Chapter 3

Important Probability Distributions

In this chapter we will use probability distributions to answer important biological questions about populations based on observed sample data. These questions include:

- If the CCND3 (Cyclin D3) gene expressions for all the ALL patients are normally distributed with mean 1.90 and standard deviation 0.5, what is the probability of observing expression values larger than 2.4?

- What is the probability of finding fourteen purines in a microRNA with a length of 22 nucleotides?

- If we cross two heterozygous carriers of the recessive allele for albinism, then what is the probability that 5 of the 6 F1 offspring will be albino?

In this chapter, several important distributions will be explored as tools to answer these questions. A probability distribution assigns a probability to each measurable subset of the possible outcomes of a random experiment, survey, or procedure. The particular distributions this chapter covers includes the *binomial* discrete probability distribution and the *normal*, t, F, and *chi-squared* continuous probability distributions. Combined, these distributions have a wealth of applications to statistically test many biological hypotheses. In addition, when relevant to our study, the *probability density function* (PDF), the *cumulative distribution function* (CDF), the mean μ (mu), and the standard deviation σ (sigma), are explicitly defined.

3.1 Discrete probability distributions

Discrete distributions model discrete (typically integral) data, for example the number of samples with a certain phenotype. The most common discrete distributions used to model such data include the Bernoulli, binomial, Poisson, and hypergeometric distributions. In this chapter, we will introduce the Bernoulli, binomial and Poisson distributions and cover the hypergeometric in later chapters. All of the discrete probability distributions above that we will cover can be defined by a *Probability Mass Function* (PMF) that defines masses of probability at each discrete value within the domain of the discrete probability distribution. A probability mass function $P(X)$ of a random discrete variable X must satisfy 2 important properties:

1. For all values k, the probability $P(X = k)$ is ≥ 0 (i.e. no negative probabilities are allowed).

2. For all values k, the sum of the probabilities $\sum_{i=1}^{k_i \in Domain(X)} P(X = k)$ is equal to 1.

3.1.1 Bernoulli distribution

The Bernoulli distribution is a very basic probability distribution of a random variable which takes the value 1 with success probability p and the value 0 with failure probability $q = 1 - p$. The Bernoulli distribution can be easily explained with the coin toss illustration. Suppose we toss a coin once, giving us either "Heads" or "Tails", and that we agree on counting Heads. Then the random variable $X = 1$ in case of a Heads and $X = 0$ in case of a Tails. Since we do not know whether the coin is fair, we let $P(X = 1) = p$ where p is a number between 0 and 1. By the complement rule it follows that $P(X = 0) = 1 - p = q$. The expectation of a Bernoulli distributed random variable is

$$E(X) = 0 \cdot P(X = 0) + 1 \cdot P(X = 1) = 0 \cdot (1 - p) + 1 \cdot p = p.$$

Therefore, in a betting situation where you would win one dollar, euro, or yen in the case of a Heads and nothing in case of a Tails, you would expect to win p of your favorite currency. Let X be a discrete random variable with values x_k having probabilities $P(X = x_k)$, $k = 1, \cdots, m$. The variance of a

Bernoulli distributed random variable is

$$\begin{aligned} \sigma^2 = \mathrm{Var}(X) &= \sum_{k=1}^{2}(x_k - E(X))^2 P(X = x_k) = (0-p)^2 \cdot (1-p) + (1-p)^2 \cdot p \\ &= p^2 - p^3 + p - 2p^2 + p^3 = p(1-p). \end{aligned} \tag{3.1}$$

3.1.2 Binomial distribution

The binomial distribution has many applications in medicine and bioinformatics. It models the outcomes of n repeated trials where each independent trial has a dichotomous outcome, for example success-failure, healthy-diseased, heads-tails, or purine-pyrimidine. When there are n trials, then the number of ways to obtain k successes is given by the binomial coefficient:

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

where $n! = n \cdot (n-1) \cdots 1$ and $0! = 1$ (Samuels & Witmer, 2003). The binomial probability of k successes out of n trials consists of the product of this coefficient with the probability of k successes and the probability of $n-k$ failures. Let p be the probability of success in a single trial and X the (random) variable denoting the number of successes. We say that X is binomially distributed and write $X \sim B(n,p)$. Then the probability P of the event $(B(n,p) = k)$ that k successes occur out of n trials can be expressed as:

$$P\left(B(n,p) = k\right) = \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k}, \qquad \text{for} \quad k = 0, \cdots, n. \tag{3.2}$$

The collection of these probabilities is called the binomial probability mass function (PMF). Also, for a binomially distributed random variable $X \sim B(n,p)$: the mean $\mu = np$, and variance $\sigma^2 = np(1-p)$, and the standard deviation $\sigma = \sqrt{np(1-p)}$. Lastly, also note that the Bernoulli distribution is a special case of the binomial distribution with $n = 1$.

Example 1: Teaching demonstration. Load the `TeachingDemos` package and execute the function `vis.binom()` to visualize the binomial distribution for diffent values of n and p. Note that the TeachingDemos require XQuartz from www.xquartz.org in order to run on the Mac platform. Click on "Show Normal Approximation" to observe that the normal approximation to the binomial improves as n increases, and as p approaches 0.5.

```
> library(TeachingDemos)
> vis.binom()
```

Example 2: Albinism inheritance. If we cross two heterozygous carriers of the recessive allele for albinism, then each F1 mouse has probability $1/4$ of being albino. What is the probability of exactly one F1 mouse out of three having albinism? To answer this question, we assign $n = 3$ (number of offspring), $k = 1$ (number of albino offspring), and $p = 0.25$ (probability of each F1 mouse being albino) in Equation (3.2) and obtain:

$$P\left(B(3,0.25) = 1\right) = \frac{3!}{1!(3-1)!}0.25^1 0.75^2 = 3 \cdot 0.140625 = 0.421875.$$

We can compute this in R using the `choose()` function:

```
> choose(3,1)* 0.25^1* 0.75^2   # binomial coefficient for "3 choose 1"
[1] 0.421875
```

The `choose(3,1)` (read as “3 choose 1”) above computes the binomial coefficient. It is more efficient to compute the above calculation by using the built-in binomial probability mass function `dbinom(k,n,p)` For example, below we use the `dbinom(k,n,p)` function to print all the probabilities of observing 0 through 3 albino F1 mice out of 3 total offspring:

```
> for (k in 0:3) {
+     print(dbinom(k,3,0.25))  # binomial probability mass function
+ }
[1] 0.421875
[1] 0.421875
[1] 0.140625
[1] 0.015625
```

From the output above, we see that the probability of no albino F1 mouse is 0.4218 and the probability that all three F1 mice are albino equals 0.0156.

The related `pbinom(k,n,p)` function (changing `d` into `p`) yields the cumulative distribution function (CDF) - which contains the cumulative probabilities that monotonically increase from 0 to 1.The values of the $B(3, 0.25)$ probability mass and cumulative distribution functions are summarized in Table 3.1.

Table 3.1: Binomial probability mass and cumulative distribution function values for S_3, with $p = 0.25$.

Number of successes	$k = 0$	$k = 1$	$k = 2$	$k = 3$
Probability mass $P\left(B(3, 0.25) = k\right)$	0.4218	0.4218	0.1406	0.0156
Cumulative distribution $P\left(B(3, 0.25) \leq k\right)$	0.4218	0.843	0.9843	1

Example 3: Four coin tosses. After four coin tosses, we can use the CDF of the binomial distribution $B(4, 0.5)$ to determine the probability $P\left(B(4, 0.5) \leq 3\right)$ that the number of heads is lower than or equal to three:

```
> pbinom(3,4,0.5)  # binomial cumulative distribution function (CDF)
[1] 0.9375
```

Example 4: Number of purines in a microRNA. RNA consists of a sequence composed of 4 nucleotides: A, G, U, and C. The first two (A, G) are purines and the last two (U, C) are pyrimidines. For illustration, let's suppose that the length of a certain microRNA is 22 nucleotides, that the probability of a purine equals 0.7, and that the process of placing purines and pyrimidines is binomially distributed. The event that our microRNA contains 14 purines is represented as $X \sim B(22, 0.7) = 14$. The probability of this event can be computed by

$$P\left(B(22, 0.7) = 14\right) = \frac{22!}{14!(22-14)!} 0.7^{14} 0.3^{8}$$

and calculated in R with the `dbinom()` function:

```
> dbinom(14,22,0.7)  # binomial probability mass function (PMF)
[1] 0.1422919
```

Thus, the value 0.14122929 is the value of the binomial probability mass function (PMF) at $B(22, 0.7) = 14$. Next, the probability of the event of 13 `or less` purines equals the value of the cumulative distribution function (CDF) at $B(22, 0.7) = 13$. The probability $P\left(B(22, 0.7) \leq 13\right)$ can be calculated with the `pbinom()` function:

```
> pbinom(13,22,0.7)  # binomial cumulative distribution function (CDF)
[1] 0.1864574
```

The probability of strictly more than 10 purines is:

$$\begin{aligned} P\left(B(22, 0.7) \geq 11\right) &= \sum_{k=11}^{22} P(S_{22} = k) \\ &= 1 - \sum_{k=1}^{10} P(S_{22} = k) \end{aligned}$$

and can be calculated with either the `dbinom()` or `pbinom()` function:

```
> sum(dbinom(11:22,22,0.7))  # integral over the binomial PMF
[1] 0.9859649
> 1 - pbinom(10, 22, 0.7)
[1] 0.9859649
```

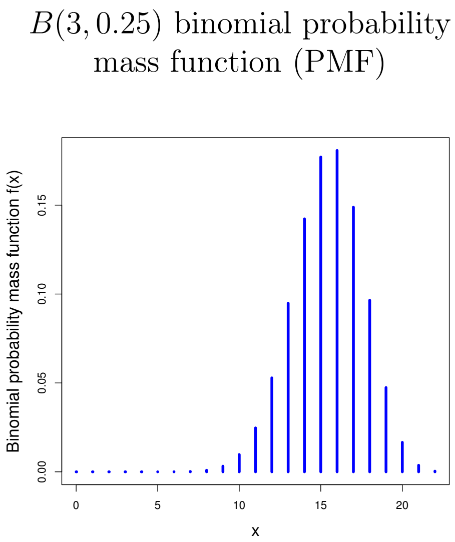

$B(3, 0.25)$ binomial probability mass function (PMF)

Figure 3.1: Binomial probabilities with $n = 22$ and $p = 0.7$

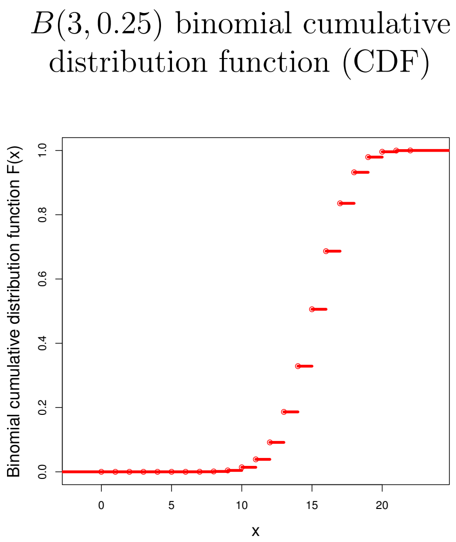

$B(3, 0.25)$ binomial cumulative distribution function (CDF)

Figure 3.2: Binomial cumulative probabilities with $n = 22$ and $p = 0.7$.

