Introduction to Statistics for Engineers with Python

Prepared by:

Dr. Gokhan Bingol (gbingol@hotmail.com)

December 13, 2024

Document version: 1.0

Engineering Documents: https://www.pebytes.com/pubs

Follow on GitHub: https://github.com/gbingol

1. Introduction

With the increasing digitalization of the process industry, engineers must be equipped not only with core engineering principles but also with strong digital and computational skills (Proctor & Chiang, 2023)[1]. The growing popularity of machine learning (ML) and its broader field, artificial intelligence (AI), has further highlighted the need for engineers to develop a solid foundation in descriptive and inferential statistics, as well as in supervised and unsupervised modeling techniques (Pinheiro & Patetta, 2021). Statistical tests, long-standing tools in data analysis, offer interpretable results and well-defined hypothesis testing (Montgomery, 2012). They are particularly valuable for analyzing small datasets and determining the significance of relationships (Box *et al*. 2005).

On the other hand, ML techniques excel at uncovering patterns in complex datasets with intricate relationships that traditional statistics may overlook (Hastie *et al*., 2009). However, these methods often require larger datasets and computational resources. Moreover, the "black-box" nature of certain ML models, such as deep learning, can limit their interpretability, which is crucial for making informed decisions in process engineering (Rudin, 2019)[2].

Random variables—such as the lifespan of a pump, the time required to complete a task, or the occurrence of natural phenomena like earthquakes—play a pivotal role in both everyday life and engineering applications (Forbes *et al*., 2011). The probability distribution of a random variable provides a mathematical description of how probabilities are assigned across its possible values. While statistical literature describes a vast array of distributions (Wolfram MathWorld)[3], only a limited subset is commonly used in engineering, as highlighted by Forbes *et al*. (2011) and Bury (1999).

Statistical tools and tests are indispensable in engineering analysis. Common parametric tests like t-tests and ANOVA are widely used for comparing means and analyzing variance (Montgomery, 2012). Non-parametric tests, such as the Kruskal-Wallis test or the sign test, are particularly useful when data fail to meet the assumptions of normality (Gopal, 2006; Kreyszig *et al*., 2011). Regression analysis, another critical tool, enables the investigation of relationships between variables (Montgomery *et al*., 2021).

---

1 https://www.thechemicalengineer.com/features/data-science-and-digitalisation-for-chemical-engineers/

2 **Rudin C** (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), 206-215.

3 https://mathworld.wolfram.com/topics/StatisticalDistributions.html

The current work emphasizes the application of statistics in engineering, leveraging Python as the computational tool of choice. Furthermore, it relies extensively on Python packages such as *numpy* and *scisuit*[4]. The *scisuit's* statistical library draws inspiration from *R*[5], enabling readers to transfer the knowledge gained here to *R*, a widely used software in the data science domain.

4 **scisuit** at least v1.4.3.
5 https://www.r-project.org

2. Probability & Random Variables

2.1. Permutations / Combinations

2.1.1. Permutations

Any ordered sequence of *k* objects taken *without replacement* from a set of *n* objects is called a permutation of size *k* of the objects (Devore *et al*., 2021). There are two cases:

A) *Objects are Distinct:* The set contains only distinct objects, such as A, B, C… Then the number of permutations of length *k*, that can be formed from the set of *n* elements is:

$$n P_k = n \cdot (n-1) \cdot \ldots \cdot (n-k+1) = \frac{n!}{(n-k)!} \qquad \textbf{(2.1)}$$

The interpretation of Eq. (2.1) is fairly straightforward: Initially there are *n* objects in the set and once one is taken out (since without replacement), *n-1* objects are left and then the sequence continues in similar fashion.

Example 2.1

How many permutations of length k=3 can be formed from the elements A, B, C and D (Adapted from Larsen & Marx, 2011)?

Solution:

Mathematically the solution is: $\frac{4!}{(4-3)!} = 24$

Script 2.1

```
from itertools import permutations
print(list ( permutations(["A", "B", "C", "D"], 3) ))
```

This will printout 24 tuples, each representing a permutation.

B) *Objects are NOT Distinct:* The set contains *n* objects, n_1 being one kind, n_2 of second kind … and n_r of r^{th} kind, then:

$$\frac{n!}{n_1! \cdot n_2! \cdot n_r!} \quad \textbf{(2.2)}$$

where $n_1+n_2+...+n_r=n$

Example 2.2

A biscuit in a vending machine cost 85 cents. In how many ways can a customer put 2 quarters, 3 dimes and 1 nickel (Adapted from Larsen & Marx, 2011)?

Solution:

$n_1=2$, $n_2=3$ and $n_3=1$ therefore

$$n = n_1+n_2+n_3 = 2+3+1=6$$

Eq. (2.2) can now be used:

$$\frac{6!}{2!3!1!}=60$$

2.1.2. Combinations

The number of different combinations of n different things taken, k at a time, *without repetitions*, is computed by (Kreyszig et al., 2011):

$$\binom{n}{k}=\frac{n!}{k!(n-k)!} \quad \textbf{(2.3)}$$

and if *repetitions allowed*:

$$\binom{n+k-1}{k} \quad \textbf{(2.4)}$$

Example 2.3

Given a set of elements A, B, C and D list the combinations of *unique* elements of size 2.

Solution:

There are 4 unique characters, therefore: $\binom{4}{2}=\frac{4!}{2!(4-2)!}=6$

Script 2.2

```
from itertools import combinations
print (list( combinations(["A", "B", "C", "D"], 2) ))
```

('A', 'B'), ('A', 'C'), ('A', 'D'), ('B', 'C'), ('B', 'D'), ('C', 'D')

Note that each tuple contains *k=2* different "things" and none of the tuples contains exactly the same *2* things, i.e. there is no ('A', 'A'). Please also note that unlike permutations, there is no ('B', 'A'), ('C', 'A') since there is already ('A', 'B') and ('A', 'C'), respectively.

Note that if *repetitions were allowed* Eq. (2.4) would be used:

$$\binom{4+2-1}{2}=\binom{5}{2}=\frac{5\cdot 4}{2\cdot 1}=10$$

adding (AA, BB, CC, DD) to the above list. ■

A summary of permutations and combinations for k elements from a set of n candidates is given by Liben-Nowell (2022) as follows:

1) Order matters and repetition is allowed: n^k

2) Order matters and repetition is **not** allowed: $\frac{n!}{(n-k)!}$

3) Order does **not** matter and repetition is allowed: $\binom{n+k-1}{k}$

4) Order does **not** matter and repetition is **not** allowed: $\binom{n}{k}$

2.2. Random Variables

A random variable is a variable therefore can assume different values; however, the value depends on the outcome of a chance experiment (Peck *et al*. 2016; Devore *et al*. 2021).

For example, when two dice are tossed a sample space of a set of 36 ordered pairs, $S(i,j)=[(1,1),(1,6),...,(6,1),(6,6)]$ is obtained. In many cases, the set of 36 ordered pairs is not of interest to us, for example for some of the games only the *sum of the numbers* is of interest to us, therefore, we are only interested in eleven possible sums (2, 3,…, 11, 12), i.e. if we were interested in sum being 7, then it does not matter if the outcome was (4, 3) or (6, 1). Therefore, in this case, the random variable can be defined as $X(i,j)=i+j$ (Larsen & Marx, 2011).

There are two types of random variables:

1. *Discrete:* Takes values from either a finite set or a countably infinite set.
2. *Continuous:* Takes values from uncountably infinite number of outcomes, i.e. all numbers in a single interval on the number line.

2.2.1. Discrete Random Variables

With each discrete random variable X a probability density function is associated:

$$p_X(k)=P(s\in S|X(s)=k) \tag{2.5}$$

where,

- X = the random variable
- k = a specified number the random variable can assume
- $P(X=k)$, the probability that X equals to k (Utts and Heckard, 2007).

For the dice example, let's say we are interested in the sum of numbers being 2. Then the notation would be $P(X=2) = 1/36$.

2.2.2. Continuous Random Variables

With each continuous random variable Y a probability density function is associated:

$$P(a \le Y \le b) = P(s \in S | a \le Y(s) \le b) = \int_a^b f_Y(y)\,dy \tag{2.6}$$

It is seen that unlike Eq. (2.5) which gives the probability at a particular value, Eq. (2.6) yields probability at an interval *[a, b]*.

2.2.3. Cumulative Distribution Function

Unlike PDF, the cumulative distribution function (CDF) for discrete or continuous random variable is the same, that is:

$$F_W(w) = P(W \le w) \tag{2.7}$$

Example 2.4

A fair die is rolled 4 times. Let X denote the number of sixes that appear. Find PDF and CDF (Adapted from Larsen & Marx, 2011).

Solution:

X has a binomial distribution (see chapter 3.2) with n=4 and p=1/6. Therefore, the PDF:

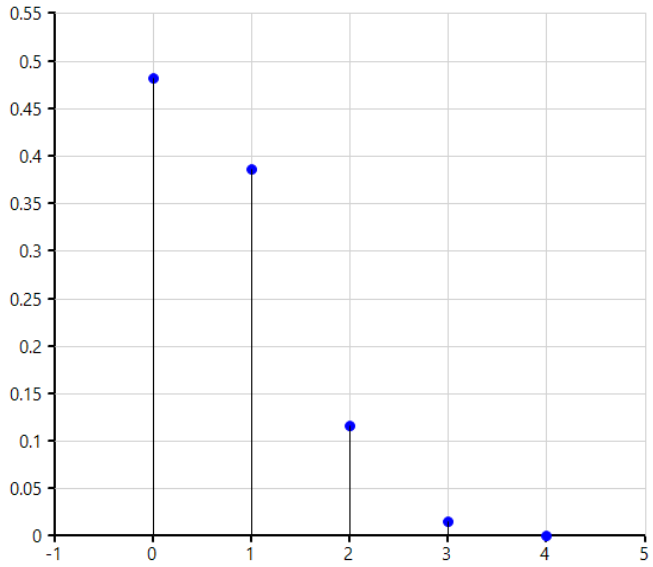
$$p(X=4) = \binom{n}{4} \cdot \left(\frac{1}{6}\right) \cdot \left(\frac{5}{6}\right)^{4-k}, \quad k = 0,1,2,3,4$$

Let's see how the probability of getting number of sixes changes with a simple plot:

Script 2.3

```
import scisuit.plot as plt
import scisuit.plot.gdi as gdi
from scisuit.stats import dbinom, pbinom

k = range(0, 5)
x = [*k]
y = [dbinom(x=i, size=4, prob=1/6) for i in x]
```

```
plt.scatter(x=x, y=y)
for i,v in enumerate(k):
    gdi.line(p1=(v, 0), p2=(x[i], y[i]))

plt.show(antialiasing=True)
```



As expected the probability of getting number of sixes decrease as *k* increases.

Fig 2.1: *Probability of getting number of sixes*

Note that Fig. (2.1) only shows probabilities for individual data points, i.e., for *k=0, 1, 2, 3* and *4* sixes. However, it does not immediately show the probability for *P(X≤2)*. The cumulative distribution function is:

$$F_X(x)=\begin{cases} 0 & x<0 \\ \left(\frac{5}{6}\right)^4=0.482 & 0\le x<1 \\ \left(\frac{5}{6}\right)^4+4\left(\frac{1}{6}\right)\left(\frac{5}{6}\right)^3=0.868 & 1\le x<2 \\ \left(\frac{5}{6}\right)^4+4\left(\frac{1}{6}\right)\left(\frac{5}{6}\right)^3+6\left(\frac{1}{6}\right)^2\left(\frac{5}{6}\right)^2=0.984 & 2\le x<3 \\ \left(\frac{5}{6}\right)^4+4\left(\frac{1}{6}\right)\left(\frac{5}{6}\right)^3+6\left(\frac{1}{6}\right)^2\left(\frac{5}{6}\right)^2+4\left(\frac{1}{6}\right)^3\left(\frac{5}{6}\right)=0.999 & 3\le x<4 \\ 1 & 4\le x \end{cases}$$

With minor changes to Script (2.3):

Script 2.4

```
y = [pbinom(q=i, size=4, prob=1/6) for i in x]

plt.scatter(x=x, y=y)

for i,v in enumerate(k):
    gdi.line(p1=(v, y[i]), p2=(i+1, y[i]))
    gdi.marker(xy=(i+1, y[i]))

plt.show(antialiasing=True)
```

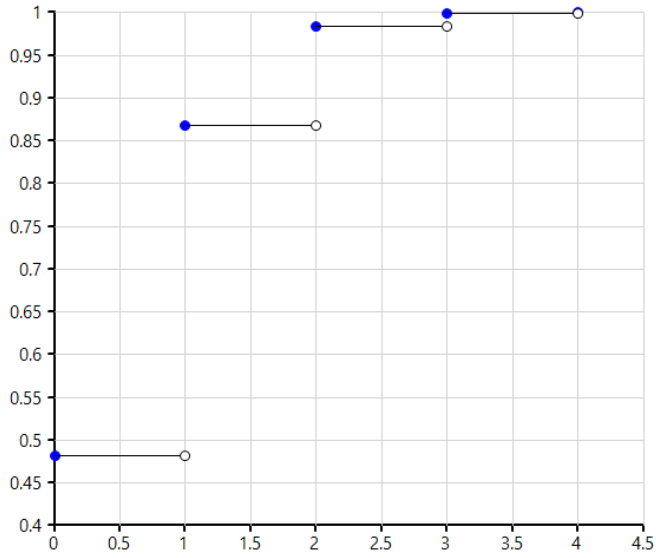


Fig 2.2: *Cumulative distribution*

What is the probability of getting at least 2 sixes?

Now it is easier to answer this question using the cumulative distribution plot. It is seen that $P(X \leq 2) = 0.98$.

2.2.4. Empirical Distribution Function

The empirical distribution function for a random sample $X_1, X_2, \ldots, X_n$ from a distribution F is the function defined by:

$$F_n(x) = \frac{1}{n} \sum_{i=1}^{n} 1_{\{x_i \leq x\}} \qquad \textbf{(2.8)}$$

where $1_{\{xi \leq x\}}$ is an indicator function that is equal to 1 if $x_i \leq x$ and 0 otherwise.

Example 2.5

A random sample of $n=8$ people yields the following counts of the number of times they exercised in the past 2 weeks:

$$0, 2, 1, 2, 7, 6, 4, 6$$

Calculate $F_n(x)$ (adapted from Anon. 2024)[6].

Solution:

The general equation for the given data is:

$$F_8(x)=\frac{1}{8}\sum_{i=1}^{8} 1_{\{x_i\le x\}}$$

For example, for $x\le 2$,

$$F_8(x)=\frac{1}{8}\sum_{i=1}^{8} 1_{\{x_i\le x\}}=\frac{1}{8}\cdot(1+1+1+1+0+0+0+0)=\frac{4}{8}$$

As demonstrated below for larger datasets, it is considerably more convenient to use Numpy:

Script 2.5

```
import numpy as np

x = np.array([0, 2, 1, 2, 7, 6, 4, 6])
f2 = np.sum(x<=2)/len(x)
print(f2)
```

```
0.5
```

2.2.5. Moment-Generating Function

Let X be a random variable. Then the moment-generating function for X is denoted by $M_x(t)$ and expressed as:

$$M_W(t)=E(t^{tW})=\begin{cases}\sum_{all\ k} e^{tk}\,p_W(k) & \text{if W is discrete}\\ \int_{-\infty}^{\infty} e^{tw}\,f_W(w) & \text{if W is continuous}\end{cases} \qquad \textbf{(2.9)}$$

6 https://online.stat.psu.edu/stat415/lesson/empirical-distribution-functions

Theorem: Let W_1, W_2, …, W_n be independent random variables with mgfs $M_{W1}(t)$, $M_{W2}(t)$, …, $M_{Wn}(t)$, respectively. Let $W = W_1+W_2+, …, W_n$. Then,

$$M_W(t)=M_{W_1}(t)\cdot M_{W_2}(t)\ldots M_{W_n}(t) \quad \textbf{(2.10)}$$

Example 2.6

Find the moment-generating function of a Bernoulli random variable:

$$X_i=\begin{cases}1 & p\\ 0 & 1-p\end{cases},\ 0<p<1$$

Solution:

Note that Bernoulli random variable is a discrete random variable, therefore condensing Eq. (2.9) for only discrete random variables yields:

$$M_X(t)=\sum_{all\ k} e^{tk}\,p_X(k)$$

One should notice the condition in the equation which states that the summation should be performed for "*all k*". Note that for Bernoulli random variable there exists only 2 *k*'s, therefore:

$$M_X(t) \;=\; e^{t\cdot 0}\cdot p(X=0)+e^{t\cdot 1}p(X=1) \;=\; (1-p)+p\cdot e^t$$

Example 2.7

Find the MGF of a binomial random variable given by the following equation:

$$p_x(k)=P(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$$

Solution:

Binomial random variable is a discrete random variable, therefore $M_X(t)$ is:

$$M_X(t) = E(e^{tX}) = \sum_{k=0}^{n} e^{tk} \binom{n}{k} p^k (1-p)^{n-k}$$

Rewriting the equation yields:

$$M_X(t) = \sum_{k=0}^{n} \binom{n}{k} (pe^t)^k (1-p)^{n-k}$$

Newton's binomial expansion formula:

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k \cdot y^{n-k}$$

Observing that mgf and binomial expansion are exactly the same if we replace x and y with $x=pe^t$ and $y=1-p$. Therefore the moment-generating function is:

$$M_X(t) = \left(1 - p + pe^t\right)^n$$

2.2.6. Expected Value

It is the most frequently used statistical measure to describe central tendency (Larsen & Marx, 2011). Let X and Y be discrete and continuous random variables, respectively. The expected values of X and Y are denoted by E(X) and E(Y), respectively, and given by the following equations:

$$E(X) = \mu = \sum_{\text{all k}} k \cdot p_X(k) \qquad \textbf{(2.11)}$$

$$E(Y) = \mu = \int_{-\infty}^{\infty} y f_Y(y) dy \qquad \textbf{(2.12)}$$

One notable property of expected value is that it is a linear operator and therefore,

$$E(aX + bY) = a \cdot E(X) + b \cdot E(Y) \qquad \textbf{(2.13)}$$

Example 2.8

Below table shows the number of courses a student registered in a university with 15,000 students. Find the average number of courses per student (Adapted from Carlton & Devore, 2014).

x	1	2	3	4	5	6	7
# Students	150	450	1950	3750	5850	2550	300

Solution #1:

The simplest approach: $\bar{x} = \frac{1\times 150 + 2\times 450 + ... + 7\times 300}{15000} = 4.57$

Solution #2:

We define a random variable X as the number of courses a student has enrolled. The mean value (weighted average) of a random variable is its expected value. Furthermore, since the random variable is discrete, Eq. (2.11) will be applied. However, we first need to compute the probabilities:

p(x)	0.01 = 150/15000	0.03	0.13	0.25	0.39	0.17	0.02 = 300/15000

Eq. (2.11) can now be applied: $\bar{x} = 1\times 0.01 + 2\times 0.03 + ... + 7\times 0.02 = 4.57$ ■

Although Eqs. (2.11 & 2.12) can be used to find the expected value of a random variable, it is not always very convenient to do so.

If $M_W(t)$ is the moment-generating function (mgf) of the random variable *W*, then the following relationship holds as long as the rth derivative of mgf exists:

$$M_W^{(r)}(0) = E(W^r) \tag{2.14}$$

Let's prove for *r=1*.

$$M_Y^{(1)}(0) = \frac{d}{dt}\int_{-\infty}^{\infty} e^{ty} f_Y(y)\,dy$$

Placing the derivative as the integrand, then equation can be rewritten as:

$$M_Y^{(1)}(0)=\int_{-\infty}^{\infty}\frac{d}{dt}e^{ty}f_Y(y)dy$$

Noting that only e^{ty} is a function of t and performing the derivation yields:

$$M_Y^{(1)}(0)=\int_{-\infty}^{\infty}y\,e^{ty}f_Y(y)dy$$

Replacing $t=0$ gives:

$$M_Y^{(1)}(0)=\int_{-\infty}^{\infty}y\,f_Y(y)dy=E(Y)$$

Note that the last equation is exactly the same as Eq. (2.12), which is the expected value, $E(Y)$. Therefore, the first-derivative of mgf with respect to $t=0$ gives $E(Y)$ and second-derivative $E(Y^2)$ and so on…

Example 2.9

Find the expected value of the binomial random variable.

Solution:

The moment-generating function was already computed in example (2.7) as:

$$M_X(t)=\left(1-p+pe^t\right)^n$$

Taking the derivative with respect to t:

$$M_X^{(1)}(t)=n\left(1-p+pe^t\right)^{n-1}\cdot pe^t$$

Replacing $t=0$ yields the final answer:

$$M_X^1(t=0)=E(X)=np$$

■

Example 2.10

The PDF of the maximum order statistics is given by:

$$f_{Y_{(n)}}(y)=n\left[F_Y(y)\right]^{n-1}f_Y(y)$$

Find the expected maximum for uniform distribution in the interval of [0, 1].

Solution:

Substituting above equation in Eq. (2.12) yields:

$$E[Y_{(n)}]=\int_{-\infty}^{\infty} y\cdot n[F_Y(y)]^{n-1}f_Y(y)\,dy$$

The maximum PDF for uniform distribution in the interval of [0, 1] is $f_Y(y)=1$, therefore the cumulative distribution function is $F_Y(y)=y$. Substituting these knowns to above equation and integrating in the interval of [0, 1] yields $\frac{n}{n+1}$.

Note that as n increases, the expected maximum values approaches to 1, which we would expect if we draw large number of samples from a uniform distribution. ■

2.2.7. Variance

Although the expected value is an effective statistical measure of central tendency, it gives no information about the spreadout of a probability density function. Although the spread can be calculated using $X-\mu$, it is immediately noted that negative deviations will cancel positive ones (Larsen & Marx, 2011). The variance of a random variable is defined as the expected value of its squared deviations. In mathematical terms,

$$Var(X)=E[(X-\mu)^2] \tag{2.15}$$

Noting the following property of expected value for the random variable X and $g(X)$ any function,

$$E[g(X)]=\sum_{\text{all k}} g(k)\cdot p_x(k) \tag{2.16}$$

If $g(X)$ in Eq. (2.16) is replaced with $(X-\mu)^2$, then Eq. (2.15) can also be expressed as,

$$Var(X)=\sum_{\text{all k}}(k-\mu)^2\cdot p_X(k) \tag{2.17}$$

If Y is a continuous random variable with PDF $f_Y(y)$, then

$$Var(Y)=E[(Y-\mu)^2]=\int_{-\infty}^{\infty}(y-\mu)^2\cdot f_Y(y)\,dy \tag{2.18}$$

Let W be any random variable, discrete or continuous, and a and b any two constants. Then,

$$Var(a\cdot W+b)=a^2\cdot Var(W) \tag{2.19}$$

Let W_1, W_2, …., W_n be a set of independent random variables. Then,

$$Var(W_1+W_2+...+W_n)=Var(W_1)+Var(W_2)+...+Var(W_n) \tag{2.20}$$

Example 2.11

Test whether Eq. (2.15) represents population or sample variance.

Solution:

Let's work on an arbitrarily chosen dataset: [4, 7, 6, 2, 7, 6].

Spreadsheet's have two equations for computing sample and population variance, namely *Var.S* and *Var.P*, respectively. Computation with *Var.S* and *Var.P* yielded 3.86667 and 3.22222, respectively. Let's investigate using Python libraries:

Script 2.6

```
import statistics as stat

x=np.array([4, 7, 6, 2, 7, 6]) #arbitrary numbers

#returns sample variance
varS1 = stat.variance(x.tolist())
varS2 = np.var(x, ddof=1) #notice ddof=1

#Using Equation
```

```
EX, EX2 = np.mean(x), np.mean(x**2)
varEq = EX2 - EX**2
varP = np.var(x, ddof=0) #notice ddof=0

print(f"Sample: statistics pkg= {varS1} and Numpy={varS2}")
print(f"Population: Equation= {varEq} and Numpy={varP}")
```

Sample: statistics pkg= 3.8666 and Numpy=3.8666

Population: Equation= 3.2222 and Numpy=3.2222

Notice that the number of samples was intentionally kept low to see the difference between sample and population variance since for large samples the difference becomes negligible.

Although Eqs. (2.17 & 2.18) can be used to find variances of discrete and continuous random variables, respectively, using MGF (if known/available) to find the variance can be more convenient as demonstrated in the following example.

Example 2.12

Find the variance of the binomial random variable.

Solution:

From example (2.7) the moment-generating function:

$$M_X(t)=\left(1-p+pe^t\right)^n$$

From example (2.9) the expected-value:

$$E(X)=np$$

From Eq. (2.14) we know that the second-derivative of mgf with respect to t gives $E(X^2)$, therefore:

$$M_X^2(t)=pe^t\cdot n\cdot(n-1)\cdot\left(1-p+pe^t\right)^{n-2}pe^t+n\left(1-p+pe^t\right)^{n-1}pe^t$$

Replacing $t=0$ gives $E(X^2)$:

$$E(X^2)=n(n-1)p^2+np$$

From Eq. (2.15) remembering that:

$$E[(X-\mu)^2]=E(X^2)-E(X)^2$$

Now all we have to do is to replace $E(X^2)$ and $E(X)$ which yields:

$$Var(X)=n(n-1)p^2+np-(np)^2$$

Tidying up the equation gives the final answer:

$$Var(X)=np(1-p)$$

3. Discrete Probability Distributions

All discrete probability distributions has the following properties:

1. For every possible x value, $0 \leq x \leq 1$.
2. $\sum_{\text{all x values}} p(x) = 1$

The general characteristics of a discrete probability distribution can be visualized using a probability histogram.

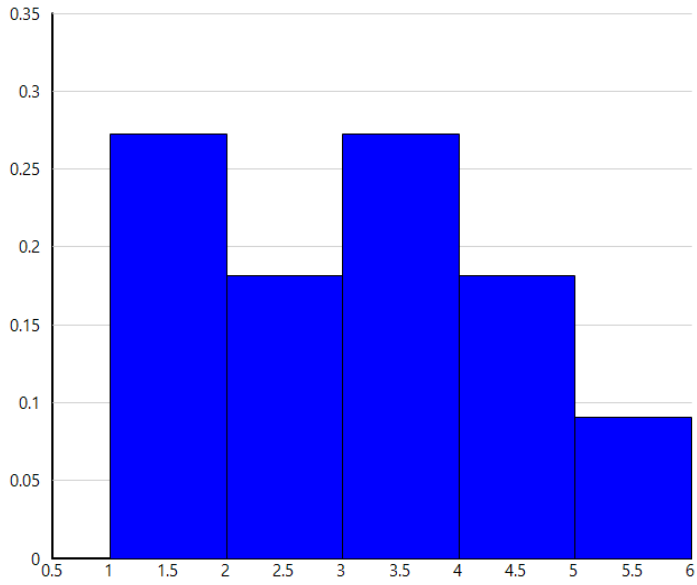


Fig 3.1: *Density histogram for a random data*

Script 3.1

```
import scisuit.plot as plt

plt.histogram([1, 2, 3, 4, 5, 3, 4, 2, 5, 4, 6])
plt.show()
```

Note that after the histogram has been plotted, the density option was selected and the number of bins were adjusted to 5.

3.1. Bernoulli Distribution

A Bernoulli trial can have one of the two outcomes, success or failure. The probability of success is *p* and therefore the probability of failure is *1-p* (Forbes *et al*., 2011). It is the simplest discrete distribution; however, it serves as the building block for other complicated discrete distributions (Weisstein 2023)[7].

The PDF is:

$$X_i = \begin{cases} 1 & p \\ 0 & 1-p \end{cases}, \quad 0 < p < 1 \tag{3.1}$$

MGF, Mean and Variance

$$M_X(t) = (1-p) + p \cdot e^t \tag{3.2}$$

$$E(X) = p \tag{3.3}$$

$$Var(X) = E(X^2) - E(X)^2 = p - p^2 = p(1-p) \tag{3.4}$$

7 Weisstein, Eric W. "Bernoulli Distribution." From https://mathworld.wolfram.com/BernoulliDistribution.html

3.2. Binomial Distribution

The outcome of the experiment is either a *success* or a *failure*. The term *success* is determined by the random variable of interest (*X*). For example, if *X* counts the number of female births among the next *n* births, then a female birth can be considered as a *success* (Peck *et al.*, 2016).

We run *n* independent trials and define probability as $p = P(success\,occurs)$ and assume *p* remains *constant* from trial to trial (Larsen & Marx, 2011). However, since we are only interested in the total number of *successes*, we therefore define *X* as the total number of successes in *n* trials. This definition then leads to binomial distribution and is expressed as:

$$p_x(k) = P(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{3.5}$$

Imagine 3 coins being tossed, each having a probability of *p* of coming up heads. Then the probability of all heads (HHH) coming up is p^3 and all tails (no heads, TTT) is $(1\text{-}p)^3$ and HHT is $3p^2(1\text{-}p)$.

Observe that in Eq. (3.5) the combination part shows the number of ways to arrange *k* heads and *n*-*k* tails (section 2.1), therefore:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{3.6}$$

the remaining part of Eq. (3.5), $p^k \cdot (1-p)^{n-k}$, is the probability of any sequence having *k* heads and *n*-*k* tails.

Example 3.1

An IT center uses 9 drives for storage. The probability that any of them is out of service is 0.06. For the center at least 7 of the drives must function properly. What is the probability that the computing center can get its work done (Adapted from Larsen & Marx, 2011)?

Solution #1:

$$\binom{9}{7}0.94^{7}0.06^{2}+\binom{9}{8}0.94^{8}0.06^{1}+\binom{9}{7}0.94^{9}0.06^{0}=0.986$$

```
sum( dbinom(x=[7, 8, 9], size=9, prob=0.94) )
```

```
0.986
```

Solution #2:

$$\binom{9}{7}0.94^{7}0.06^{2}+\binom{9}{8}0.94^{8}0.06^{1}+\binom{9}{7}0.94^{9}0.06^{0}=1-\sum_{i=0}^{6}\binom{9}{i}0.94^{i}0.06^{(9-i)}$$

```
1 - pbinom(q=6, size=9, prob=0.94)
```

```
0.986
```

Example 3.2

Find the 10% quantile of a binomial distribution with 10 trials and probability of success on each trial is 0.4?

```
qbinom(p=0.10, size=10, prob=0.4)
```

```
2.0
```

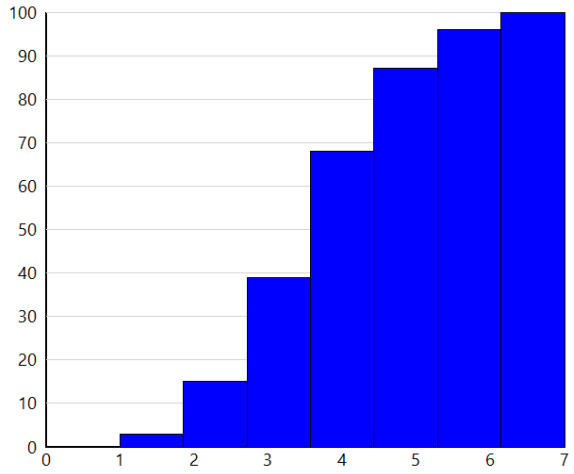


Fig 3.2: *Cumulative histogram of 100 random data points*

Figure shows the results of a simulation run by generating 100 random data points from the binomial distribution.

