operatorname{E}[y])] \\
&= \operatorname{Var}[x] + \operatorname{Var}[y] + 2\operatorname{E}[(x - \operatorname{E}[x])(y - \operatorname{E}[y])]
\end{aligned}
$$

where the third term is

$$
\begin{aligned}
\operatorname{E}[(x - \operatorname{E}[x])(y - \operatorname{E}[y])] &= \operatorname{E}[xy - x\operatorname{E}[y] - y\operatorname{E}[x] + \operatorname{E}[x]\operatorname{E}[y]] \\
&= \operatorname{E}[xy] - \operatorname{E}[x]\operatorname{E}[y]
\end{aligned}
$$

and for independent variables x and y the third term is zero. Thus the variance of the sum of N independent dice rolls is

$$
\operatorname{Var}[N] = N\sigma_1^2 = 100 \cdot 2.9167 = 291.67
$$

and the standard deviation is

$$
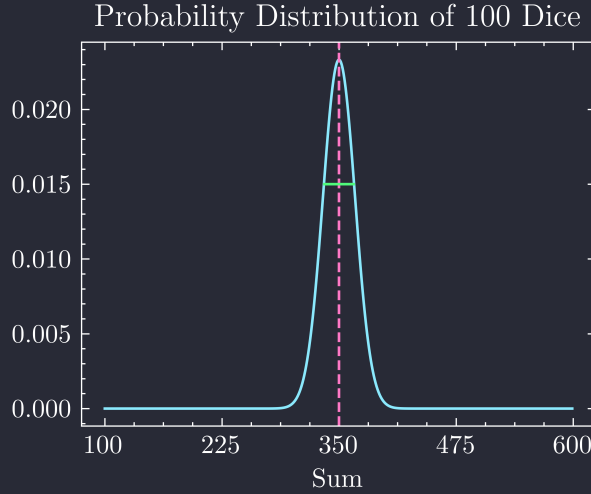\sigma_N = \sqrt{\operatorname{Var}[N]} = \sqrt{291.67} = \boxed{17.08}
$$

Figure 1.1: Probability distribution of the sum of 100 dice rolls. The mean is 350 and the standard deviation is ≈ 17.

The sketch of probability distribution of the sum of 100 dice rolls is shown in Figure 1.1.

4. (a) Assuming that there is an equal likelihood of the order of the age of the three brothers being any of the 6 possible permutations:

$$
\operatorname{Age}_{A,B,F} = \{ABF, AFB, BAF, BFA, FAB, FBA\}
$$

where we denote the first element of a permutation as the oldest brother and the last element as the youngest brother.
The probability that Fred (F) is older than Bob (B) is $\boxed{1/2}$ from both the 3 possible permutations or by realizing that there are only two equal outcomes when looking at only the age of Fred vs Bob.
(b) Given that Fred is older than Alex (A), we can eliminate the 3 permutations where Alex is older. Thus the probability that Fred is older than Bob is $\boxed{2/3}$.

5. (a) Given that the probability of choosing a black ball from an urn is $f_B = \frac{B}{K}$, the probability distribution of choosing n_B black balls from N draws is

$$\boxed{P(n_B|N, f_B) = \binom{N}{n_B} f_B^{n_B}(1 - f_B)^{N-n_B}}$$

(b) Finding the mean and standard deviation of n_B is quite similar to Problem 3. Since each draw is independent, the mean of n_B is the sum of the means of each draw! In the case of drawing one ball has a binary outcome

$$n_B(N = 1) = \begin{cases} 1 & \text{black ball with probability } f_b \\ 0 & \text{white ball } (1 - f_B) \end{cases}$$

thus the mean of n_B for one draw is

$$\mu_1 = 1(f_B) + 0(1 - f_B) = f_B$$

and the variance is

$$\sigma_1^2 = (1 - f_B)^2 f_B + (0 - f_B)^2(1 - f_B) = f_B(1 - f_B)$$

for N draws the means add up to

$$\boxed{\mu = Nf_B}$$

and the variances add only due to the independent nature of the draws

$$\sigma^2 = Nf_B(1 - f_B)$$

thus the standard deviation is

$$\boxed{\sigma = \sqrt{Nf_B(1 - f_B)}}$$

For $K = 20$, and $B = K$; $f_B = 5/20 = 0.25$. And for $N = 5$ we have the ratio

$$\frac{\sigma}{\mu} = \frac{\sqrt{5 \cdot 0.25 \cdot 0.75}}{5 \cdot 0.25} = \frac{\sqrt{15}}{5} \approx \boxed{0.77}$$

and for $N = 20$

$$\frac{\sigma}{\mu} = \frac{\sqrt{1000 \cdot 0.25 \cdot 0.75}}{1000 \cdot 0.25} = \frac{\sqrt{30}}{100} \approx \boxed{0.05}$$