We can plot the binomial probability mass function (PMF) by using the `dbinom()` function:

```
> x <- 0:22
> plot(x,             # X-values
+      dbinom(x,size=22,prob=.7), # binomial probability mass function
+      type="h",       # histogram
+      col="blue",
+      lwd=4,          # make lines thicker
+      xlab="x",
+      ylab="Binomial probability mass function f(x)",
+      cex.lab=1.5)     # make axis labels big
```

The resultant Figure 3.1 shows that the largest probabilities occur near the expectation $E\,(B(3, 0.25)) = 15.4$.
Similarly, the binomial cumulative distribution function (CDF) can be plotted using the step function `stepfun()` in conjunction with the binomial CDF function `pbinom()`:

```
> binomialCDF = stepfun(x, c(0,pbinom(x,size=22,prob=.7))) # create a step function from
    ↪ binomial CDF
> plot(binomialCDF, # binomial cumulative function
+      col="red",
+      vertical=FALSE,
+      lwd=4,         # make lines thicker
+      xlab="x",
+      ylab="Binomial cumulative distribution function F(x)",
+      main=NULL,
+      cex.lab=1.5)    # make axis labels big)
```

The resultant Figure 3.2 illustrates that the cumulative distribution function (CDF) is an increasing step function, with x on the horizontal axis and $P\,(B(22, 0.7) \leq x)$ on the vertical. In general, the CDF is very useful for quickly ascertaining $P(X \leq x)$ for any discrete or continuous random variable X and any value x.

Lastly, we can simulate the number of purines in 1000 randomly produced microRNAs where the probability of a purine is $p = 0.7$ and the length is $n = 22$. In other words, we can produce a random sample of size 1000 from the $B(22, 0.7)$ binomial cumulative distribution. We use the `rbinom()` function to perform the sampling.

```
> rbinom(1000,22,0.7)    # random sampling from the binomial CDF
   [1] 16 16 21 18 17 18 15 16 18 16 11 16 19 12 16 17 15 15 15 14 16 12 15 12
  [25] 16 17 15 13 15 17 17 17 12 16 15 16 16 15 11 16 16 17 17 14 13 13 13 16
  [49] 17 16 18 17 17 15 15 16 20 16 21 19 21 12 11 14 17 14 14 17 10 15 14 12
  ....
```

3.1.3 Poisson Distribution

When n is large and p is small, computations involving $\binom{n}{k}$ may become practically infeasible. In such a case the binomial distribution can be approximated by the Poisson distribution. Since the values of the Poisson distribution consist of non-negative integers, it potentially provides a useful model for many phenomena in bioinformatics. Examples occur from observations of the number of occurrences of a certain event within a unit of time, space, or length. The Poisson distribution can be used to model a Poisson process whereby random events occur continuously and independently at a constant mean rate λ. The density of a Poisson distributed random variable is

$$f(x) = P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!}, \quad x = 0, 1, 2, \cdots, \quad \text{and} \quad \lambda > 0.$$

Here $0! = 1$. This gives a whole class of Poisson distributions because any distinct value of the parameter λ gives a distinct density function. To check Property 1 of a density function, note that $e^{-\lambda}$, λ^x and $x!$ are all positive. This implies that $f(x)$ is positive for all positive integers x. By adding over x the distribution function becomes

$$F(y) = \sum_{x=0}^{y} P(X = x) = \sum_{x=0}^{y} \frac{e^{-\lambda}\lambda^x}{x!}.$$

Let's check that Property 2 for densities holds. From calculus it is known that $e^\lambda = \sum_{x=0}^{\infty} \lambda^x / x!$ (e.g. Goldstein, Lay, & Schneider, 2001, p. 582), so that

$$\sum_{x=0}^{\infty} P(X = x) = \sum_{x=0}^{\infty} \frac{e^{-\lambda}\lambda^x}{x!} = e^{-\lambda} \sum_{x=0}^{\infty} \frac{\lambda^x}{x!} = e^{-\lambda} e^{\lambda} = 1.$$

To compute its expectation

$$E(X) = \sum_{x=0}^{\infty} \frac{e^{-\lambda}\lambda^x}{x!} x = e^{-\lambda}\lambda \sum_{x=1}^{\infty} \frac{\lambda^{x-1}}{(x-1)!} = \lambda e^{-\lambda} \sum_{x=0}^{\infty} \frac{\lambda^x}{x!} = \lambda e^{-\lambda} e^{\lambda} = \lambda.$$

In a similar manner it can also be shown that $\text{Var}(X) = \lambda$.

Poisson approximation to the binomial. To see how well the Poisson($\lambda < 5$) distribution can approximate the binomial distribution $B(n, p)$ as n increases and p decreases, we can plot different binomial distributions on top of the Poisson($\lambda < 5$) probability mass function (PMF). For example, Figure 3.3 illustrates how well the Poisson($\lambda = 4$) distribution can approximate the binomial distribution $B(n, p)$ for $n > 50$ and $p < 0.1$.

The Poisson($\lambda = 4$) approximation to the binomial distribution $B(n, p)$

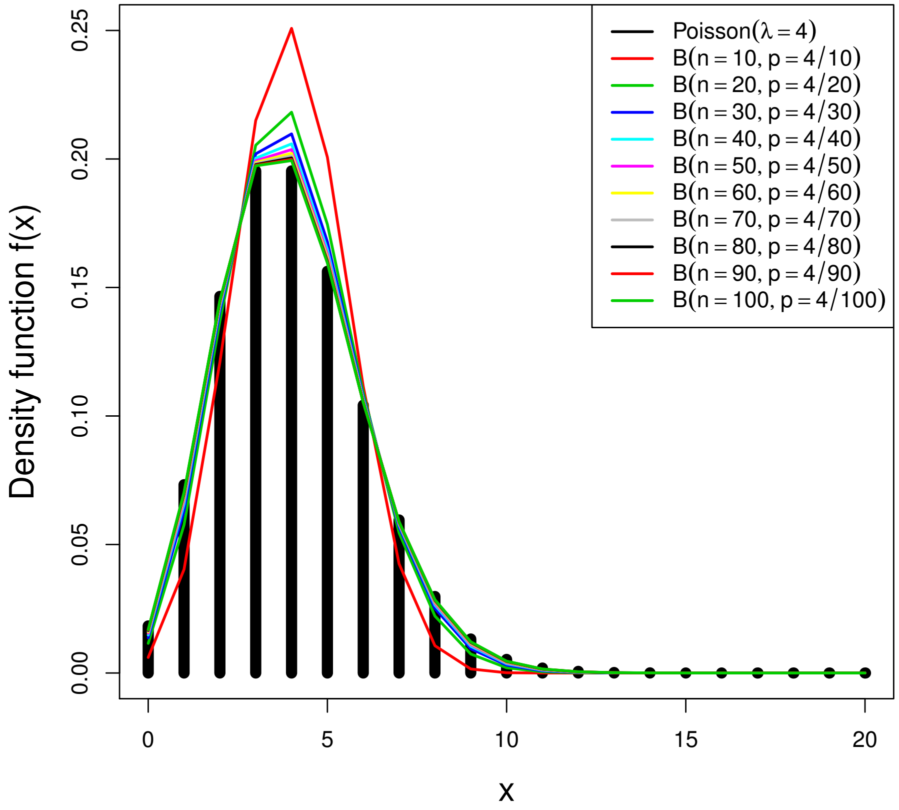


Figure 3.3: Multiple binomial distributions $B(n, p)$ with increasing n and decreasing p are plotted on top of the Poisson($\lambda = 4$) probability mass function (PMF). We can see that the Poisson($\lambda = 4$) distribution can well approximate the binomial distribution $B(n, p)$ for $n > 50$ and $p < 0.1$.