Script 3.2

```
import scisuit.plot as plt
from scisuit.stats import rbinom

data = rbinom(n=100, size=10, prob=0.4)

plt.hist(data, cumulative=True)
plt.show()
```

It is seen that 10% quantile is somewhere around 1.8, less than 2; however, when reporting it is rounded up[8]. The following two commands shines more light on this policy:

```
pbinom(q=1, size=5, prob=0.3)
```
```
0.52822
```
```
pbinom(q=2, size=5, prob=0.3)
```
```
0.83692
```
```
qbinom(p=[0.53, 0.80], size=5, prob=0.3)
```
```
[2, 2]
```

It can be seen that although p=0.53 is closer to q=1 (0.52822) whereas p=0.80 is closer to q=2 (p=0.83692), any number in between p=0.52822 and p=0.83692 will be reported as q=2 by the *qbinom* function.

MGF, Mean and Variance

The derivations of MGF, *E(X)* and *Var(X)* were already presented in Examples (2.7), (2.9) and (2.12), respectively. Although approaches presented in the examples work very well, one can also keep in mind that each binomial trial is actually a Bernoulli trial, therefore the random variable W for binomial distribution is a function of Bernoulli random variables: $X_1, X_2, \ldots, X_n$, *yielding* $W=X_1+X_2+\ldots+X_n$. Thus Eq. (3.2) and Eq. (2.10) can be combined to derive Eq. (3.7). Remembering the linearity of expected value, similar approach can be used for *E(W)* and *Var(W)* to obtain Eqs. (3.7 & 3.8).

$$M_X(t)=\left(1-p+pe^t\right)^n \tag{3.7}$$

$$E(X)=np \tag{3.8}$$

$$Var(X)=np(1-p) \tag{3.9}$$

8 https://www.boost.org/doc/libs/1_40_0/libs/math/doc/sf_and_dist/html/math_toolkit/policy/pol_tutorial/understand_dis_quant.html

Let’s run a simple simulation to test Eq. (3.8):

Script 3.3

```
from scisuit.stats import rbinom

for size in [5, 10]:
    x = rbinom(n=1000, size=size, prob=0.3)
    print(f"size={size}, mean= {np.mean(x)}")
```

```
size=5, mean= 1.489
size=10, mean= 2.983
```

We have intentionally run large number of experiments (*N=1000*) for the simulation. Note that Eq. (3.8) and rbinom function match when *n=size* and *p=prob*. Therefore for the first case $E=5\times0.3=1.5$, which is close to 1.49.

To test Eq. (3.9), the following simulation can be run:

Script 3.4

```
p, n = 0.3, 10
x = rbinom(n=5000, size=10, prob=0.3)

print(f"variance = {np.var(x, ddof=0)}")
print(f"equation = {n*p*(1-p)}")
```

```
variance = 2.108
equation = 2.099
```

Finally, let’s test our understanding in the meaning of randomness of Binomial distribution. First let’s generate 10 random numbers from a Binomial distribution.

```
rbinom(n=10, size=5, prob=0.5)
```

```
[1, 2, 2, 1, 2, 3, 2, 2, 3, 2]
```

What do the numbers returned by the function mean?

In an analogy, we flip 5 coins (*size=5*) and count the number of heads (*prob=0.5*) which we consider as success. We run this experiment for 10 times (*n=10*). In the first experiment we have 1 heads, in the second 2 heads and so on.

3.3. Hypergeometric Distribution

Suppose that an urn contains r good chips and w defective chips (total number of chips $N = r + w$). If n chips are drawn out at random *without replacement*, and X denotes the total number of good chips selected, then X has a hypergeometric distribution and,

$$P(X = k) = \frac{\binom{r}{k} \cdot \binom{w}{n-k}}{\binom{N}{n}} \qquad \textbf{(3.10)}$$

Notes:

1. If the selected chip was returned back to the population, that is the chips were drawn *with replacement*, then X would have a binomial distribution (see Example 3.3).
2. Since we are interested in total number of good chips, it does not matter if it is $r_1 r_2 r_3 \ldots$ or $r_2 r_1 r_3 \ldots$. Therefore $\frac{r!}{(r-k)!}$ was divided by k! and we used $\binom{r}{k} = \frac{r!}{k! \cdot (r-k)!}$.

Example 3.3

An urn has 100 items, 70 good and 30 defective. A sample of 7 items is drawn. What is the probability that it has 3 good and 4 defective items? (adapted from Tesler 2017)[9]

Solution #1: Sampling with replacement

$$p(X = 3) = \binom{7}{3} \cdot 0.7^3 \cdot (1 - 0.7)^4 = 0.0972$$ (dbinom(x=3, size=7, prob=0.7))

Solution #2: Sampling without replacement

$$P(3 \text{ good and } 4 \text{ bad}) = \frac{\binom{70}{3} \cdot \binom{30}{4}}{\binom{100}{7}} = 0.0937$$ (dhyper(x=3, m=70, n=30, k=7))

9 https://mathweb.ucsd.edu/~gptesler/186/slides/186_hypergeom_17-handout.pdf

MGF, Mean and Variance

The MGF, mean and variance of hypergeometric distribution are presented by Walck (2007) and derivation of expected-value is given by Hogg *et al*. (2019).

$$M_X(t)=\frac{\binom{W}{n}}{\binom{N}{n}}\cdot {}_2F_1(-n,-r;w-n+1;t) \tag{3.11}$$

where F is hypergeometric function.

Let $p=\frac{r}{N}$ and $q=1-p$ then,

$$E(X)=np \tag{3.12}$$

$$Var(X)=npq\frac{N-n}{N-1} \tag{3.13}$$

Let's demonstrate Eq. (3.12) with a simple code:

Script 3.5

```
x = rhyper(nn=1000, m=70, n=30, k=7)
avg = np.mean(x)

print(f"mean = {avg}")
```

```
mean = 4.931
```

Explicitly expressing Eq. (3.12):

$$E(X)=np=n\cdot\frac{r}{N}$$

Transforming above equation to the notation used by the rhyper function:

$$E(X)=k\cdot\frac{m}{m+n}=7\cdot\frac{70}{30+70}=4.9$$

3.4. Geometric Distribution

It is similar to binomial distribution such that trials have two possible outcomes: success or failure. However, unlike binomial distribution where we were interested in the total number of successes, now we are only interested in the trial where *first success occurs*. Therefore, if *k* trials were carried out, *k-1* trials end up in failures and the k^{th} one occurs with success. Thus we define the random variable *X* as the trial at which the first success occurs (Larsen & Marx, 2011).

In more explicit terms, we have thus far said that: "first *k-1* trials end up in failure" *and* "k^{th} trial ends in success". Mathematically expressing,

$$P(X=k) = P(\text{first success on } k^{th} \text{ trial})$$

$$= P(\text{first k-1 ends in failure}) \cdot P(k^{th} \text{ trial ends in success})$$

which then leads to the following equation:

$$P(X=k)=(1-p)^{k-1}\cdot p \tag{3.14}$$

MGF, Mean and Variance

$$M_x(t)=\frac{p\,e^t}{1-(1-p)e^t} \tag{3.15}$$

$$E(X)=\frac{1}{p} \tag{3.16}$$

$$var(X)=E(X^2)-E(X)^2=\frac{1-p}{p^2} \tag{3.17}$$

Example 3.4

A political pollster randomly selects persons on the street until he encounters someone who voted for the Fun-Party. What is the probability he encounters 3 people who did not vote for the Fun-Party before he encounters one who voted. It is known that 20% of the population voted for the Fun-Party (adapted from Foley[10] 2019)?

Solution:

The probability of success (voted for Fun-Party) is: $p=\frac{20}{100}=0.2$

Since 3 have not voted for the Fun-Party (failure) and the next one voted, 4 trials carried out.

$$P(X=4)=(1-0.2)^{3}\cdot 0.2^{1}=0.1024$$

Using Python code:

```
dgeom(x=3, prob=0.2)
```

```
0.1024
```

Note that, in the definition of the function dgeom *x* is the number of failures, therefore, instead of *x=4*, *x=3* was used.

10 https://rpubs.com/mpfoley73/458721

3.5. Negative Binomial Distribution

In section (3.4) the geometric distribution was introduced where we defined the random variable X as the trial at which the *first* success occurs. Therefore the trials were discontinued as soon as a success occurred. Now instead of first success, we are interested in r^{th} success. Similar to geometric distribution each trial has a probability p of ending in success.

Therefore, we might have a sequence of {S, F, F, S, S, S} if we were interested in the r=4th success out of $k=6$ trials. Putting it in more mathematical terms,

3 successes before the 4th success: r-1

2 failures before the 4th success in $k-1=5$ trials: (k-1) – (r-1) = k-r

Now if we define the random variable X as the trial at which the r^{th} success occurs, then all the background work to obtain the probability density function has been done.

Before proceeding with the final pdf, also note that before the r^{th} success occurs, $k-1$ trials might have various different sequences having $r-1$ successes, such as {SFFSS} or {FSFSS} or so on… Note that this is indeed very similar to the idea presented in section (3.2) by Eq. (3.6). Therefore,

I) Before the r^{th} success occurs ($k-1$ trials), number of different sequences with $r-1$ successes:

$$\binom{k-1}{r-1}$$

II) ($r-1$ success in the first $k-1$ trials) and (success on k^{th} trial):

$$p^{r-1}(1-p)^{k-1-(r-1)}$$

Putting the equations in (I) and (II) together gives the pdf for negative binomial distribution:

$$p_X(k)=\binom{k-1}{r-1}p^r\cdot(1-p)^{k-r} \qquad \textbf{(3.18)}$$

Example 3.5

A process engineer wishes to recruit 4 interns to aid in carrying out lab tests for the development of a new technology. Let *p= P(randomly chosen CV is a fit)*. If *p* is 0.2, what is the probability that exactly 15 CVs must be examined before 4 interns can be recruited (Adapted from Carlton & Devore, 2014)?

Solution:

The pdf for negative-binomial distribution is:

$$p_X(k)=\binom{k-1}{r-1}p^r\cdot(1-p)^{k-r}$$

where *k=15*, *r=4* and *p=0.2*.

Substituting *k* and *r* in the equation:

$$p(X=15)=\binom{15-1}{4-1}0.2^4\cdot(1-0.2)^{15-4}=0.050$$

Using Python:

```
dnbinom(x=15-4, size=4, prob=0.2)
```

```
0.050
```

Note that in dnbinom function the argument *x* represents the number of failures (*k*-*r*).

Now, let's ask ourselves a simple question? Does the probability increase or decrease if the number of CVs to be examined increase or decrease?

```
for k in [4, 5, 10, 15, 20, 25, 50]:
    print(dnbinom(x=k-4, size=4, prob=0.2))
```

```
0.0016  0.005  0.035  0.05  0.043  0.029  0.001
```

It is seen that the probability rises to maximum at around 15 and then decreases. In this context, this means that it is very unlikely to find 4 suitable candidates by examining 5 CVs and it is not so much necessary to examine more than 25 CVs. Finally note that,

```
dbinom(x=4, size=4, prob=0.2)
```

```
0.0016  #0.2**4
```

3.5.1. Relationship to Geometric Distribution

Let G and B be random variables for geometric and negative-binomial distributions. The definitions of random variables are then as follows:

G: Trial at which the *first* success occurs

B: Trial at which the r^{th} success occurs

It is clearly seen that if $r=1$ then $B=G$, it can therefore be said that the *negative-binomial distribution generalizes the geometric distribution*.

Larsen & Marx (2011) expresses the relationship between negative-binomial and geometric distributions in the following way which is easier to derive a mathematical relationship between the random variables:

X = total number of trials to achieve *rth* success

= number of trials to achieve 1st success +

number of additional trials to achieve 2nd +

… +

number of additional trials to achieve *rth* success.

$$X = X_1 + X_2 + \ldots + X_r \tag{3.19}$$

where $X_1, X_2, \ldots, X_r$ are random variables for geometric distributions.

It should be observed that until the 1st success occurs the trials overlaps with the definition of geometric random variable. However, after the 1st success we are interested in the *additional* trials (please note the word *additional*) to observe the 2nd success and therefore the trials *between the 1st and 2nd success* fits again with the definition of geometric random variable. Continuing in this fashion the rationale for Eq. (3.19) is justified.

3.5.2. MGF, Mean and Variance

$$M_X(t)=\left[\frac{pe^t}{1-(1-p)e^t}\right]^r \tag{3.20}$$

$$E(X)=\frac{r}{p} \tag{3.21}$$

$$var(X)=\frac{r(1-p)}{p^2} \tag{3.22}$$

Although above-given equations can be derived directly from the PDF of negative-binomial distribution, Eq. (3.19) paves the way to combine Eqs. (2.10 & 3.15) to derive MGF in a very straightforward fashion. Also by using Eqs. (3.16 & 3.17) expected-value and variance can be derived conveniently as shown below:

1) $M_X(t) = M_{X_1}(t)\cdot M_{X_2}(t)\cdot ....M_{X_r}(t) = \left[\frac{pe^t}{1-(1-p)e^t}\right]^r$

2) $E(X) = E(X_1)+E(X_2)+...+E(X_r) = 1/p+1/p+...+1/p=r/p$

3) $var(X) = var(X_1)+var(X_2)+...+var(X_r) = \frac{1-p}{p^2}+\frac{1-p}{p^2}+...+\frac{1-p}{p^2}=\frac{r(1-p)}{p^2}$

3.6. Poisson Distribution

Poisson distribution is a consequence of Poisson limit, which is an approximation to binomial distribution when $n \to \infty$ and $p \to 0$.

3.6.1. Poisson Limit

The Poisson limit states that, if $n \to \infty$ and $p \to 0$ such that $\lambda = np$ remains constant, then for $k \geq 0$, the following relationship holds (Larsen & Marx, 2011):

$$\lim_{n \to \infty} \binom{n}{k} p^k (1-p)^{n-k} = \frac{e^{-np}(np)^k}{k!} \tag{3.23}$$

A proof of Eq. (3.23) is presented in various textbooks (Devore *et al*., 2021; Larsen & Marx, 2011).

Let's inspect the accuracy of Eq. (3.23) using Python code. There are two tests where each has different probabilities (p); however for both tests $\lambda = np$ remains constant as 1.

Test #1:

Script 3.6

```
n, kmax = 5, 5
```

Test #2

```
n, kmax = 100, 10
```

```
p = 1/n #probability

binom = dbinom(x=x, size=n, prob=p)
pois = dpois(x=x, mu=n*p) #lambda=1

D = np.abs(np.array(binom)-np.array(pois)) #difference

print(f"min:{min(D)} at k={np.argmin(D)}")
print(f"max:{max(D)} at k={np.argmax(D)}")
```

```
Test #1: min:0.0027 at k=5 & max:0.0417 at k=1,
Test #2: min:3.13e-08 at k=10, max:0.0018 at k=1
```

It is clearly seen that in both tests the Poisson limit approximates binomial probabilities fairly well. However, as evidenced from Test #2 where n was larger and p was smaller, the agreement between Poisson limit and binomial probabilities became remarkably good for all k.

Example 3.6

When data is transmitted over a data link, there is a possibility of errors being introduced. Bit error rate is defined as the rate (errors/total number of bits) at which errors occur in a transmission system[11]. Assume you have a 4 MBit modem with bit error probability 10^{-8}. What is the probability of exactly 3 bit errors in the next minute (adapted from Devore *et al.* 2021)?

Solution:

In a minute $4 \cdot 10^6 \frac{bits}{s} \times 60\,s = 240 \cdot 10^6$ bits will be transferred and probability of error is 10^{-8}. The errors can be at any sequence and we are interested in *total number* of errors, which is by definition is the binomial probability:

$$P(3) = \binom{240 \cdot 10^6}{3} (10^{-8})^3 (1 - 10^{-8})^{240 \cdot 10^6 - 3}$$

Since n is very large (240,000,000) and p is very small (10^{-8}) the above computation is an excellent candidate for Poisson's limit: $\lambda = np = 2.4 \cdot 10^8 \times 10^{-8} = 2.4$

```
#Binomial probability
dbinom(x=3, size=240000000, prob=1E-8)
```

```
0.2090142
```

```
#Poisson limit
dpois(x=3, mu=2.4)
```

```
0.2090142
```

If we pose the question, "*what is the probability at most 3 bit errors in the next minute*?", then the solution is:

$$P(X \leq 3) = \sum_{k=0}^{3} \binom{240 \cdot 10^6}{k} (10^{-8})^k (1 - 10^{-8})^{240 \cdot 10^6 - k}$$

```
#Poisson limit
ppois(q=3, mu=2.4)
```

```
0.7787229
```

11 https://www.electronics-notes.com/articles/radio/bit-error-rate-ber/what-is-ber-definition-tutorial.php

3.6.2. Poisson Distribution

The random variable X is said to have a Poisson distribution if,

$$P_X(k)=\frac{e^{-\lambda}\lambda^k}{k!} \tag{3.24}$$

where λ>0.

Example 3.7

7 cards drawn (with replacement) from a deck containing numbers from 1 to 10. Success is considered when 5 is drawn. Can the produced data be described by the Poisson distribution?

Solution: Simulation will be run using the following script:

Script 3.7

```
#size=7 cards, prob=1/10
XX = rbinom(n=10000, size=7, prob=0.1)

#Get unique elements (e.g. [0, 1, 2, 3, 4, 5]) and their frequencies
unique, Frequencies = np.unique(XX, return_counts=True)

total = float(np.sum(Frequencies))

#frequencies / total is the weighted average
aver = sum(Frequencies*unique)/total

probabilities = Frequencies/total
poisson = [dpois(x=float(i), mu=aver) for i in unique]

print(probabilities)
print(poisson)
```

```
[0.4781     0.3733     0.1253     0.0209     0.0021     0.0003]
[0.4983,    0.34708,   0.12087,   0.02806,   0.0048,    0.0007]
```

It is seen from the output that the probabilities can be well described by Poisson distribution. It should be noted that when the probability value in the simulation was increased to 0.5, the difference between actual and predicted probabilities increased.

3.6.3. MGF, Mean and Variance

$$M_X(t) = e^{\lambda \cdot (e^t - 1)} \tag{3.25}$$

$$E(X) = \lambda \tag{3.26}$$

$$Var(X) = \lambda \tag{3.27}$$

Derivation of Eq. (3.25) can be found in mathematical statistic textbooks (Devore *et al*. 2021; Wackerly *et al*. 2008).

3.6.4. Poisson Process

It is widely used counting processes (the number of accidents in an area; the outbreaks of diseases; …) and mostly used in situations where we *only* know the rate of occurrence of an event but the events occur completely at random, for example using historic data knowing that earthquakes occurring in a certain area with a rate of 3 per year. Note that we only know the rate of earthquakes and do not have any information on timings of the earthquakes as they occur completely at random (Anon[12]. 2023). If an event satisfies the above-mentioned conditions we can assume that Poisson process might be a good candidate to model such event.

12 https://www.probabilitycourse.com/chapter11/11_1_2_basic_concepts_of_the_poisson_process.php

3.7. Multinomial Distribution

The multinomial distribution is a generalization of the binomial distribution (Forbes *et al.*, 2011; Larsen & Marx, 2011). Let X_i show the number of times the random variable Y equals y_i, $i=1,2,\ldots,k$ in a series of n independent trials where $p_i=P(Y=y_i)$. Then,

$$P(X_1=x_1, X_2=x_2, \ldots, X_k=x_k)=\frac{n!}{x_1!\cdot x_2!\ldots x_k!}p_1^{x_1}\cdot p_2^{x_2}\ldots p_k^{x_k} \quad \textbf{(3.28)}$$

where $i=0, 1, \ldots, k$ and $\sum_{i=1}^{k} x_i=n$.

Notes:

1. The rationale for $\frac{n!}{x_1!\cdot x_2!\ldots x_k!}$ part is directly related to Eq. (2.2) in section (2.1).
2. Thinking along the lines of probability events:

   Trial #1: Event 1 (E_1) probability $p_1 \rightarrow$ n independent trials x_1 successes

   Trial #2: Event 2 (E_2) probability $p_2 \rightarrow$ n independent trials x_2 successes

   Trial #k: Event k (E_k) probability $p_k \rightarrow$ n independent trials x_k successes

   Since trials are ***independent*** then $P(E_1\cap E_2\cap\ldots\cap E_k)=p_1^{x_1}\cdot p_2^{x_2}\ldots p_k^{x_k}$

Example 3.8

A die is tampered such that the probability of each of its face appearing is $p_i=P(\text{face i appears})=ki$ where k is a constant. If the die is tossed 12 times, what is the probability that each face will appear exactly twice? Compute the probability for the case of a normal die (Adapted from Larsen & Marx, 2011)

Solution:

Since a die has 6 faces and the sum of probabilities must be equal to 1.0, it is straightforward to compute the constant k: $\sum_{i=1}^{6} k\cdot i=k\cdot\sum_{i=1}^{6} i=k\cdot\frac{6\times7}{2}=1\rightarrow k=\frac{1}{21}$

Since the question is asking that each face should appear twice, then all is left to apply Eq. (3.28):

$$P(X_1=2,\ldots,X_6=2)=\frac{12!}{2!2!2!2!2!2!}\cdot\left(\frac{1}{21}\right)^2\cdot\left(\frac{2}{21}\right)^2\ldots\left(\frac{6}{21}\right)^2=0.0005$$

With a normal die, each face would have probability of 1/6 and therefore:

$$P(X_1=2,\ldots,X_6=2)=\frac{12!}{2!2!2!2!2!2!}\cdot\left(\frac{1}{6}\right)^2\cdot\left(\frac{1}{6}\right)^2\ldots\left(\frac{1}{6}\right)^2=\frac{12!}{2^6}\cdot\left(\frac{1}{6}\right)^{12}=0.0034$$

Script 3.8

```
from scisuit.stats import dmultinom

#Tempered die
probs = [1/21*i for i in range(1,7)]
x = [2]*6

p = dmultinom(x=x, size=12, prob=probs)
print(f"probability (tempered)={p}")

#Normal die
probs = [1/6]*6

p = dmultinom(x=x, size=12, prob=probs)
print(f"probability (normal)={p}")
```

```
probability (tempered) = 0.00052
probability (normal) = 0.0034
```

3.7.1. Binomial/Multinomial Relationship

At the beginning of this section (3.7) it has already been mentioned that multinomial distribution is a generalization of the binomial distribution. Binomial distribution is characterized by two outcomes: *success* or *failure*, where the probability of success is p. In the language of multinomial distribution this corresponds to two events: $p_1=p$ and $p_2=1-p$. Furthermore if there are n trials $x_1=k$ will end up with success and $x_2=n-k$ with failure. Therefore replacing p_1, p_2 and x_1, x_2 in Eq. (3.28) yields:

$$P(X_1=k,X_2=n-k)=\frac{n!}{k!(n-k)!}p^k\cdot(1-p)^{n-k}$$

Noting that $\frac{n!}{k!(n-k)!}=\binom{n}{k}$, then one can see that above equation is exactly the same as Eq. (3.5).

3.7.2. MGF, Mean and Variance

The moment-generating function, mean and variance of multinomial distribution is given in various textbooks (Forbes *et al*., 2011; Larsen & Marx, 2011).

$$M_X(t)=\left(\sum_{i=1}^{k} p_i e^{t_i}\right)^n \tag{3.29}$$

$$E(X)=np_i \tag{3.30}$$

$$Var(X)=n\,p_i(1-p_i) \tag{3.31}$$

A proof of Eq. (3.29) is given by Taboga[13] (2024).

Let's simulate Eq. (3.30):

Script 3.9

```
import numpy as np
from scisuit.stats import rmultinom

n=10

#testing probabilities
p = np.array([0.05, 0.15, 0.30, 0.50 ])

#2D array
arr = np.array(rmultinom(n=1000, size=n, prob=p))

#4 means, each is mean of 1000 random numbers with probabilities 0.05, 0.15 ...
means = np.mean(arr, axis=1)

#expected value (n*p[i])
E_X = n*p

print(f"Difference = {means - E_X}")
```

```
Difference = [0.003 0.013 0.064 0.042]
```

It is seen that the *difference* for each probability is less than 0.1, therefore for a reasonably large number of random samples Eq. (3.30) predicts the mean adequately well.

13 https://www.statlect.com/probability-distributions/multinomial-distribution

3.8. Summary

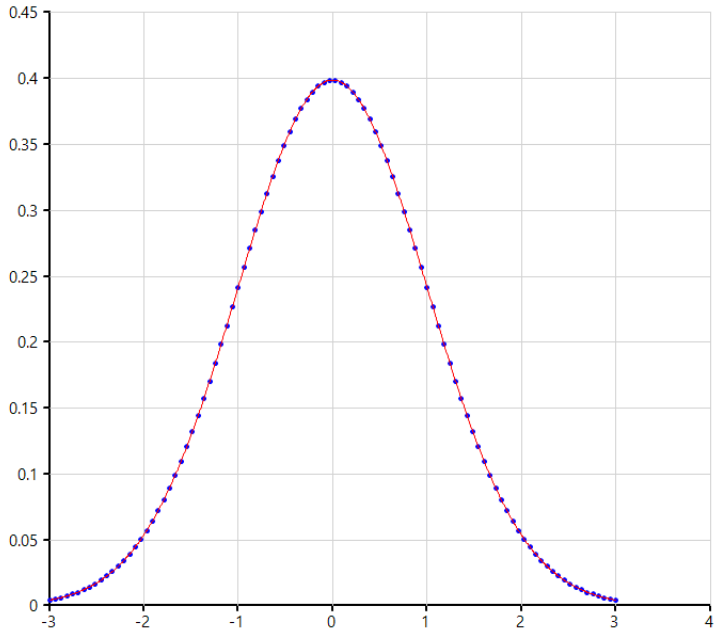
Name	Description	Equation
Bernoulli	One of the two outcomes, success (p) or failure (1-p).	$X_i=\begin{cases}1 & p\\ 0 & 1-p\end{cases},\ 0<p<1$
Binomial	Two possible outcomes: success (p) or failure (1-p). In *n* independent trials where p remains *constant*, we are only interested in the *total number of successes (k).*	$p_x(k)=\binom{n}{k}p^k(1-p)^{n-k}$
Hypergeometric	*n* chips are drawn out at random *without replacement*, and *X* denotes the total number of good chips ($N=r+w$).	$P(X=k)=\frac{\binom{r}{k}\cdot\binom{w}{n-k}}{\binom{N}{n}}$
Geometric	Two possible outcomes: success or failure. However, unlike binomial distribution in *k* trials we are only interested in the trial where *first success occurs*.	$P(X=k)=(1-p)^{k-1}\cdot p$
Negative Binomial	In geometric distribution we defined the random variable *X* as the trial at which the *first* success occurs. Now instead of first success, we are interested in r^{th} success. Therefore, it generalizes the geometric distribution.	$p_X(k)=\binom{k-1}{r-1}p^r\cdot(1-p)^{k-r}$
Poisson	We *only* know the rate of occurrence of an event but the events occur completely at random. The Poisson limit states that, if $n\to\infty$ and $p\to 0$ such that $\lambda=np$ remains constant, then for $k\geq 0$,	$P_X(k)=\frac{e^{-\lambda}\lambda^k}{k!}$
Multinomial	It is a generalization of the binomial distribution. In the language of multinomial distribution this corresponds to two events: $p_1=p$ and $p_2=1-p$. Furthermore if there are *n* trials $x_1=k$ will end up with success and $x_2=n-k$ with failure.	
$P(X_1=x_1,\ldots,X_k=x_k)=\frac{n!}{x_1!\cdot\ldots x_k!}p_1^{x_1}\cdot\ldots p_k^{x_k}$		

4. Continuous Probability Distributions

Continuous probability distributions have the following properties:

1. $f(x) \geq 0,$
2. $\int_{-\infty}^{\infty} f(x) = 1$

Continuous probability distributions can be visualized by a curve called a density curve. The function that defines this curve is called the density function.



Script 4.1

```
from numpy import linspace
from scisuit.plot import scatter, plot, show
from scisuit.stats import dnorm

x = linspace(start=-3, stop=3, num=100)
y = dnorm(x)

scatter(x=x, y=y, marker="c", markersize=3)
plot(x=x, y=y)

show()
```

Fig 4.1: *Density curve of standard normal distribution*

Using the following rationale in the above-given script, probability density curve for other distributions can be obtained.

4.1. Uniform Distribution

If you generate random numbers between 0 and 1 using a computer, you will get observations from a uniform distribution since there will be *almost* same amount of numbers in each equally spaced sub-interval, i.e. 0-0.2 or 0.2-0.4. Let's run a simulation:

Script 4.2

```
import random

x = np.array([random.random() for i in range(1000)])

start, dx=0.0, 0.2

while start<1.0:
    L = len( np.where( np.logical_and(x>=start, x<(start+dx)) )[0] )
    print(f"({start}, {start+dx}): {L}")

    start += dx
```

Output[14] is: *(0.0, 0.2): 218*, *(0.2, 0.4): 202*, *(0.4, 0.6): 192*, *(0.6, 0.8): 209*, *(0.8, 1.0): 179*

Although number of samples drawn was relatively small, it is seen that each sub-interval in the range of [0, 1] has similar amount of numbers. Instead of 1000 samples, if the simulation was run with 10,000,000 samples the difference between amount of numbers in each sub-interval would have been negligible.

A random variable *Y* has a continuous uniform probability distribution on the interval (a, b) if the PDF is defined as follows:

$$f(y)=\begin{cases}\dfrac{1}{b-a} & a\le y\le b\\ 0 & elsewhere\end{cases} \qquad \textbf{(4.1)}$$

The uniform distribution is very important for theoretical studies (Wackerly *et al*., 2008). For example if *F(y)* is a distribution function, it is often possible to transform uniform distribution to *F(y)*. For example, it is possible to transform it to standard normal distribution using Box-Muller transform[15].