6. (a) Dividing the time period T in to M intervals where each interval has a probability $r\,dt$ or $dt = T/M$. The probability of no events occuring in time T is

$$\lim_{M \to \infty} (1 - r\,dt)^M = \lim_{M \to \infty} \left(1 - \frac{rT}{M}\right)^M = \boxed{e^{-rT}}$$

(b) For $n_T = x$ events occuring in M this is similar to the binomial distribution as $M \to \infty$.

$$\lim_{M \to \infty} \frac{M!}{x!(M - x)!} \left(\frac{rT}{M}\right)^x \left(1 - \frac{rT}{M}\right)^{M-x}$$

canceling out some terms...

$$\frac{M!}{(M - x)!} = \frac{M(M - 1) \cdots (M - x + 1)(M - x)!}{(M - x)!} = M(M - 1) \cdots (M - x + 1)$$

and now the facorial has x terms, so we can write it as

$$\frac{M(M-1)\cdots(M-x+1)}{M^x} = \frac{M}{M}\frac{M-1}{M}\cdots\frac{M-x+1}{M}$$

$$= 1\cdot\left(1-\frac{1}{M}\right)\cdots\left(1-\frac{x-1}{M}\right)$$

and as $M\to\infty$ the terms in the product go to 1, so the product goes to 1. Thus we are left with

$$\lim_{M\to\infty}\frac{(rT)^x}{x!}\left(1-\frac{rT}{M}\right)^{M-x} = \lim_{M\to\infty}\frac{(rT)^x}{x!}\left(1-\frac{rT}{M}\right)^{M}\left(1-\frac{rT}{M}\right)^{-x}$$

the second term is the limit of the exponential function

$$\lim_{M\to\infty}\left(1-\frac{rT}{M}\right)^{M} = e^{-rT}$$

and the third term tends to 1 as $M\to\infty$. Thus the probability of x events occuring in time T is

$$\boxed{P(x) = \frac{(rT)^x}{x!}e^{-rT}}$$

where $x = n_T$ for the sake of brevity in notation.

(c) The mean of n_T is

$$\mu = \sum_{x=0}^{\infty} xP(x)$$

$$= \sum_{x=0}^{\infty} x\frac{(rT)^x}{x!}e^{-rT}$$

$$= e^{-rT}\sum_{x=1}^{\infty} x\frac{(rT)(rT)^{x-1}}{x(x-1)!}$$

$$= e^{-rT}(rT)\sum_{x=1}^{\infty}\frac{(rT)^{x-1}}{(x-1)!}$$

the the first term of the sum is zero which is why the sum starts at $x=1$. The sum is also the Taylor series expansion of e^{rT} if we let $n = x-1$ so

$$\sum_{x=1}^{\infty}\frac{(rT)^{x-1}}{(x-1)!} = \sum_{n=0}^{\infty}\frac{(rT)^n}{n!} = e^{rT}$$

Therefore the mean of n_T is

$$\mu = (rT)e^{-rT}e^{rT} = rT$$

The variance of n_T is

$$\sigma^2 = \mathrm{E}[x^2] - \mathrm{E}[x]^2$$

the first term is solved similarly to the mean

$$E[x^2] = \sum_{x=0}^{\infty} x^2 P(x)$$

$$= \sum_{x=0}^{\infty} x^2 \frac{(rT)^x}{x!} e^{-rT}$$

$$= e^{-rT} \sum_{x=1}^{\infty} x^2 \frac{(rT)(rT)^{x-1}}{x(x-1)!}$$

$$= (rT)e^{-rT} \sum_{x=1}^{\infty} x \frac{(rT)^{x-1}}{(x-1)!}$$

$$= (rT)e^{-rT} \left[\sum_{x=1}^{\infty} (x-1) \frac{(rT)^{x-1}}{(x-1)!} + \sum_{x=1}^{\infty} \frac{(rT)^{x-1}}{(x-1)!} \right] \qquad x = [(x-1)+1]$$

$$= (rT)e^{-rT} \left[(rT) \sum_{x=2}^{\infty} \frac{(rT)^{x-2}}{(x-2)!} + \sum_{n=0}^{\infty} \frac{(rT)^n}{n!} \right] \qquad n = x-1$$