In Figure 3.3, we use the `lines()` function in a *for loop* to add 10 additional binomial curves on top of the original plot of the Poisson($\lambda = 4$) probability mass function (PMF). We also assign the PMF and each curve to a numbered color from 1 to 11. The R engine will assign each of the numbers to a different color as determined by a predefined color palette. Lastly, we use the `apply()` function in combination with the `expression()` and `bquote()` functions to create the color legend for the Poisson PMF and the 10 additional binomial curves:

```
> lambda=4
> plot(x,              # X-values
+      dpois(x,lambda),# Poisson density function
+      type="h",       # histogram
+      col=1,
+      lwd=8,          # make lines thicker
+      ylim=c(0,0.25), # range of the y-axis
+      xlab="x",
+      ylab="Density function f(x)",
+      cex.lab=1.5)    # make axis labels big
> for (counter in 1:10) {
+     lines(x, dbinom(x, size=counter*10, prob=lambda/(counter*10)), col=counter+1, lwd=2)
+ }
> legend("topright", lty=rep(1,11), lwd=rep(2,11), col=1:11,
+        legend=c(expression(Poisson(lambda == 4)),
+              as.expression(sapply(seq(10,100, 10), function(x) bquote(B(n ==.(x), p == 4/.(
    ↪ x)))))))
```

Example 1: Teaching demonstration. To visualize that the Poisson distribution converges to that of the binomial, load the package `TeachingDemos` and give the command `vis.binom()`. Click on "Show Poisson Approximation" and adjust the parameters of the slider box.

```
> library(TeachingDemos)
> vis.binom()
```

Example 2: Daily lottery. The daily lottery is an example of the Poisson distribution in action. Assume that the probability to win is about 1 in 40 million and the number n of sold tickets is a large 80 million. Then $\lambda = np = 80 \cdot 10^6/(40 \cdot 10^6) = 2$. Hence, the probability of no winners is

$$P(X = 0) = \frac{e^{-2}2^0}{0!} = 0.1353$$

and the probability of one winner is

$$P(X = 1) = \frac{e^{-2}2^1}{1!} = 0.2706.$$

We can compute these values directly in R. However, it is more efficient to use the built-in function `dpois()` with input (x, λ):

```
> exp(-2)*2^0/factorial(0)
[1] 0.1353353
> dpois(0,2)
[1] 0.1353353
> exp(-2)*2^1/factorial(1)
[1] 0.2706706
> dpois(1,2)
[1] 0.2706706
```

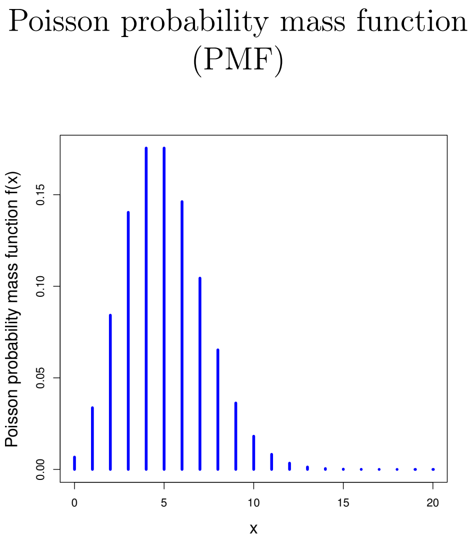


Figure 3.4: Poisson probabilities with $\lambda = 5$.

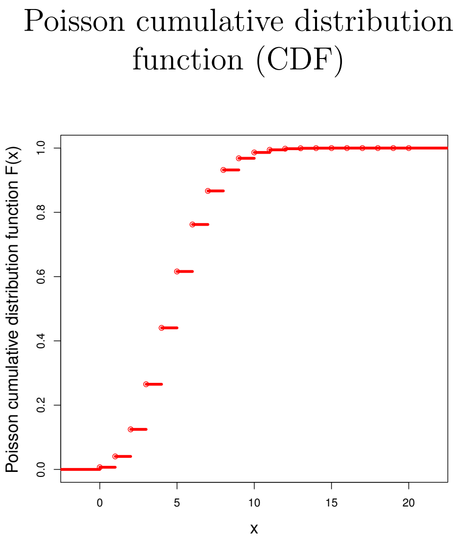


Figure 3.5: Poisson cumulative probabilities with $\lambda = 5$.

We can plot the Poisson probability mass function (PMF) for $\lambda = 5$ by using the `dpois()` function:

```
> x <- 0:20
> plot(x,           # X-values
+      dpois(x,5),  # Poisson mass function
+      type="h",    # histogram
+      col="blue",
+      lwd=4,        # make lines thicker
+      xlab="x",
+      ylab="Poisson probability mass function f(x)",
+      cex.lab=1.5)    # make axis labels big
```

The resultant Figure 3.4 shows that the largest probabilities occur near the expectation $E(X) = 5$ and that the distribution is also quite symmetric around $E(X) = 5$. Similarly, the Poisson cumulative distribution function

(CDF) can be plotted using the step function `stepfun()` in conjunction with the Poisson CDF function `ppois()`:

```
> poissonCDF = stepfun(x, c(0,ppois(x,5))) # create a step function from poisson CDF
> plot(poissonCDF, # poisson cumulative function
+      col="red",
+      vertical=FALSE,
+      lwd=4,         # make lines thicker
+      xlab="x",
+      ylab="Poisson cumulative distribution function F(x)",
+      main=NULL,
+      cex.lab=1.5)    # make axis labels big)
```

The resultant Figure 3.5 illustrates that the cumulative distribution function (CDF) is an increasing step function, with x on the horizontal axis and $P\left(Poisson(5) \leq x\right)$ on the vertical.

Example 3: Finding k-mers. Let's compute the probability of observing the k-mer or "word" CAGTAC consisting of $k = 6$ letters (nucleic acids) in a large DNA sequence of length 10^7 nucleotides. If the probability of each letter equals 0.25, then the probability of observing the 6-mer CAGTAC is $0.25^6 = 2.44 \cdot 10^{-6} = p$. Suppose we only look at consecutive, non-overlapping 6-mers of length 6 in the reading frame whereby the first 6-mer starts at the beginning of the sequence. In this 6-mer reading frame, the DNA sequence contains $10^7/6 = 1.6 \cdot 10^6 = n$ consecutive, non-overlapping 6-mers. And the number of CAGTACs we expect to find is $\lambda = np = 2.44 \cdot 1.6 = 3.9$. Hence, the probability to find no CAGTAC 6-mer in this reading frame is

$$P(X = 0) = \frac{e^{-3.9} \cdot 3.9^0}{0!} = e^{-3.9} = 0.020.$$

Again, we can compute this value directly in R or use the built-in function `dpois()` with input (x, λ):

```
> exp(-3.9)*2^0/factorial(0)
[1] 0.02024191
> dpois(0,3.9)
[1] 0.02024191
```

And by the complement rule the probability of finding at least one CAGTAC 6-mer is $1 - 0.020 = 0.980$.

```
> 1 - dpois(0,3.9)
[1] 0.9797581
```

Example 4: Gaps in an open reading frame. Suppose that, within a certain window length on an assembly of a DNA sequence, the mean number of random gaps within an open reading frame is $\lambda = 5$. We can compute the probability of two gaps $P(X = 2) = e^{-5}5^2/2!$ by direct calculation:

```
> exp(-5)*5^2/factorial(2)
[1] 0.08422434
```

where factorial$(2) = 2!$ However, it is more efficient to use the built-in function `dpois()` with input (x, λ):

$$P(X = 2) = \texttt{dpois}(2, 5) = 0.08422434.$$

```
> dpois(2,5)
[1] 0.08422434
```

To compute the probability of strictly less than 4 gaps, we use the built-in function `ppois()` as follows:

$$F(3) = P(X \leq 3) = \sum_{x=0}^{3} P(X = x) = \sum_{x=0}^{3} \frac{e^{-5}5^x}{x!} = \texttt{ppois}(3, 5) = 0.2650259.$$

```
> ppois(3,5)
[1] 0.2650259
```

Since $\lambda = 5$, it follows that $E(X) = \mathrm{Var}(X) = 5$. To verify this by random sampling, a random sample of size 1000 can be drawn from the Poisson distribution with $\lambda = 5$ by the command `rpois(1000,5)`. Then computing the sample mean and variance reveals that these are close to their population counterparts:

```
> y <- rpois(1000,5)   # 1000 random samples with lambda=5
> mean(y)
[1] 5.081
> var(y)
[1] 5.555995
```

We can also compute the quantiles $x_{0.025}$ and $x_{0.975}$ which are the x-values for which the $P(X \leq x) = 0.025$ and $P(X \leq x) = 0.975$, respectively. Thus, in R the quantile function is the inverse of the cumulative function. We can compute these quantiles of the Poisson distribution by using the built-in function `qpois()` as follows:

```
> qpois(c(0.025,0.975), lambda=5, lower.tail = TRUE)
[1]  1 10
```

As illustrated in Figure 3.5 the distribution function is a step function, so that there is no exact solution for the quantiles. To check this use `sum(dpois(1:10,5))`:

```
> sum(dpois(1:10,5))
[1] 0.9795668
```

3.2 Continuous probability distributions

Continous distributions are used to model real-valued (decimal) data, for example gene expression fold-change or protein concentration. The most common continuous distributions used to model biological data include the *exponential*, *normal*, *t*, *F*, and *chi-squared* distributions. In this chapter, we will introduce all five distributions and describe the important characteristics of each. Analogous to probability mass functions (PMFs), the five continuous probability distributions above can be defined by a "Probability Density Function" (PDF) that defines the density of the probability of random variable X between any two real values. A probability density function $f(x)$ of a random continuous variable X must satisfy 2 important properties:

1. For all real values x between x_{min} and x_{max}, the density $f(x)$ is > 0 In other words, only positive densities are allowed over all real values within the domain of X where $x_{min} \leq x \leq x_{max}$.

2. For all real values x between x_{min} and x_{max}, the infinite sum of all their probabilities $P(x_{min} \leq X \leq x_{max}) = \int_{x_{min}}^{x_{max}} f(x)dx$ is equal to 1. Graphically, we say that the area under the graphed curve $f(x)$, over the interval from x_{min} to x_{max}, is equal to one.

The expression $\int_{x_{min}}^{x_{max}} f(x)dx$ is called the definite integral of $f(x)$ with respect to x over the interval (x_{min}, x_{max}), and it represents the area under the function $f(x)$ in that interval (see Appendix).

In addition, the antiderivative $F(x)$ of $f(x)$ is called the cumulative distribution function (CDF) in the context of probability and statistics. In statistics, the antiderivative $F(x)$ gives the probability $P(X \leq x)$ that the random variable X takes a value less than or equal to x. When the antiderivative $F(x)$ is known for a given density function $f(x)$, it is very useful for calculating the probability $P(x_1 \leq X \leq x_2)$ that the random variable X takes a value within the interval (x_1, x_2):

$$P(x_1 \leq X \leq x_2) = \int_{x_1}^{x_2} f(x)dx = [F(x)]_{x_1}^{x_2} = F(x_2) - F(x_1)$$

Fortunately, R knows the antiderivative $F(x)$ for many probability density functions in order to allow for fast probability calculations.