14 It should be reminded that in random sampling each run will produce different results.
15 https://en.wikipedia.org/wiki/Box%E2%80%93Muller_transform

MGF, Mean and Variance

For *t≠0:*

$$M_Y(t)=\int_{-\infty}^{a} 0\cdot e^{ty}\,dy+\int_{a}^{b}\frac{e^{ty}}{b-a}+\int_{b}^{\infty} 0\cdot e^{ty}\,dy \;=\; \frac{e^{tb}-e^{ta}}{t(b-a)}$$

and for *t=0*:

$$M_Y(t)=\int_{-\infty}^{\infty} e^{ty}\cdot f_Y(y)\,dy \;=\; \int_{-\infty}^{\infty} 1\cdot\frac{1}{b-a}\,dy \;=\; 1$$

Therefore moment-generating function is:

$$f(y)=\begin{cases}\dfrac{e^{tb}-e^{ta}}{t(b-a)} & t\neq 0\\ 1 & t=0\end{cases} \qquad \textbf{(4.2)}$$

$$E(Y)=\frac{a+b}{2} \qquad \textbf{(4.3)}$$

$$Var(Y)=\frac{(b-a)^2}{12} \qquad \textbf{(4.4)}$$

It should be noted that the derivation (presented by Wolfram[16]) of Eq. (4.3) from Eq. (4.2) might pose challenges for many. Instead, it is recommended to use Eq. (2.12) as then the derivation becomes considerably more convenient.

16 https://mathworld.wolfram.com/UniformDistribution.html

Example 4.1

As evidenced from above a random number generator will spread its output uniformly across the entire interval from 0 to 1. What is the probability that the numbers will be in between 0.3 and 0.7?

Solution

This is a rather straightforward question and the answer is $P(0.3 \leq X \leq 0.7)=0.4$. Let's demonstrate it with a short script:

Script 4.3

```
from random import  random
from numpy import array, logical_and, where

#[10, 100, 1000, ...]
arr = array( [10**i for i in range(1, 6)] )

#helper function to create 1D list with j random numbers
func = lambda j: [random() for _ in range(j)]


# x[0] has 10, x[1] has 100 elements
x = list ( map(func, arr ) )

L = []
for lst in x:
    cond = logical_and(array(lst)>=0.3, array(lst)<0.7)
    length = len( where( cond )[0])
    L.append(length)

print(np.array(L)/arr)
```

```
[0.3   0.41   0.396   0.4022   0.39924]
```

As evidenced from the above output, as the number of samples in the array (*arr*) increased from 10 to 10^5, the simulated probability approached to the computed probability.

4.2. Normal Distribution

Normal distributions are bell-shaped and symmetric curves. They are widely used and are the single most important probability model in all of statistics since:

1. They provide a reasonable approximation to the distribution of many different variables,
2. They play a central role in many of the inferential procedures (Larsen & Marx, 2011; Peck *et al.*, 2016).

In section (3.6) it was shown that the Poisson limit approximated binomial probabilities when $n \to \infty$ and $p \to 0$. Historically, this was not the only approximation [interested reader can find a historical evolution of the normal distribution in the paper from Stahl (2006)]. Abraham DeMoivre showed that when X is a binomial random variable and n is large the probability for $P\left(a \le \frac{X-np}{\sqrt{np(1-p)}} \le b\right)$ can be estimated using the following equation:

$$f_Z(z) = \frac{1}{\sqrt{2\pi}} e^{\frac{-z^2}{2}}, -\infty < z < \infty \tag{4.5}$$

The formal statement of the approximation is known as DeMoivre-Laplace limit theorem (Larsen & Marx, 2011):

$$\lim_{n \to \infty} P\left(a \le \frac{X-np}{\sqrt{np(1-p)}} \le b\right) = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-z^2/2} dz \tag{4.6}$$

Eq. (4.5) is referred as the *standard normal curve* where μ=0 and σ=1. If μ≠0 and σ≠1 then the equation is expressed as follows:

$$f_Z(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-1}{2}\cdot\left(\frac{x-\mu}{\sigma}\right)^2}, -\infty < x < \infty \tag{4.7}$$

In order to show DeMoivre's idea, let's write a fairly short Python script. rbinom function was used to sample 1000 experiments where each experiment consists of 60 trials with a probability of success of 0.4 (adapted from Larsen & Marx, 2011).

Script 4.4

```
import scisuit.plot as plt
from scisuit.stats import rbinom

n, p = 60, 0.4

#Generate random numbers from a binomial distribution
x = np.array(rbinom(n=1000, size=n, prob=p))

#z-ratio
z = (x - n*p)/math.sqrt(n*p*(1-p))

#DeMoivre's equation
f = 1.0/math.sqrt(2*math.pi)*np.exp(-z**2/2.0)

#Density scaled histogram
plt.hist(z, density=True, breaks=5)

#Overlay scatter plot
plt.scatter(x=z, y=f)
plt.show()
```

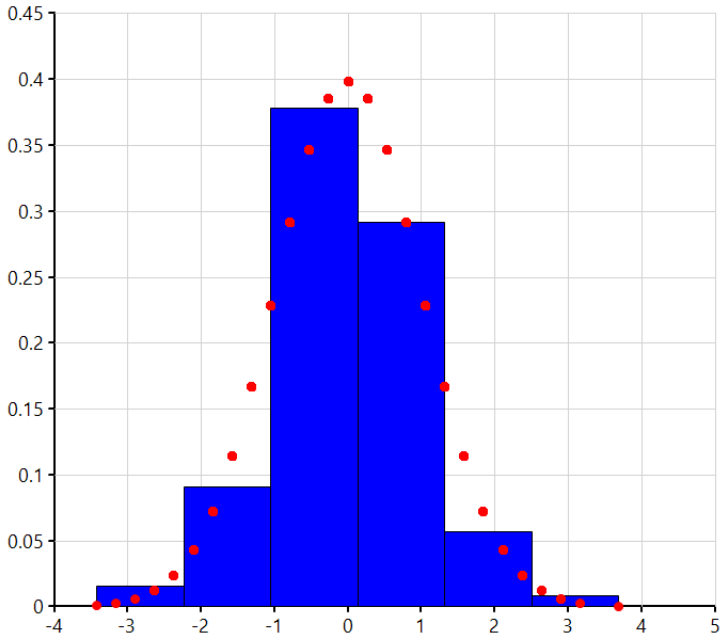


Fig 4.2: *Density scaled histogram and scatter plot (x-axis: z-ratio, y-axis: density)*

It is seen that the curve generated by DeMoivre's approximation equation is fairly well describing the variation of the histogram generated by the binomial data.

4.2.1. MGF, Mean and Variance

$$M_Y(t) = e^{\mu t + \sigma^2 t^2/2} \tag{4.8}$$

$$E(Y) = \mu \tag{4.9}$$

$$Var(Y) = \sigma^2 \tag{4.10}$$

4.2.2. Sampling Variability

When we would like to estimate the mean value of a population, we would take samples of size *n* from the population and try to make inferences based on the sample. It is natural that the average value of samples will *change from sample to sample*. This is known as *sampling variability.*

In order to simulate this we will generate a sample space of size 250 from an exponential distribution. Then we will draw samples of size 5, 10, 20 and 30 (250 times) from the sample space and compute the average of each sample. It will reveal us how the choice of sample size affects sampling distribution. We will run the following script:

Script 4.5

```
import numpy as np
import scisuit.plot as plt
import scisuit.stats as st

N = 250

#Generate random numbers from an exponential distribution
SS = np.array(st.rexp(n=N))

plt.layout(3, 2)

#Density scaled histogram of exponential distribution
plt.subplot(0,0)
plt.hist(SS)

n = [5, 10, 20, 30]
colors=["#FF0000", "#FFA500", "#00FF00", "#964B00"]
```

```
r, c = 1, 0
for i , v in enumerate(n):
    #take samples of size n[i] from the sample space (SS)
    x = [np.mean(np.random.choice(SS, size = v, replace=False)) for _ in range(N)]

    plt.subplot(r, c)
    plt.hist(x, fc = colors[i])
    plt.title(f"{chr(65+i)}) n={v}")

    c += 1
    if c%2 == 0:
        r += 1 ; c = 0

plt.show()
```

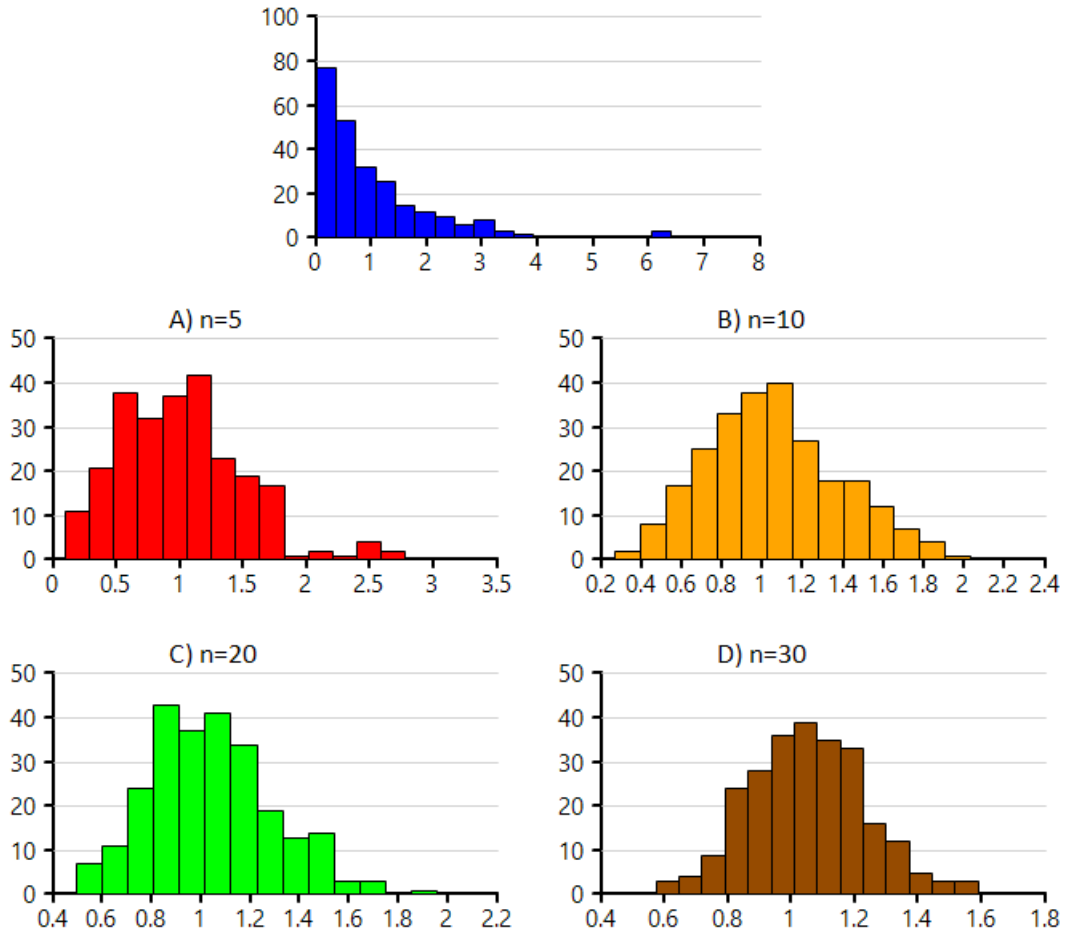


Fig 4.3: *Frequency histogram of sample space and different sample sizes*

The following inferences can be made from Fig. (4.3):

1. Although the histogram of sample space (variable *SS*) does not look like normal in shape, each of the four histograms is resembles to normal in shape,
2. Each of the histogram (*A*-*D*) has an average value close to the sample space's average value. Generally, $\bar{x}$ based on a larger sample size is closer to the mean value of the population.
3. The smaller the value of sample size, the greater the sampling distribution spreads out (compare the limits of x-axis for *A* and *D* where sample sizes were 5 and 30, respectively).

4.2.3. Central Limit Theorem

When n is sufficiently large (n≥30), the sampling distribution of $\bar{x}$ is well approximated by a normal curve (Peck *et al*., 2016). Formally expressing, let W_1, W_2, … be an infinite sequence of independent random variables each with the same distribution. Then,

$$\lim_{n\to\infty} P\left(a \le \frac{W_1+\ldots+W_n-n\mu}{\sqrt{n}\sigma} \le b\right) = \frac{1}{\sqrt{2\pi}}\int_a^b e^{-z^2/2}dz \qquad \textbf{(4.11)}$$

$$E\left[\frac{1}{n}(W_1+\ldots+W_n)\right] = E(\bar{W}) = \mu \qquad \textbf{(4.12)}$$

$$Var\left[\frac{1}{n}(W_1+\ldots+W_n)\right] = \frac{\sigma^2}{n} \qquad \textbf{(4.13)}$$

The implication of Eq. (4.13) could be observed from Fig. (4.3) where increasing the sample size decreased the variability of the distribution. In order to show how Eq. (4.11) works we will be generating an array with 5 columns and 250 rows from a standard uniform distribution. Then, sum of 5 columns will be computed to generate another array (250 rows). Since for a standard uniform distribution μ=0.5 and σ^2=1/12, z-ratio will be computed using $\frac{y-5/2}{\sqrt{5/12}}$.

Script 4.6

```
from math import  sqrt, pi
from numpy import array, exp, sum

import scisuit.plot as plt
from scisuit.stats import runif

n = 5

#For uniform distribution
mu, sigma = 0.5, sqrt(1/12)

#generate a list of random numbers from uniform distribution
G = lambda _: runif(n=250)

#2D list (5 rows and 250 columns) Python list
L = list(map(G, [None]*n))

#2D Numpy array (250*5)
W = array(L).transpose()

#W1+W2+...
x = sum(W, axis=1) #len=250

#z-ratio
z = (x - n*mu)/(sqrt(n)*sigma)

#DeMoivre's equation
f = 1.0 / sqrt(2*pi)*exp(-z**2/2.0)

#Density scaled histogram
plt.hist(z, density=True,)

#Overlay scatter plot
plt.scatter(x=z, y=f)

plt.show()
```

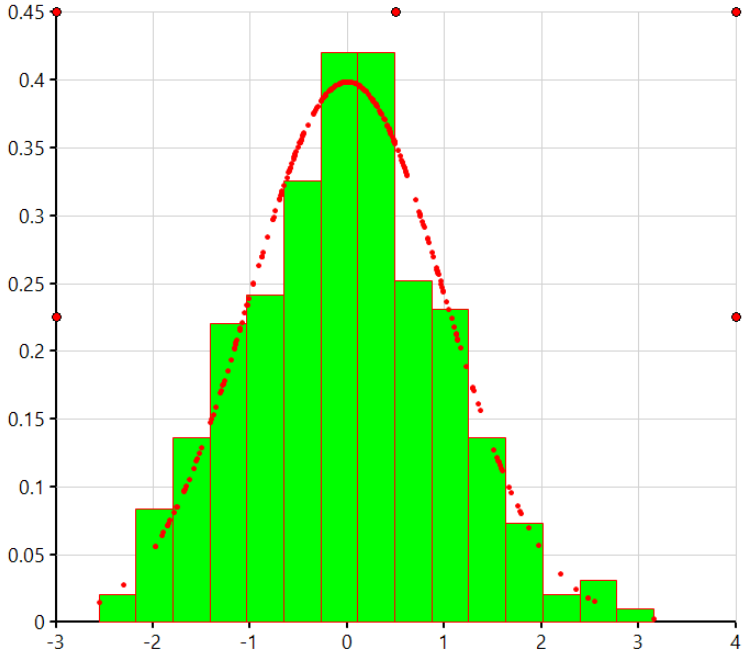


Fig 4.4: *Density scaled histogram and scatter plot (x-axis: z-ratio, y-axis: density)*

It is seen that even the number of samples were small (n=5), the sums yielded a distribution closely resembling to normal distribution.

Larsen & Marx (2011) states that samples from symmetric distributions will produce sums that will quickly converge to the theoretical limit (normal dist). However, if samples come from a skewed distribution then larger *n* is needed (see section on *sampling variability*)

4.2.4. The 68-95-99.7 Rule

A normal distribution with mean μ and standard deviation σ:

1. Approximately 68% of the observations fall within σ of the mean μ.
2. Approximately 95% of the observations fall within 2σ of μ.
3. Approximately 99.7% of the observations fall within 3σ of μ.

Script 4.7

```
N = 10000 #number of samples

#sample from standard normal distribution
x = np.array(rnorm(n=N))

for sigma in [1, 2, 3]:
    L = len(np.where(np.logical_and(x>=-sigma, x<=sigma))[0])

    print(f"{sigma} sigma= {L/N*100}%")
```

```
1 sigma= 68.61%, 2 sigma= 95.42%, 3 sigma= 99.75%
```

Note that *rnorm(n=,)* function samples from standard normal distribution where μ=0 and σ=1.

Example 4.2

A producer claims that bottles contain μ=12 deciliters of soda with σ=0.16 deciliters. To verify this claim as a quality control engineer you have randomly selected 16 bottles and measured the volume in each bottle. What is the probability that the average value of 16 bottles is in between 11.96 and 12.08 deciliters (adapted from Peck *et al*., 2016)?

Solution:

It is reasonable to assume that the samples come from a normal distribution. Standard deviation of sample:

$$\sigma_{\bar{x}}=\frac{\sigma}{n}=\frac{0.16}{\sqrt{16}}=0.04$$

Approach #1

Standardizing the given limits:

$$z_1=\frac{11.96-12}{0.04}=-1.0 \quad z_2=\frac{12.08-12}{0.04}=2.0$$

Probability that sample average will be between 11.96 and 12.08 is:

$$P\left(z_1\leq\bar{x}\leq z_2\right)=P\left(-1.0\leq\bar{x}\leq 2.0\right)=0.8185$$

Since the limits have been standardized we can use standard normal distribution to compute probabilities:

```
pnorm(q=2) – pnorm(q=-1)
```

```
0.8186
```

Approach #2

If not using the standard normal distribution then mean and standard deviation must be specified.

```
pnorm(q=12.08, mean=12, sd=0.04) - pnorm(q=11.96, mean=12, sd=0.04)
```

```
0.8186
```

4.3. Exponential Distribution

In section (3.6.4) it was mentioned that in situations where we only know the rate of occurrence (λ) of an event where the events occur completely at random might be a good candidate to be modeled by a Poisson model. However, situations might arise where the time interval between consecutively occurring event is an important random variable. The exponential distribution has many applications:

- The time to decay of a radioactive atom,
- The time to failure of components with constant failure rates,
- In the theory of waiting lines or queues (for example, time taken for an ambulance to arrive at the scene of an accident) (Forbes *et al*., 2011).

Suppose a series of events satisfying the Poisson process are occurring at a rate of λ per unit time. Let random variable Y denote the interval between consecutive events. Then,

$$f_Y(y)=\lambda e^{-\lambda y}, y>0 \tag{4.14}$$

MGF, Mean and Variance

$$M_Y(t)=\frac{\lambda}{\lambda-t} \tag{4.15}$$

$$E(Y)=\frac{1}{\lambda} \tag{4.16}$$

$$Var(Y)=\frac{1}{\lambda^2} \tag{4.17}$$

Example 4.3

During the period of 1832 to 1950, the following data was collected for the eruptions of a volcano:

126	73	3	6	37	23	73	23	2	65	94	51
26	21	6	68	16	20	6	18	6	41	40	18
41	11	12	38	77	61	26	3	38	50	91	12

Can the data be described by an exponential distribution model? (Adapted from Larsen & Marx, 2011)

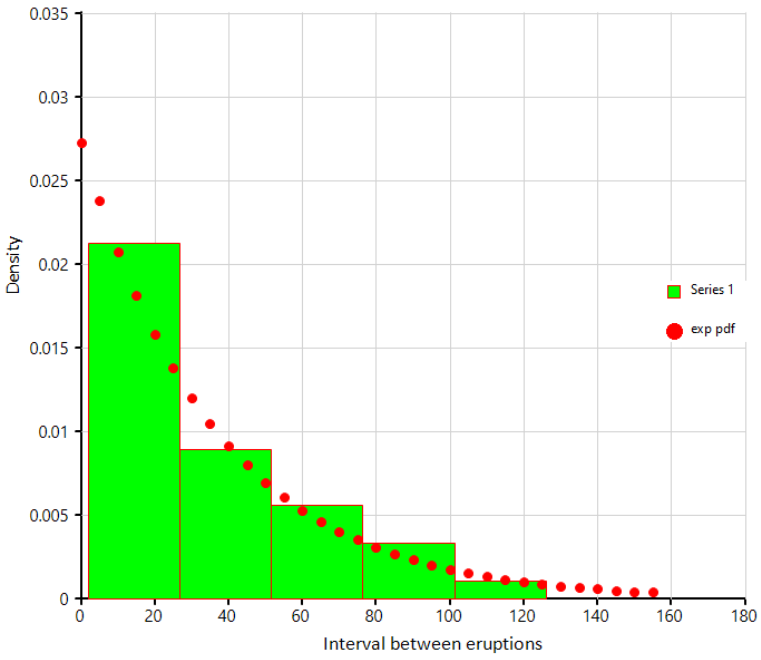
Solution:

In order to test whether exponential distribution is an adequate choice, first a density histogram of the data needs to be plotted. Then a scatter plot in the domain of the data using Eq. (4.14) will be overlaid. The following script handles both tasks:

Script 4.8

```
from numpy import array, linspace, average
import scisuit.plot as plt
from scisuit.stats import dexp

Data = array([126, 73, 3, 6, 37, 23, 73, 23, 2, 65, 94, 51, 26, 21, 6, 68, 16, 20, 6,
              18, 6, 41, 40, 18, 41, 11, 12, 38, 77, 61, 26, 3, 38, 50, 91, 12])

xvals = range(0, 160, 5)

plt.hist(Data, density=True, fc = "0 255 0", ec="255 0 0", label="data")
plt.scatter(x=xvals, y=dexp(x=xvals, rate=1/average(Data)), label="exp pdf")
plt.show()
```

The script will produce the following figure:



It is seen that the shape of the histogram is consistent with the theoretical model (exponential distribution).

Fig 4.5: *Histogram of raw data and exponential distribution describing data*

4.4. Gamma Distribution

In section (4.3), it was mentioned that if a series of events satisfying the Poisson process are occurring at a rate of λ per unit time and the random variable Y denote the interval between consecutive events it could be modeled with exponential distribution. Here the random variable Y can also be interpreted as the *waiting time* for the *first* occurrence.

This is similar to geometric distribution (section 3.4) where we were only interested in the trial where *first success occurs*. In section (3.5), in negative-binomial distribution instead of first success, we were interested in r^{th} success. Therefore, it was mentioned that the negative-binomial distribution generalizes the geometric distribution.

In a similar fashion, gamma distribution generalizes the exponential distribution such that we are now interested in the occurrence of (waiting time of) r^{th} event. However, before we proceed with the probability density function of gamma distribution we need to define the gamma function.

4.4.1. Gamma Function

It is a commonly used extension of factorial function and defined as:

$$\Gamma(z)=\int_0^{\infty} t^{z-1}\cdot e^{-t}\,dt \qquad \textbf{(4.18)}$$

With minor calculus, one can quickly see that $\Gamma(1)=1$. Using integration by parts[17], it is seen that: $\Gamma(z+1)=z\cdot\Gamma(z)$. Using induction one can further see that $\Gamma(n)=(n-1)!$.

Script 4.9

```
from math import  gamma, factorial

for i in [1, 2, 3, 4]:
    print(f"T({i})={gamma(i)}, ({i}-1)!={factorial(i-1)}")
```

```
T(1)=1.0,  (1-1)!=1
T(2)=1.0,  (2-1)!=1
T(3)=2.0,  (3-1)!=2
T(4)=6.0,  (4-1)!=6
```

17 https://en.wikipedia.org/wiki/Gamma_function

4.4.2. Probability Density Function

Suppose that Poisson events are occuring at constant rate of λ. Let random variable Y denote the waiting time for r^{th} event. Then,

$$f_Y(y)=\frac{\lambda^r}{(r-1)!}y^{r-1}e^{-\lambda y}, y>0 \quad \textbf{(4.19)}$$

A proof of Eq. (4.19) can be found in mathematical statistics textbooks (Larsen & Marx, 2011). Eq. (4.19) is often expressed in the following form (Devore *et al*., 2021; Miller & Miller, 2014; R-Documentation[18]) :

$$f(x;\alpha,\beta)=\frac{1}{\beta^{\alpha}\Gamma(\alpha)}x^{\alpha-1}e^{-x/\beta}, x>0 \quad \textbf{(4.20)}$$

Softwares such as *R* and *scisuit* Python package calls α as the shape and β as the scale parameter. Note that in Eq. (4.19) $r=\alpha$ and $\lambda=1/\beta$.

When $\beta=1$ the distribution is called the standard gamma distribution.

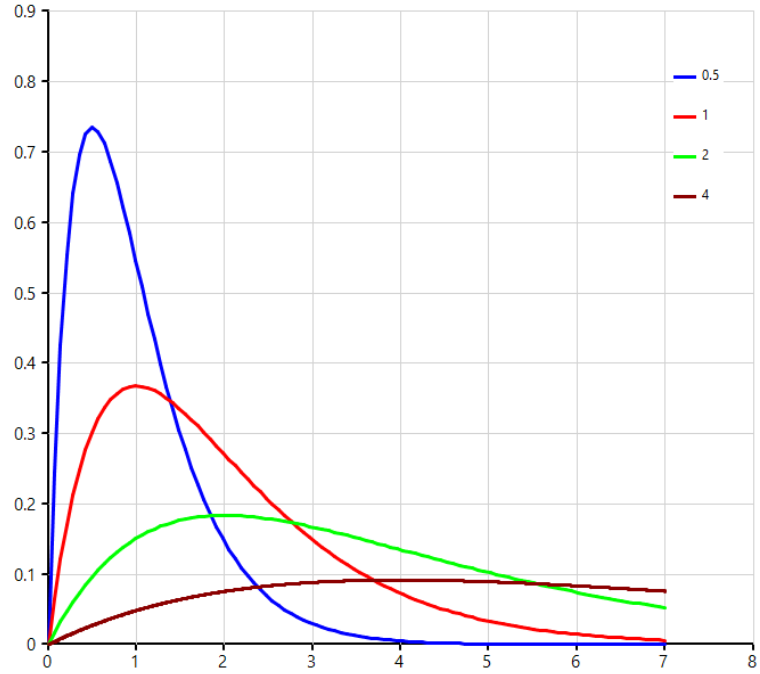
Devore *et al*. (2021) states that the parameter β is called a scale parameter because values other than 1 either stretch or compress the pdf in the x-direction. Let's visualize this using a constant shape factor, *shape=2*:

Script 4.10

```
from numpy import linspace
from scisuit.plot import scatter, show, legend
from scisuit.stats import dgamma

x=linspace(0, 7, num=100)

for beta in [0.5, 1, 2, 4]:
    scatter(x, dgamma(x=x, shape=2, scale=beta), label=str(beta), lw=3, ls="-")

legend()
show()
```

The following figure will be generated:

18 https://search.r-project.org/CRAN/refmans/ExtDist/html/Gamma.html



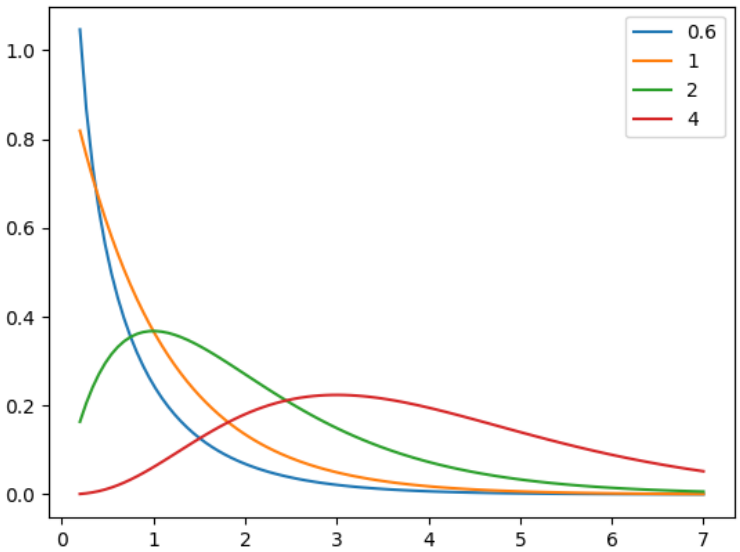
It is seen that for β=1, max value is around 0.35.

For a smaller β value, the curve is "compressed" and therefore became narrower and the max value increased to ~0.7.

For a larger β value the curve is "stretched" and therefore became wider and max value decreased to ~0.2 for β=2.

Fig 4.6: *Gamma density curves for different scale (β) values (α=2)*

With minor editing if the same script is run for different values of α=[0.6, 1, 2, 4], where β=1, then the following figure will be obtained:



It is seen that:

1) when α≤1, the curve is strictly decreasing as x increases.

2) when α> 1, f(x; α) rises to a maximum and then decreases as x increases.

Fig 4.7: *Standard gamma (β=1) density curves for different shapes (α)*

4.4.3. MGF, Mean and Variance

$$M_Y(t)=\frac{1}{(1-\beta t)^{\alpha}} \tag{4.21}$$

$$E(Y)=\alpha\cdot\beta \tag{4.22}$$

$$Var(Y)=\alpha\cdot\beta^2 \tag{4.23}$$

Example 4.4

As a process engineer you are given the task of designing a system to pump fluid from a reservoir to the processing plant. As this is important for the manufacturing to continue smoothly you have included two pumps, one active and one as a backup to be brought on line.

The manufacturer of the pump specifies that the pump is expected to fail once every 100 hours. What are the chances that the whole manufacturing will not remain functioning for 50 hours? (Adapted from Larsen & Marx, 2011)

Solution:

For the whole manufacturing to be interrupted, 2 pumps should fail, for example first after 10 hours and second after 40 hours… *Failure rate*: λ= 0.01 failure/hour

Approach #1:

We are going to use Eq. (4.19) where λ= 0.01 and r=2.

$$P(\text{manufacturing fails to last for 50 hours})=\int_0^{50}\frac{0.01^2}{(2-1)!}y^{2-1}e^{-0.01y}=0.09$$

Approach 2: We are going to use Eq. (4.20) where β= 100 and α=2.

```
pgamma(q=50, shape=2, scale=100)
```

```
0.09
```

Assume that 9% probability is too high for you. Another manufacturer claims that the pump they are offering is expected to fail once every 200 hours, but the price is double, therefore your costs will double. Would you use 3 pumps where each is expected to fail once every 100 hours or 2 pumps where each is expected to fail once every 200 hours to minimize the probability of 9%?