$$= (rT)e^{-rT} \left[(rT) \sum_{l=0}^{\infty} \frac{(rT)^l}{l!} + \sum_{n=0}^{\infty} \frac{(rT)^n}{n!} \right] \qquad l = x-2$$

$$= (rT)e^{-rT} \left[(rT)e^{rT} + e^{rT} \right]$$

$$= (rT)^2 e^{-rT} e^{rT} + (rT)e^{-rT} e^{rT}$$

$$= (rT)^2 + rT$$

and the variance is

$$\sigma^2 = (rT)^2 + rT - (rT)^2 = rT$$

Therefore the mean and standard deviation of n_T are

$$\boxed{\mu = rT \quad \text{and} \quad \sigma = \sqrt{rT}}$$

7. Using Bayes' theorem for the outcome $X = \{7, 3, 4, 2, 5, 3\}$ is

$$P(A|7,3,4,2,5,3) = P(A|X) = \frac{P(X|A)P(A)}{P(X)}$$

where the probability of choosing dice A is 1 in 3—$P(A) = 1/3$. The conditional probabilty $P(X|A)$ is the probability of rolling the outcome X given that dice A is chosen:

$$P(X|A) = \frac{1 \times 4 \times 2 \times 4 \times 2 \times 4}{20^6} = \frac{256}{20^6}$$

and the probability of rolling the outcome X is given by the sum rule

$$P(X) = P(X|A)P(A) + P(X|B)P(B) + P(X|C)P(C)$$

where $P(B) = P(C) = 1/3$ and the conditional probabilities $P(X|B)$ and $P(X|C)$ are

$$P(X|B) = \frac{2 \times 3 \times 2 \times 2 \times 2 \times 3}{20^6} = \frac{144}{20^6}$$

$$P(X|C) = \frac{2^6}{20^6} = \frac{64}{20^6}$$

thus the probability of choosing dice A given the outcome X is

$$P(A|X) = \frac{\frac{256}{20^6} \cdot \frac{1}{3}}{\frac{256}{20^6} \cdot \frac{1}{3} + \frac{144}{20^6} \cdot \frac{1}{3} + \frac{64}{20^6} \cdot \frac{1}{3}} = \frac{256}{464} \approx \boxed{0.55}$$

with the knowledge that terms cancel out, the probability of the die being B is

$$P(B|X) = \frac{144}{464} \approx \boxed{0.31}$$

and the probability of the die being C is

$$P(C|X) = \frac{64}{464} \approx \boxed{0.14}$$

8. (a) Given that the bus arrives on average every 5 minutes, the average wait time is 5 minutes. And the bus that just left Sally would have left an average of 5 minutes ago. From the code, taking the mean value of the wait times is also $\approx$ 5 minutes.
(b) Therefore the average time between two buses is the sum in the time Sally is waiting for the bus and how long the missed bus has been gone: 10 minutes.
(c) The paradox is that 'we' think that after waiting for 5 minutes the bus will arrive, but the average time between buses is 10 minutes, so we are waiting longer than we expect to intuitively. This is because the conditional probability of Sally getting to the bus stop where the interval between buses is less than 5 minutes given that she has waited for a time t is less as time goes on. And the probability that Sally arrived at the bus stop where the interval between buses is more than 5 minutes given that she has waited for a time t is more as time goes on.
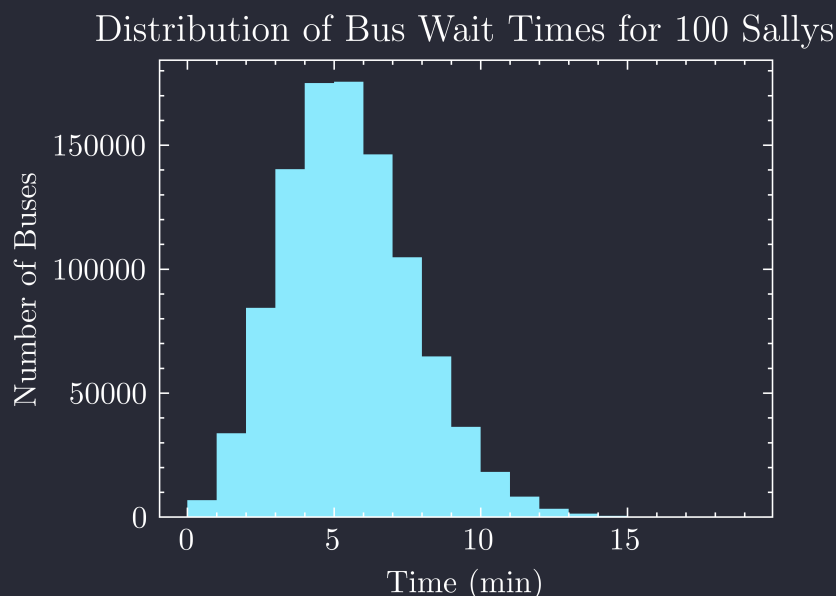(d)

Figure 1.2: Mean of 5.00 min and Time between buses of 10.00 min.