3.2.1 Exponential distribution

The number of occurrences of a certain phenomenon within a fixed time interval is Poisson distributed with mean λ, depending on the length of the interval. The probability of the waiting time being greater than t is $P(X > t) = e^{-\lambda t}$. This formula is closely related to the density function of the exponential distribution which is $f(x) = \lambda e^{-\lambda x}$, where λ is a positive number. For this density function it is straightforward to check Property 1 and 2. In particular, an exponential function with a positive base has positive function values (see Appendix). Hence, we have that $0 < e^{-\lambda x}$ for all real x and Property 1 holds. To check Property 2, we compute

$$\int_0^\infty f(x)dx = \int_0^\infty \lambda e^{-\lambda x}dx = \left[-e^{-\lambda x}\right]_0^\infty = \lim_{x\to\infty} -e^{-\lambda x} - (-e^{-\lambda \cdot 0}) = 0-(-1) = 1.$$

Hence, for any positive λ it holds that $f(x)$ is a density function of a random variable X. Since this holds for each distinct positive value of λ, a whole family of exponentially distributed random variables is defined. The formulation of the cumulative distribution function $F(x)$ is

$$F(x) = \int_0^x f(y)dy = \int_0^x \lambda e^{-\lambda y}dy = \left[-e^{-\lambda y}\right]_0^x = -e^{-\lambda x} - (-e^0) = 1 - e^{-\lambda x}.$$

Next, we compute the expectation $E(X)$ of an exponentially distributed random variable X:

$$E(X) = \int_0^\infty x \cdot f(x)dx = \int_0^\infty \lambda x e^{-\lambda x}dx = \frac{1}{\lambda},$$

where the last equality follows from integration by parts (see Appendix). By the same integration technique it can also be shown that the variance $\text{Var}[X] = 1/\lambda^2$.

Example 1: Suppose that $\lambda = 5$. The probability $P(X \leq 0.5)$ that X is less than or equal to 0.5 is:

$$\begin{aligned} P(X \leq 0.5) &= \int_0^{0.5} f(x)dx = \int_0^{0.5} 5e^{-5x}dx = \left[-e^{-5x}\right]_0^{0.5} \\ &= -e^{-5\cdot 0.5} - (-e^{-5\cdot 0}) = -0.082085 + 1 = 0.917915. \end{aligned}$$

Graphically, the probability $P(X \leq 0.5)$ is equal to the magenta area under the curve of the density function $f(x)$ over the interval between 0 and 0.5 in Figure 3.6. Also, figure 3.7 illustrates how easy it is to retrieve the value $P(X \leq 0.5) = 0.917915$ from the cumulative distribution function. The value of $P(X \leq 0.5)$ can be computed in R by using `pexp(.5,5)`:

```
> pexp(.5,5)
[1] 0.917915
```

Exponential probability density function (PDF)

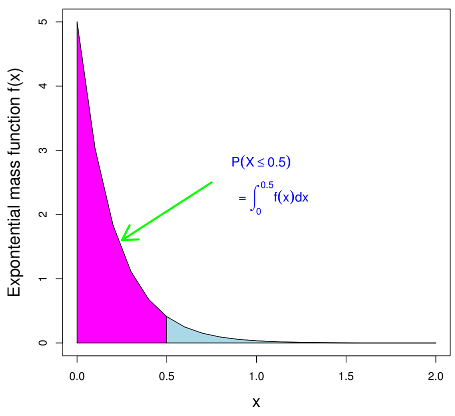


Figure 3.6: Graph of the exponential probability density function (PDF) with Poisson mean ($\lambda = 5$).

Exponential cumulative distribution function (CDF)

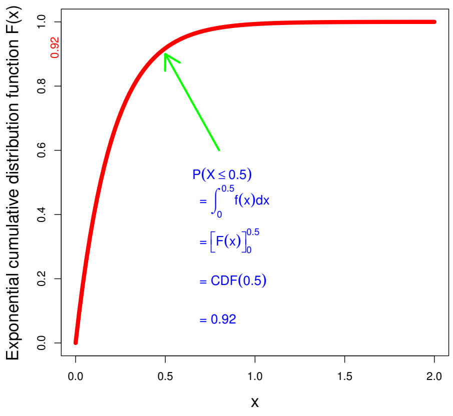


Figure 3.7: Graph of the exponential cumulative distribution function (CDF) with Poisson mean ($\lambda = 5$).

We can plot Figure 3.6 by using `dexp()` with the `polygon()`, `arrows()`, and `text()` functions:

```
> f <-function(x) { dexp(x,5) }
> plot(f,  # function to plot
+      0,  # first x-value
+      2,  # last x-value
+      xlab="x",
+      ylab="Expontential mass function f(x)",
+      cex.lab=1.5)  # make axis labels bigger
> x1 <- seq(0,0.5,.1) # set of magenta x-values
> x2 <- seq(0.5,2,.1) # set of blue x-values
> polygon(c(0,x1,.5), c(0,f(x1),0), col="magenta")
> polygon(c(.5,x2,2), c(0,f(x2),0), col="lightblue")
> arrows(0.75,2.5,0.25,1.6, lwd=3, col="green")
> text(0.86, 2.7 - c(0,0.7), cex = 1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 0.5)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 0.5)))))
```

Similarly, we can plot Figure 3.7 by using the `pexp()` instead of the `dexp()` function:

```
> F <- function(x) { pexp(x,5) }
> plot(F,             # function
+      0,             # start x
+      2,             # end x
+      cex.lab=1.5,   # make axis labels big
+      col="red",
+      lwd=6,         # make line thicker
+      xlab="x",
+      ylab="Exponential cumulative distribution function F(x)")
> mtext("0.92",side=2,at=0.92, col="red")
> arrows(0.8,0.6,0.5,0.9, lwd=3, col="green")
> text(0.65, 0.5 - c(0,.11,.22,.33,.44), cex=1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 0.5)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 0.5))),
+        expression(paste("  = ", bgroup("[", F(x) ,"]")[0]^0.5)),
+        expression(paste("  = ", CDF(0.5))),
+        expression(paste("  = ", 0.92))))
```

The probability that X is larger than 2 is:

$$P(X \geq 2) = 1 - P(X \leq 2) = 1 - \texttt{pexp}(2,5) = 0.000453.$$

```
> 1-pexp(2,5)
[1] 4.539993e-05
```

The probability that X is between 0.25 and 2 is equal to:

$$\begin{aligned} P(0.25 \leq X \leq 2) &= \int_{.25}^{2} f(x)dx \\ &= \int_{0}^{2} f(x)dx - \int_{0}^{0.25} f(x)dx \\ &= \texttt{pexp}(2,5) - \texttt{pexp}(.25,5) = 0.2864594. \end{aligned}$$

```
> pexp(2,5)-pexp(.25,5)
[1] 0.2864594
```

To illustrate that the exponential distribution function is strictly increasing, the graph of its cumulative distribution function is plotted, see Figure 3.7. The exact values for the quantiles $x_{0.025}$ and $x_{0.975}$ can be computed by the following code:

```
> qexp(c(0.025,0.975), rate = 5, lower.tail = TRUE, log.p = FALSE)
[1] 0.005063562 0.737775891
> pexp(0.737775891,5)-pexp(0.005063562,5)
[1] 0.95
```

Above, it is verified that 95% of the values of the distribution are between $x_{0.025} = 0.005063562$ and $x_{0.975} = 0.737775891$, that is $P(x_{0.025} \leq X \leq x_{0.975}) = 0.95$.

3.2.2 Normal distribution

The normal distribution is widely used to model many types of biological data (and other phenomena). For example, the normal distribution is commonly used to model (preprocessed) gene expression values. That is, the data values $x_1, \cdots, x_n$ are often modeled as a relatively small sample that is randomly selected from a much larger normally distributed population. Equivalently, we can say that the data values are members of a normally distributed population with mean μ (mu) and variance σ^2 (sigma squared). Typically, Greek letters are used to signify the population properties and $\mathcal{N}(\mu, \sigma^2)$ is used to uniquely signify the normal population distribution with mean μ and variance σ^2. Additionally, usually the letters $\overline{x}$ and s are used to signify the sample mean and sample standard deviation, respectively.

The equation for the normal distribution $\mathcal{N}(\mu, \sigma^2)$ is:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2 / 2\sigma^2}$$

Various properties of the normal distribution are illustrated by the examples below.

Example 1: Random sampling. When population X is distributed as $\mathcal{N}(1.90, 0.5^2)$, then the population mean is 1.9 and the population standard deviation is 0.5. If we randomly sample many times from this population, then the resultant sample distribution should have the same normal distribution as the underlying population. To verify this we draw a random sample of size 1000 from the population using the `rnorm()` function and then calculate the `mean()` and `sd()` of our sample:

```
> x <- rnorm(1000,1.9,0.5)  # 1000 random samples with mean=1.9 and sd=0.5
> mean(x)
[1] 1.915186
> sd(x)
[1] 0.4918254
```

Indeed, the sample mean $\overline{x} = 1.9152$ and sample standard deviation $s = 0.4918$ are close to their population values $\mu = 1.9$ and $\sigma = 0.5$.
[1]

Example 2: Teaching demonstration. Load the TeachingDemos package and execute the command vis.normal() to view members of the normal distribution family in an interactive display.

```
> library(TeachingDemos)
> vis.binom()
```

These bell-shaped curves are called normal probability densities and all share the common characterisitic that the area under their curves equal 1. The curves are symmetric around the mean μ and attain a unique maximum at $x = \mu$. As values of x move away from the mean μ, the curves asymptotically approach the $f(x) = 0$ horizontal axis indicating that extreme values occur with small probability. In the interactive display, move the Mean and the Standard Deviation from the left to the right to explore their effects on the shape of the normal distribution. In particular, when the mean μ increases, then the distribution shifts to the right. When the σ is small or large, then the distribution scales to be either steep or flat, respectively.

Example 3: Theoretical gene expression. Suppose that the CCND3 (Cyclin D3) gene expression values, reprented as X, is distributed as $\mathcal{N}(1.90, 0.5^2)$. Then we can write $X \sim \mathcal{N}(1.90, 0.5^2)$. From the $\mathcal{N}(1.90, 0.5^2)$ probability density function in Figure 3.8, we see that the PDF is symmetric and bell-shaped around the mean $\mu = 1.90$.

