STAT 260 - Introduction to R Assignment 2

- For each family of probability distributions, R has several functions.
  - ▷ Functions that begin with "d" give the probability density function. (i.e. the pmf or pdf.)
  - ▷ Functions that begin with "p" give the cumulative density function. (i.e. the cdf.)
  - ▷ Functions that begin with "r" give a random sample from the family.
  - ▷ Functions that begin with "q" are used to find percentiles.
- R can complete basic arithmetic operations such as addition $(+)$, subtraction $(-)$, multiplication $(*)$, and division $(/)$. These can be used in conjunction with the probability calculations discussed below.

1 Binomial Distribution

- Suppose that we want to work with a binomial random variable X with 18 trials and probability of success 0.171. That is, $n = 18$, and $p = 0.171$.
- The four functions for the binomial distribution are **dbinom**, **pbinom**, **rbinom**, and **qbinom**.
- **Calculating Probabilities**:
  - ▷ If we wanted to find $P(X = 2)$, we would use the **dbinom** command and enter:

    ```
    dbinom(2, size=18, prob=0.171)
    ```

  - ▷ If we wanted to find $P(X \leq 3)$, we would use the **pbinom** command and enter:

    ```
    pbinom(3, size=18, prob=0.171)
    ```

  - ▷ As with the cdf tables, if we want to find $P(2 \leq X \leq 5)$ or $P(X \geq 2)$, we would first need to rewrite the probabilities in a way so that the cdf could be used. For example, if we wanted $P(2 \leq X \leq 5)$, we would instead find $P(X \leq 5) - P(X \leq 1)$. We would then use the **pbinom** command and enter:

    ```
    pbinom(5, size=18, prob=0.171) - pbinom(1, size=18, prob=0.171)
    ```

- **Generating Datasets**:
  - ▷ If we want to simulate the values of X that we might see if our binomial experiment were to be carried out 6 times, we would then use the **rbinom** command:

    ```
    rbinom(6, size=18, prob=0.171)
    ```

    This would return a list of 6 values between 0 and 18. The first value represents the number of successes seen on the first simulation of the experiment (i.e. number of successes in the first 18 trials), the second value represents the number of successes seen in the second simulation of the experiment (i.e. number of successes in the next 18 trials), etc.
  - ▷ If we want to do anything with the simulated data, we should define a vector containing the data. For example:

    ```
    bin.simulation <- rbinom(6, size=18, prob=0.171)
    ```

    (Note that this creates the vector of values, but does not tell you which values are stored in the vector. If we wanted to see which values were stored in the vector we would then call upon our vector name and enter `bin.simulation`, which would then give us the output of the 6 values which were stored in the vector.)
  - ▷ Now that the observations are contained in the vector named `bin.simulation`, we can make histograms, find the mean or variance or standard deviation the same way that we did in R Assignment 1. (See R Assignment 1 for any commands needed to do this.)

2 Poisson Distribution

- Suppose that we want to work with a Poisson random variable X with a mean of $\lambda = 5$.
- The four functions for the Poisson distribution are **dpois**, **ppois**, **rpois**, and **qpois**.
- **Calculating Probabilities**:
  - ▷ To find $P(X = 2)$, enter:

    ```
    dpois(2, lambda=5)
    ```

  - ▷ To find $P(X \leq 4)$, enter:

    ```
    ppois(4, lambda=5)
    ```

  - ▷ As with the binomial distribution, if we wanted to find something like $P(X \geq 7)$ or $P(3 \leq X \leq 7)$, we need to rewrite the probability statement first with the CDF.
    For example, to find $P(X \geq 7)$, we would use $P(X \geq 7) = 1 - P(X \leq 6)$, and enter:

    ```
    1-ppois(6, lambda=5)
    ```

- **Generating Datasets**:
  - ▷ To simulate 10 repetitions of the Poisson experiment with rate $\lambda = 5$, we would enter:

    ```
    rpois(10, lambda=5)
    ```

3 Normal Distribution

- The four functions for the normal distribution are **dnorm**, **pnorm**, **rnorm**, and **qnorm**.
- **Calculating Probabilities**:
  - ▷ If $X \sim Normal(\mu = 7, \sigma = 0.6)$ and we wanted to calculate $P(X \leq 8)$, enter:

    ```
    pnorm(8, mean=7, sd=0.6)
    ```

    If we wanted to calculate $P(6 \leq X \leq 8)$ we would first rewrite as $P(6 \leq X \leq 8) = P(X \leq 8) - P(X \leq 6)$ and enter:

    ```
    pnorm(8, mean=7, sd=0.6) - pnorm(6, mean=7, sd=0.6)
    ```

  - ▷ If $X \sim Normal(\mu = 7, \sigma = 0.6)$ and we wanted to find the X value c for which 5% of values are less than or equal to c, enter:

    ```
    qnorm(0.05, mean=7, sd=0.6)
    ```

- **Generating Datasets**:
  - ▷ If we want to simulate 50 values of a random variable X, where $X \sim Normal(\mu = 7, \sigma = 0.6)$, we would then use the

    ```
    rnorm(50, mean=7, sd=0.6)
    ```

4 Other Continuous Distributions

4.1 Gamma, Exponential, and Uniform Distributions

- The four functions for the gamma distribution are **dgamma**, **pgamma**, **rgamma**, and **qgamma**.
- The four functions for the exponential distribution are **dexp**, **pexp**, **rexp**, and **qexp**.
- The four functions for the uniform distribution are **dunif**, **punif**, **runif**, and **qunif**.
- Note that the normal, the gamma, the exponential, and the uniform distributions are all continuous distributions, and so to calculate our probabilities we will only use the cdf functions: **pnorm**, **pgamma**, **pexp**, and **punif**.
- **Calculating Probabilities:**
  - ▷ If $X \sim Gamma(\alpha = 2, \beta = 3)$ and we wanted to calculate $P(X \leq 1.5)$, enter:

    ```
    pgamma(1.5, shape=2, scale=3)
    ```

    (Note that in this command $shape = \alpha$ and $scale = \beta$.)
  - ▷ If $X \sim exponential(\lambda = 2)$ and we wanted $P(X \leq 1)$, enter:

    ```
    pexp(1, rate=2)
    ```

    (Note that in this command $rate = \lambda$.)
  - ▷ If $X \sim uniform(a = 10, b = 30)$ and we wanted $P(X \leq 15)$, enter:

    ```
    punif(15, min=10, max=30)
    ```

- **Generating Datasets**:
  - ▷ To simulate 50 values of a random variable X, where $X \sim Gamma(\alpha = 7, \beta = 3)$, we would enter:

    ```
    rgamma(50, shape=7, scale=3)
    ```

  - ▷ To simulate 200 values of a random variable X, where $X \sim exponential(\lambda = 4)$, we would enter:

    ```
    rexp(200, rate=4)
    ```

  - ▷ To simulate 100 values of a random variable X, where $X \sim uniform(a = 2, b = 5)$, we would enter:

    ```
    runif(100, min=7, max=3)
    ```