We will use a short script to generate the probability density curves and inspect the pdf's.

Script 4.11

```
from numpy import linspace
from scisuit.plot import plot, legend, show
from scisuit.stats import dgamma

x = linspace(0, 500, num=100)

plot(x, dgamma(x=x, shape=2, scale=100), label="2 pumps, 100h")
plot(x, dgamma(x=x, shape=3, scale=100), label="3 pumps, 100h")
plot(x, dgamma(x=x, shape=2, scale=200), label="2 pumps, 200h")

legend()
show()
```

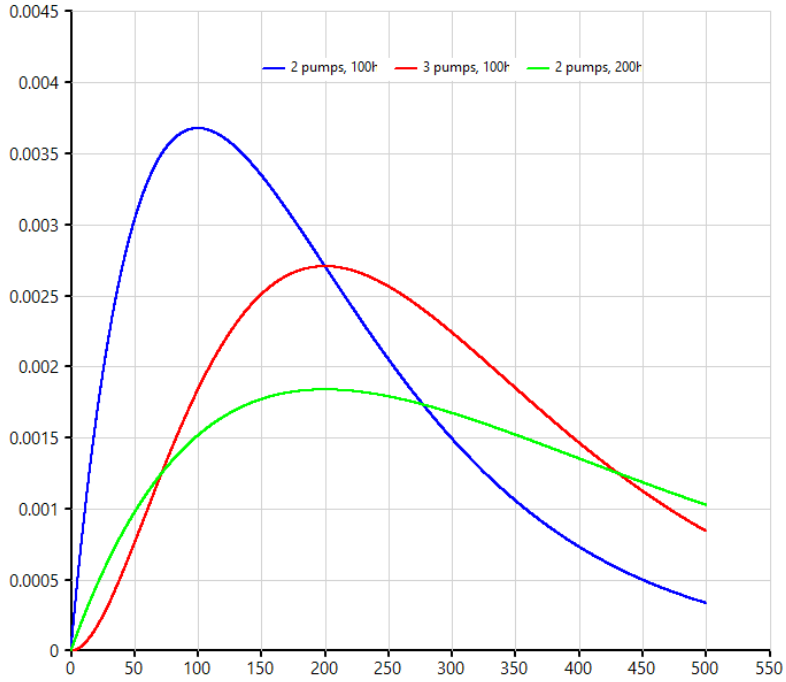


Fig 4.8: *Gamma curves for different number of pumps and failure rates.*

It is seen that using 3 pumps where each pump is expected to fail once every 100 hours will have a lower probability than using 2 pumps where each pump is expected fail once every 200 hours. Therefore, there is no need to double the cost.

However, also note that if instead of 50 hours we would like 80 hours or greater than using 2 pumps where each pump is expected fail once every 200 hours is a more reasonable approach in terms of lowering the probability.

4.5. Chi-Square Distribution

Chi-squared distribution is the sum of the squares of a number of normal distribution and this fact gives to important applications of it, i.e. analysis of contingency tables (Forbes *et al*., 2011).

4.5.1. One-way Frequency Table

Categorical univariate data consists of non-numerical observations which maybe placed in categories (Wikipedia[19]) and are most conveniently summarized in a one-way frequency table (Peck *et al*., 2016).

Suppose 100 people being surveyed whether they will go to a certain movie and choices (categories) are: *Definitely, Probably, Probably not, Definitely not*. Now a table can be formed from counting the observations:

Table 4.1: Results of the hypothetical survey

	Definitely	Probably	Probably not	Definitely not
Frequency	20	40	25	15

Let k be the number of categories of a categorical variable and p_k population proportion for category $k>0$. Then,

H_0: p_k is the hypothesized proportion for category k

H_a: H_0 is not true (at least one of the population category proportions differs from the corresponding hypothesized value).

$$X^2 = \frac{\sum_{\text{all cells}} (\text{Observed cell count} - \text{Expected cell count})^2}{\text{Expected cell count}} \quad \textbf{(4.24)}$$

where X^2 has approximately a chi-square distribution with $df= k-1$.

19 https://en.wikipedia.org/wiki/Univariate_(statistics)

Example 4.5

Lunar Phase	Number of Days	Number of Births
Phase 1	24	7680
Phase 2	152	48442
Phase 3	24	7579
Phase 4	149	47814
Phase 5	24	7711
Phase 6	150	47595
Phase 7	24	7733
Phase 8	152	48230

An urban legend claims that more babies are born during certain phases of the lunar cycle, especially near the full moon. Data for a sample of randomly selected births occurring during 24 lunar cycles are given in the table. Test whether the data support the urban legend claim (Adapted from Peck *et al*., 2016).

Solution:

There are 699 total days and a total of 222,784 births. The probability of a birth to happen at *Phase 1* is *24/699=0.0343* and at *Phase 8* is *152/699=0.2175*.

So if lunar phase did not have any effect, then we expect that at *Phase 1* there would be *0.0343×222784=7649.23* births. We continue our computations in this fashion and then use Eq. (4.24) to compute X^2 value.

Script 4.12

```
from scisuit.stats import pchisq

#Lunar periods
days = np.array([24, 152, 24, 149, 24, 150, 24, 152])

#Observed births at each lunar cycle
observed = np.array([7680, 48442, 7579, 47814, 7711, 47595, 7733, 48230])

#probabilities (ratios)
probs = days / np.sum(days)
#expected birth numbers
expected = np.sum(observed)*probs
```

```
chisq = (expected-observed)**2 / expected

#pchisq gives left tails probability
pval = 1 - pchisq(q=np.sum(chisq), df=len(chisq) - 1)

print(f"p-value: {round(pval, 3)}")
```

The output is: *p-value: 0.504.* Therefore, we cannot accept H_a (population category proportions differs from the corresponding hypothesized value). Thus, the claim is not supported by statistical evidence.

Forbes et al. (2011) states that to be able to use Eq. (4.24), the data produced from the differences between observed and expected values should be normally distributed. We will use QQ plot to check whether the data is normally distributed.

Script 4.13

```
import scisuit.plot as plt
from scisuit.stats import test_norm_ad

diff = observed-expected
print(f"Anderson-Darling test: {test_norm_ad(x=diff)}")

plt.qqnorm(data=diff)
plt.show()
```

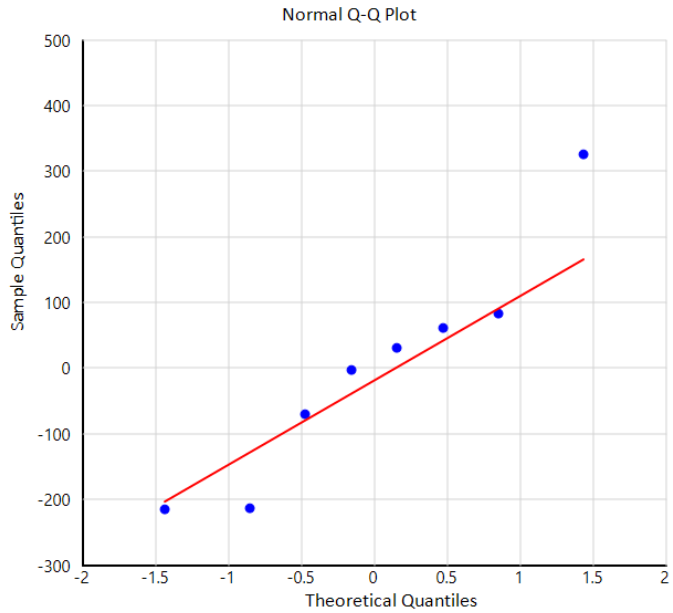


Fig 4.9: *QQ plot of the differences between observed and expected birth rates.*

Although there is an outlier point, the rest of the data follows the QQ-Line fairly well.

Moreover, the p-value reported by Anderson-Darling test is 0.448, therefore we cannot reject H_0 that "the data follows normal distribution".

4.5.2. Probability Density Function

A random variable *Y* is said to have a chi-square distribution with *n* degrees of freedom (*n>0*), if

$$f_Y(y)=\frac{1}{2^{n/2}\Gamma(n/2)}y^{(n/2)-1}e^{-y/2}, y>0 \quad \textbf{(4.25)}$$

Please note that Eq. (4.25) is a special case of Eq. (4.19) where $r=n/2$ and $\lambda=1/2$. Substituting these values in Eq. (4.19) and tidying up slightly yields:

$$f_Y(y)=\frac{1}{2^{(n/2)}(n/2-1)!}y^{n/2-1}e^{-1/2\,y}$$

Noticing that $(n/2\text{-}1)!=\Gamma(n/2)$, then one can see that above equation is equal to Eq. (4.25).

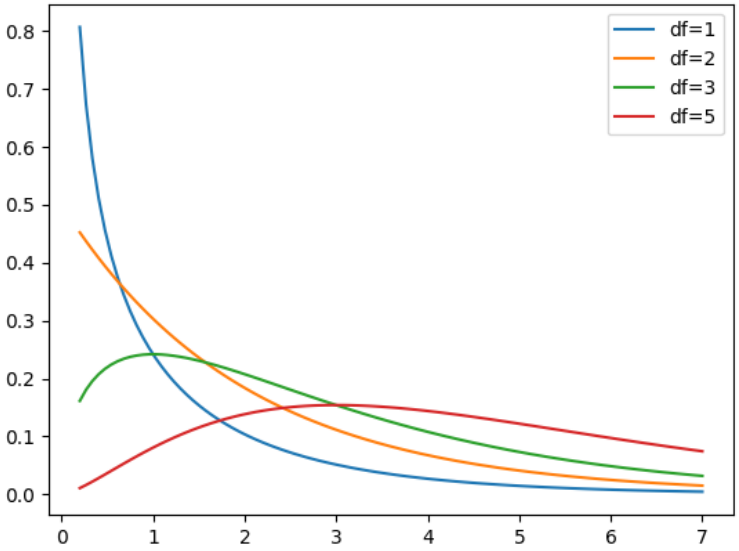


Fig 4.10: *Chi-square distribution with different degrees of freedoms*

The shape of chi-square distribution depends on the value of degrees of freedom(df):
df<3: decreases strictly as x increases,
df≥3: increases to a maximum and then decreases.

It should also be noted that regardless of *df*, all chi-square distributions are skewed to right.

Theorem: Let $Z_1, Z_2, \ldots, Z_n$ be *n* independent standard normal random variables. Then,

$$\sum_{i=1}^{n} Z_i^2$$

has chi-square distribution with *n* degrees of freedom. A proof of the theorem can be found in mathematical statistics textbooks (Larsen & Marx, 2011).

4.5.3. MGF, Mean and Variance

$$M_Y(t)=(1-2t)^{-n/2}, t<1/2 \tag{4.26}$$

$$E(Y)=n \tag{4.27}$$

$$Var(Y)=2n \tag{4.28}$$

Script 4.14

```
from scisuit.stats import rchisq

#number of samples
N = 1000

#arbitrary values for degrees of freedom
df = [1, 3, 5, 10]

#2D array of random values for each degrees of freedom
X=np.array([rchisq(N, x) for x in df])

#mean and variance
print(f"mean = {np.average(X, axis=1)}")
print(f"variance = {np.var(X, axis=1, ddof=0)}")
```

```
mean = [1.019    2.968    4.914    9.817]
variance = [2.131    5.696    9.669    19.553]
```

Notice how close the values are to the values that would be computed by Eqs. (4.27 & 4.28). For example for *df=1*, *E(Y)=1* and *Var(Y)=2*.

4.6. The Student's *t* distribution

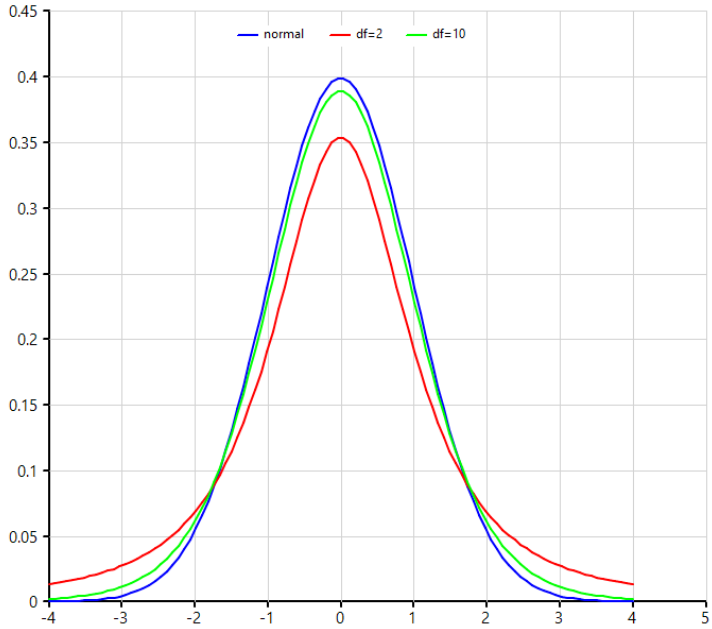
The t distribution is used to test whether the difference between the means of two samples of observations is statistically significant assuming they were drawn from the same population (Forbes *et al*., 2011).

In sections (4.2.2 & 4.2.3) it was shown that if y_1, y_2, …, y_n is a random sample from a normal distribution with mean μ and standard deviation ρ then $\frac{\bar{Y}-\mu}{\rho/n}$ has a standard normal distribution (SND). However Gosset (Student, 1908) realized that $\frac{\bar{Y}-\mu}{s/n}$ does not have a SND and derived the probability density function.

Let's see the differences between SND and t-distribution using the short Python code:

Script 4.15

```
from numpy import linspace
from scisuit.stats import dnorm, dt
from scisuit.plot import plot, show, legend

x = linspace(-4, 4, num=100)

plot(x, dnorm(x=x), label="normal")

for n in [2, 10]:
    scatter(x, dt(x=x, df=n), label=f"df={n}")

show()
```



1) Both dists are symmetric.

2) Both dists have a mean of 0.

3) t-dist is characterized by the degrees of freedom (df). As df increases, t-dist becomes more similar to a normal dist.

4) The curves of t-dist with larger df are taller and have thinner tails.

5) t-dist is most useful for small sample sizes.

Fig 4.11: *Standard normal distribution and t-distribution with different degrees of freedoms*

In comparison of t-distribution with SND it was mentioned that t-distribution is most useful for *small* sample sizes, but have not explained what is meant by *small*. Larsen & Marx (2011) states that many tables providing probability values for t-distribution will have it for degrees of freedom in the range of [1, 30]. Furthermore, elsewhere[20] it was mentioned that for a sample size of at least 30, SND can be used instead of t-distribution.

Let Z be a standard normal random variable and V an independent chi-square random variable with n degrees of freedom. The Student t ratio with n degrees of freedom is,

$$T_n = \frac{Z}{\sqrt{V/n}} \tag{4.29}$$

In line with observations from Fig. (4.11), Eq. (4.29) is symmetric: $f_{T_n}(t) = f_{T_n}(-t)$

The PDF for a Student t random variable with n degrees of freedom is,

20 https://www.jmp.com/en_no/statistics-knowledge-portal/t-test/t-distribution.html

$$f_{T_n}(n)=\frac{\Gamma\left(\frac{n+1}{2}\right)}{\sqrt{n\pi}\,\Gamma\left(\frac{n}{2}\right)\left(1+\frac{t^2}{n}\right)^{\frac{(n+1)}{2}}},-\infty<t<\infty \tag{4.30}$$

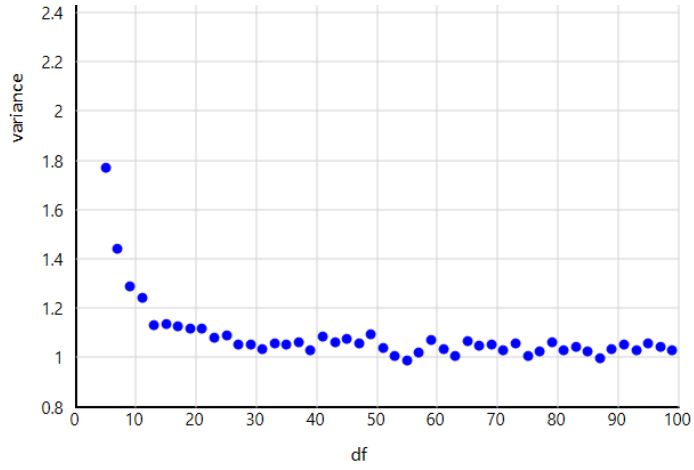
MGF, Mean and Variance

The moment-generating function of t-distribution is undefined[21] and its mean is 0 as can be observed from Fig. 4.11 for different degrees of freedom.

$$Var(Y)=\frac{n}{n-2},n>2 \tag{4.31}$$

Script 4.16

```
from scisuit.plot import scatter, show
from scisuit.stats import rt
from statistics import pvariance

var = []
for df in range(3, 100, 2):
    var.append(pvariance( rt(n=5000, df=df) ))

scatter(x=list(dfs), y=var)
show()
```



It is seen that for df>2 the variance is always larger than 1 and for large df the variance is close to 1 (this can also be observed from the equation).

Devore *et al*. (2021) states that for small dfs the t-dist curve spreads out more than the standard normal dist curve; however, for large dfs the t-dist curve approaches to standard normal dist curve (μ=0, ρ=1) (see above figure).

Fig 4.12: *Variance of t-distribution with different df values*

21 https://en.wikipedia.org/wiki/Student's_t-distribution

4.7. F (Fisher–Snedecor) Distribution

It is the ratio of independent chi-square random variables. Many experimental scientists use the technique called analysis of variance (ANOVA) (Forbes *et al.*, 2011). ANOVA analyzes the variability in the data to see how much can be attributed to differences in the means and how much is due to variability in the individual populations (Peck *et al.*, 2016). In one-way ANOVA, F is the ratio of variation among the samples to variation within the samples.

Suppose that U and V are independent chi-square random variables with m and n degrees of freedom, respectively. Then,

$$F = \frac{U/m}{V/n} \tag{4.32}$$

The PDF for F distribution is:

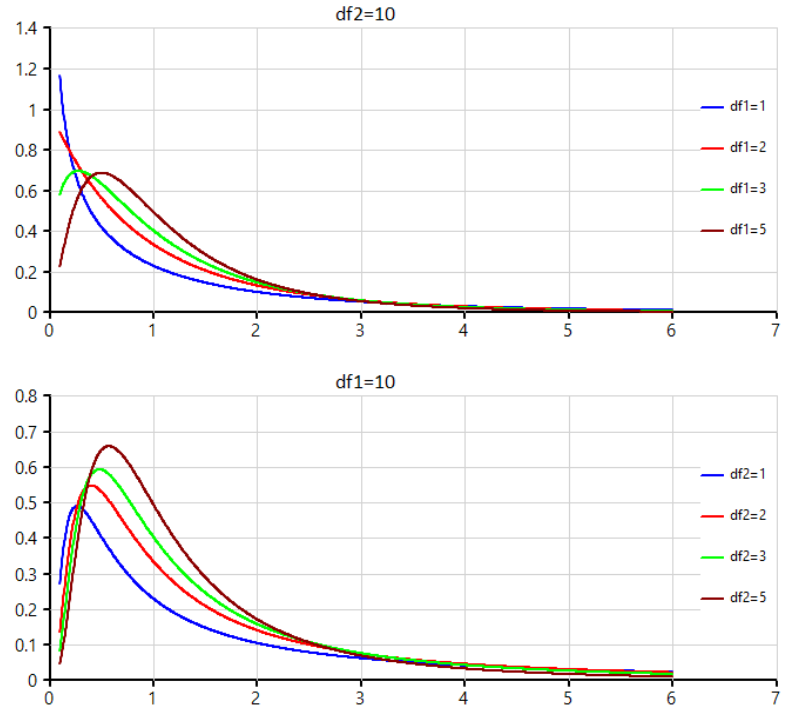
$$f_{F_{m,n}}(r) = \frac{\Gamma\left(\frac{m+n}{2}\right) m^{m/2} n^{n/2} r^{(m/2)-1}}{\Gamma\left(\frac{m}{2}\right)\Gamma\left(\frac{n}{2}\right)\left(n+mr\right)^{(m+n)/2}}, r>0 \tag{4.33}$$

The derivation of Eq. (4.33) is detailed in the textbook from Larsen & Marx (2011). Let's use a fairly short script to generate F-distribution curves for constant m (df_1) and varying n (df_2) and for constant df_2 and varying df_1.

Script 4.17

```
from scisuit.stats import df
from numpy import linspace
import scisuit.plot as plt

x_axis=linspace(0.1, 6, num=500)
dfree = 10

plt.layout(2,1)
plt.subplot(0,0)
for x in [1, 2, 3, 5]:
    plt.plot(x_axis, df(x_axis, df1=x, df2=dfree), label=f"df1={x}")
plt.title("df2=10")
plt.legend()
```

```
plt.subplot(1,0)
for x in [1, 2, 3, 5]:
   plt.plot (x_axis, df(x_axis, df1=dfree, df2=x), label=f"df2={x}")
plt.title("df1=10")
plt.legend()

plt.show()
```



It is seen that when **df2** is constant the F dist curves looks very much like a typical chi-square dist curves.

When **df1** is constant, all F dist curves rapidly rises to a maximum and then decreases in value as x increases.

In all cases, F values are never negative and sharply skewed to the right.

Fig 4.13: *F-distribution curves for constant A) df2, B) df1*

MGF, Mean and Variance

The moment-generating function of F distribution does not exist[22].

$$E(Y)=\frac{n}{n-2}, n>2 \tag{4.34}$$

$$Var(Y)=\frac{2n^2(m+n-2)}{m(n-2)^2(n-4)} \tag{4.35}$$

22 https://en.wikipedia.org/wiki/F-distribution

4.8. Weibull Distribution

Weibull distribution is commonly used as a lifetime distribution in reliability applications (Forbes *et al*., 2011). It is of great interest to statisticians and to practitioners because of its ability to fit to data from various fields including engineering sciences (Rinne, 2009).

A random variable X has Weibull distribution ($\alpha>0$, $\beta>0$), if the PDF is defined as follows:

$$f(x;\alpha,\beta)=\begin{cases}\frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1}e^{-(x/\beta)^{\alpha}} & x\geq 0\\ 0 & x<0\end{cases} \qquad \textbf{(4.36)}$$

where α and β are the shape and scale parameters, respectively. According to Rinne (2009), Eq. (4.36) is the most often used two-parameter (third parameter, the location was assumed to be 0) Weibull distribution.

4.8.1. Effect of parameters

When $\alpha=1$, the Eq. (4.36) reduces to the exponential distribution (Eq. 4.14). Therefore, exponential distribution is a special case of both the gamma and Weibull distributions.

Replacing α with 1 in Eq. (4.36) gives:

$$f(x;\alpha=1,\beta)=\frac{1}{\beta}e^{-(x/\beta)}$$

If $\lambda=1/\beta$ then

$$f(x;\beta)=\lambda e^{-\lambda y}$$

which is exactly the same as Eq. (4.14).

The shape parameter (α) can be interpreted in the following way too:

- $0<\alpha<1$ → the failure rate decreases over time (waiting time between two subsequent stock exchange transactions of the same stock),
- $\alpha=1$ → the failure rate is constant over time (radioactive decay of unstable atoms),
- $\alpha>1$ → the failure rate increases over time (wind speeds, distribution of the size of droplets)

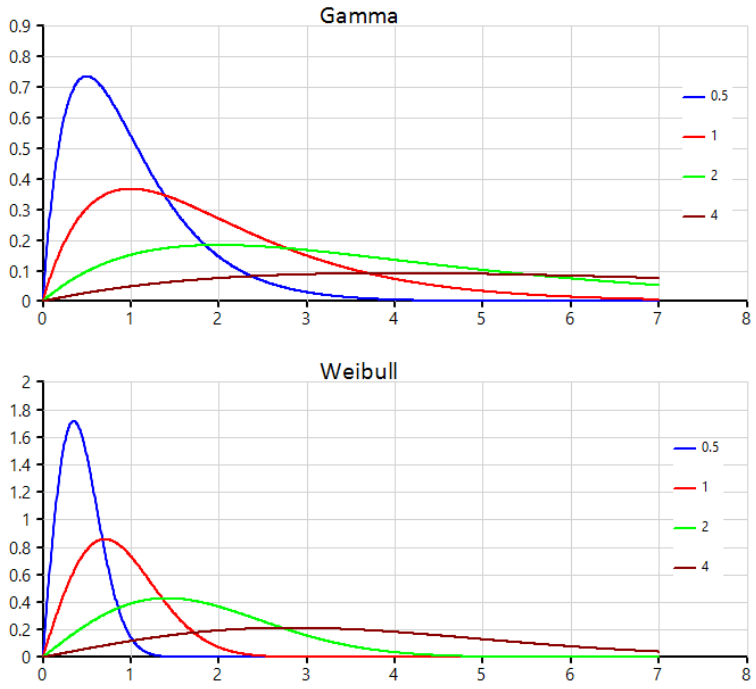
A “bathtub” diagram and the α-values for above-mentioned examples are presented by Kızılersü[23] *et al*. (2018).

Now, let’s remember that in section (4.4.2), it was mentioned that Gamma distribution has *shape* and *scale* parameters, which is similar to Weibull distribution. Let’s investigate the similarities and differences:

Script 4.18

```
from numpy import linspace
from scisuit.stats import dgamma, dweibull
import scisuit.plot as plt

x=linspace(0, 7, num=1000)

plt.layout(nrows=2, ncols=1)

plt.subplot(0,0)
for beta in [0.5, 1, 2, 4]:
   plt.plot(x, dgamma(x=x, shape=2, scale=beta), label=str(beta))
plt.title("Gamma")
plt.legend()

plt.subplot(1,0)
for beta in [0.5, 1, 2, 4]:
   plt.plot(x, dweibull(x=x, shape=2, scale=beta), label=str(beta))
plt.title("Weibull")
plt.legend()

plt.show()
```

23 **Kızılersü A, Kreer M, Thomas AW**. The Weibull Distribution. *Significance*, April 2018.



Similarities:

1) For a smaller β value, the curve is "compressed" and therefore became narrower.

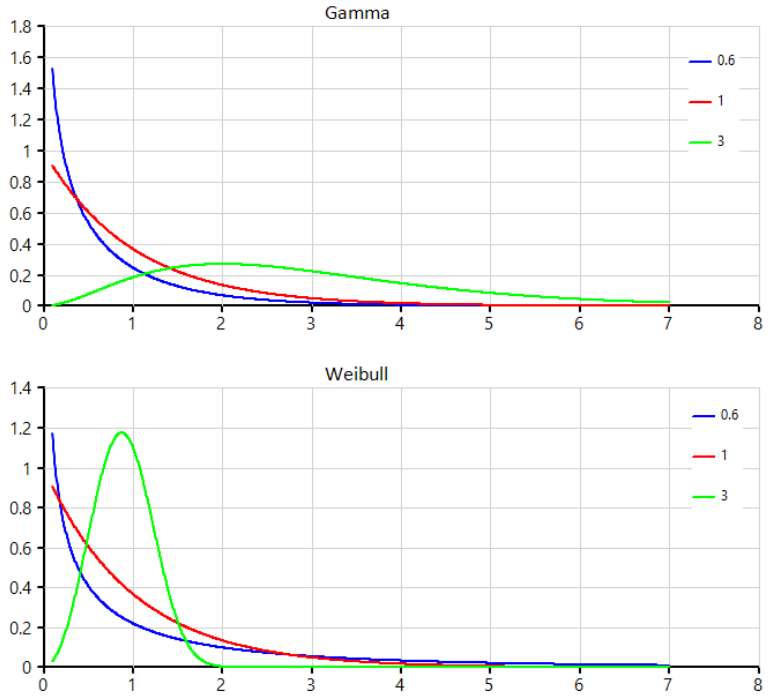
2) For a larger β value the curve is "stretched" and therefore became wider.

Differences:

1) Weibull is compressed more and stretched less.

2) Both are right-skewed; however, Gamma distribution has longer tail.

Fig 4.14: *Gamma and Weibull density curves for different scale (β) values (α=2)*



Similarities:

1) when α≤1, the curve is strictly decreasing as x increases.

2) when α> 1, f(x; α) rises to a maximum and then decreases as x increases.

3) when α=1, both distribution shows exactly the same characteristics (Why?).

Differences:

1) when α> 1, f(x; α) rises to a maximum; however, Weibull dist decreases sharply whereas Gamma dist decreases gradually as x increases.

2) when α> 1, by modifying the script it was observed that Weibull dist has the maximum peak approximately around x=0.5-1.0 whereas for Gamma dist x-values ranged considerably.

Fig 4.15: *Gamma and Weibull density curves for different shape (α) values (β=1)*

4.8.2. MGF, Mean and Variance

$$\sum_{n=0}^{\infty} \frac{t^n \beta^n}{n!} \Gamma\left(1+\frac{n}{\alpha}\right), \alpha \geq 1 \tag{4.37}$$

$$E(Y) = \beta \Gamma\left(1+\frac{1}{\alpha}\right) \tag{4.38}$$

$$Var(Y) = \beta^2 \left\{ \Gamma\left(1+\frac{2}{\alpha}\right) - \left[\Gamma\left(1+\frac{1}{\alpha}\right)\right]^2 \right\} \tag{4.39}$$

Example 4.6

The article[24] by Field and Blumenfeld (2016) investigates modeling the time to repair for reusable shipping containers, which are fairly expensive and need to be monitored carefully. The random variable X defined as the time required for repairing in months. The authors recommended the Weibull distribution with parameters α=10.0 and β=3.5. What is the probability that a container requires repair within the first 3 months?

Solution:

Script 4.19

```
from scisuit.stats import pweibull

for m in [2, 3, 4]:
    print(f"P ({m} months) = {pweibull(q=m, shape=10, scale=3.5)}")
```

```
P (2 months) = 0.0037
P (3 months) = 0.193
P (4 months) = 0.978
```

Note that we are almost certain that a container will not require a repair the first two months but will definitely require a repair within first 4 months. *Why is that*?