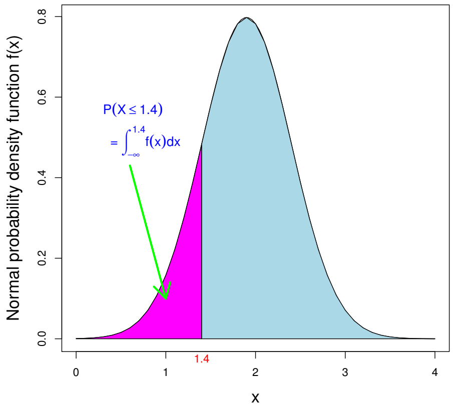
The probability that the expression values are less then 1.4 is written as $P(\mathcal{N}(1.90, 0.5^2) < 1.4)$ and can be calculated with the pnorm() function:

```
> pnorm(1.4,1.9,0.5)     # left-side tail of the Normal cumulative density function (CDF)
[1] 0.1586553
```

Note that it may help to visualize the probability density function (PDF) as a histogram with arbitrarily small bars (intervals) in the ideal case of infinite datapoints. In other words, as our sample size increases the resultant histogram should increasely match the probability density function of the underlying population.

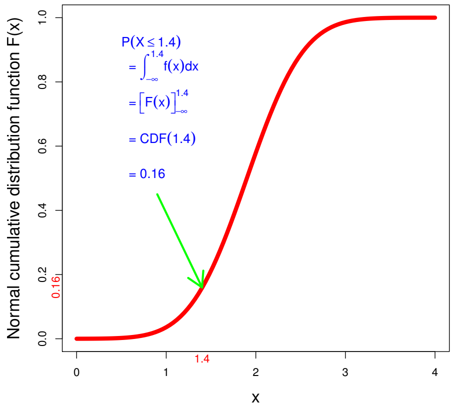
The value of the cumulative distribution function is given by $P\left(\mathcal{N}(1.90, 0.5^2) \leq x\right)$ which represents the probability of the population to have values smaller than or equal to x. Figure 3.9 illustrates how easy it is to retrieve the

[1] Use the function round to print the mean in a desired number of decimal places.



$\mathcal{N}(1.90, 0.5^2)$ normal probability density function (PDF)

Figure 3.8: Graph of the normal probability density function (PDF) with mean 1.9 and standard deviation 0.5.



$\mathcal{N}(1.90, 0.5^2)$ normal cumulative distribution function (CDF)

Figure 3.9: Graph of the normal cumulative distribution function (CDF) with mean 1.9 and standard deviation 0.5.

value $P\left(\mathcal{N}(1.90, 0.5^2) \leq 1.4\right) = 0.16$ from the cumulative distribution function. The $P\left(\mathcal{N}(1.90, 0.5^2) \leq 1.4\right)$ also corresponds to the magenta area under the probability density function (PDF) curve in Figure 3.8. We can plot Figure 3.8 by using `dnorm()` with the `polygon()`, `arrows()`, `mtext()`, `expression()`, and `text()` functions:

```
> dnormFun <- function(x) { dnorm(x,1.9,0.5) }
> x1<- seq(0,1.4,0.1)
> x2<- seq(1.4,4,0.1)
> plot(dnormFun,  # function
+      0,             # start
+      4,             # end
+      cex.lab=1.5,   # make axis labels big
+      xlab="x",
+      ylab="Normal probability density function f(x)")
> polygon(c(0.0,x1,1.4), c(0,dnormFun(x1),0), col="magenta")
> polygon(c(1.4,x2,4.0), c(0,dnormFun(x2),0), col="lightblue")
> mtext("1.4",side=1,at=1.4, col="red")
> arrows(0.6,0.43,1.0,0.1, lwd=3, col="green")
> text(0.3, 0.55 - c(0,.10), cex = 1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 1.4)),
+        expression(paste("  = ", integral(f(x) * dx, -infinity, 1.4)))))
```

Similarly, we can plot Figure 3.9 by using the `pnorm()` instead of the `dnorm()` function:

```
> pnormFun <- function(x) { pnorm(x,1.9,0.5) }
> plot(pnormFun,      # function
+      0,             # start
+      4,             # end
+      cex.lab=1.5,   # make axis labels big
+      col="red",
+      lwd=6,         # make line thicker
+      xlab="x",
+      ylab="Normal cumulative distribution function F(x)")
> mtext("1.4",side=1,at=1.4, col="red")
> mtext("0.16",side=2,at=0.16, col="red")
> arrows(0.9,0.45,1.4,0.16, lwd=3, col="green")
> text(0.5, 0.9 - c(0,.1,.2,.3,.4), cex=1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 1.4)),
+        expression(paste("  = ", integral(f(x) * dx, -infinity, 1.4))),
+        expression(paste("  = ", bgroup("[", F(x) ,"]")[-infinity]^1.4)),
+        expression(paste("  = ", CDF(1.4))),
+        expression(paste("  = ", 0.16))))
```

The probability that the expression values are larger than 2.4 is $P\left(\mathcal{N}(1.90, 0.5^2) \geq 2.4\right)$ and can be calculated with the `pnorm()` function:

```
> 1-pnorm(2.4,1.9,0.5)    # right-side tail of normal cumulative density function (CDF)
[1] 0.1586553
```

The probability that $X \sim \mathcal{N}(1.90, 0.5^2)$ is between 1.4 and 2.4 equals $P(1.4 \leq \mathcal{N}(1.90, 0.5^2) \leq 2.4)$ which can be calculated as the difference between two areas using the `pnorm()` function:

```
> pnorm(2.4,1.9,0.5)-pnorm(1.4,1.9,0.5)   # central area of the normal cumulative density
    ↪ function (CDF)
[1] 0.6826895
```

Figure 3.9 illustrates that the cumulative distribution function CDF is strictly increasing.

The exact value for the quantile $x_{0.025}$, which are the x-value for which the $P(\mathcal{N}(1.90, 0.5^2) \leq x) = 0.025$, of the $\mathcal{N}(1.90, 0.5^2)$ distribution can be computed using the `qnorm()` function:

```
> qnorm(0.025,1.9,0.5)   # Normal quantile function
[1] 0.920018
```

Hence, it holds that the probability of values smaller than 0.920018 equals 0.025, that is $P(\mathcal{N}(1.90, 0.5^2) \leq 0.920018) = 0.025$, as can be verified with the `pnorm()` function:

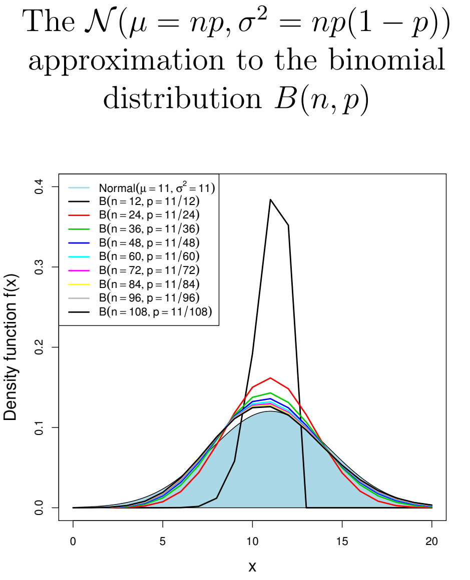
```
> pnorm(0.920018,1.9,0.5)    # left-side tail of the Normal cumulative density function (
    ↪ CDF)
[1] 0.025
```

For any X distributed as $\mathcal{N}(\mu, \sigma^2)$, it holds that $(X - \mu)/\sigma = Z$ is distributed as $\mathcal{N}(0, 1)$. Thus, by subtracting μ and dividing the result with σ

any normally distributed variable can be *standardized* into a standard normally distributed Z having mean zero and standard deviation one.

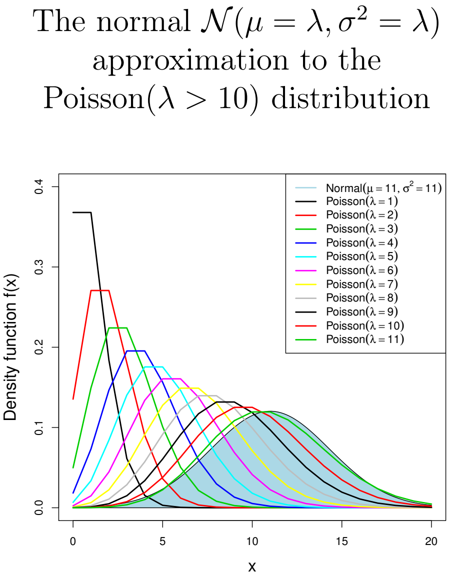
Normal approximation to the binomial and Poisson. The continuous normal distribution $\mathcal{N}(\mu, \sigma^2)$ can be used to approximate the discrete binomial and Poisson distributions in certain parameter regimes. To see how well the normal distribution $\mathcal{N}(\mu = np, \sigma^2 = np(1-p))$ can approximate the binomial distribution $B(n, p)$ as n increases and p decreases, we can plot different binomial distributions on top of the normal $\mathcal{N}(\mu = np, \sigma^2 = np(1-p))$ probability density function (PDF). For example, Figure 3.10 illustrates how well the normal distribution $\mathcal{N}(\mu, \sigma^2)$ can approximate the binomial distribution $B(n, p)$ for $n > 50$ and $np > 10$.

Likewise, to see how well the normal distribution $\mathcal{N}(\mu = \lambda, \sigma^2 = \lambda)$ can approximate the Poisson(λ) distribution as λ increases, we can plot different Poisson distributions on top of the $\mathcal{N}(\mu = \lambda, \sigma^2 = \lambda)$ probability density function (PDF). For example, Figure 3.11 illustrates how well the normal distribution $\mathcal{N}(\mu = \lambda, \sigma^2 = \lambda)$ can approximate the Poisson(λ) distribution for $\lambda > 10$.



The $\mathcal{N}(\mu = np, \sigma^2 = np(1-p))$ approximation to the binomial distribution $B(n,p)$

Figure 3.10: Multiple binomial distributions $B(n,p)$ with increasing n and decreasing p are plotted on top of the normal $\mathcal{N}(\mu = np, \sigma^2 = np(1-p))$ probability density function (PDF). We can see that the normal $\mathcal{N}(\mu = np, \sigma^2 = np(1-p))$ distribution can well approximate the binomial distribution $B(n,p)$ for $n > 50$ and $np > 10$.