PYTHON CODE BELOW

hw1_python

January 30, 2024

```python
import numpy as np
import scipy as sp
import matplotlib.pyplot as plt
import matplotlib as mpl
import scienceplots

# Science plot package + Dracula theme
plt.style.use(['science', 'dark_background'])
plt.rcParams['axes.facecolor'] = '#282a36'
plt.rcParams['figure.facecolor'] = '#282a36'
colorcycle = ['#8be9fd', '#ff79c6', '#50fa7b', '#bd93f9', '#ffb86c', '#ff5555',
 '#f1fa8c',
'#6272a4']
plt.rcParams['axes.prop_cycle'] = mpl.cycler(color=colorcycle)
white = '#f8f8f2' # foreground

# change dpi
plt.rcParams['figure.dpi'] = 1024
```

```python
# highly optimized function to count the number of ways to get a sum of s with
 d dice
def count_ways_to_sum(sum_target, num_dice):
    # Initialize a 2D array to store results of subproblems
    dp = [[0] * (sum_target + 1) for _ in range(num_dice + 1)]

    # Base case: there is one way to get a sum of 0 (no dice)
    dp[0][0] = 1

    # Fill the dp table using the convolution formula
    for d in range(1, num_dice + 1):
        for s in range(1, sum_target + 1):
            for k in range(1, 7):
                if s - k >= 0:
                    dp[d][s] += dp[d-1][s-k]

    return dp[num_dice][sum_target]
```

```python
# Example usage:
sum_100_ways = count_ways_to_sum(100, 100)
sum_101_ways = count_ways_to_sum(101, 100)
sum_102_ways = count_ways_to_sum(102, 100)
sum_350_ways = count_ways_to_sum(350, 100)

print(f"Number of ways to get a sum of 100 with 100 dice: {sum_100_ways}")
print(f"Number of ways to get a sum of 101 with 100 dice: {sum_101_ways}")
print(f"Number of ways to get a sum of 102 with 100 dice: {sum_102_ways}")
print(f"Number of ways to get a sum of 350 with 100 dice: {sum_350_ways:.2e}")

# for loop to get a function of sum to combinations
sums = np.arange(100, 600, 1)
ways = []
for i in sums:
    ways.append(count_ways_to_sum(i, 100) / 6**100)

# plotting
plt.figure()
plt.plot(sums, ways)
plt.xlabel('Sum')
plt.title('Probability Distribution of 100 Dice')

# Add x-axis label at 350 as "mean"
plt.axvline(x=350, color=colorcycle[1], linestyle='--')

# Add tick label at 500/4 intervals
plt.xticks(np.arange(100, 601, 125))

# Add standard deviation at x= 350 +/- 17 as horizontal line
plt.hlines(y=0.015, xmin=350-17, xmax=350+17, color=colorcycle[2])

plt.show()

print (f"Probability of getting a sum of 350 with 100 dice: {sum_350_ways /
    ↪6**100:.2e}")
# sum up ways
print (f"Probability distribution adds to 1: {sum(ways):.2e}")
```
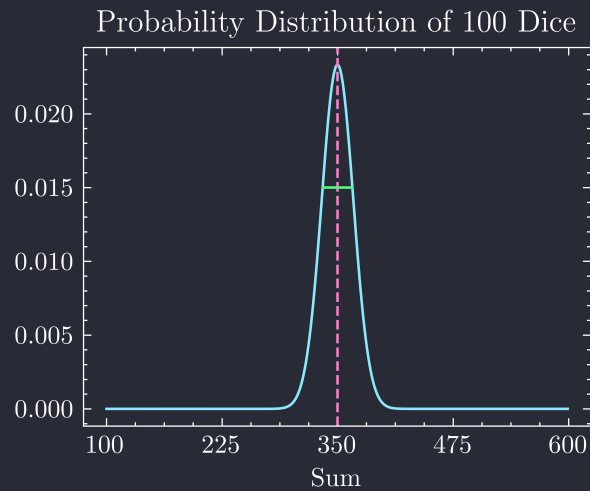
Number of ways to get a sum of 100 with 100 dice: 1
Number of ways to get a sum of 101 with 100 dice: 100
Number of ways to get a sum of 102 with 100 dice: 5050
Number of ways to get a sum of 350 with 100 dice: 1.52e+76