24 **Field DA, Blumenfeld D.** Supply Chain Inventories of Engineered Shipping Containers. International Journal of Manufacturing Engineering. Available at: https://doi.org/10.1155/2016/2021395

1. First of all in the previous section (4.8.1) it was mentioned that if α>1 then the failure rate increases over time, which coincides with our observation.
2. Secondly let's compute the mean and standard deviation of the specific distribution:

$$\mu = 3.5\,\Gamma(1+1/10) = 3.33$$

and the standard deviation is:

$$\sigma^2 = 3.5^2\left\{\Gamma(1+2/10) - [\Gamma(1+1/10)]^2 = 0.16 \rightarrow \sigma = 0.4\right\}$$

Thus it is reasonable to expect a high probability of requirement for repair in the range *3.33-0.4* ≤ *x* ≤ *3.33+0.4*. This is also evidenced in the following figure:

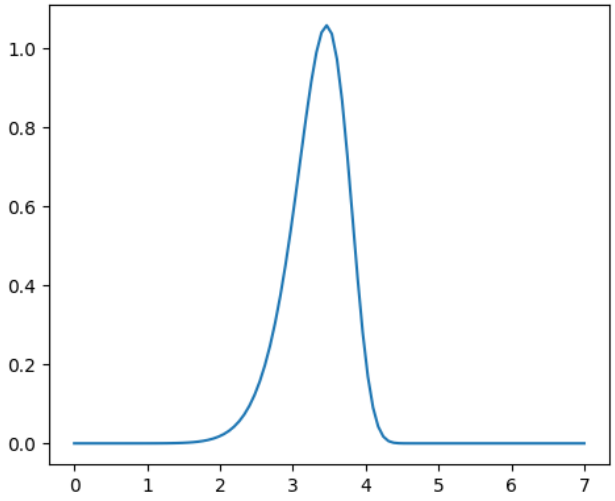


Fig 4.16: Weibull pdf α=10.0 and β=3.5

It is seen that the first 2 months the probability is very low and then the probability drastically increases between 2 to 4 months. The difference after 4 months can be considered as negligible for many practical purposes.

P (4 months) = 0.978
P (5 months) = 0.9999
P (6 months) = 1.0

4.9. Beta Distribution

Beta distribution is defined on the interval [0, 1] or (0, 1) in terms of two parameters, $\alpha>0$ and $\beta>0$ which control the shape of the distribution (Wikipedia[25], 2023). It is frequently used as a prior distribution for binomial proportions in Bayesian analysis (Forbes *et al*., 2011) and often used as a model for proportions, i.e. proportion of impurities in a chemical product or the proportion of time that a machine is under repair (Wackerly *et al*., 2008).

A random variably *Y* is said to have beta distribution with parameters *α*, *β*, *A* and *B* if the pdf is,

$$f(y;\alpha,\beta,A,B)=\frac{1}{B-A}\cdot\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\cdot\left(\frac{y-A}{B-A}\right)^{\alpha-1}\cdot\left(\frac{B-y}{B-A}\right)^{\beta-1},A\le y\le B \qquad \textbf{(4.40)}$$

If *A=0* and *B=1* then Eq. (4.40) gives *standard*[26] beta distribution:

$$f(y;\alpha,\beta)=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\cdot y^{\alpha-1}\cdot(1-y)^{\beta-1} \qquad \textbf{(4.41)}$$

Eq. (4.41) is sometimes expressed as (Wackerly *et al.*, 2008):

$$f(y;\alpha,\beta)=\frac{y^{\alpha-1}\cdot(1-y)^{\beta-1}}{\mathrm{B}(\alpha,\beta)} \qquad \textbf{(4.42)}$$

where B is:

$$\int_0^1 y^{\alpha-1}\cdot(1-y)^{\beta-1}dy=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \qquad \textbf{(4.43)}$$

25 https://en.wikipedia.org/wiki/Beta_distribution

26 **R** (https://stat.ethz.ch/R-manual/R-devel/library/stats/html/Beta.html) and **scisuit** uses standard beta distribution where parameters *shape1*, *shape2* corresponds to α and β, respectively.

Let's demonstrate the relationship between beta and gamma functions:

Script 4.20

```
#Python's built-in math library does not have the beta function
from scipy.special import beta
from math import gamma
from random import randint

a, b = randint(1,10), randint(1,10)

print(f"beta({a}, {b}) = {beta(a,b)}")
print(f"Using gamma = {gamma(a)*gamma(b)/gamma(a+b)}")
```

```
beta(9, 4) = 0.00050505
Using gamma = 0.00050505
```

4.9.1. MGF, Mean and Variance

Moment-generating function of beta distribution is fairly complex and is given by Wikipedia[27] as follows:

$$M_Y(t)=1+\sum_{k=1}^{\infty}\left(\prod_{r=0}^{k-1}\frac{\alpha+r}{\alpha+\beta+r}\right)\frac{t^k}{k!} \tag{4.44}$$

Series expansion of Eq. (4.44) can be conveniently obtained using hypergeometric function at Wolfram Alpha[28]. Let's present the first 3 terms of the series:

$$M_Y(t)=1+\frac{\alpha t}{\alpha+\beta}+\frac{\alpha(\alpha+1)t^2}{2(\alpha+\beta)(\alpha+\beta+1)}+\cdots$$

Taking the first derivative and setting $t=0$:

$$\frac{d}{dt}\left(M_Y(t=0)\right)=\frac{\alpha}{\alpha+\beta}$$

Therefore the expected value is:

27 https://en.wikipedia.org/wiki/Beta_distribution
28 https://www.wolframalpha.com/input?i=1F1%28%CE%B1%2C+%CE%B1+%2B+%CE%B2%2C+t%29+series

$$\mu = E(Y) = \frac{\alpha}{\alpha+\beta} \tag{4.45}$$

Taking the second derivative and setting $t=0$ yields $E(X^2)$:

$$E(Y^2) = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}$$

Using Eq. (2.15) and some mathematical manipulation:

$$Var(Y) = \frac{\alpha \cdot \beta}{(\alpha+\beta)^2 \cdot (\alpha+\beta+1)} \tag{4.46}$$

4.9.2. Effect of parameters

Standard beta distribution has two parameters, namely α and β. Let's simulate the effect of these two parameters on the shape of the beta curves:

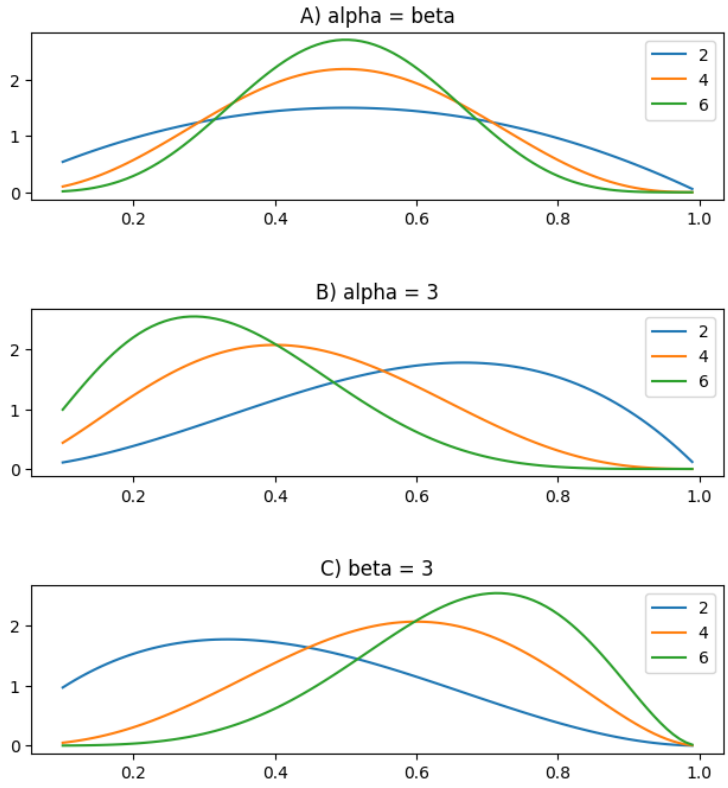


Fig 4.17: *Pdf of beta dist for different α and β values*

It is seen from **Fig. A** that when α=β, the curves are symmetric and has a peak value at $x=0.5$ (Why?).

It is also observed that larger α=β values spread less than lower α=β values since if $x=\alpha=\beta$ then Eq. (4.46) gives:

$$Var(Y; x=\alpha=\beta) = \frac{1}{4 \cdot (2x+1)}$$

In **Figs. B&C** one can observe that when $\alpha>\beta$ then the curves are left-skewed whereas when $\alpha<\beta$ then the curves are right-skewed.

Finally note that in Fig. (4.17-A) $\alpha=\beta$ values started from 2. It is recommended to try to get an insight how the curve would look like when $\alpha=\beta=1$. [Tip: use Eq. (4.41)].

Example 4.7

A wholesale distributor has storage tanks to hold fixed supplies of gasoline which are filled at the beginning of the week. The wholesaler is interested in the proportion of the supply that is sold during the week. After several weeks of data collection it is found that the proportion that is sold could be modeled by a beta distribution with α=4 and β=2. Find the probability that the wholesaler will sell *at least* 90% of her stock in a given week (Adapted from Wackerly *et al*., 2008).

Solution:

A simple script will yield the solution (remember that *shape1*= α and *shape2*= β).

Script 4.21

```
from scisuit.stats import pbeta
prob = pbeta(q=0.9, shape1=4, shape2=2)
print(f"P(Y>0.9) = {1 - prob}")
```

```
P(Y>0.9) = 0.081
```

Imagine also the wholesaler is interested in up to what proportions of gasoline could be sold 50, 75 and 95% of the time?

Script 4.22

```
from scisuit.stats import qbeta
probs = [0.5, 0.75, 0.95]
for p in probs:
    print(f"{p*100}%: {qbeta(p=p, shape1=4, shape2=2)}")
```

```
50%: 0.68, 75%: 0.81, 95%: 0.92
```

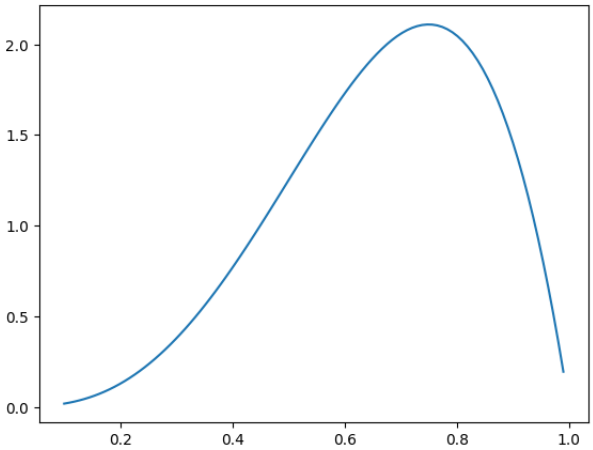


Fig 4.18: *pdf of beta dist for α=4 and β=2*

It is seen that the curve is left-skewed (Why?). This might be considered as “lucky” for the wholesaler as lower probabilities is associated with lower gasoline sale proportions.

4.10. Summary

Name	Description	Equation
Uniform	There is *almost* same amount of numbers in each equally spaced sub-interval	$f(y)=\begin{cases}\frac{1}{b-a} & a\le y\le b\\ 0 & elsewhere\end{cases}$
Normal	Poisson limit approximated binomial probabilities when $n\to\infty$ and $p\to 0$. Abraham DeMoivre showed that when X is a binomial random variable and n is large the probability $P\left(a\le\frac{X-np}{\sqrt{np(1-p)}}\le b\right)$ can be estimated	$f_Z(z)=\frac{1}{\sqrt{2\pi}}e^{\frac{-z^2}{2}},-\infty<z<\infty$
Exponential	Poisson model is a good candidate when we only know the rate of occurrence (λ) of an event where the events occur completely at random. However, situations might arise where the time interval between consecutively occurring event is an important random variable.	$f_Y(y)=\lambda e^{-\lambda y},y>0$
Gamma	Events satisfying the Poisson process occurring at a rate of λ could be modeled with exponential distribution. Here the random variable Y can also be interpreted as the *waiting time* for the *first* occurrence. gamma distribution generalizes the exponential distribution such that we are now interested in the occurrence of (waiting time of) r^{th} event.	$f=\frac{1}{\beta^\alpha\Gamma(\alpha)}x^{\alpha-1}e^{-x/\beta},x>0$
Chi-square	The chi-squared distribution with k degrees of freedom is the distribution of a sum of the squares of k independent standard normal random variables.	$f_Y(y)=\frac{1}{2^{n/2}\Gamma(n/2)}y^{(n/2)-1}e^{-y/2}$
Student t	If $y_1, y_2, \ldots, y_n$ is a random sample from a normal distribution with mean μ and standard deviation ρ then $\frac{\bar{Y}-\mu}{\rho/n}$ has a standard normal distribution (SND). However Gosset realized that $\frac{\bar{Y}-\mu}{s/n}$ does not have a SND and derived the probability density function.	$T_n=\frac{Z}{\sqrt{V/n}}$

F	Ratio of independent chi-square random variables.	$F = \frac{U/m}{V/n}$
Weibull	Commonly used as a lifetime distribution in reliability applications. It is of great interest to statisticians and to practitioners because of its ability to fit to data from various fields.	$f = \begin{cases} \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} e^{-(x/\beta)^{\alpha}} & x \geq 0 \\ 0 & x < 0 \end{cases}$
Beta	Defined on the interval [0, 1] or (0, 1) in terms of two parameters, $\alpha>0$ and $\beta>0$ which control the shape of the distribution. It is frequently used as a prior distribution for binomial proportions in Bayesian analysis and often used as a model for proportions, i.e. proportion of impurities in a chemical product.	$f = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \cdot y^{\alpha-1} \cdot (1-y)^{\beta-1}$

5. Estimation and Hypothesis Testing

5.1. Point Estimation

A point estimate is a single value (i.e., mean, median, proportion, ...) based on sampled data to represent a plausible value of a population characteristics (Peck *et al*. 2016).

Example 5.1

Out of 7421 US College students 2998 reported using internet more than 3 hours a day. What is the proportion of *all* US College students who use internet more than 3 hours a day? (Adapted from Peck et al. 2016).

Solution:

The solution is straightforward: $p = \frac{2998}{7421} = 0.40$

Based on the statistics it is possible to claim that approximately 40% of the students in US spend more than 3 hours a day using the internet. Please note that based on the survey result, we made a claim about the population, students in US. ■

Now that we made an estimate based on the survey, we should ask ourselves: "*How reliable is this estimate?*". We know that if we had another group of students, the percentage might not have been 40, maybe it would be 45 or 35. There are no perfect estimators but we expect that *on average* the estimator should gives us the right answer.

5.1.1. Unbiased estimators

Definition: A statistic Θ is an unbiased estimator of the parameter θ of a given distribution if and only if,

$$E(\Theta) = \theta \qquad (5.1)$$

for all possible values of θ (Miller and Miller 2014).

Example 5.2

If X has binomial distribution with the parameters *n* and *p*, show that the sample proportion, *X/n* is an unbiased estimator of *p*.

Before we proceed with the solution, let's refresh ourselves with a simple example: Suppose we conduct an experiment where we flip a coin 10 times. We already know that the probability of getting heads (success) is *p*=0.5. However, we want to estimate *p* by flipping the coin and calculating the sample proportion, *X/n*. If we flip the coin 10 times and get *X=6* heads, *p=0.6*. However, after many experiments *p* will be found as 0.5. Therefore, *X/n* is an unbiased estimator.

$$E\left(\frac{X}{n}\right)=\frac{1}{n}E(X)=\frac{1}{n}\cdot np=p$$

therefore, *X/n* is an unbiased estimator of *p*. ■

Example 5.3

Prove that $E\left[\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2\right]$ is unbiased estimator of population variance (σ^2).

Solution:

$$E(S^2)=E\left[\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})^2\right]$$

Now we are going to add and subtract μ inside the parenthesis:

$$=\frac{1}{n-1}E\left[\sum_{i=1}^{n}\left((X_i-\mu)-(\bar{X}-\mu)\right)^2\right]$$

After a straightforward algebraic manipulation,

$$=\frac{1}{n-1}E\left[\sum_{i=1}^{n}\left((X_i-\mu)^2-(\bar{X}-\mu)^2\right)\right]$$

$$=\frac{1}{n-1}\left[\sum_{i=1}^{n}E(X_i-\mu)^2-n(\bar{X}-\mu)^2\right]$$

Note that $E(X_i-\mu)^2=\sigma^2$ and $E(\bar{X}-\mu)^2=\frac{\sigma^2}{n}$. Putting the knowns in the last equation,

$$=\frac{1}{n-1}\left(n\cdot\sigma^2-n\cdot\frac{\sigma^2}{n}\right)=\sigma^2$$

Therefore, $E(S^2)$ is an unbiased estimator of population variance. ■

Example 5.4

For the uniform probability distribution $f_Y(y;\theta)=\frac{1}{\theta}$ there are two estimates for θ:

1. $\hat{\theta}_1=\frac{2}{n}\sum_{i=1}^{n}Y_i$
2. $\hat{\theta}_2=Y_{max}$

Which one is an unbiased estimator of θ? (Adapted from Larsen & Marx 2011)

Solution #1:

$$E(\hat{\theta}_1)=E\left(\frac{2}{n}\sum_{i=1}^{n}Y_i\right)=\frac{2}{n}\sum_{i=1}^{n}E(Y_i)$$

expected value of uniform distribution is *θ/2*, therefore

$$E(\hat{\theta}_1)=\frac{2}{n}\sum_{i=1}^{n}\frac{\theta}{2}=\frac{2}{n}\cdot n\cdot\frac{\theta}{2}=\theta$$

Therefore is an unbiased estimator.

Solution #2:

Using the equation given in Example (2.10), the PDF can be found as follows:

$$f_{Y_{max}}(u)=n\cdot\frac{1}{\theta}\cdot\left(\frac{u}{\theta}\right)^{n-1}$$

$$E(\hat{\theta}_2)=\int_0^{\theta}u\cdot n\cdot\frac{1}{\theta}\cdot\left(\frac{u}{\theta}\right)^{n-1}du=\frac{n}{n+1}\theta$$

It is seen that as *n* increases, the “bias” decreases and for large *n* it becomes unbiased.

5.1.2. Efficiency

It is seen in section (5.1.1) that a parameter can have more than one unbiased estimators. Which one should we choose? We should choose one with higher precision, in other words, with smaller variance.

Definition: Let $\hat{\theta}_1$ and θ_2 be two unbiased estimators for parameter θ. If,

$$Var(\hat{\theta}_1) < Var(\hat{\theta}_2) \tag{5.2}$$

$\hat{\theta}_1$ is more efficient than θ_2 (Larsen & Marx 2011).

Example 5.5

Given the estimators in Example (5.4), which one is more efficient?

Solution:

A tedious mathematical derivation is presented by Larsen & Marx (2011). The results are as follows:

$$Var(\hat{\theta}_1) = \frac{\theta^2}{3n}$$

$$Var(\hat{\theta}_2) = \frac{\theta^2}{n(n+2)}$$

For $n>1$, it is seen that second estimator has a smaller variance than the first one. Therefore, it is more efficient. ■

There are more properties of estimators: **i)** minimum variance, **ii)** robustness, **iii)** consistency, **iv)** sufficiency. Interested readers can refer to textbooks on mathematical statistics (Devore *et al*. 2021; Larsen & Marx 2011; Miller & Miller 2014).

5.2. Statistical Confidence

Suppose you want to estimate the SAT scores of students. For that purpose, a randomly selected 500 students have been given an SAT test and a mean value of 461 is obtained (adapted from Moore *et al.* 2009). Although it is known that the sample mean is an *unbiased* estimator of the population mean (μ), we already know that had we sampled another 500 students, the mean could (most likely would) have been different than 461. Therefore, how confident are we to claim that the population mean will be 461.

Suppose that the standard deviation of the population is known (σ=100). We know that if we repeat sampling 500 samples, the mean of these samples will follow the N(μ, $\frac{100}{\sqrt{500}}=4.5$) curve.

Script 5.1

```
import scisuit.plot as plt
from scisuit.stats import rnorm
aver = []
for i in range(1000):
    sample = rnorm(n=500, mean= 461, sd= 100)
    aver.append(sum(sample)/500)

plt.hist(data=aver, density=True)
plt.show()
```

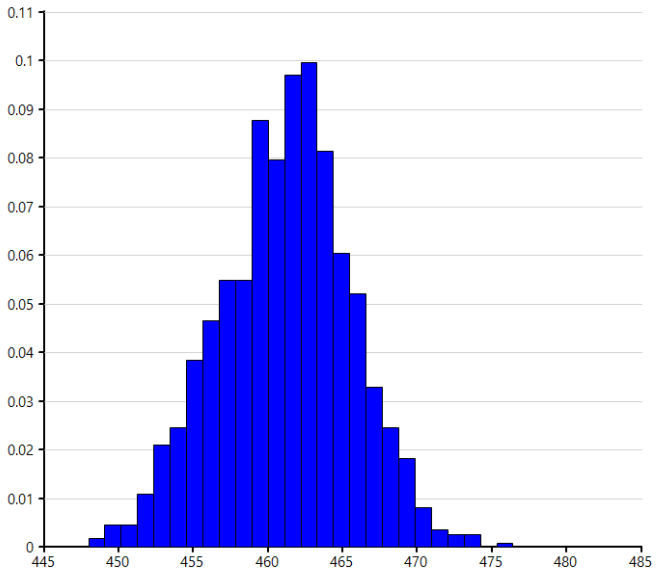


Fig 5.1: *Density scaled histogram*

It is seen that the interval (447.5, 474.5) represents almost all possible mean values. Therefore we are 99.7% (3σ) confident (confidence level) that the population mean will be in this interval. Note also that, as a natural consequence our confidence level decreases as the interval length decreases.

$461-3\times4.5=447.5$

$461+3\times4.5=474.5$

5.3. Confidence Intervals

A way to quantify the amount of uncertainty in a point estimator is to construct a confidence interval (Larsen & Marx 2011). The definition of confidence interval is as follows: "... *an interval computed from sample data by a method that has probability C of producing an interval containing the true value of the parameter.*" (Moore *et al*. 2009). Peck *et al*. (2016) gives a general form of confidence interval as follows:

$$\begin{pmatrix}\text{point estimate using a}\\ \text{specified statistic}\end{pmatrix} \pm (\text{critical value}) \cdot \begin{pmatrix}\text{estimated standard deviation}\\ \text{of the statistic}\end{pmatrix} \tag{5.3}$$

Note that the estimated standard deviation of the statistic is also known as *standard error.* In other words, when the standard deviation of a statistic is estimated from the data (because the population's standard deviation is not known), the result is called the standard error of the statistic (Moore *et al*. 2009).

Example 5.6

Establish a confidence interval for binomial distribution.

Solution:

We already know (chapter 4.2) that Abraham DeMoivre showed that when X is a binomial random variable and n is large the probability can be approximated as follows:

$$\lim_{n\to\infty} P\left(a \le \frac{X-np}{\sqrt{np(1-p)}} \le b\right) = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-z^2/2}\, dz$$

To establish an approximate 100(1−α)% confidence interval,

$$P\left[-z_{\alpha/2} \le \frac{X-np}{\sqrt{np(1-p)}} \le z_{\alpha/2}\right] = 1-\alpha$$

Rewriting the equation,

$$P\left[-z_{\alpha/2} \le \frac{X/n-p}{\sqrt{\frac{(X/n)(1-X/n)}{n}}} \le z_{\alpha/2}\right] = 1-\alpha$$

Rewriting the equation by isolating *p* leads to,

$$\left[\frac{k}{n}-z_{\alpha/2}\sqrt{\frac{(k/n)(1-k/n)}{n}},\frac{k}{n}+z_{\alpha/2}\sqrt{\frac{(k/n)(1-k/n)}{n}},\right]$$

■

If a 95% confidence interval to be established, then $z_{\alpha/2}$ would be ≈1.96.

Script 5.2

```
alpha1 = 0.05
alpha2 = 0.01
print(qnorm(alpha1/2), qnorm(1-alpha1/2))
print(qnorm(alpha2/2), qnorm(1-alpha2/2))
```

```
-1.95996        1.95996
-2.57583        2.57583
```

Note that if a 95% confidence interval (CI) yields an interval (0.52, 0.57), it is *tempting* to say that there is a probability of 0.95 that *p* will be in between 0.52 and 0.57. Larsen & Marx (2011) and Peck *et al*. (2016) warns against this *temptation*. A close look at Eq. (5.3) reveals that from sample to sample the constructed CI will be different. However, *in the long* run 95% of the constructed CIs will contain the true *p* and 5% will not. This is well depicted in the figure (Figure 9.4 at pp. 471) presented by Peck *et al*. (2016).

Note also that a 99% CI will be wider than a 95% CI. However, the higher reliability causes a loss in precision. Therefore, Peck *et al*. (2016) remarks that many investigators consider a 95% CI as a reasonable compromise between reliability and precision.

5.4. Hypothesis Testing

Confidence intervals and statistical tests are the two most important ideas in the age of modern statistics (Kreyszig *et al*. 2011). The confidence interval is carried out when we would like to estimate population parameter. Another type of inference is to assess the evidence provided by data against a claim about a parameter of the population (Moore *et a*l. 2009). Therefore, after carrying out an experiment conclusions must be drawn based on the obtained data. The two competing propositions are called the *null hypothesis* (H_0) and the *alternative hypothesis* (H_1) (Larsen & Marx 2011).

We initially assume that a particular claim about a population (H_0) is correct. Then based on the evidence from data we either reject **H_0** and accept **H_1** if there is *compelling* evidence or accept **H_0** in favor of **H_1** (Peck *et al*. 2016).

An example from Larsen & Marx (2011) would clarify the concepts better: Imagine as an automobile company you are looking for additives to increase gas mileage. Without the additives, the cars are known to average 25.0 mpg with a σ=2.4 mpg and with the addition of additives, it was found (experiment involved 30 cars) that the mileage increased to 26.3 mpg.

Now, in terms of null and alternative hypothesis, **H_0** is 25 mpg and **H_1** claims 26.3 mpg. We know that if the experiments were carried out with another 30 cars, the result would be different (lower or higher) than 26.3 mpg. Therefore, *"is an increase to 26.3 mpg due to additives or not?"*. At this point we should rephrase our question: *"if we sample 30 cars from a population with μ=25.0 mpg and σ=2.4, what are the chances that we will get 26.3 mpg on average?"*. If the chances are high, then the additive is not working; however, if the chances are low, then it must be due to the additives that the cars are getting 26.3 mpg. Let's evaluate this with a script (note the similarity to Script 5.1):

Script 5.3

```
aver = []
for i in range(10000):
    sample = rnorm(n=30, mean= 25, sd= 2.4)
    aver.append(sum(sample)/30)

filtered = list(filter(lambda x: x>=26.5, aver))
print(f"probability = {len(filtered)/len(aver)}")
```

```
probability = 0.0002
```

We observe that the probability is too low for this to happen by chance (random sampling from the population). Therefore, we conclude that in light of the statistical evidence the additives indeed work (H_1 wins) and reject H_0.

Directly computing the probability:

$$P\left(\frac{26.50-25.0}{2.4/\sqrt{30}}\right)=0.0003$$

which is very close to the simulation result of Script (5.3).

Wackerly *et al*. (2008) lists the elements of a statistical test as follows:

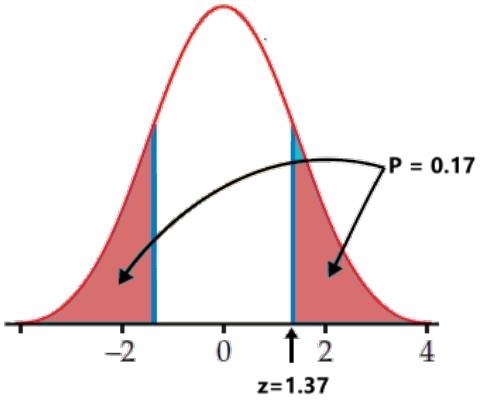
1. Null hypothesis (μ=25.0 mpg),
2. Alternative hypothesis (26.5 mpg),
3. Test statistic ($\frac{\bar{x}-25.0}{2.4/\sqrt{30}}$),
4. Rejection region ($\bar{x}\geq 25.718$ for $\alpha=0.05$)

■

5.4.1. The P-Value

We have seen that using a level of significance a critical region (H_0 being rejected) can be identified (Larsen & Marx 2011); for example, $z\geq 2$ is a rejection criteria for the Supreme Court of the United States (Moore et al. 2009). However, not all test statistics are normal and therefore a new strategy is to calculate the p-value, which is defined as (Larsen & Marx 2011): "… *the probability of getting a value for that test statistic as extreme as or more extreme than what was actually observed given that H_0 is true.*". The term *extreme* is also used by Moore *et al*. (2009) and is explained as "*far from what we would expect if H_0 were true*".

If for example the test statistics yield $Z=1.37$ and we are carrying out a two-sided test, the p-value would be, $P(Z\leq -1.37 \text{ or } Z\geq 1.37)$ where Z has a standard normal distribution.




```
z=1.37

#pnorm computes left-tailed probability
pvalue = pnorm(-z) + (1-pnorm(z))

print(f"p-value = {pvalue}")
```

```
p-value = 0.17
```

Fig 5.2: *P-value (i.e. z≥1.37 is considered extreme) (adapted from Moore et al. 2009)*

A simpler definition is given by Miller & Miller (2014): "*… the lowest level of significance at which the null hypothesis could have been rejected*". Let's rephrase Miller & Miller (2014) definition: once a level of significance is decided (e.g. α=0.05), if the computed *p-value* is less than the α, then we reject H_0. For example, in the gasoline additive example, p-value was computed as 0.0003 and if α=0.05, then since $p < \alpha$, we reject H_0 in favor of H_1 (i.e., additive has effect).

In terms of a standard normal distribution, there are 3 cases of computing p-values:

1. $H_1: \mu > \mu_0 \rightarrow P(Z \geq z)$ (alternative is greater than)
2. $H_1: \mu < \mu_0 \rightarrow P(Z \leq z)$ (alternative is smaller than)
3. $H_1: \mu \neq \mu_0 \rightarrow 2P(Z \geq |z|)$ (alternative is not equal)

Example 5.7

A bakery claims on its packages that its cookies are 8 g. It is known that the standard deviation of the 8 g packages of cookies is 0.16 g. As a quality control engineer, you collected 25 packages and found that the average is 8.091 g. Is the production process going alright? (adapted from Miller & Miller 2014).

Solution:

The null hypothesis is $H_0: \mu=8$ g,

The alternative hypothesis $H_1: \mu \neq 8$ g.