The normal $\mathcal{N}(\mu = \lambda, \sigma^2 = \lambda)$ approximation to the Poisson($\lambda > 10$) distribution

Figure 3.11: Multiple Poisson(λ) distributions with increasing λ are plotted on top of the normal $\mathcal{N}(\mu = 11, \sigma^2 = 11)$ probability density function (PDF). We can see that the normal $\mathcal{N}(\mu = \lambda, \sigma^2 = \lambda)$ distribution can well approximate the Poisson(λ) distribution for $\lambda > 10$.

Similar to Figure 3.3, in Figures 3.10 and 3.11 we use the `lines()` function in a *for loop* to add N additional curves on top of the original plot of the normal $\mathcal{N}(\mu, \sigma^2)$ probability density function (PDF). We also assign the PDF and N curves to a numbered color from 1 to $N+1$. The R engine will assign each of the numbers to a different color as determined by a predefined color palette. Lastly, we use the `apply()` function in combination with the `expression()` and `bquote()` functions to create the color legend for the normal PDF and the N additional curves:

```
> x <- 0:20
> np=11
> xPolygon<- seq(0,20,0.1)
> dnormFun <- function(x) { dnorm(x,np,sqrt(np)) }
> plot(dnormFun,       # normal density function
+      0,              # start
+      20,             # end
+      col="lightblue",
+      ylim=c(0,0.4), # range of the y-axis
+      xlab="x",
+      ylab="Density function f(x)",
+      cex.lab=1.5)   # make axis labels big
> polygon(c(0.0,xPolygon,20), c(0,dnormFun(xPolygon),0), col="lightblue")
> for (counter in 1:9) {
+     lines(x, dbinom(x, size=counter*12, prob=np/(counter*12)), col=counter, lwd=2)
+ }
> legend("topleft", lty=rep(1,10), lwd=rep(2,10), col=c("lightblue",1:9),
+        legend=c(expression(Normal(mu == 11, sigma^2 == 11)),
+            as.expression(sapply(seq(12,108, 12), function(x) bquote(B(n ==.(x), p == 11/
    ↪ .(x))))))
```

```
> x <- 0:20
> xPolygon<- seq(0,20,0.1)
> dnormFun <- function(x) { dnorm(x,11,sqrt(11)) }
> plot(dnormFun,      # normal density function
+      0,             # start
+      20,            # end
+      col="lightblue",
+      ylim=c(0,0.4), # range of the y-axis
+      xlab="x",
+      ylab="Density function f(x)",
+      cex.lab=1.5)   # make axis labels big
> polygon(c(0.0,xPolygon,20), c(0,dnormFun(xPolygon),0), col="lightblue")
> for (counter in 1:11) {
+     lines(x, dpois(x,counter), col=counter, lwd=2)
+ }
> legend("topright", lty=rep(1,12), lwd=rep(2,12), col=c("lightblue",1:11),
+        legend=c(expression(Normal(mu == 11, sigma^2 == 11)),
+            as.expression(sapply(1:11, function(x) bquote(Poisson(lambda ==.(x)))))))
```

3.2.3 Chi-squared distribution

The chi-squared distribution plays an important role in testing hypotheses about frequencies. Let $\{Z_1, \cdots, Z_m\}$ be independent and standard normally distributed random variables. Then the sum of their squares

$$\chi^2_m = Z_1^2 + \cdots + Z_m^2 = \sum_{i=1}^{m} Z_i^2,$$

is the so-called chi-squared distributed (random) variable with m degrees of freedom.

Example 1: Teaching demonstration. Load the `TeachingDemos` package to view various members of the χ^2 distribution. Execute the command `vis.gamma()` to open an interactive display of various distributions. For different distribution examples, click on "Visualizing the gamma", "Visualizing the Chi-squared", and adapt "Xmax". Also, moving the "Shape" button to the right will increase the degrees of freedom. Observe that the graphs of chi-squared densities change from heavily skewed to the right into a more bell-shaped, normal distribution as the degrees of freedom increases.

```
> library(TeachingDemos)
> vis.gamma()
```

χ^2_5 probability density function (PDF)

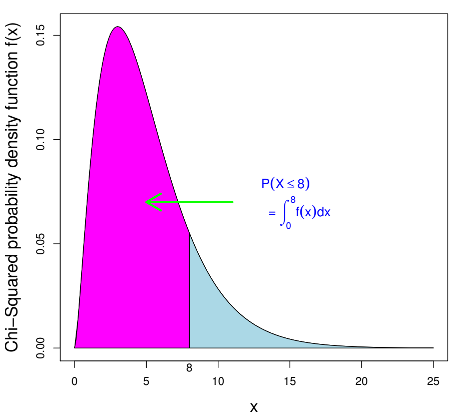


Figure 3.12: Graph of the χ^2_5 probability density function (PDF) with 5 degrees of freedom.

χ^2_5 cumulative distribution function (CDF)

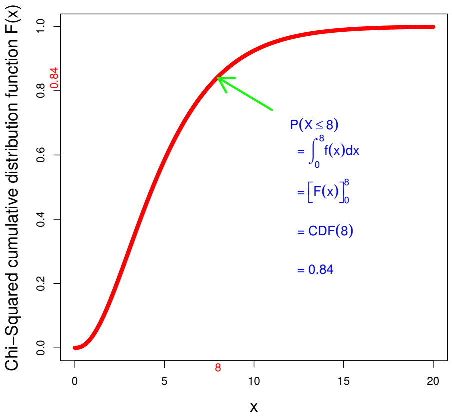


Figure 3.13: Graph of the χ^2_5 cumulative distribution function (CDF) with 5 degrees of freedom.

Example 2: Five degrees of freedom. Let's consider the chi-squared variable with 5 degrees of freedom; $\chi^2_5 = Z_1^2 + \cdots + Z_5^2$. To compute the probability $P\left(\chi^2_5 \leq 8\right)$ of observing values smaller than eight, we use the function `pchisq()` as follows:

```
> pchisq(8,5)    # left-side tail of the Chi-squared cumulative density function (CDF)
[1] 0.8437644
```

This yields the value of the cumulative distribution function (CDF) at $x = 8$ (see Figure 3.13). This value corresponds to the magenta area below the probability density function (PDF) curve in Figure 3.12. We can plot Figure 3.12 by using the `dchisq()` function:

```
> dchisqFun<-function(x) { dchisq(x,5) }
> plot(dchisqFun,     # function
+      0,              # start
+      25,             # end
+      cex.lab=1.5,    # make axis labels big
+      xlab="x",
+      ylab="Chi-Squared probability density function f(x)")
> x1 <- seq(0,8,0.1)
> x2 <- seq(8,25,0.1)
> polygon(c(0,x1,8),  c(0,dchisqFun(x1),0), col="magenta")
> polygon(c(8,x2,25), c(0,dchisqFun(x2),0), col="lightblue")
> mtext("8",side=1,at=8)
> arrows(11,0.07,5,0.07, lwd=3, col="green")
> text(13, 0.075 - c(0,.018), cex = 1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 8)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 8)))))
```

Similarly, we can plot Figure 3.13 by using the `pchisq()` function:

```
> pchisqFun<-function(x) { pchisq(x,5) }
> plot(pchisqFun,     # function
+      0,              # start
+      20,             # end
+      cex.lab=1.5,    # make axis labels big
+      col="red",
+      lwd=6,          # make line thicker
+      xlab="x",
+      ylab="Chi-Squared cumulative distribution function F(x)")
> mtext("8",   side=1,at=8, col="red")
> mtext("0.84",side=2,at=0.84, col="red")
> arrows(11,0.74,8,0.84, lwd=3, col="green")
> text(12, 0.67 - c(0,.11,.22,.33,.44), cex=1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 8)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 8))),
+        expression(paste("  = ", bgroup("[", F(x) ,"]")[0]^8)),
+        expression(paste("  = ", CDF(8))),
+        expression(paste("  = ", 0.84))))
```

Often, we are interested in the value for the quantile $x_{0.025}$, where $P(\chi^2_5 \leq x_{0.025}) = 0.025$. [2] The quantile $x_{0.025}$ can be computed by using the `qchsq()` function:

```
> qchisq(0.025, 5, lower.tail=TRUE)   # Chi-squared quantile function
[1] 0.8312116
```

Example 3: Goodness of fit. The chi-squared distribution is frequently used as a so-called "goodness of fit" measure. For example, say that someone has hypothesized that the expression values of the CCND3 (Cyclin D3) gene for ALL patients are distributed as $\mathcal{N}(1.90, 0.50^2)$. If that hypothesis is true, then the probability of observing values greater than the sum of the squared, standardized, observed values should be about a 1/2, that is

[2] If the cumulative distribution is strictly increasing, then there exists an exact and unique solution for the quantiles.

$P\left(\chi^2_{27} \geq \sum_1^{27} z_i^2\right) \approx 0.5$. We can use the Golub et al. data to test this hypothesis. Let $x_1, \cdots, x_{27}$ be the observed gene expression values for CCND3 (Cyclin D3). Then the standardized values are $z_i = (x_i - 1.90)/0.50$ and their sum of squares is $\sum_1^{27} z_i^2 = 25.03312$. The probability of observing larger values is $P\left(\chi^2_{27} \geq 25.03312\right) = 0.5726 \approx 0.5$, which indicates that this normal distribution fits the data well. Hence, it is likely that the specified normal distribution is indeed correct. We can make these calculations by using the `pchisq()` function:

```
> library(multtest); data(golub)
> golubFactor <- factor(golub.cl,levels=0:1, labels= c("ALL","AML"))
> ccnd3 = grep("CCND3",golub.gnames[ ,2], ignore.case = TRUE)
> ccnd3
[1] 1042
> x <- golub[ccnd3,golubFactor=="ALL"]
> z <- (x-1.90)/0.50
> sum(z^2)
[1] 25.03312
> pchisq(sum(z^2),27, lower.tail=FALSE)
[1] 0.5726059
```

In the next chapter, we will explore hypothesis tests that provide a measure of how well the normal distribution $\mathcal{N}(1.90, 0.50^2)$ models (or fits) the observed CCND3 gene expression data for the ALL patients.

