Statistics & Probability
Statistical Programming with R
Class 3 – Continuous Random Variables & Confidence Intervals

In today's class we will use **R** to perform calculations regarding continuous random variables. Later we will calculate confidence intervals for the mean of a population given a large sample.

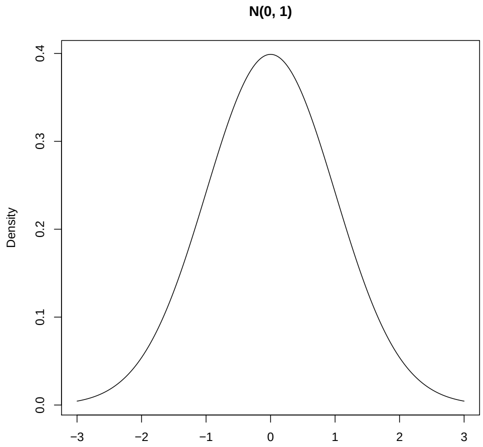
1 Continuous Random Variables

1.1 The Normal Distribution

- The normal distribution is the most important distribution in statistics. The probability density of the normal distribution is of the form:

$$f(x) = \frac{1}{\sqrt{(2\pi\sigma^2)}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

The graph below shows the density curve when $\mu = 0$ and $\sigma^2 = 1$.



- The expected value and variance are μ and σ^2, respectively.

- The `rnorm` function can be used to simulate values from a normal distribution. To simulate 1000 values from the $N(0, 1)$ distribution illustrated above run:

  ```
  norm_samp <- rnorm(1000, mean=0, sd=1)
  ```

- **NB:** The standard deviation is the square root of the variance σ^2. We usually characterise a Normal distribution as $X \sim N(\mu, \sigma^2)$ but `rnorm` takes the standard deviation as an input, not the variance.

- Plot a histogram of these values and note the shape. Recall that for normally distributed data the histogram should be bell shaped and symmetric.

- **Note:** You can adjust the number of bins used in plotting a histogram in **R** by adding the `breaks` input. For example if you want 30 bins in your histogram run:

  ```
  hist(norm_samp, breaks = 30)
  ```

- The cumultative distribution function for the normal distribution does not have an easily computed functional form but it is used in many applications. Computing probabilties from the normal distribution is easy to do using **R**.

- Suppose you wanted to know the area between 0.6 and 1.2 under the density curve of a normal distribution with mean 0 and standard deviation 1. To do this with **R**, you would use a function called `pnorm`. This function takes three parameters

  1. An observation value (e.g. 1.2).
  2. The mean of the normal distribution.
  3. The standard deviation of the normal distribution.

- `pnorm` then returns the area under the pdf to the left of the given point. The syntax looks like this:

  ```
  pnorm(1.2, mean = 0, sd = 1)
  ```

  And to find the area to the left of 0.6,

  ```
  pnorm(0.6, mean = 0, sd = 1)
  ```

  To find the area between the two, use **R** to subtract one from the other.

  Complete part A on your answer sheet

- The lengths of baby elephants' trunks follow a normal distribution with mean 1.8 metres and standard deviation 0.4 metres. What is the probability that a baby elephant will have a trunk between 1.6 and 2.1 metres long?

- If we were using the New Cambridge Statistical Tables, we would have to standardise before answering this question. But with **R** we don't need to do that! The first area you need is given by:

  ```
  area1 <- pnorm(2.1, mean = 1.8, sd = 0.4)
  ```

- Now complete the rest of the probability calculation yourself.

  Complete part B on your answer sheet*

- Look at the help files for the `qnorm()` function. This can be used to find percentiles of the normal distribution like we did in class. Using the function, find the values A and B for which the probability of $X \sim N(10, 3)$ lying between A and B is 0.95 and $\mathbb{P}(X < A) = \mathbb{P}(X > B) = 0.025$.

- Don't forget to sketch a picture before trying this.

- Download the dataset called `variables.txt` from blackboard. Save it in your Documents folder and make this folder your Working Directory (for instructions see Software Class 1). Now import it into **R** – remember, you do that by using the `read.table` function and then the data.frame function, like this:

```
variables <- read.table("variables.txt", header = TRUE)
variables <- data.frame(variables)
```

- You will have to alter the file address above to exactly match the location where you have saved the data, including case sensitivity. There are some notes on reading data in the lab sheet from the first **R** class if you run into difficulty.

- It has been suggested that both columns in this set contain data which follow normal distributions. Using graphs and descriptive statistics, decide whether you agree with this claim and explain why or why not for each column.

- Remember, the theoretical qualities of the Normal distribution that you are looking for are that the data should produce a bell-shaped curve, symmetric around the mean; and where the mean, median and mode are all equal. A function which might help is the `summary` function you used in the first **R** class.