The test statistic: $z = \frac{8.091-8}{0.16/\sqrt{25}} = 2.84$

```
1-pnorm(2.84) + pnorm(-2.84) #2*(1- pnorm(2.84))
```

```
0.0045
```

Since $p<0.05$, we reject the null hypothesis. Therefore, the process should be checked and suitable adjustments should be made.

6. Z-Test for Population Means

The fundamental limitation to applying z-test is that the population variance must be known in advance (Kanji 2006; Moore *et al*. 2009; Peck *et al*. 2016). The test is accurate when the population is normally distributed; however, it will give an approximate value even if the population is not normally distributed (Kanji 2006). In most practical applications, population variance is unknown and the sample size is small therefore a *t-test* is more commonly used.

6.1. One-sample z-test

From a population with known mean (μ) and variance (σ), a random sample of size *n* is taken (generally n≥30) and the sample mean ($\bar{x}$) calculated. The test statistic:

$$Z = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}} \qquad \textbf{(6.1)}$$

Example 6.1

A filling process is set to fill tubs with powder of 4 g on average. For this filling process it is known that the standard deviation is 1 g. An inspector takes a random sample of 9 tubs and obtains the following data: *Weights = [3.8, 5.4, 4.4, 5.9, 4.5, 4.8, 4.3, 3.8, 4.5]*

Is the filling process working fine? (Adapted from Kanji 2006).

Solution:

The average of 9 samples is: 4.6 g

Test statistic: $Z = \frac{4.6 - 4}{1/\sqrt{9}} = 1.8,$

Since 1.8 is in the range of -1.96<Z<1.96, we cannot reject the null hypothesis, therefore the filling process works fine (i.e. there is no evidence to suggest it is different than 4 g).

Is it over-filling?

Now, we are going to carry out 1-tailed z-test and therefore acceptance region is Z<1.645. Since the test statistic is greater than 1.645, we reject the null hypothesis and have evidence that the filling process is over-filling.

Script 6.1

```
import scisuit.plot as plt
from scisuit.stats import test_z

data = [3.8, 5.4, 4.4, 5.9, 4.5, 4.8, 4.3, 3.8, 4.5]
result = test_z(x=data, sd1=1, mu=4)
print(result)
```

```
N=9, mean=4.6, Z=1.799
p-value = 0.072 (two.sided)
Confidence interval (3.95, 5.25)
```

Since $p>0.05$, we cannot reject H_0.

Script 6.1 requires minor change to analyze whether it is over-filling or not. We will set the parameter, namely *alternative*, to "greater" whose default value was set to "two.sided".

Script 6.2

```
result = test_z(x=data, sd1=1, mu=4, alternative="greater")
print(result)
```

```
p-value = 0.036 (greater)
Confidence interval (4.052, inf)
```

Since $p<0.05$, we reject the null hypothesis in favor of alternative hypothesis.

6.2. Two-sample z-test

In essence, two-sample is very similar to one-sample z-test such that we take n_1 and n_2 samples from two populations with means (μ_1 and μ_2) and variances (σ_1 and σ_2). Therefore, the test statistic is computed as:

$$Z=\frac{\left(\bar{x}_1-\bar{x}_2\right)-\left(\mu_1-\mu_2\right)}{\left(\frac{\sigma_1^2}{n_1}+\frac{\sigma_2^2}{n_2}\right)^{\frac{1}{2}}} \qquad \textbf{(6.2)}$$

Example 6.2

A survey has been conducted to see if studying over or under 10 h/week has an effect on overall GPA. For those who studied less (x) and more (y) than 10 h/week the GPAs were:

x=[2.80, 3.40, 4.00, 3.60, 2.00, 3.00, 3.47, 2.80, 2.60, 2.0]

y = [3.00, 3.00, 2.20, 2.40, 4.00, 2.96, 3.41, 3.27, 3.80, 3.10, 2.50].

respectively. It is known that the standard deviation of GPAs for the whole campus is σ=0.6. Does studying over or under 10 h/week has an effect on GPA? (Adapted from Devore et al. 2021)

Solution:

We have two groups (those studying over and under 10 h/week) from the same population (whole campus) whose standard deviation is known (σ=0.6).

We will solve this question directly using a Python script and the mathematical computations are left as an exercise to the reader.

Script 6.3

```
x = [2.80, 3.40, 4.00, 3.60, 2.00, 3.00, 3.47, 2.80, 2.60, 2.0]
y = [3.00, 3.00, 2.20, 2.40, 4.00, 2.96, 3.41, 3.27, 3.80, 3.10, 2.50]
mu = 0
sd1, sd2 = 0.6, 0.6

result = test_z(x=x, y=y, sd1=sd1, sd2=sd2, mu=0)
print(result)
```

```
n1=10, n2=11, mean1=2.967, mean2=3.058
Z=-0.3478
p-value = 0.728 (two.sided)
Confidence interval (-0.605, 0.423)
```

Since $p>0.05$ there is no statistical evidence to reject H_0 ($\mu_1 - \mu_2 = 0$) and therefore there is no statistically significant different between studying over or under 10 h/week.

7. Student's t-test for Population Means

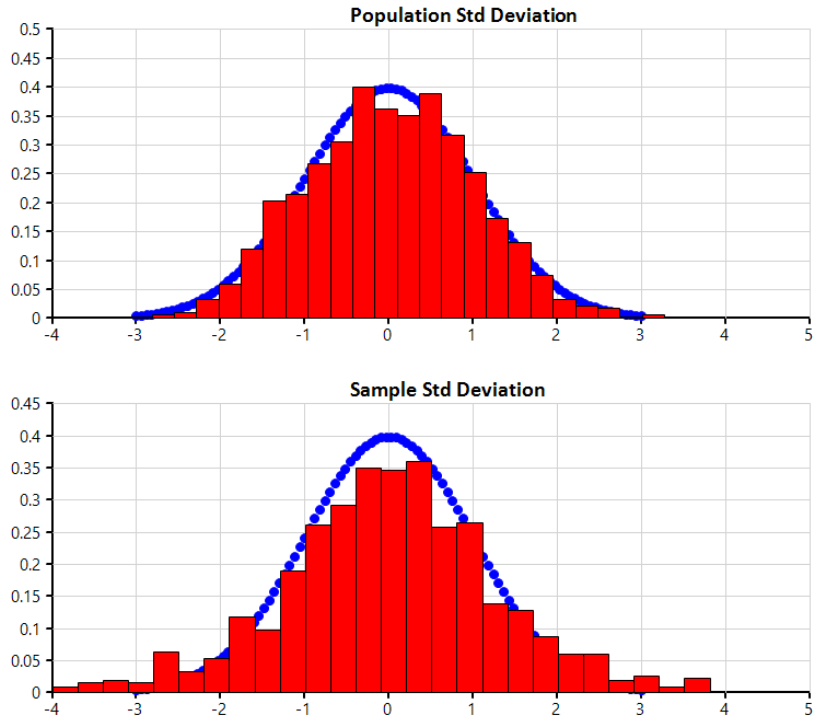
In Chapter 6, we have seen that *z-test* is only possible when standard deviation (σ) of the population is known. Therefore, the z-statistic is not commonly used (Peck *et al*. 2016). When standard deviation(s) are not known they must be estimated from samples and in Example (2.3) it was already shown that S^2 is an unbiased estimator of σ^2. Therefore, one might immediately be *tempted* to use Eq. (6.1).

Then, comes the question: *What effect does replacing σ with S have on Z ratio*? (Larsen & Marx 2011). In order to answer this question, let's demonstrate the effect of replacing σ with S on Z ratio with a script:

Script 7.1

```
import numpy as np
import scisuit.plot as plt
from scisuit.stats import dnorm, rnorm

#plotting f(z) curve
x = np.linspace(-3, 3, num=100)
y = dnorm(x)

N = 4
sigma, mu = 1.0,  0.0 #stdev and mean of population
z, t = [], []
for i in range(1000):
   sample = rnorm(n=N)
   aver = sum(sample)/N

   #using population stdev
   z_ratio = (aver-mu)/(sigma/sqrt(N))
   z.append(z_ratio)

   #computing stdev from sample
   s = float(np.std(sample, ddof=1))
   z_ratio = (aver-mu)/(s/sqrt(N))

   #filter out too big and too small ones
   if(-4<z_ratio<4):
      t.append(z_ratio)

plt.layout(nrows=2, ncols=1)

plt.subplot(row=0, col=0)
plt.scatter(x=x, y=y)
```

```
plt.hist(data=z, density=True)
plt.title("Population Std Deviation")

plt.subplot(row=1, col=0)
plt.scatter(x=x, y=y)
plt.hist(data=t, density=True)
plt.title("Sample Std Deviation")

plt.show()
```



In the top figure, it is seen that when the standard deviation of the population (σ) is known *f(z)* is consistent with $\frac{\bar{x}-\mu}{\sigma/\sqrt{n}}$.

However, when σ is not known and instead *S* is used to compute z-ratio, $\frac{\bar{x}-\mu}{S/\sqrt{n}}$, it is seen that *f(z)* underestimates the ratios much less than zero as well as the ratios much larger than zero.

Credit for recognizing this difference goes to *William Sealy Gossett*.[29]

Fig 7.1: $\frac{\bar{x}-\mu}{\sigma/\sqrt{n}}$ *(top) vs* $\frac{\bar{x}-\mu}{S/\sqrt{n}}$

Note that in Script (7.1), *N* was intentionally chosen a small value (*N=4*). It is recommended to change *N* to a greater number, such as 10, 20 or 50 in order to observe the effect of large samples.

29 **Student** (1908). The probable error of a mean. Biometrika, 6(1), 1-25.

7.1. One-sample t-test

Let $\bar{x}$ and s be the mean and standard deviation of a random sample from a normally distributed population. Then,

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}} \tag{7.1}$$

has a *t* distribution with *df=n-1*. Here s is the sample's standard deviation and computed as:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2 \tag{7.2}$$

Example 7.1

In 2006, a report revealed that UK subscribers with 3G phones listen on average 8.3 hours/month full-track music. The data for a random sample of size 8 for US subscribers is x=[5, 6, 0, 4, 11, 9, 2, 3]. Is there a difference between US and UK subscribers? (Adapted from Moore et al. 2009).

Solution:

Script 7.2

```
from statistics import stdev
from scisuit.stats import qt

x=[5, 6, 0, 4, 11, 9, 2, 3]
n = len(x)
df = n-1 #degrees of freedom
aver = sum(x)/n
stderr = stdev(x)/sqrt(n) #standard error

#construct a 95% interval
tval = qt(0.025, df=df) #alpha/2=0.025
v_1 = aver - tval*stderr
v_2 = aver + tval*stderr
print(f"Interval: ({min(v_1, v_2)}, {max(v_1, v_2)})")
```

```
Interval: (1.97, 8.03)
```

Since the confidence interval does not contain 8.3 and furthermore since its upper limit is smaller than 8.3, it can be concluded that US subscribers listen less than UK subscribers.

Directly solving using *scisuit*'s built-in function:

Script 7.3

```
from scisuit.stats import test_t
x=[5, 6, 0, 4, 11, 9, 2, 3]
result = test_t(x=x, mu=8.3)
print(result)
```

```
One-sample t-test for two.sided
N=8, mean=5.0
SE=1.282, t=-2.575
p-value =0.037
Confidence interval: (1.97, 8.03)
```

Since $p<0.05$ we reject H_0 and claim that there is statistically significant difference between US and UK subscribers. [If in `test_t` function H_1 was set to "less" instead of "two.sided" then $p=0.018$. Therefore, we would reject the H_0 in favor of H_1, i.e. US subscribers indeed listen less than UK's.]

7.2. Two-sample t-test

7.2.1. Equal Variances

Assume we are drawing n and m samples from two populations, namely X and Y, with equal variances, s^2, but with different means μ_1 *and* μ_2. Let S_P^2 be the pooled variance, then:

$$S_P^2=\frac{\sum_{i=1}^{n}(X_i-\bar{X})^2+\sum_{i=1}^{n}(Y_i-\bar{Y})^2}{n+m-2} \tag{7.3}$$

and test statistic is defined as:

$$T_{n+m-2}=\frac{\bar{X}-\bar{Y}-(\mu_X-\mu_Y)}{S_p\sqrt{\frac{1}{n}+\frac{1}{m}}} \tag{7.4}$$

has a Student's t-distribution with $n+m-2$ degrees of freedom.

Example 7.2

Student surveys are important in academia. An academic who scored low on a student survey joined workshops to improve "enthusiasm" in teaching. X and Y are survey scores from his fall and spring semester classes which he selected to *have the same demographics*.

X= [3, 1, 2, 1, 3, 2, 4, 2, 1]

Y = [5, 4, 3, 4, 5, 4, 4, 5, 4]

Is there a difference in scores of both semester? (Adapted from Larsen & Marx 2011).

Solution:

We can make the following assumptions:

1. The variance of the populations are not known, therefore z-test cannot be applied.
2. It is reasonable to assume equal variances since the X and Y have the same demographics.

Script 7.4

```
from scisuit.stats import test_t
x = [3, 1, 2, 1, 3, 2, 4, 2, 1]
y = [5, 4, 3, 4, 5, 4, 4, 5, 4]
result = test_t(x=x, y=y, varequal=True)
print(result)
```

```
Two-sample t-test assuming equal variances
n1=9, n2=9, df=16
s1=1.054, s2=0.667
Pooled std = 0.882
t = -5.07
p-value = 0.0001 (two.sided)
Confidence interval: (-2.992, -1.230)
```

Since $p<0.05$, the difference between the scores of fall and spring are statistically significant.

7.2.2. Unequal Variances

Similar to section 7.2.1, we are drawing random samples of size n_1 and n_2 from normal distributions with means μ_X and μ_Y, but with standard deviations σ_X and σ_Y, respectively.

$$S_1^2=\frac{\sum_{i=1}^{n}(X_i-\bar{X})^2}{n_1-1} \text{ and } S_2^2=\frac{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}{n_2-1} \quad \textbf{(7.5)}$$

The test statistic is computed as follows:

$$t=\frac{\bar{X}-\bar{Y}-(\mu_X-\mu_Y)}{\sqrt{\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}}} \quad \textbf{(7.6)}$$

In 1938 Welch[30] showed that t is approximately distributed as a Student's t random variable with df:

$$df=\frac{\left(\frac{s_1^2}{n_1}+\frac{s_2^2}{n_2}\right)^2}{\frac{s_1^4}{n_1^2(n_1-1)}+\frac{s_2^4}{n_2^2(n_2-1)}} \quad \textbf{(7.7)}$$

Example 7.3

A study by Larson and Morris[31] (2008) surveyed the annual salary of men and women working as purchasing managers subscribed to *Purchasing* magazine. The salaries are (in thousands of US dollars):

Men = [81, 69, 81, 76, 76, 74, 69, 76, 79, 65]

Women = [78, 60, 67, 61, 62, 73, 71, 58, 68, 48]

Is there a difference in salaries between men and women? (Adapted from Peck et al. 2016)

30 https://www.jstor.org/stable/2332010

31 **Larson PD & Morris M** (2008). Sex and Salary: A Survey of Purchasing and Supply Professionals, Journal of Purchasing and Supply Management, 112–124.

Solution:

Following assumption can be made:

1. Z-test cannot be applied because the variance of the populations are not known.
2. Although the samples were selected from the subscribers of *Purchasing* magazine, Larson and Morris (2008) considered two populations of interest, i.e. male and female purchasing managers. Therefore, equal variances should not be applied.

Script 7.5

```
from scisuit.stats import test_t
Men = [81, 69, 81, 76, 76, 74, 69, 76, 79, 65]
Women = [78, 60, 67, 61, 62, 73, 71, 58, 68, 48]
result = test_t(x=Women, y=Men, varequal=False)
print(result)
```

```
Two-sample t-test assuming unequal variances
n1=10, n2=10, df=15
s1=8.617, s2=5.399
t = -3.11
p-value = 0.007 (two.sided)
Confidence interval: (-16.7, -3.1)
```

Since p<0.05, there is statistically significant difference between salaries of each group.

7.3. Paired t-test

In essence a paired t-test is a two-sample t-test as there are two samples. However, the two samples are *not independent* as one of the factors in the first sample is paired in a meaningful way with a particular observation in the second sample (Larsen & Marx 2011; Peck et al. 2016).

The equation to compute the test statistics is similar to one-sample t-test, Eq. (7.1):

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}} \tag{7.8}$$

where $\bar{x}$ and s are mean and standard deviation of the sample differences, respectively. The degrees of freedom is: *df=n-1*.

Example 7.4

In a study where 6th grade students who had not previously played chess participated in a program in which they took chess lessons and played chess daily for 9 months. Below data demonstrates their memory test score before and after taking the lessons:

Pre = [510, 610, 640, 675, 600, 550, 610, 625, 450, 720, 575, 675]

Post = [850, 790, 850, 775, 700, 775, 700, 850, 690, 775, 540, 680]

Is there evidence that playing chess increases the memory scores? (Adapted from Peck *et al*. 2016).

Solution:

Before we attempt to solve the question, we make the following assumptions:

1. Z-test cannot be applied since population variance is not known,
2. Pre- and post-test scores are not independent since they were applied to the same subjects.

Script 7.6

```
from scisuit.stats import test_t
Pre = [510, 610, 640, 675, 600, 550, 610, 625, 450, 720, 575, 675]
Post = [850, 790, 850, 775, 700, 775, 700, 850, 690, 775, 540, 680]
result = test_t(x=Post, y=Pre, paired=True)
print(result)
```

```
Paired t-test for two.sided
N=12, mean1=747.9, mean2=603.3, mean diff=144.6
t =4.564
p-value =0.0008
Confidence interval: (74.9, 214.3)
```

Since $p<0.05$, there is statistical evidence that playing chess indeed made a difference in increasing the memory scores.

If the parameter, namely *alternative*, was set to "less", then $p=0.99$. Therefore, we would reject the alternative hypothesis (*Post*<*Pre*). However, on the other hand, alternative was set to "greater" then $p=0.0004$, therefore we would reject the H_0 and accept H_1 (*Post*>*Pre*).

8. F-Test for Population Variances

Assume that a metal rod production facility uses two machines on the production line. Each machine produces rods with thicknesses μ_X and μ_Y which are not significantly different. However, if the variabilities are significantly different, then some of the produced rods might become unacceptable as they will be outside the engineering specifications.

In Section (7.2), it was shown that there are two cases for two-sample *t-tests*: whether variances were equal or not. To be able to choose the right procedure, Larsen & Marx (2011) recommended that *F test* should be used prior to testing for $\mu_X=\mu_Y$.

Let's draw random samples from populations with normal distribution. Let $X_1, \dots, X_m$ be a random sample from a population with standard deviation σ_1 and let $Y_1, \dots, Y_n$ be another random sample from a population with standard deviation σ_2. Let S_1 and S_2 be the sample standard deviations. Then the test statistic is:

$$F=\frac{S_1^2/\sigma_1}{S_2^2/\sigma_2} \tag{8.1}$$

has an F distribution with $df_1=m-1$ and $df_2=n-1$, (Devore et al. 2021).

Example 8.1

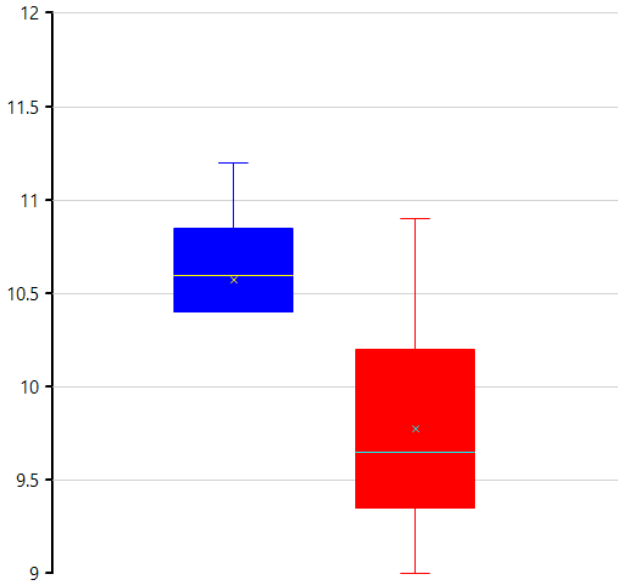
α-waves produced by brain have a characteristic frequency from 8 to 13 Hz. The subjects were 20 inmates in a Canadian prison who were randomly split into two groups: one group was placed in solitary confinement; the other group was allowed to remain in their own cells. Seven days later, α-wave frequencies were measured for all twenty subjects are shown below:

non-confined = [10.7, 10.7, 10.4, 10.9, 10.5, 10.3, 9.6, 11.1, 11.2, 10.4]
confined = [9.6, 10.4, 9.7, 10.3, 9.2, 9.3, 9.9, 9.5, 9, 10.9]

Is there a significant difference in variability between two groups?

Solution:

Using a box-whisker plot, let's first visualize the data as shown in Fig. (8.1).



It is seen that inmates placed in solitary confinement (red box) show a clear decrease in the α-wave frequency.

Furthermore, the variability of that particular group seems higher than non-confined inmates.

Fig 8.1: *Non-confined (blue) vs solitary confined (red)*

Script 8.1

```
from scisuit.stats import test_f, test_f_Result

nonconfined = [10.7, 10.7, 10.4, 10.9, 10.5, 10.3, 9.6, 11.1, 11.2, 10.4]
confined = [9.6, 10.4, 9.7, 10.3, 9.2, 9.3, 9.9, 9.5, 9, 10.9]
result = test_f(x=confined, y=nonconfined)
print(result)
```

```
F test for two.sided
df1=9, df2=9, var1=0.357, var2=0.211
F=1.696
p-value =0.443
Confidence interval: (0.42, 6.83)
```

Since $p>0.05$, we cannot reject H_0 ($\sigma_1=\sigma_2$). Therefore, there is no statistically significant difference between the variances of two groups.

9. Analysis of Variance (ANOVA)

In Section (7.2) we have seen that when exactly two means needs to be compared, we could use two-sample t-test. The methodology for *comparing several means* is called analysis of variance (ANOVA). When there is only a single factor with multiple levels, i.e. color of strawberries subjected to different power levels of infrared radiation, then we can use *one-way ANOVA*. However, besides infrared power if we are also interested in different exposure times, then *two-way ANOVA* needs to be employed.

9.1. One-Way ANOVA

There are 3 essential assumptions for the test to be accurate (Anon 2024)[32]:

1. Each group comes from a normal population distribution.
2. The population distributions have the same standard deviations ($\sigma_1=\sigma_2=\ldots=\sigma_n$).

   It is reasonable to expect that standard deviations of populations have some differences in values. Therefore, Peck *et al*. (2016) suggest that if $\sigma_{max} \leq 2 \cdot \sigma_{min}$ ANOVA still can safely be used.
3. The data are independent.

A similarity comparison of two-sample t-test and ANOVA is given by Moore *et al*. (2009). Suppose we are analyzing whether the means of two different groups of same size are different. Then we would employ two-sample t-test with equal variances (due to assumption #2):

$$t=\frac{\bar{X}-\bar{Y}}{S_p\sqrt{\frac{1}{n}+\frac{1}{n}}}=\frac{\sqrt{\frac{n}{2}}\left(\bar{X}-\bar{Y}\right)}{S_p} \tag{9.1}$$

The square of test statistic is:

$$t^2=\frac{\frac{n}{2}\left(\bar{X}-\bar{Y}\right)^2}{S_p^2} \tag{9.2}$$

32 https://online.stat.psu.edu/stat500/lesson/10/10.2/10.2.1

If we had used ANOVA, the *F-statistic* would have been exactly equal to t^2 computed using Eq. (9.2). A careful inspection of Eq. (9.2) reveals couple things:

1. The numerator measures the variation *between* the groups (known as fit).
2. The denominator measures the variation *within* groups (known as residual), see Eq. (7.3).

The null- and alternative-hypothesis for ANOVA are:

$$\begin{aligned} H_0&: \quad \mu_1 = \mu_2 = \ldots = \mu_n \\ H_a&: \quad \text{At least two of the } \mu\text{'s are different} \end{aligned} \tag{9.3}$$

Therefore the basic idea is, to test H_0, we simply compare the variation *between* the means of the groups with the variation *within* groups. A graphical example adapted from Peck *et al*. (2016) can cement our understanding:

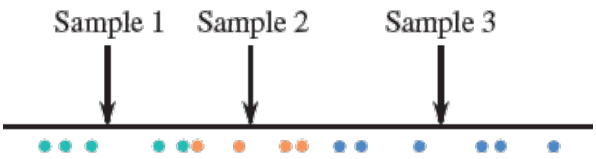


Fig 9.1-A: *A dataset with small within variability*

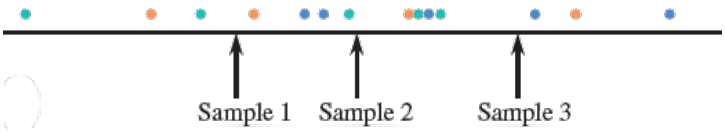


Fig 9.1-B: *A dataset with high within variability*

It is clearly seen from Fig. (9.1-A) that H_0 can be rejected as the means of 3 samples are different. The variability within each sample is smaller than the differences between the sample means.

In Fig. (9.1-B), the difference between sample means are as same as Fig. (9.1-A); however, there is considerable overlap between the samples. Therefore, *the difference between the means of the samples could simply be due to variability in sampling rather than the differences in population means.*

Computing the statistics:

Let k be the number of populations being compared [in Fig. (9.1) k=3] and $n_1, n_2, \ldots, n_k$ be the sample sizes:

1. Total number of observations:

   $N = n_1 + n_2 + \ldots + n_k$

2. Grand total (the sum of all observations):

$$T=\sum_{k=1}^{k}\sum_{i=1}^{n}X_{k,i}$$

3. Grand mean (average of all observations):

$$\bar{x}=\frac{T}{N}$$

4. Sum of squares of treatment:

$$SS_{TR}=n_1\cdot(\bar{x}_1-\bar{x})^2+n_2\cdot(\bar{x}_2-\bar{x})^2+\ldots+n_k\cdot(\bar{x}_k-\bar{x})^2$$

where *df=k-1*

5. Sum of squares of error:

$$SS_{Error}=(n_1-1)\cdot s_1^2+(n_2-1)\cdot s_2^2+\ldots+(n_k-1)\cdot s_k^2$$

where *df = N-k*

6. Mean squares:

$$MS_{TR}=\frac{SS_{TR}}{k-1}\quad\text{and}\quad MS_{Error}=\frac{SS_{Error}}{N-k}$$

The test statistics:

$$F=\frac{MS_{TR}}{MS_{Error}} \tag{**9.4**}$$

with $df_1=k-1$ and $df_2=N-k$.

Before proceeding with an example on ANOVA, let's further investigate Eq. (9.4). Remember that *F* distribution is the ratio of independent chi-square random variables and is given with the following equation:

$$F=\frac{U/m}{V/n} \tag{**9.5**}$$

where *U* and *V* are independent chi-square random variables with *m* and *n* degrees of freedom.

The following theorem establishes the link between Eqs. (9.4 & 9.5):

Theorem: Let $Y_1, Y_2, \ldots, Y_n$ be random sample from a normal distribution with mean μ and variance σ^2. Then,

$$\frac{(n-1)S^2}{\sigma^2} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(Y_i - \bar{Y})^2 \qquad \textbf{(9.6)}$$

has a chi-square distribution with *n-1* degrees of freedom. A proof of Eq. (9.6) is given by Larsen & Marx (2011) and is beyond the scope of this study.

Using Eq. (9.6), now it is easy to see that when sum of squares of treatment (or error) is divided by σ, it will have a chi-square distribution. Therefore Eq. (9.4) is indeed equivalent to Eq. (9.5) and therefore gives an F distribution with $df_1 = k-1$ and $df_2 = N-k$.

Example 9.1

In most of the integrated circuit manufacturing, a plasma etching process is widely used to remove unwanted material from the wafers which are coated with a layer of material, such as silicon dioxide. A process engineer is interested in investigating the relationship between the radio frequency power and the etch rate. The etch rate data (in Å/min) from a plasma etching experiment is given below:

160 W	180 W	200 W	220 W
575	565	600	725
542	593	651	700
530	590	610	715
539	579	637	685
570	610	629	710

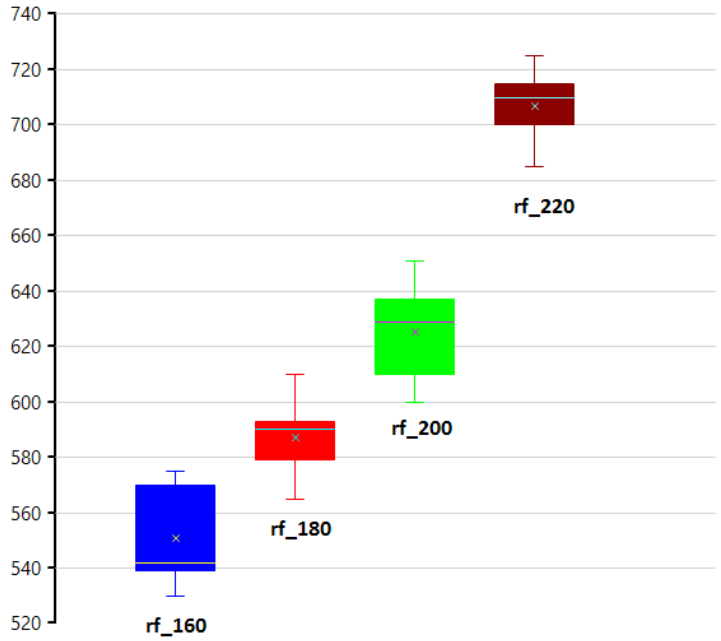
Does the RF power affect etching rate? (Adapted from Montgomery 2012)

Solution:

Before attempting any numerical solution, let's first visualize the data using box-whisker plot generated with a Python script:

Script 9.1

```
import scisuit.plot as plt

rf_160 = [575, 542, 530, 539, 570]
rf_180 = [565, 593, 590, 579, 610]
rf_200 = [600, 651, 610, 637, 629]
rf_220 = [725, 700, 715, 685, 710]

for dt in [rf_160, rf_180, rf_200, rf_220]:
    _name = [ k for k,v in locals().items() if v == dt][0]
    plt.boxplot(data=dt, label=_name)
plt.show()
```



It is immediately seen from the figure that μ_{220} is considerably different than other means. It can thus be inferred that the null hypothesis will be rejected since H_0 claims:

$$\mu_{160}=\mu_{180}=\mu_{200}=\mu_{220}$$

Fig 9.2: *The etch rate data at different RFs*

Before using *scisuit*'s built-in function, let's compute *F-value* using a Python script so that above-shown steps to calculate test statistics become clearer.