3.2.4 t-Distribution

The t-distribution (short for Student's t-distribution) has many useful applications for testing hypotheses about one or two normally distributed populations based on the means and standard deviations of one or two samples. For example, we will use the t-Distribution to model the difference between two observed gene expression means. If the population is normally distributed, then the values of $\sqrt{n}(\overline{x} - \mu)/s$ from the sample data follow a t-distribution with $n-1$ degrees of freedom. The t-Distribution is very helpful particularly when the sample size is < 30. For small sample sizes, the t-Distribution is similar to the normal distribution except for that the tails are "fatter" (i.e. the tails have more probability density). However, as the sample size n increases the tails of the t-Distribution get skinnier and approach the tails of the normal distribution. And when the sample size is ≥ 30 then the t-distribution is approximately equal to the normal distribution.

Example 1: Teaching demonstration. Load the `TeachingDemos` package and execute `vis.t()` to explore a visualization of the t-distribution. Click on

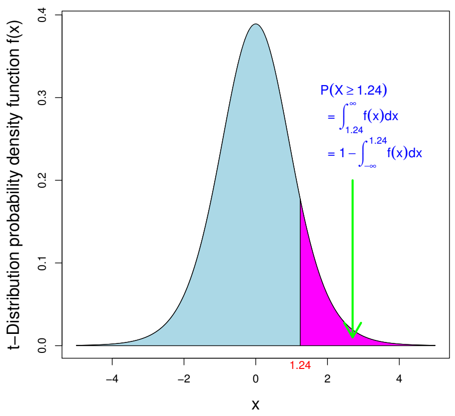

t_{10} probability density function (PDF)

Figure 3.14: Graph of the t_{10} probability density function (PDF) with 10 degrees of freedom.

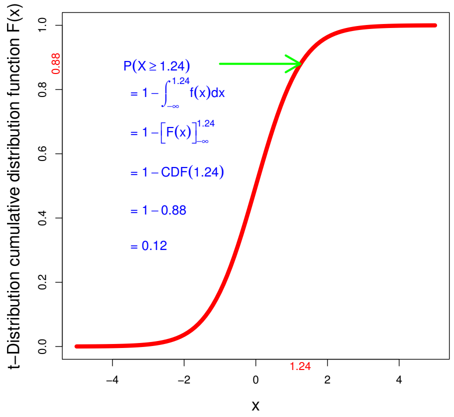

t_{10} cumulative distribution function (CDF)

Figure 3.15: Graph of the t_{10} cumulative distribution function (CDF) with 10 degrees of freedom.

"Show Normal Distribution" and increase the number of degrees of freedom to verify that $df = 30$ is sufficient for the normal approximation to be quite precise.

```
> library(TeachingDemos)
> vis.t()
```

Example 2: Gdf5 gene expression. A quick NCBI scan makes it reasonable to assume that the gene Gdf5 has no direct relation with leukemia. For this reason, we will assume that the mean change in Gdf5 expression for the AML patients is 0, that is $\mu_{Gdf5} = 0$. With this assumption, we can attempt to use the t-Distribution to model our data. The `grep()` function can be used to find the row of expression values for the Gdf5 gene in the `golub` matrix. Then we can compute the sample t-value $\sqrt{n}(\overline{x} - \mu_{Gdf5})/s$:

```
> n <- 11
> gdf5 = grep("GDF5",golub.gnames[ ,2], ignore.case = TRUE)
> gdf5
[1] 2058
> x <- golub[gdf5, golubFactor=="AML"]
> t.value <- sqrt(n)*(mean(x)-0)/sd(x)
> t.value
[1] 1.236324
```

From above, we now know that the t-values for multiple samples of Gdf5 gene expression should follow the t-distribution with $n - 1 = 10$ degrees of freedom, represented as t_{10}. The probability of observing our t-value of 1.236324 or greater in the t_{10} distribution is computed as follows:

$$P(t_{10} \geq 1.236324) = 1 - P(t_{10} \leq 1.236324)$$

and can be calculated with:

```
> 1 - pt(1.236324,10)    # right-side tail of the t-distribution cumulative density
    ↪ function (CDF)
[1] 0.1222945
```

The probability $P(t_{10} \geq 1.236324)$ corresponds to the magenta area below the probability density function (PDF) curve in Figure 3.14. We can create Figure 3.14 by using the `dt()` function:

```
> f<-function(x) { dt(x,10) }
> plot(f,             # function
+      -5,            # start
+      5,             # end
+      cex.lab=1.5,   # make axis labels big
+      xlab="x",
+      ylab="t-Distribution probability density function f(x)")
> x1 <- seq(-5,1.24,0.01)
> x2 <- seq(1.24,5,0.01)
> polygon(c(-5,x1,1.24), c(0,f(x1),0), col="lightblue")
> polygon(c(1.24,x2,5),  c(0,f(x2),0), col="magenta")
> mtext("1.24",side=1,at=1.24, col="red")
> arrows(2.7,0.20,2.7,0.01, lwd=3, col="green")
> text(1.8, 0.3 - c(0,.043, .086), cex = 1.2, adj=c(0,0), col="blue",
+      c(expression(P(X >= 1.24)),
+        expression(paste("  = ", integral(f(x) * dx, 1.24, infinity))),
+        expression(paste("  = ", 1 - integral(f(x) * dx, -infinity, 1.24)))))
```

The t-distribution cumulative distribution function (CDF) with ten degrees of freedom is illustrated in Figure 3.15. Similar to Figure 3.14, we can create Figure 3.15 by using the `pt()` function:

```
F<-function(x) { pt(x,10) }
> plot(F,             # function
+      -5,            # start
+      5,             # end
+      cex.lab=1.5,   # make axis labels big
+      col="red",
+      lwd=6,         # make line thicker
+      xlab="x",
+      ylab="t-Distribution cumulative distribution function F(x)")
> mtext("1.24",side=1,at=1.24, col="red")
> mtext("0.88",side=2,at=0.88, col="red")
> arrows(-1,0.88,1.24,0.88, lwd=3, col="green")
> text(-3.7, 0.85 - c(0,.11,.22,.33,.44,.55), cex=1.2, adj=c(0,0), col="blue",
+      c(expression(P(X >= 1.24)),
+         expression(paste("  = ", 1 - integral(f(x) * dx, -infinity, 1.24))),
+         expression(paste("  = ", 1 - bgroup("[", F(x) ,"]")[-infinity]^1.24)),
+         expression(paste("  = ", 1 - CDF(1.24))),
+         expression(paste("  = ", 1 - 0.88)),
+         expression(paste("  = ", 0.12))))
```

The probability $P(-2 \leq t_{11} \leq 2)$ that the random variable t_{10} is between -2 and 2 can be calculated as the difference between two CDFs using the `pt()` function:

```
> pt(2,10)-pt(-2,10)   # t-distribution cumulative density function (CDF) - central area
[1] 0.926612
```

Lastly, the 2.5% quantile $x_{0.025}$ of t_{10}, which is the x-value for which the $P(t_{10} \leq x) = 0.025$, can be computed by using the `qt()` function:

```
> qt(0.025,n-1)    # t-distribution quantile function
[1] -2.228139
```

3.2.5 F-Distribution

The F-distribution is widely used for testing whether the population variances of two normal populations are equal, based on the sample standard deviations. It can be shown that the ratio of variances between two independent samples from two normally distributed random variables follows an F-distribution. More specifically, if the two population variances are equal ($\sigma_1^2 = \sigma_2^2$), then for two samples s_1 and s_2 the ratio s_1^2/s_2^2 follows an F-distribution with $n_1 - 1, n_2 - 1$ degrees of freedom, where s_1^2 is the variance of the first set, s_2^2 that of the second, and n_1 is the number of observations in the first set and n_2 in the second.[3]

Example 1: CCND3 gene expression. When the two population variances are in fact equal, then the probability is great that the ratio of their sample variances is near one. Using the Golub et. al. (1999) data, we can calculate the probability of observing the CCND3 sample variances for the ALL and AML patients after assuming that their population variances are the same. First, we compute the ratio between the sample CCND3 expression variances among the ALL and AML patients:

```
> var(golub[1042,golubFactor=="ALL"]) / var(golub[1042,golubFactor=="AML"])
[1] 0.7116441
```

[3] It is more correct to define S_1^2/S_2^2 for certain random variables S_1^2 and S_2^2. We shall, however, not bother.

$F_{26,10}$ probability density function (PDF)

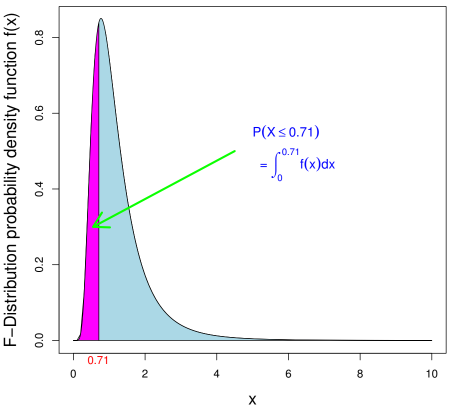


Figure 3.16: Graph of the $F_{26,10}$ probability density function (PDF) with (26,10) degrees of freedom.

$F_{26,10}$ cumulative distribution function (CDF)

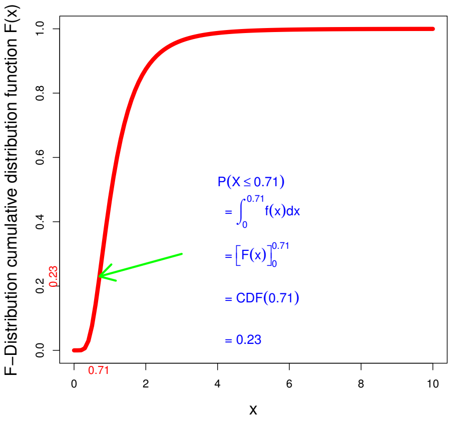


Figure 3.17: Graph of the $F_{26,10}$ cumulative distribution function (CDF) with (26,10) degrees of freedom.