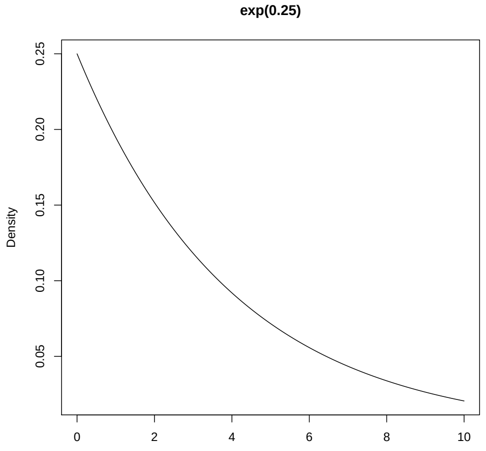
  Complete part C on your answer sheet

1.2 The Exponential Distribution

- We looked at the exponential distribution in lectures and discussed how the lifetimes of machine components could be well modelled by this distribution.

- Suppose that the lifetime of an iPad (let's call this random variable *iTime*) is well modelled by an exponential distribution with a mean of 4 years. Thus $iTime \sim \exp(\lambda = 0.25)$.

- The density curve for this distribution looks like this:



- We can simulate some sample lifetimes from this distribution using the `rexp` function. We need to specify how many samples we want and the rate parameter of the exponential distribution.

  ```
  samp_exp <- rexp(1000, rate=0.25)
  ```

  Where did the value of 0.25 for the rate parameter come from?

- Plot a histogram of this data. Does the histogram resemble the curve above? Simulate samples from other exponential distributions by changing the value for `rate`.

- We can calculate probabilties from an exponential distribution using the `pexp` function. Suppose we are interested in knowing what the probability is that an iPad will function for less than 5 years, `pexp` can quickly calculate this:

  ```
  pexp(5, rate=0.25)
  ```

- Now suppose that *iTime* is actually exponentially distributed with a mean of 5 years. What is the probability that an iPad is functional for between 6 and 8 years?

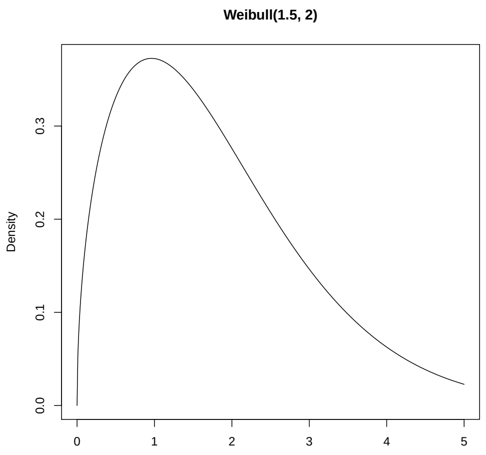
  Complete part D on your answer sheet

1.3 The Weibull Distribution

- The Weibull distribution is another waiting time distribution which is often used in reliability analyses.

- Suppose the time to failure for the latest smartphone follows a Weibull distribution with shape parameter $\alpha = 1.5$ and scale parameter $\beta = 2$.

- The density curve for this distribution looks like this:



- Again we can simulate lifetimes from this distribution and compare a histogram of the sample to the theoretical density. The function which samples from the Weibull distribution is called `rweibull`. Generate a sample of 1,000 smartphone lifetimes by running the following code:

```
rweibull(1000, shape=1.5, scale=2)
```

- How does a histogram of this sample compare to the theoretical density? The shape parameter can drastically change the shape of the density curve. Experiment with different values of the shape parameter and note how the histogram of the sample changes shape.

- Probabilities concerning the Weibull distribution can be calculated, in a similar way to that for the other distributions we looked at, using the `pweibull` function.

- Using the help files can you work out the probability that the latest smartphone functions for more than 3 years before failing?

  Complete part E on your answer sheet

2 Confidence Intervals

- In class we have seen how to calculate confidence intervals for the mean of a population given a large sample. The formulae involved are simple and so confidence intervals of this kind can be calculated in **R** using the simple summary functions we have already seen.

- A $100(1-\alpha)\%$ confidence interval for μ the mean of a population is given by:

  $$\bar{X} \pm Z_{\frac{\alpha}{2}} \frac{s}{\sqrt{n}}$$

  where $n > 30$.

- We can calculate $\bar{X}$ using the `mean` function, s using the `sd` function and $Z_{\frac{\alpha}{2}}$ using `qnorm`. We can calculate n for a particular variable using the `length` function which simply returns the length of the vector.

- Refer again to the `variables` data set from earlier. The following code will calculate the lower and upper bounds of a confidence interval for the mean of `var1`.

```
xbar <- mean(variables[, 1])
s <- sd(variables[, 1])
z <- qnorm(0.975)
n <- length(variables[, 1])
lower <- xbar - z*(s/sqrt(n))
upper <- xbar + z*(s/sqrt(n))
```

- Calculate a 90% confidence interval for the mean of `var2`.

  Complete part F on your answer sheet

Make sure you save your R script to a USB key or to Google drive before you log off.

Statistics & Probability
3 – Continuous Random Variables
& Confidence Intervals

A. Area left of 1.2:

Area left of 0.6:

Area between 0.6 and 1.2:

B. What is the probability that the trunk is between 1.6 and 2.1 metres long?

What code did you use?

C. Is `var1` normally distributed? Explain.

What code did you use?

Is `var2` normally distributed? Explain.

What code did you use?

D. What is the probability that an iPad functions for between 6 and 8 years?

What code did you use?

E. What is the probability that the probability that the latest smartphone functions for more than 3 years before failing?

What code did you use?

F. 90% confidence interval for the mean of `var2`:

What code did you use?