Script 9.2

```
import numpy as np
from scisuit.stats import qf

#create a 2D array
data = np.array([rf_160, rf_180, rf_200, rf_220]) #see Script (9.1)

#compute grand mean
grandmean = np.mean(data)

ss_tr, ss_error = 0, 0
for dt in data:
    n = len(dt) #size of each sample
    ss_tr += n*(np.mean(dt)-grandmean)**2
    ss_error += (n-1)*np.var(dt, ddof=1) #note ddof=1, the sample variance

row, col = data.shape
df_tr = row - 1
df_error = row*(col - 1)

Fvalue = (ss_tr/df_tr) / (ss_error/df_error)
Fcritical = qf(1-0.05, df1=df_tr, df2=df_error)

print(f"F={Fvalue}, F-critical={Fcritical}")
```

```
F=66.8, F-critical=3.24
```

Since the computed *F-value* is considerably greater than *F-critical*, we can safely reject H_0. Using *scisuit*'s built-in *aov* function:

Script 9.3

```
aovresult = aov(rf_160, rf_180, rf_200, rf_220)
print(aovresult)
```

One-Way ANOVA Results

Source	**df**	**SS**	**MS**	**F**	**p-value**
Treatment	3	66870.55	22290.18	66.80	2.8829e-09
Error	16	5339.20	333.70		
Total	19	72209.75			

Since $p<0.05$, we can reject H_0 in favor of H_1.

Now, had we not plotted Fig. (9.2), we would not be able to see why H_0 has been rejected. As a matter of fact, among other reasons due to overlap in whiskers and boxes or outliers a box-whisker plot does not always clearly show whether H_0 will be rejected. Therefore, we need to use post hoc tests along

with ANOVA. There are several tests[33] for this purpose, here we will be using Tukey's test[34]. Continuing from Script (9.3):

```
tukresult = tukey(alpha=0.05, aovresult=aovresult)
print(tukresult)
```

Tukey Test Results (alpha=0.05)

Pairwise Diff	i-j	Interval
1 - 2	-36.20	(-69.25, -3.15)
1 - 3	-74.20	(-107.25, -41.15)
1 - 4	-155.80	(-188.85, -122.75)
2 - 3	-38.00	(-71.05, -4.95)
2 - 4	-119.60	(-152.65, -86.55)
3 - 4	-81.60	(-114.65, -48.55)

Since none of the pairs contain the value 0.0, the Tukey procedure shows that means of all pairs are significantly different. Thus it can be concluded that each power level has an effect on etch rate that is different from the other power levels.

9.2. Two-Way ANOVA

In one-way ANOVA, the populations were classified according to a single factor; whereas in two-way ANOVA, as the name implies, there are two factors, each with different number of levels. For example, a baker might choose 3 different baking temperatures (150, 175, 200°C) and 2 different baking times (45 and 60 min) to optimize a cake recipe. In this example we have two factors (baking time and temperature) each with different number of levels (Devore *et al*. 2021; Moore *et al*. 2009).

Moore *et al*. (2009) lists the following advantages for using two-way ANOVA:

1. It is more efficient (i.e., less costly) to study two factors rather than each separately,
2. The variation in residuals can be decreased by the inclusion of a second factor,
3. Interactions between factors can be explored.

33 https://en.wikipedia.org/wiki/Post_hoc_analysis
34 https://en.wikipedia.org/wiki/Tukey%27s_range_test

In order to analyze a data set with two-way ANOVA the following assumptions must be satisfied (Field 2024; Moore 2012):

1. The response variable must be continuous (e.g., weight, height, yield, …),
2. The two independent variables must consist of discrete levels (e.g., type of treatment, brand of product) and each factor must have at least two levels,
3. In order to analyze interaction effects between independent variables, there should be replicates,
4. The observations must be independent,
5. It is desirable that the design should be balanced.

Let's start from **#5** and take a look at what it means balanced or unbalanced. In ANOVA or design of experiments, a balanced design has equal number of observations for all possible combinations of factor levels. For example[35], assume that the independent variables are A, B, C with 2 levels. Table (9.1) shows a balanced design whereas Table (9.2) shows an unbalanced design of the same factors (since the combination [1, 0, 0] is missing).

Table 9.1: *Balanced Design*

A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Table 9.2: *Unbalanced Design*

A	B	C
0	0	0
0	1	0
0	1	0
0	0	1
0	1	0
1	0	1
1	1	0
1	1	1

Note that if Table (9.1) was re-designed such that each row displayed a factor level (0 or 1) and each column displayed a factor (A, B or C) then there would be *no empty cells* in that table. If the data includes multiple observations for each treatment, the design includes *replication*.

35 https://support.minitab.com/en-us/minitab/help-and-how-to/statistical-modeling/anova/supporting-topics/anova-models/balanced-and-unbalanced-designs/

Example 9.2

A study by Moore and Eddleman[36] (1991) investigated the removal of marks made by erasable pens on cotton and cotton/polyester fabrics. The following data compare three different pens and four different wash treatments with respect to their ability to remove marks on. The response variable is based on the color change and the lower the value the more marks were removed.

Table 9.3: *Effect of washing treatment and different pen brands on color change*

	Wash 1	**Wash 2**	**Wash 3**	**Wash 4**
Pen #1	0.97	0.48	0.48	0.46
Pen #2	0.77	0.14	0.22	0.25
Pen #3	0.67	0.39	0.57	0.19

Is there any difference in color change due either to different brands of pen or to the different washing treatments? (Adapted from *Devore et al*. 2021)

Solution:

The data satisfies the requirements to be analyzed with two-factor ANOVA, since:

1. There are two independent factors (pen brands and washing treatment),
2. The independent variables consist of discrete levels (e.g., brand #1, #2 and #3)
3. There are no empty cells (data is balanced),
4. There are ***no replicates*** (interaction cannot be explored),
5. Observations are independent.

Once a table similar to Table (9.3) is prepared, finding the F-values for both factors is fairly straightforward if a spreadsheet software is used.

Grand mean (T) = 0.466

36 **Moore MA, Eddleman VL** (1991). An Assessment of the Effects of Treatment, Time, and Heat on the Removal of Erasable Pen Marks from Cotton and Cotton/Polyester Blend Fabrics. *J. Test. Eval.*. 19(5): 394-397

Averages of treatments ($\mu_{treatments}$) = [0.803, 0.337, 0.423, 0.3]

$$SS_{treatment}=\sum_{i=1}^{4}(\mu_{treatments}[i]-T)^2\times 3=0.48 \quad \text{and} \quad MS_{treatment}=\frac{SS_{treatment}}{df}=\frac{0.48}{4-1}=0.16$$

Averages of brands (μ_{brands}) = [0.598, 0.345, 0.455]

$$SS_{brand}=\sum_{i=1}^{3}(\mu_{brands}[i]-T)^2\times 4=0.128 \quad \text{and} \quad MS_{brand}=\frac{SS_{brand}}{df}=\frac{0.128}{3-1}=0.06$$

$$SS_{Error}=\sum\sum(\mu_{ij}-T)-SS_{Treatment}-SS_{brand}=0.087 \quad \text{and} \quad MS_{Error}=\frac{SS_{Error}}{df}=\frac{0.087}{(3-1)\times(4-1)}=0.014$$

$$F_{treatment}=\frac{MS_{treatment}}{MS_{Error}}=\frac{0.16}{0.014}=11.05$$

$$F_{brand}=\frac{MS_{brand}}{MS_{Error}}=\frac{0.06}{0.014}=4.15$$

Although the solution is straightforward, it is still cumbersome and error-prone; therefore, it is best to use functions dedicated for this purpose:

Script 9.4

```
brand = [1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3]
treatment = [1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4]
removal = [0.97, 0.48, 0.48, 0.46, 0.77, 0.14, 0.22, 0.25, 0.67, 0.39, 0.57, 0.19]

result = aov2(y=removal, x1=treatment, x2=brand)
print(result)
```

Two-way ANOVA Results

Source	df	SS	MS	F	p-value
x1	3	0.48	0.16	11.05	7.40e-03
x2	2	0.13	0.06	4.43	6.58e-02

■

Unlike Example (9.2) in which the data does not have replicates, the following example will demonstrate a data set which have replicates. It should be noted that when replicates are involved the

solution becomes slightly more tedious and therefore the following example will be directly solved using *scisuit*'s built-in function. Interested readers can consult to textbooks (Devore *et al*. 2021) for a detailed solution.

Example 9.3

A process engineer is testing the effect of catalyst type (A, B, C) and reaction temperature (high, medium, low) on the yield of a chemical reaction. She designs an experiment with 3 replicates for each combination as shown in the following data. Do both catalyst type and reaction temperature have an effect on the reaction yield?

Catalyst = [A, A, A, A, A, A, A, A, A, B, B, B, B, B, B, B, B, B, C, C, C, C, C, C, C, C, C]

Temperature = [L, L, L, M, M, M, H, H, H, L, L, L, M, M, M, H, H, H, L, L, L, M, M, M, H, H, H]

%Yield = [85, 88, 90, 80, 82, 84, 75, 78, 77, 90, 92, 91, 85, 87, 89, 80, 83, 82, 88, 90, 91, 84, 86, 85, 79, 80, 81]

Solution:

If one wishes to use a spreadsheet for the solution, a table of averages needs to be prepared as shown below:

Table 9.4: *Effect of temperature and catalyst type on reaction rate*

	Temperature		
Catalyst	**L**	**M**	**H**
A	87.667	82	76.667
B	91	87	81.667
C	89.667	85	80

After preparing the above-shown table, a methodology similar to Example (9.2) can be followed.

Let’s solve the question directly by using *scisuit*’s built-in function:

Script 9.5

```
from scisuit.stats import aov2

Catalyst = ["A", "A", "A", "A", "A", "A", "A", "A", "A",
            "B", "B", "B", "B", "B", "B", "B", "B", "B",
            "C", "C", "C", "C", "C", "C", "C", "C", "C"]

Temperature = ["L", "L", "L", "M", "M", "M", "H", "H", "H",
               "L", "L", "L", "M", "M", "M", "H", "H", "H",
               "L", "L", "L", "M", "M", "M", "H", "H", "H"]

Yield = [85, 88, 90, 80, 82, 84, 75, 78, 77, 90, 92, 91,
         85, 87, 89, 80, 83, 82, 88, 90, 91, 84, 86, 85, 79, 80, 81]

result = aov2(y=Yield, x1=Temperature, x2=Catalyst)
print(result)
```

Two-way ANOVA Results

Source	**df**	**SS**	**MS**	**F**	**p-value**
x1	2	450.30	225.15	83.27	7.9886e-10
x2	2	90.74	45.37	16.78	7.7004e-05
*x1*x2*	4	3.04	0.76	0.28	8.8654e-01

From the ANOVA results, it is seen that both temperature and catalyst have significant ($p<0.05$) effect on reaction yield.

10. Linear Regression

Based on the amount of error associated with data, there are two general approaches for *curve fitting* (Chapra & Canale 2013):

1. *Regression:* When data shows a significant degree of error or "noise" (generally originates from experimental measurements), we want a curve that represents the *general trend* of the data.
2. *Interpolation:* When the noise in data can be ignored (generally originates from tables), we would like a curve(s) that pass directly through each of the data points.

In terms of mathematical expressions, interpolation (Eq. 10.1) and regression (Eq. 10.2) can be shown as follows:

$$Y = f(X) \tag{10.1}$$

$$Y = f(X) + \epsilon \tag{10.2}$$

Peck *et al*. (2016) used the terms *deterministic* and *probabilistic* relationships for Eq. (10.1) and Eq. (10.2), respectively. Therefore a *probabilistic relationship is actually a deterministic relationship with noise* (random deviations).

To further our understanding on Eq. (10.2), a simple example from Larsen & Marx (2011) can be helpful: Consider a tooling process where the initial weight of the sample determines the finished weight of the steel rods. For example, in a simple experiment if the initial weight was measured as 2.745 g then the finished weight was measured as 2.080 g. However, even if the initial weight is controlled and is exactly 2.745 g, in reality the finished weight would fluctuate around 2.080 g. and therefore, with each x (independent variable) there will be a range of possible y values (dependent variable), which Eq. (10.2) exactly tells us.

10.1. Simple Linear Regression

When there is only a single explanatory (independent) variable, the model is referred to as "simple" linear regression. Therefore, Eq. (10.2) can be expressed as:

$$Y = \beta_0 + \beta_1 x + \epsilon \tag{10.3}$$

where regardless of the x value, the random variable ε is assumed to follow a N(0, σ) distribution.

Let x^* show a particular value of x, then:

$$E(\beta_0 + \beta_1 x^* + \epsilon) = \beta_0 + \beta_1 x^* + E(\epsilon) = \beta_0 + \beta_1 x^* = \mu_{Y|x^*} \tag{10.4}$$

$$Var(\beta_0 + \beta_1 x^* + \epsilon) = Var(\epsilon) = \sigma^2_{Y|x^*} \tag{10.5}$$

where the notation $Y|x^*$ should be read as the value of Y when $x=x^*$, i.e., the mean value of Y when $x=x^*$. Note also that Eq. (10.4) tells us something important that the population regression line is the line of mean values of Y.

The following assumptions are made for a linear model (Larsen & Marx, 2011):

1. $f_{Y|x}(y)$ is a normal probability density function for all x (i.e., for a known x value, there is a probability density function associated with y values)
2. The standard deviations, σ, of y-values are same for all x values.
3. For all x-values, the distributions associated with $f_{Y|x}(y)$ are independent.

Example 10.1

Suppose that the relationship between applied stress (x) and time to fracture (y) is given by the simple linear regression model with $\beta_0=65$, $\beta_1=-1.2$, and $\sigma=8$. What is the probability of getting a fracture value greater than 50 when the applied stress in 20? (Adapted from Devore et al. 2021)

Solution:

Let's compute y when x=20:

$$y = 65 - 1.2x = 65 - 1.2 \times 20 = 41$$

Note that if this was a curve fitting problem in nature, then whenever the stress value was 20, the fracture time would have always been equal to 41. However, since Eq. (10.2) tells us that random deviations are involved, this cannot be the case. We already know that the random deviations, namely ε, follows a normal distribution. Therefore, it becomes straightforward to compute the probability:

$$P\left(Z>\frac{50-41}{8}\right)=P\left(Z>1.125\right)=1-pnorm\left(1.125\right)=0.13 \qquad \blacksquare$$

In Example (10.1), the coefficients, namely β_0 and β_1, of the regression line was given. However, in practice we need to estimate these coefficients. It should be noted that there are two commonly[37] used methods for estimating the regression coefficients (please note that we use the word, *estimate*):

1. Least squares estimation method,
2. Maximum likelihood estimation method.

10.1.1. Least Squares Estimation

Let $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ represent n observation pairs, from the measurement of X and Y. Our goal is to find β_0 and β_1 in Eq. (10.3) such that the drawn line is *as close as possible* to all data points.

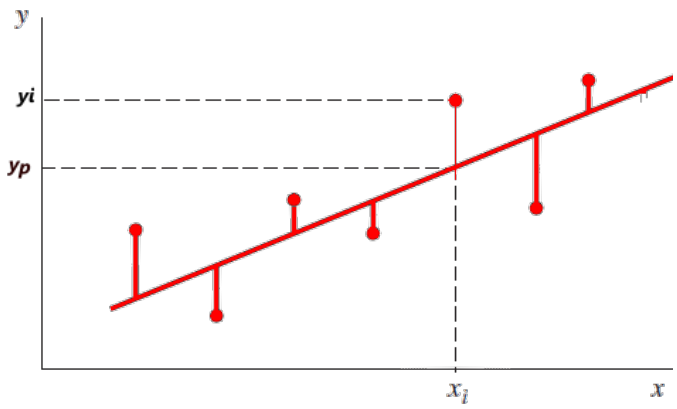


Fig 10.1: *Fitting a line through a set of data points*

In the figure y_i is the measured data point whereas y_p is the predicted value both of which corresponds to x_i. The associated error (also known as *residual*) is with this prediction is:

$$e_i = y_i - y_p$$

Since by definition we want the line as close as possible to all data points, therefore our goal is to minimize the sum of e_i's by varying β_0 and β_1.

37 https://support.minitab.com/en-us/minitab/help-and-how-to/statistical-modeling/reliability/supporting-topics/estimation-methods/least-squares-and-maximum-likelihood-estimation-methods/

The residual sum of squares (RSS) also known as sum of squares of error (SSE):

$$RSS=\sum_{i=1}^{n} e_i^2=e_1^2+e_2^2+...+e_n^2 \qquad \textbf{(10.6)}$$

If the coefficients of the best line passing through the data points are β_0 and β_1 then:

$$L=RSS=\sum_{i=1}^{n} (y_i-\beta_0-\beta_1 x)^2 \qquad \textbf{(10.7)}$$

The partial derivatives of Eq. (10.7) with respect to β_0 and β_1 are:

$$\frac{\partial L}{\partial \beta_0}=\sum_{i=1}^{n} -2(y_i-\beta_0-\beta_1 \cdot x_i)=0$$

$$\frac{\partial L}{\partial \beta_1}=\sum_{i=1}^{n} 2x_i(y_i-\beta_0-\beta_1 \cdot x_i)=0$$

Dropping the constants -2 and 2 from both equations and simply rearranging the terms yields:

$$\sum_{i=1}^{n} y_i=n\beta_0+\beta_1\sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i \cdot y_i=\beta_0\sum_{i=1}^{n} x_i+\beta_1\sum_{i=1}^{n} x_i^2$$

We have two equations and two unknowns, therefore it is possible to solve this system of equations. Here, one can use the elimination method; however, Cramer's rule provides a direct solution. Let's solve for β_1 and leave β_0 as an exercise:

$$\hat{\beta}_1=\frac{\begin{vmatrix} n & \sum y_i \\ \sum x_i & \sum x_i y_i \end{vmatrix}}{\begin{vmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{vmatrix}}$$

If one takes the determinants in numerator and denominator, then:

$$\hat{\beta}_1 = \frac{n\sum x_i y_i - (\sum x_i)(\sum y_i)}{n\sum x_i^2 - (\sum x_i)^2} \quad \textbf{(10.8)}$$

β_1 can be further simplified if a notation S_{xy} and S_{xx} and S_{xy} are defined as:

$$S_{xx} = \sum (x_i - \bar{x})^2 = \sum x_i^2 - \frac{1}{n}(\sum x_i)^2$$

$$S_{yy} = \sum (y_i - \bar{y})^2 = \sum y_i^2 - \frac{1}{n}(\sum y_i)^2$$

$$S_{xy} = \sum (x_i - \bar{x})(y_i - \bar{y}) = \sum x_i y_i - \frac{1}{n}(\sum x_i)(\sum y_i)$$

Then $\hat{\beta}_1$ can be simplified as:

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} \quad \textbf{(10.9)}$$

and $\hat{\beta}_0$ is equal to:

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \cdot \bar{x} \quad \textbf{(10.10)}$$

and the estimated variance is:

$$\hat{\sigma^2} = \frac{1}{n}\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \quad \textbf{(10.11)}$$

where $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \cdot x_i, \quad i = 1, 2, \ldots, n$

10.1.2. Maximum likelihood estimation

Before proceeding with the derivation based on maximum likelihood estimation (MLE), let's work on a simple example.

Example 10.2

Suppose you have been tasked with finding the probability of heads (H) and tails (T) of an unknown coin. You flipped the coin for 3 times and the sequence is HTH. What is the probability, *p*? (Adapted from Larsen & Marx)

Solution:

It makes sense with defining a random variable, X, as follows:

$$X=\begin{cases}1 & \text{heads come up}\\ 0 & \text{tails come up}\end{cases}$$

Then a probability model is defined:

$$p_X(k)=p^k(1-p)^{1-k}=\begin{cases}p & \text{k=1}\\ 1-p & \text{k=0}\end{cases}$$

Therefore, based on the probability model the function is that defines the sequence HTH is:

$$p_X(k)=p^2(1-p)$$

Using calculus, it can easily be computed that the value that maximizes the probability model is: $p=2/3$. ■

Now, instead of the sequence HTH (Example 10.2) we have data pairs (x_1, y_1), (x_2, y_2), … , (x_n, y_n) obtained from a random experiment. Furthermore, it is known that the y_i's are normally distributed with mean $\beta_0+\beta_1 x_i$ and variance σ^2 (Eqs. 10.4 & 10.5).

The equation for normal distribution is:

$$f_Z(z)=\frac{1}{\sqrt{2\pi\sigma}}e^{\frac{-1}{2}\cdot\left(\frac{x-\mu}{\sigma}\right)^2},-\infty<x<\infty \qquad \textbf{(10.12)}$$

Replacing x and μ in Eq. (10.12) with y_i and Eq. (10.4), respectively, yields the probability model for a single data pair:

$$f_Z(z)=\frac{1}{\sqrt{2\pi\sigma}}e^{\frac{-1}{2}\cdot\left(\frac{y_i-\beta_0-\beta_1 x_i}{\sigma}\right)^2} \quad \textbf{(10.13)}$$

For *n* data pairs, the maximum likelihood function is:

$$L=\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi\sigma}}e^{\frac{-1}{2}\cdot\left(\frac{y_i-\beta_0-\beta_1 x_i}{\sigma}\right)^2} \quad \textbf{(10.14)}$$

In order to find MLE of β_0 and β_1 partial derivatives with respect to β_0 and β_1 must be taken. However, Eq. (10.14) is not easy to work with as is. Therefore, as suggested by Larsen and Marx (2011), taking the logarithm will make it more convenient to work with.

$$-2lnL=n\cdot\ln(2\pi)+n\ln(\sigma^2)+\frac{1}{\sigma^2}\sum_{i=1}^{n}(y_i-\beta_0-\beta_1\cdot x_i)^2 \quad \textbf{(10.15)}$$

Taking the partial derivatives of Eq. (10.15) with respect to β_0 and β_1 and solving the resulting set of equations similar to as shown in section (10.1.1) will yield Eqs. (10.9 & 10.10).

10.1.3. Properties of Linear Estimators

Due to the assumptions made for a linear model (section 10.1), the estimators, $\hat{\beta}_0, \hat{\beta}_1$ and $\hat{\sigma}$, are random variables (i.e., probability distribution functions are associated with them). Then,

1. $\hat{\beta}_0$ and $\hat{\beta}_1$ are normally distributed.
2. $\hat{\beta}_0$ and $\hat{\beta}_1$ are unbiased, therefore, $E(\hat{\beta}_0)=\beta_0$ and $E(\hat{\beta}_1)=\beta_1$
3. $Var(\hat{\beta}_1)=\dfrac{\sigma^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$
4. $Var(\hat{\beta}_0)=\dfrac{\sigma^2\sum_{i=1}^{n}x_i^2}{n\sum_{i=1}^{n}(x_i-\bar{x})^2}$

Proof of #2:

In section (5.1.1), it was mentioned that to be an unbiased estimator, $E(\Theta) = \theta$ must be satisfied. In the case of $\hat{\beta}_1$, we need to show that $E(\hat{\beta}_1)=\beta_1$. If Eq. (10.8) is divided by n, the following equation is obtained:

$$\hat{\beta}_1=\frac{\sum x_i y_i-\frac{1}{n}(\sum x_i)(\sum y_i)}{\sum x_i^2-\frac{1}{n}(\sum x_i)^2} \quad \textbf{(I)}$$

Noting that $\bar{x}=\frac{\sum x_i}{n}$ the Eq. (I) can be rewritten as:

$$\hat{\beta}_1=\frac{\sum x_i y_i-\bar{x}\sum y_i}{\sum x_i^2-n\bar{x}^2} \quad \textbf{(II)}$$

Rearranging the terms in the numerator:

$$\hat{\beta}_1=\frac{\sum y_i(x_i-\bar{x})}{\sum x_i^2-n\bar{x}^2} \quad \textbf{(III)}$$

Note that due to the assumption of the linear model, in Eq. (III) except y_i, the other terms can be treated as constant. Therefore, replacing the expected value of y_i with Eq. (10.4) gives:

$$E(\hat{\beta}_1)=\frac{\sum(\beta_0+\beta_1 x_i)(x_i-\bar{x})}{\sum x_i^2-n\bar{x}^2} \quad \textbf{(IV)}$$

Expanding the terms in the numerator:

$$E(\hat{\beta}_1)=\frac{\beta_0\sum(x_i-\bar{x})+\beta_1\sum(x_i-\bar{x})x_i}{\sum x_i^2-n\bar{x}^2} \quad \textbf{(V)}$$

Noting that the first term in the numerator equals to 0 and the remaining terms in the numerator (except β_1) equals to the denominator, the proof is completed.

$$E(\hat{\beta}_1)=\beta_1 \quad \textbf{(VI)}$$

A similar proof can be obtained for β_0. For cases *#3* and *#4*, Larsen & Marx (2011) presented a detailed proof.

Example 10.3

It seems logical that riskier investments might offer higher returns. A study by Statman *et al*. (2008)[38] explored this by conducting an experiment. One group of investors rated the risk (**x**) of a company's stock on a scale from 1 to 10, while a different group rated the expected return (**y**) on the same scale. This was done for 210 companies, and the average risk and return scores were calculated for each. Data for a sample of ten companies, ordered by risk level, is given below:

x = [4.3, 4.6, 5.2, 5.3, 5.5, 5.7, 6.1, 6.3, 6.8, 7.5]

y = [7.7, 5.2, 7.9, 5.8, 7.2, 7, 5.3, 6.8, 6.6, 4.7]

How does the risk of an investment related to its expected return? (Adapted from Devore et al. 2021)

Solution:

Let's first visualize the data using a scatter plot.

Script 10.1

```
import scisuit.plot as plt

x = [4.3, 4.6, 5.2, 5.3, 5.5, 5.7, 6.1, 6.3, 6.8, 7.5]
y = [7.7, 5.2, 7.9, 5.8, 7.2, 7, 5.3, 6.8, 6.6, 4.7]
plt.scatter(x=x, y=y)
plt.show()
```

38 **Statman M, Fisher KL, Anginer D** (2008). Affect in a Behavioral Asset-Pricing Model. Financial Analysts Journal, 64-2, 20-29.

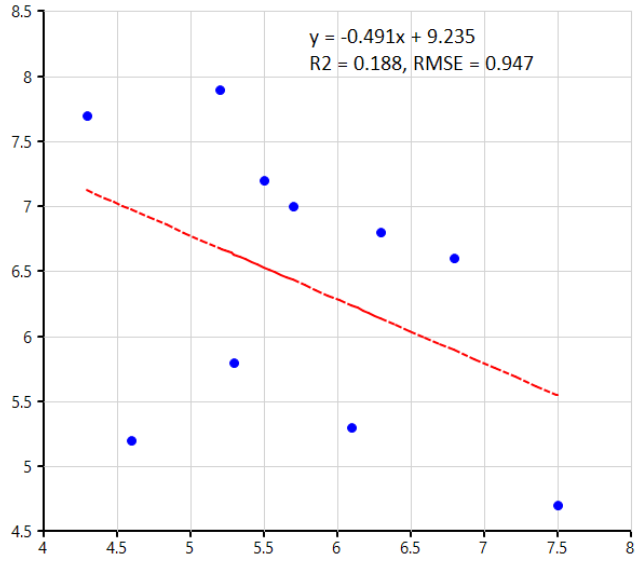


Fig 10.2: *Relationship between risk and expected return*

It is seen that there is a weak inverse relationship between the perceived risk of a company's stock and its expected return value.

Note: After plotting the data, since **scisuit**'s charts are interactive, the trendline was added by first selecting the data and then selecting "Add trendline" option.

Fig. (10.2) shows that there is no convincing relationship between risk and expected return of an investment. Let's take a look if this is numerically the case. Continuing from Script (10.1):

Script 10.2

```
from scisuit.stats import linregress
result = linregress(yobs=y, factor=x)
print(result)
```

```
Simple Linear Regression
F=1.85, p-value=0.211, R2=0.19

The regression equation: Y = 9.235 - 0.491·X
```

Predictor	**Coeff**	**StdError**	**T**	**p-value**
Intercept	9.235	2.10	4.40	0.0023
Slope	-0.491	0.36	-1.36	0.2110

Since $p>0.05$, we cannot reject the null hypothesis ($H_0: \beta_1=0$) in favor of H_1.

Have we carried out a reliable analysis, i.e., is there no relationship between risk and expected returns? Devore *et al*. (2021) suggested that with small number of observations, it is possible not to detect a relationship because when the sample size is small hypothesis tests do not have much power. Also note

that the original study uses 210 observations where Statman *et al*. (2008) concluded that risk is a useful predictor of expected return, although the risk only accounted for 19% of expected returns. ■

10.2. Multiple Linear Regression

Suppose the taste of a fruit juice is related to sugar content and pH. We wish to establish an empirical model, which can be described as follows:

$$y=\beta_0+\beta_1 x_1+\beta_2 x_2+\epsilon \tag{10.16}$$

where y is the response variable (taste) and x_1 and x_2 are independent variables (sugar content and pH). Unlike simple linear regression (SLR) model, where only one independent variable exists, in multiple linear regression (MLR) problems at least 2 independent variables are of interest to us. Therefore, in general, the response variable maybe related to k independent (regressor) variables. The model is:

$$y=\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_k x_k+\epsilon \tag{10.17}$$

This model describes a hyperplane and the regression coefficient, β_j, represents the expected change in response to per unit change in x_j when all other variables are held constant (Montgomery 2012). If one enters the data in a spreadsheet, it would generally be in the following format:

Table 10.1: *Data for multiple linear regression*

y	**x_1**	**x_2**	**…**	**x_k**
y_1	x_{11}	x_{12}	…	x_{1k}
y_2	x_{21}	x_{22}	…	x_{2k}
y_n	x_{n1}	x_{n2}	…	x_{nk}

y is the response variable and *x* are the regressor variables. It is assumed that $n>k$.

The model equation for the data in Table (10.1):

$$y=\beta_0+\sum_{j=1}^{k}\beta_j x_{ij}+\epsilon_i,\quad i=1,2,\ldots,n \tag{10.18}$$

For example, for the 1st row (i=1) in Table (10.1), Eq. (10.18) yields, $y_1 = \beta_0 + \beta_1 \cdot x_{11} + \beta_2 \cdot x_{12} + \ldots + \beta_k \cdot x_{1k}$.