Since $n_1 = 27$ and $n_2 = 11$, our ratio of the sample variances should be a realization of the $F_{26,10}$ distribution. The probability of observing our ratio 0.7116441 or smaller in the $F_{26,10}$ distribution is $P(F_{26,10} \leq 0.7116441)$. We can calculate this probability using the `pf()` function:

```
> pf(0.7116441,26,10)    # left-side tail of the F-distribution cumulative density function
    ↪  (CDF)
[1] 0.2326147
```

In Figure 3.16, the probability $P(F_{26,10} \leq 0.7116441)$ corresponds to the magenta area below the probability density function (PDF) curve. We can create Figure 3.16 by using the `df()` function:

```
> f<-function(x) { df(x,26,10) }
> plot(f,             # function
+      0,             # start
+      10,            # end
+      cex.lab=1.5,   # make axis labels big
+      xlab="x",
+      ylab="F-Distribution probability density function f(x)")
> mtext("0.71",side=1,at=.7,cex=1, col="red")
> x1 <- seq(0,0.71,0.01)
> x2 <- seq(0.71,10,0.01)
> polygon(c(0,x1,.71),  c(0,f(x1),0), col="magenta")
> polygon(c(.71,x2,10), c(0,f(x2),0), col="lightblue")
> arrows(4.5,.50,0.55,.3, lwd=3, col="green")
> text(5.0, 0.53 - c(0,.11), cex = 1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 0.71)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 0.71)))))
```

Figure 3.17 gives the values of the cumulative distribution function. Similary, we can create Figure 3.17 by using the `pf()` function:

```
> f<-function(x) { pf(x,26,10) }
> plot(f,             # function
+      0,             # start
+      10,            # end
+      cex.lab=1.5,   # make axis labels big
+      col="red",
+      lwd=6,          # make line thicker
+      xlab="x",
+      ylab="F-Distribution cumulative distribution function F(x)")
> mtext("0.71",side=1,at=.7, cex=1, col="red")
> mtext("0.23",side=2,at=.23,cex=1, col="red")
> arrows(3,0.3,0.71,0.23, lwd=3, col="green")
> text(4.0, 0.5 - c(0,.12,.24,.36,.48), cex=1.2, adj=c(0,0), col="blue",
+      c(expression(P(X <= 0.71)),
+        expression(paste("  = ", integral(f(x) * dx, 0, 0.71))),
+        expression(paste("  = ", bgroup("[", F(x) ,"]")[0]^0.71)),
+        expression(paste("  = ", CDF(0.71))),
+        expression(paste("  = ", 0.23))))
```

Lastly, to find the quantile $x_{0.025}$ of the $F_{26,10}$ distribution, which is the x-value for which the $P(F_{26,10} \leq x) = 0.025$, we can use the `qf()` function:

```
> qf(.025,26,10)    # F-distribution quantile function
[1] 0.3861673
```

This subject is taken further in Section 4.1.6.

3.3 Overview and concluding remarks

R has many built-in functions for probability calculations that use the binomial, normal, t, F, χ^2-distributions, where `d` stands for probability Density, `p` for cumulative Probability distribution, `q` for Quantiles, and `r` for drawing Random samples (see Table 3.2). The values of the density, expectation, and variance of most of the distributions in this chapter are summarized in Table 3.3.

Although the above distributions are without a doubt among the most important, there are several additional distributions available such as the Gamma, beta, or Dirichlet that can also be used to model different types of biological data. We encourage the reader to learn more about how and when to use them. The free encyclopedia *wikipedia* often gives a useful good first, though incomplete, introduction to the characteristics of these distributions.

Table 3.2: Built-in functions for random variables used in this chapter.

	para-				random
Distribution	meters	density	cumulative	quantiles	sampling
Binomial	n, p	`dbinom`(x, n, p)	`pbinom`(x, n, p)	`qbinom`(α, n, p)	`rbinom`$(10, n, p)$
Poisson	λ	`dpois`	`ppois`	`qpois`	`rpois`
Exponential	λ	`dexp`	`pexp`	`qexp`	`rexp`
Normal	μ, σ	`dnorm`(x, μ, σ)	`pnorm`(x, μ, σ)	`qnorm` (α, μ, σ)	`rnorm`$(10, \mu, \sigma)$
Chi-squared	m	`dchisq`(x, m)	`pchisq`(x, m)	`qchisq`(α, m)	`rchisq`$(10, m)$
t	m	`dt`(x, m)	`pt`(x, m)	`qt`(α, m)	`rt`$(10, m)$
F	m,n	`df`(x, m, n)	`pf`(x, m, n)	`qf`(α, m, n)	`rf`$(10, m, n)$

Note that a distribution acts as a population from which a sample can be drawn. Hence, distributions can be seen as models of data generating procedures. For a more technical treatment of distributions we refer the reader to Bain & Engelhardt (1992), Johnson et al. (1992), and Miller & Miller (1999).

Table 3.3: Density, mean, and variance of distributions used in this chapter.

Distribution	parameters	probability mass or density	expectation	variance
Bernoulli	p	$p^k(1-p)^{1-k}$ for $k \in \{0,1\}$	p	$p(1-p)$
Binomial	n, p	$\frac{n!}{k!(n-k)!}p^k(1-p)^{n-k}$	np	$np(1-p)$
Poisson	λ	$e^{-\lambda}\lambda^x/x!$	λ	λ
Exponential	λ	$\lambda e^{-\lambda x}$	$1/\lambda$	$1/\lambda^2$
Normal	μ, σ	$\frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{1}{2}(\frac{x-\mu}{\sigma})^2)$	μ	σ^2
Chi-squared	df$=k$	$\frac{1}{2^{\frac{k}{2}}\Gamma(\frac{k}{2})}x^{\frac{k}{2}-1}e^{-\frac{x}{2}}$	k	$2k$

3.4 Exercises

It is important to obtain some familiarity with the computation of probabilities and quantiles.

1. **Binomial distribution.** Let X be binomially distributed with $n = 60$ and $p = 0.4$. Compute the following.

   (a) $P(X = 24)$, $P(X \leq 24)$, and $P(X \geq 30)$.

   (b) $P(20 \leq X \leq 30)$, $P(20 \leq X)$.

(c) $P(20 \leq X \text{ or } X \geq 40)$, and $P(20 \leq X \text{ and } X \geq 10)$.

(d) The mean and standard deviation of X.

(e) The quantiles $x_{0.025}$, $x_{0.5}$, and $x_{0.975}$.

2. **Standard normal distribution.** Compute the following probabilities and quantiles.

   (a) $P(1.6 < Z < 2.3)$.

   (b) $P(Z < 1.64)$.

   (c) $P(-1.64 < Z < -1.02)$.

   (d) $P(0 < Z < 1.96)$.

   (e) $P(-1.96 < Z < 1.96)$.

   (f) The quantiles $z_{0.025}$, $z_{0.05}$, $z_{0.5}$, $z_{0.95}$, and $z_{0.975}$.

3. **Normal distribution.** Compute for X distributed as $\mathcal{N}(10, 2)$ the following probabilities and quantiles.

   (a) $P(X < 12)$.

   (b) $P(X > 8)$.

   (c) $P(9 < X < 10, 5)$.

   (d) The quantiles $x_{0.025}$, $x_{0.5}$, and $x_{0.975}$.

4. ***t*-distribution.** Compute the following probabilities and quantiles for the t_6 distribution.

   (a) $P(t_6 < 1)$.

   (b) $P(t_6 > 2)$.

   (c) $P(-1 < t_6 < 1)$.

   (d) $P(-2 < t_6 < -2)$.

   (e) The quantiles $t_{0.025}$, $t_{0.5}$, and $t_{0.975}$.

5. ***F* distribution.** Compute the following probabilities and quantiles for the $F_{8,5}$ distribution.

   (a) $P(F_{8,5} < 3)$.

(b) $P(F_{8,5} > 4)$.

(c) $P(1 < F_{8,5} < 6)$.

(d) The quantiles $f_{0.025}$, $f_{0.5}$, and $f_{0.975}$.

6. **Chi-squared distribution.** Compute the following for the chi-squared distribution with 10 degrees of freedom.

   (a) $P(\chi^2_{10} < 3)$.

   (b) $P(\chi^2_{10} > 4)$.

   (c) $P(1 < \chi^2_{10} < 6)$.

   (d) The quantiles $x_{0.025}$, $x_{0.5}$, and $x_{0.975}$.

7. **Purines in microRNAs.** Suppose that for certain microRNA of size 20 the probability of a purine is binomially distributed with probability 0.7.

   (a) What is the probability of 14 purines?

   (b) What is the probability of less than or equal to 14 purines?

   (c) What is the probability of strictly more than 10 purines?

   (d) What is the probability that there are between 10 and 15 purines, inclusive?

   (e) How many purines do you expect? In other words: What is the mean of the distribution?

   (f) What is the standard deviation of the distribution?

8. **Zyxin gene expression.** The distribution of the expression values of the ALL patients on the Zyxin gene are distributed according to $\mathcal{N}(1.6, 0.4^2)$.

   (a) Compute the probability that the expression values are smaller than 1.2.

   (b) What is the probability that the expression values are between 1.2 and 2.0?

   (c) What is the probability that the expression values are between 0.8 and 2.4?

   (d) Compute the exact values for the quantiles $x_{0.025}$ and $x_{0.975}$.

(e) Use `rnorm` to draw a sample of size 1000 from the population and compare the sample mean and standard deviation with that of the population.

9. **Some computations on the Golub et al. (1999) data.**

   (a) Take $\mu = 0$ and compute the t-values for the ALL gene expression values. Find the three genes with largest absolute t-values.

   (b) Compute per gene the ratio of the variances for the ALL over the AML patients. How many are between 0.5 and 1.5?

10. **Extreme value investigation.** This difficult question aims to teach the essence of an extreme value distribution. An interesting extreme value distribution is given by Pevsner (2003, p.103). To repeat this example, take the maximum of a sample (with size 1000) from the standard normal distribution and repeat this a 1000 times - so that you have sampled 1000 maxima. Next, subtract from these maxima `an` and divide by `bn`, where:

```
an <- sqrt(2*log(n)) - 0.5*(log(log(n))+log(4*pi))*(2*log(n))^(-1/2)
bn <- (2*log(n))^(-1/2)
```

    Now plot the density from the normalized maxima and add the extreme value function $f(x)$ from the Pevsner example, and add the density (`dnorm`) from the normal distribution. What do you observe?