Probability Distribution of 100 Dice

Probability of getting a sum of 350 with 100 dice: 2.33e-02
Probability distribution adds to 1: 1.00e+00

```python
# 481 Problem 8d
# Simulation of 10000 buses and 100 Sally's

# constants
N = 10000 # number of buses
t_avg = 5 # average time between buses

# using poisson distribution old way
t_poisson = np.random.poisson(t_avg, N)

# simulating 100 Sally's
for i in range(99):
    t_poisson = np.concatenate((t_poisson, np.random.poisson(t_avg, N)))

# checking if it is 100 Sally's
print(len(t_poisson) / 10000)

# plotting
plt.figure()
plt.hist(t_poisson, bins=np.max(t_poisson))
plt.xlabel('Time (min)')
plt.ylabel('Number of Buses')
plt.title('Distribution of Bus Wait Times for 100 Sallys')
plt.show()
```
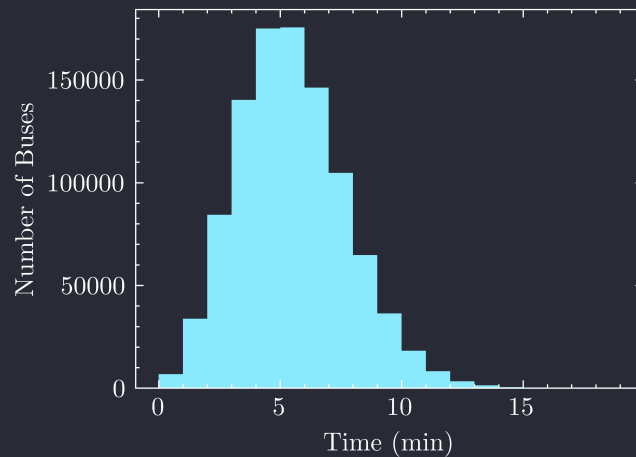
100.0

Distribution of Bus Wait Times for 100 Sallys

```python
# implementing hard code
# random number generator
rand = np.random.default_rng(seed=42)

# Probability of 1 bus given 5 min avg
p = np.exp(-5)

# wait time for 1 bus
def wait_time():
    time = 0
    prod = 1.0
    while True:
        U = rand.random()
        prod *= U
        if prod > p:
            time += 1
        else:
            return time

def wait_time_100():
    times = []
    for i in range(100):
        times.append(wait_time())
    return times

# simulating 10000 buses
times = []
for i in range(10000):
    times += wait_time_100()
```
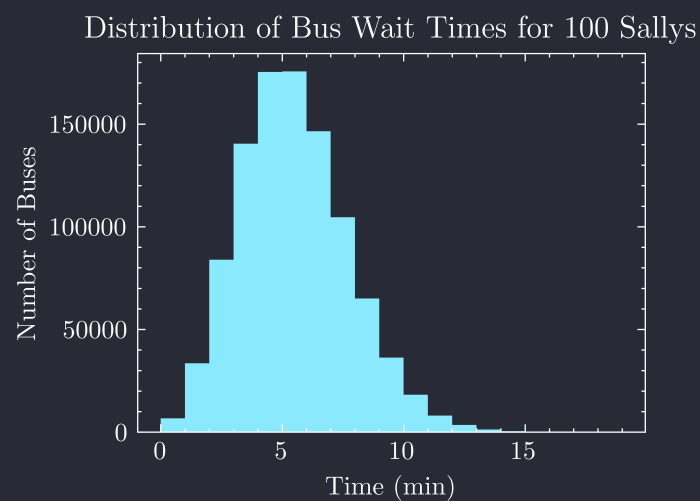
```python
print(np.mean(times))

# plotting
plt.figure()
plt.hist(times, bins=np.max(times))
plt.xlabel('Time (min)')
plt.ylabel('Number of Buses')
plt.title('Distribution of Bus Wait Times for 100 Sallys')
plt.show()
```

5.001477

Distribution of Bus Wait Times for 100 Sallys

```python
# time of bus that just passed
last_bus_times = []
for i in range(1, len(times)):
    if times[i] == 0:
        last_bus_times.append(times[i] - times[i - 1])

print(np.mean(last_bus_times))

# time in between buses
between_bus_times = []
for i in range(1, len(times)):
    between_bus_times.append(times[i] + times[i-1])

print(np.mean(between_bus_times))
```

-5.012881255552266
10.002945002945003