To find the regression coefficients, we will use a similar approach presented in section (10.1.1), such that the sum of the squares of errors, ε_i, is minimized. Therefore,

$$L=\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{k}\beta_j x_{ij}\right)^2 \tag{10.19}$$

where the function L will be minimized with respect to β_0, β_1, …, β_k which then will give the least square estimators, $\hat{\beta}_1,\hat{\beta}_2,..,\hat{\beta}_k$. The derivatives with respect to β_0 and β_j are:

$$\left.\frac{\partial L}{\partial \beta_0}\right|_{\hat{\beta}_0,\hat{\beta}_1,\ldots,\hat{\beta}_k}=-2\sum_{i=1}^{n}\left(y_i-\hat{\beta}_0-\sum_{j=1}^{k}\hat{\beta}_j x_{ij}\right) \tag{10.20-a}$$

$$\left.\frac{\partial L}{\partial \beta_j}\right|_{\hat{\beta}_0,\hat{\beta}_1,\ldots,\hat{\beta}_k}=-2\sum_{i=1}^{n}\left(y_i-\hat{\beta}_0-\sum_{j=1}^{k}\hat{\beta}_j x_{ij}\right)x_{ij} \tag{10.20-b}$$

After some algebraic manipulation, Eq. (10.20) can be written in matrix notation as follows:

$$\begin{bmatrix} n & \sum x_{i1} & \sum x_{i2} & \ldots & \sum x_{ik} \\ \sum x_{i1} & \sum x_{i1}^2 & \sum x_{i1}x_{i2} & \ldots & \sum x_{i1}x_{ik} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ \sum x_{ik} & \sum x_{ik}x_{i1} & \sum x_{ik}x_{i2} & \ldots & \sum x_{ik}^2 \end{bmatrix}\begin{bmatrix}\hat{\beta}_0\\ \hat{\beta}_1\\ \vdots\\ \hat{\beta}_k\end{bmatrix}\begin{bmatrix}\sum y_i\\ \sum x_{i1}y_i\\ \vdots\\ \sum x_{ik}yi\end{bmatrix} \tag{10.21}$$

which can be condensed to the following expression:

$$X\cdot\beta=y \tag{10.22}$$

Note that since **X** is an *i* by *k* matrix, therefore not square, the inverse does not exist and therefore the equation cannot be solved. The least-squares approach to solving Eq. (10.22) is by multiplying with transpose of **X**:

$$X^T X \cdot \beta = X^T \cdot y \tag{10.23}$$

The test of significance of regression involves the hypotheses:

$$\begin{aligned} &\boldsymbol{H_0}: \beta_1 = \beta_2 = ... = \beta_k = 0 \\ &\boldsymbol{H_1}: \beta_j \neq 0 \qquad \text{for at least for 1 j} \end{aligned} \tag{10.24}$$

Example 10.4

A process engineer who was tasked to improve the viscosity of a polymer, among the several factors, chose two process variables: reaction temperature and feed rate. She ran 16 experiments and collected the following data:

Temperature = [80, 93, 100, 82, 90, 99, 81, 96, 94, 93, 97, 95, 100, 85, 86, 87]

Feed Rate = [8, 9, 10, 12, 11, 8, 8, 10, 12, 11, 13, 11, 8, 12, 9, 12]

Viscosity = [2256, 2340, 2426, 2293, 2330, 2368, 2250, 2409, 2364, 2379, 2440, 2364, 2404, 2317, 2309, 2328]

Explain the effect of feed rate and temperature on polymer viscosity. (Adapted from Montgomery 2012).

Solution:

The solution involves several computations which can be performed by using a spreadsheet or by using *Python* with *numpy* library. Step by step solution for the coefficients can be found in the textbook from Montgomery (2012). We will be skipping all these steps and directly solve it using *scisuit*'s builtin *linregress* function.

Script 10.3

```
from scisuit.stats import linregress

#input values
temperature = [80, 93, 100, 82, 90, 99, 81, 96, 94, 93, 97, 95, 100, 85, 86, 87]
feedrate = [8, 9, 10, 12, 11, 8, 8, 10, 12, 11, 13, 11, 8, 12, 9, 12]
viscosity = [2256, 2340, 2426, 2293, 2330, 2368, 2250, 2409, 2364, 2379, 2440, 2364, 2404, 2317, 2309, 2328]

#note the order of input to factor
result = linregress(yobs=viscosity, factor=[temperature, feedrate])
print(result)
```

```
Multiple Linear Regression
```

F=82.5, p-value=4.0997e-08, R2=0.93

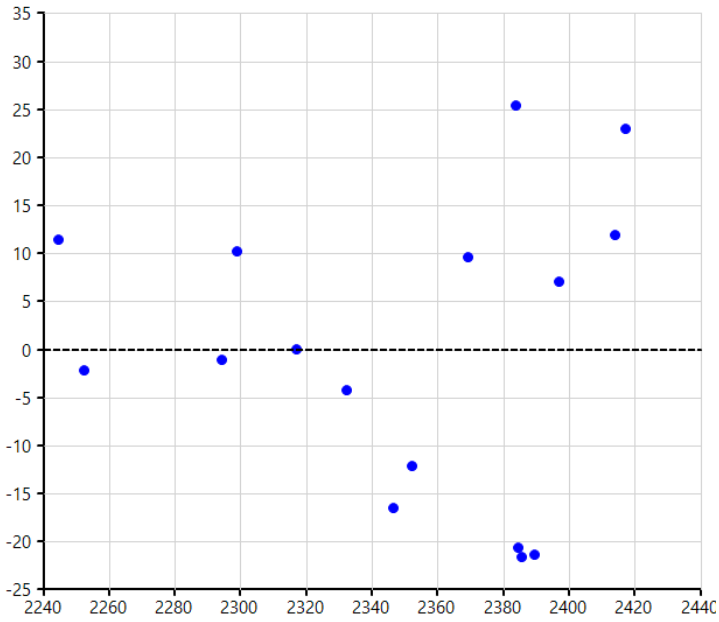
Predictor	Coeff	StdError	T	p-value
X0	1566.078	61.59	25.43	9.504e-14
X1	7.621	0.62	12.32	3.002e-09
X2	8.585	2.44	3.52	3.092e-03

Based on Eq. (10.24), the p-value tells us that at least one of the two variables (temperature and feed rate) has a nonzero regression coefficient. Furthermore, analysis on individual regression coefficients show that both temperature and feed rate have an effect on polymer's viscosity.

According to Larsen & Marx (2011), applied statisticians find residual plots to be very helpful in assessing the appropriateness of fitting. Continuing from Script (10.3), let's plot the residuals:

Script 10.4

```
import scisuit.plot as plt
import scisuit.plot.gdi as gdi

#x=Fits, y=Residuals
plt.scatter(x=result.Fits, y= result.Residuals)

#show a line at y=0
x0, x1 = min(result.Fits), max(result.Fits)*1.005
gdi.line(p1=(x0,0), p2=(x1, 0), lw=2, ls = "---" )

plt.show()
```



It is seen that the magnitudes of the residuals are comparable and they are randomly distributed. Therefore, the applied regression can be considered as appropriate.

Fig 10.3: *Fits vs residuals (y-axis)*

11. Exploring Normality

Most statistical methods are based on one basic assumption, that the observation follows normal distribution, which assumes that the samples come from normally distributed populations. Thus it is important to check normality assumptions (Das and Rahmatullah Imon, 2016). There are commonly two approaches to check normality:

1. Graphical Tests
   a) Histogram
   b) Box and Whisker Plot
   c) Normal Percent-Percent Plot
2. Analytical Test Procedures
   a) Kolmogorov-Smirnov Test
   b) Shapiro-Wilk Test
   c) Anderson-Darling Test

11.1. Graphical Tests

Let's start with ***histogram***, the easiest and simplest (in terms of interpretation) plot. A histogram provides a visual representation of the distribution of quantitative data. Before attempting to plot a histogram, let's first generate random data from normal and exponential distributions (highly skewed).

Script 11.1

```
import scisuit.plot as plt
from scisuit.stats import rnorm, rexp

n=1000
dt_norm, dt_exp = rnorm(n), rexp(n)
```

Let's see how we could visualize the data from Script (11.1) by histogram:

Script 11.2

```
plt.layout(1,2)
plt.subplot(0,0)
plt.hist(dt_norm, density=True)
plt.title("Normal")

plt.subplot(0, 1)
plt.hist(dt_exp, density=True)
plt.title("Exponential")

plt.show(antialiasing=True)
```

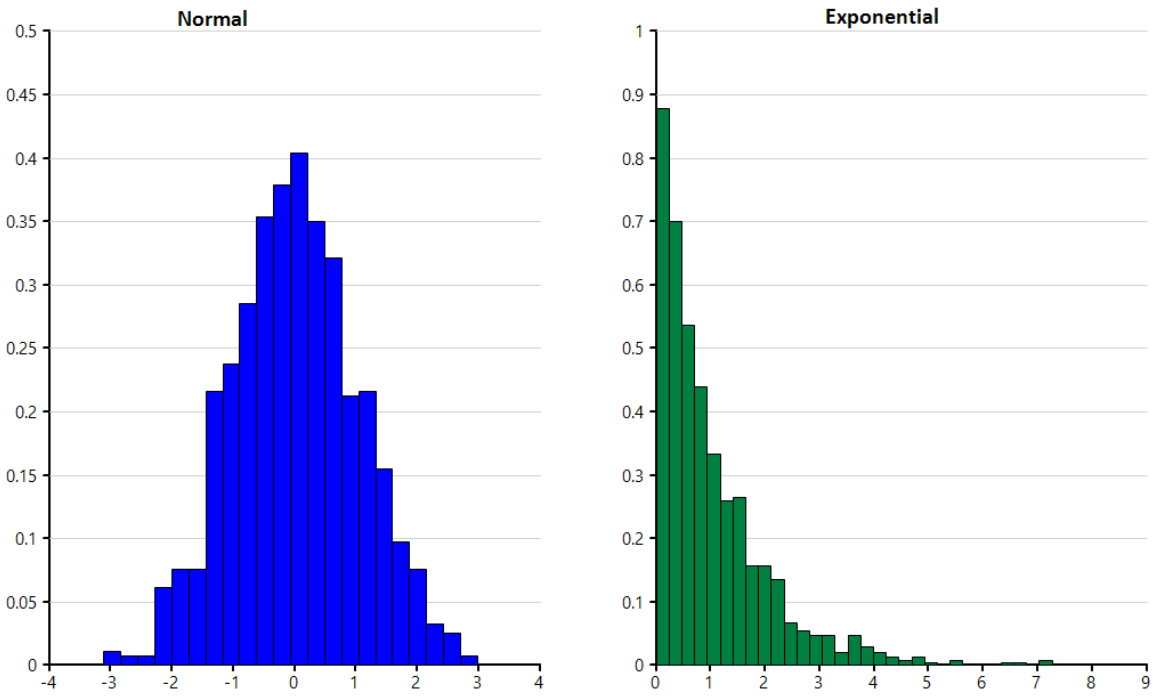


Fig 11.1: *Histogram of normal and exponential distributions*

It is clearly seen from Fig. (11.1) that normal distribution has a nearly bell-shaped distribution whereas exponential distribution is highly skewed to right.

Box-whisker plot has has another name as five number summary where it displays 1st quartile, median, 3rd quartile, min and max values. Continuing from Script (11.1):

Script 11.3

```
plt.boxplot(dt_norm)
plt.title("Normal")

plt.boxplot(dt_exp)
plt.title("Exponential")

plt.show(antialiasing=True)
```

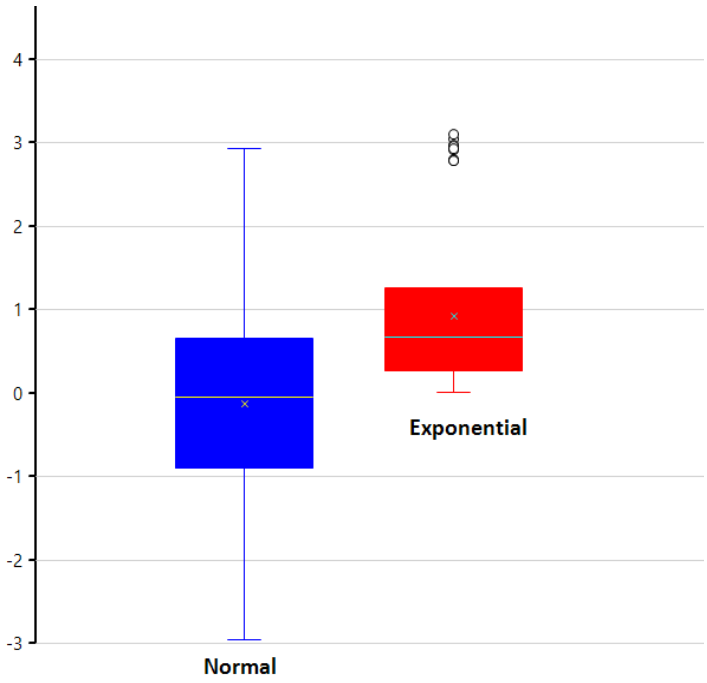


Fig 11.2: *Histogram of normal and exponential distributions (n=100)*

The following observations can be made.

Normal distribution:

1. Mean marker (x) and median line almost overlaps.
2. Mean divides the box into two equivalent halves.
3. $|Q_1\text{-min}|$ is more or less equivalent to $|Q_3\text{-max}|$.

Exponential distribution:

1. A clear separation between mean marker and median line.
2. Existence of outliers.
3. The mean marker is considerably closer to Q_3.
4. $|Q_1\text{-min}|$ is clearly different than $|Q_3\text{-max}|$.

Q–Q plot (quantile–quantile plot) is a graphical method for comparing two probability distributions by plotting their quantiles against each other[39]. On the other hand, a normal Q-Q plot is that which can be shaped by plotting quantiles of one distribution against quantiles of normal distribution. If both distributions come from normal distribution, then the data aligns itself on a straight line. To visualize Q-Q plot, we will slightly modify Script (11.2) and change ***hist*** with ***qqnorm*** function. Continuing from Script (11.1) and applying following changes, we should obtain Fig. (11.3):

Script 11.4

```
plt.qqnorm(dt_norm)
plt.qqnorm(dt_exp)
```

39 https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot

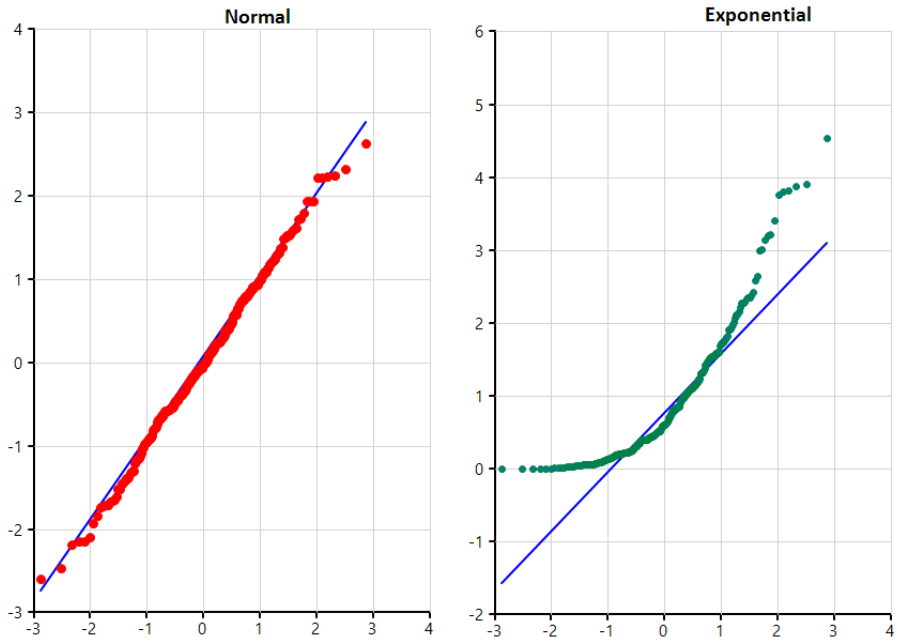


Fig 11.3: *QQ plot of normal and exponential distributions (n=250)*

It is seen from Fig. (11.3) that the data coming from a normal distribution aligns well with the straight line whereas the data from exponential distribution shows apparent deviations.

11.2. Analytical Test Procedures

11.2.1. Kolmogorov - Smirnov test

The Kolmogorov-Smirnov test was first derived by Kolmogorov[40] (1933) and more than a decade later modified by Smirnov[41] (1948). It is also known as KS test and used to test if a sample comes from a population with a specific distribution (NIST 2024)[42]. The KS test is defined by:

H_0: *The data follows a specific distribution, i.e. normal distribution*

H_1: *The data does not follow the specific distribution*

The test statistic is:

$$D = \sup_x |F_n(X) - F(X)| \tag{11.1}$$

where $F(X)$ is the theoretical cumulative distribution function (must be a continuous distribution and must be fully specified) of the normal distribution and $F_n(X)$ is the empirical CDF of the data.

Example 11.1

Does the following data come from a normal distribution:

[2.39798, -0.16255, 0.54605, 0.68578, -0.78007, 1.34234, 1.53208, -0.86899, -0.50855, -0.58256, -0.54597, 0.08503, 0.38337, 0.26072, 0.34729]

Solution:

We will write a script to run the KS test and at the same time visualize the CDF and ECDF. The following script performs both tasks:

Script 11.5

```
import numpy as np
import scisuit.plot as plt
from scisuit.stats import ks_1samp, pnorm

data = [2.39798, -0.16255, 0.54605, 0.68578, -0.78007, 1.34234, 1.53208, -0.86899, -0.50855,
-0.58256, -0.54597, 0.08503, 0.38337, 0.26072, 0.34729]
```

40 Kolmogorov A (1933). ‘‘Sulla determinazione empirica di una legge di distribuzione.’’ *G. Ist. Ital. Attuar*i, 4, 83–91

41 Smirnov N (1948). ‘‘Table for estimating the goodness of fit of empirical distributions.’’ *Annals of Mathematical Statistic*s, 19(2): 279–281.

42 https://www.itl.nist.gov/div898/handbook/eda/section3/eda35g.htm

```
mu, sd = np.mean(data), np.std(data)

"""Analytic test"""
result = ks_1samp(x=data, cdf=pnorm, args=( mu, sd))
print(result)


""" Visualization """
# Sort the data and compute ECDF
data_sorted = np.sort(data)
ecdf_y = np.arange(1, len(data) + 1) / len(data)
plt.scatter(x=data_sorted, y=ecdf_y, label="ECDF")

# Theoretical CDF for normal distribution
x = np.linspace(min(data), max(data), 100)
cdf_y = pnorm(x, mean=mu, sd=sd)
plt.scatter(x=x, y=cdf_y, label="CDF")

plt.legend(nrows=2)
plt.show()
```

```
Kolmogorov-Smirnov test
p-value: 0.885
Test statistic: 0.1414 and its sign 1
Max distance at: -0.50855
```

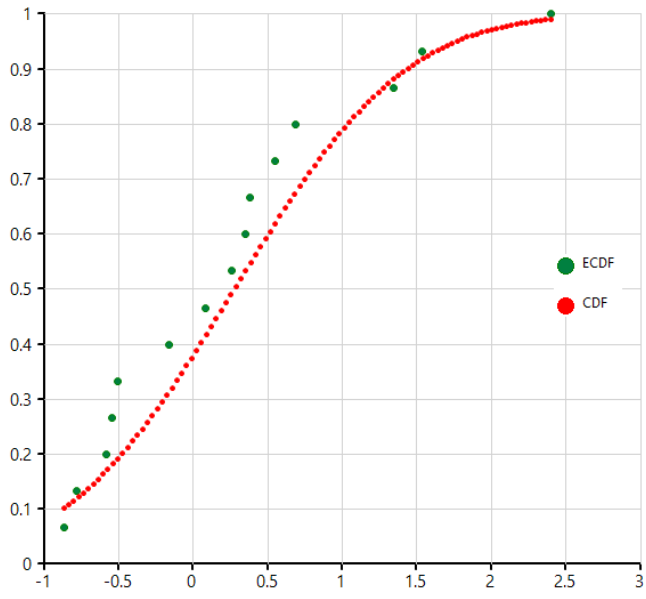


Fig 11.4: *CDF and ECDF of given data*

It is seen that the maximum vertical distance between CDF and ECDF is around -0.5 as confirmed by the ***ks_1samp*** test as -0.508.

The sign of the test is +1, which indicates that the ECDF is *above* the theoretical CDF as can be seen from Fig. (11.4).

The test statistic is calculated as the vertical distance between CDF and ECDF. The values of ECDF and CDF at max distance are roughly 0.334 and 0.192, respectively. Approximately the difference is *0.142*.

11.2.2. Shapiro-Wilk Test

The Shapiro-Wilk, introduced by Shapiro and Wilk (1965), is among the most popular tests and is particularly powerful for small to medium sample size. Similar to Kolmogorov-Smirnov test, the null and alternative hypotheses are:

H_0: *The data follows normal distribution*

H_1: *The data does not follow the normal distribution*

The test statistic is:

$$W = \frac{\left(\sum a_i y_{(i)}\right)^2}{\sum \left(y - \bar{y}\right)^2} \tag{11.2}$$

Example 11.2

Compute the test statistic (W) for the following data (from Shapiro and Wilk, 1965):

[6, 1, -4, 8, -2, 5, 0]

Solution:

1) The coefficients, a_i, in Eq. (11.2) is given by Shapiro and Wilk (1965) as:

$$a = [0.6233, 0.3031, 0.1401, 0.0, 0.1401, 0.3031, 0.6233]$$

2) The sorted sequence is: *[-4, -2, 0, 1, 5, 6, 8]*

3) In Shapiro and Wilk (1965) paper, the numerator in Eq. (11.2) is computed as follows (note that the indices start from 1):

- If the number of samples (n) is even, then $n=2k$, and numerator is:

$$b = \sum_{i=1}^{k} a_{n-i+1} \cdot \left(y_{n-i+1} - y_i\right)$$

- If n is odd then $n=2k+1$ and the computation of b is the same as above.

  Since n=7 for the example data, then k=3 and b is computed as follows:

$$b = \sum_{i=1}^{3} a_{7-i+1} \cdot \left(y_{7-i+1} - y_i\right)$$

Let’s automate these 3 steps using a Python script:

Script 11.6

```
import numpy as np
from scisuit.stats import shapiro

arr = np.array([6, 1, -4, 8, -2, 5, 0])
sorted_x = np.sort(arr)

n = len(arr)

a = np.array([0.6233, 0.3031, 0.1401, 0.0, 0.1401, 0.3031, 0.6233])

k = n/2 if n%2 == 0 else (n-1)/2

b = 0
for i in range(int(k)):
    b += a[n-i-1]*(sorted_x[n-i-1]- sorted_x[i])

W_test_stat = b**2/(np.var(arr)*n)
print(f"Test statistic: {W_test_stat}")

result = shapiro(arr)
print(result)
```

```
Test statistic: 0.95308

Shapiro-Wilk Test
p-value: 0.7612
Test statistic: 0.9535
```

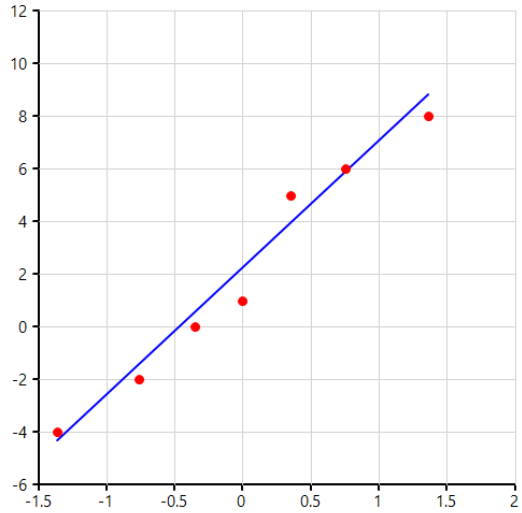
One way to visualize the Shapiro-Wilk test is through a QQ plot. If the points lie on a straight line, it indicates that the data is approximately normally distributed, matching the expectation used in the Shapiro-Wilk test. Therefore,

- When W is close to 1.0, the sample data aligns closely with the expected normal distribution, indicating that the data is likely normal.
- When W deviates significantly from 1.0, it suggests that the data does not follow a normal distribution.

Continuing from Script (11.6):

Script 11.7

```
import scisuit.plot as plt
plt.qqnorm(arr)
plt.show()
```



It is seen that the data aligns itself well with the straight line which is why the test statistic is close to 1.0.

Fig 11.5: *QQ plot of given data*

11.2.3. Anderson-Darling Test

Anderson-Darling test is similar to the KS test; however, it gives more weight to the tails (Anon. 2024)[43]. Similar to KS and Shapiro-Wilk tests, Anderson-Darling test is defined as:

H_0: *The data follows specified distribution*

H_1: *The data does not follow the specified distribution*

The test statistic is:

$$A^2 = -n - S \tag{11.3}$$

where

$$S = \sum_{i=1}^{n} \frac{(2i-1)}{n} \left[\ln F(Y_i) + \ln\left(1 - F(Y_{n+1-i})\right) \right] \tag{11.4}$$

where F is the CDF of the specified distribution and the Y_i are the ordered data.

43 https://www.itl.nist.gov/div898/handbook/eda/section3//eda35e.htm

Example 11.3

Test whether the above-given normality tests for t-distribution (see Fig. 4.11).

Solution:

Let's first remind ourselves briefly similarities and differences between t-distribution and standard normal distribution:

1) Both distributions are symmetric.

2) t- distribution is characterized by the degrees of freedom (df). As df increases, t- distribution becomes more similar to a normal dist.

3) The curves of t- distribution with larger df are taller and have thinner tails.

Script 11.8

```
from scisuit.stats import rt, anderson, ks_1samp, shapiro

n=50

data = rt(n=n, df=3)
for func in [anderson, ks_1samp, shapiro]:
    print(func(data))
    print("\n")
```

```
Anderson-Darling Test
p-value: 0.0
Test statistic: 2.3554


Kolmogorov-Smirnov test
p-value: 0.645
Test statistic: 0.1014 and its sign 1
Max distance at: 0.1982


Shapiro-Wilk Test
p-value: 0.0
Test statistic: 0.8508
```

It is seen that except KS test, both Shapiro-Wilk and Anderson-Darling tests can detect the difference between t-distribution (with small degrees of freedom) and standard normal distribution. However, when *df=10*, all of the above-mentioned tests yielded a p-value greater than 0.05.

11.2.4. Summary

The above-mentioned tests are sensitive to sample size such that if $n<20$ it can be difficult to detect deviations from normality whereas if $n>5000$ even minor departures from normality may be flagged as statistically significant (Anon. 2024)[44]. As a rule of thumb, sample sizes between 30-300 observations are recommended for reliable normality assessment.

Table 11.1: *Comparison of normality tests*

Test	Sample Size	Strengths
Anderson-Darling	Small to large	Focuses on the tails of the distribution and is applicable to various distributions.
Kolmogorov-Smirnov	Large	Focuses on entire distribution and is considered a general-purpose test
Shapiro-Wilk	Small to medium (generally $n<50$)	Focuses on entire distribution and is powerful for small samples

44 https://www.6sigma.us/six-sigma-in-focus/normality-test-lean-six-sigma/

References

Box GEP., Hunter WG, Hunter JS (2005). Statistics for Experimenters: Design, Innovation, and Discovery, 2nd Ed., Wiley.

Bury K (1999). Statistical Distributions in Engineering, Cambridge University Press.

Carlton MA, Devore JL (2014). Probability with Applications in Engineering, Science and Technology. Springer USA.

Chapra SC, Canale RP (2013). Numerical methods for engineers, seventh edition. McGraw Hill Education.

Das KR, Rahmatullah Imon AHM (2016). A Brief Review of Tests for Normality. *American Journal of Theoretical and Applied Statistics*. 5(1), 5-12.

Devore JL, Berk KN, Carlton MA (2021). Modern Mathematical Statistics with Applications. 3rd Ed., Springer.

Forbes C, Evans M, Hastings N, Peacock B (2011). Statistical Distributions, 4th Ed., Wiley.

Hastie T, Tibshirani R, Friedman J (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.

Hogg RV, McKean JW, Craig AT (2019). Introduction to mathematical statistics, 8th Ed., Pearson.

Kanji GK (2006). 100 Statistical Tests, 3rd Ed., Sage Publications.

Kreyszig E, Kreyszig H, Norminton EJ (2011). Advanced Engineering Mathematics, 10th Ed., John Wiley & Sons Inc.

Larsen RJ, Marx ML (2011). An Introduction to Mathematical Statistics and Its Applications. 5th Ed., Prentice Hall.

Liben-Nowell D. (2022). Connecting Discrete Mathematics and Computer Science (2nd Ed.). Cambridge: Cambridge University Press.

Miller I, Miller M (2014). John E. Freund's Mathematical Statistics with Applications. 8th Ed., Person New International Edition.

Montgomery DC (2012). Design and analysis of experiments, 8th Ed., John Wiley & Sons, Inc.

Montgomery DC, Peck EA, Vining GG (2021). Introduction to Linear Regression Analysis, 6th Ed., Wiley.

Moore DS, McCabe GP, Craig BA (2009). Introduction to the Practice of Statistics. 6th Ed., W. H. Freeman and Company, New York.

Peck R, Olsen C, Devore JL (2016). Introduction to Statistics and Data Analysis. 5th Ed., Cengage Learning.

Pinheiro, CAR, Patetta M (2021). Introduction to Statistical and Machine Learning Methods for Data Science. Cary, NC: SAS Institute Inc.

Rinne H (2009). The Weibull Distribution A Handbook. CRC Press.

Shapiro SS, Wilk MB (1965). An Analysis of Variance Test for Normality (Complete Samples). *Biometrika*, 52(3/4), 591-611.

Stahl S (2006). The Evolution of the Normal Distribution. *Mathematics Magazine*, 76(2), pp. 96-113. Available at: https://www.maa.org/sites/default/files/pdf/upload_library/22/Allendoerfer/stahl96.pdf

Student (1908). The probable error of a mean. *Biometrika*, 6(1), 1-25.

Utts JM, Heckard RF (2007). Mind on Statistics, 3rd Ed., Thomson/Brooks Cole.

Wackerly DD, Mendenhall W, Scheaffer RL (2008). Mathematical Statistics with Applications, 7th Ed., Thomson/Brooks Cole.

Walck C (2007). Handbook on Statistical Distributions for Experimentalists. Available at: https://s3.cern.ch/inspire-prod-files-1/1ab434101d8a444500856db124098f9c

Acronyms

CDF:	Cumulative Distribution Function
MGF	Moment-generating Function
PDF:	Probability Density Function