Module 15 – Statistics and R

therefore, many companies use R for internal analysis and regulatory testing, but use Python for any application or script that contains proprietary information.

Despite the licensing drawback, R is still a highly valuable programming language for data analysis and is used by data professionals at all levels across many fields.

Similar to Jupyter Notebooks, RStudio enables users to test their analysis scripts line by line while allowing users to view different environment variables and outputs.

The two fundamental components to programming in R are creating **data structures** and using **functions.**

R is a object-oriented programming language.

The most straightforward R data structures are named values and vectors.

**Vectors** are R's version of arrays, where a list of numbers are assigned a location and stored as a single data structure

The **assignment operator** (<-) tells R to assign whatever is right of the arrow to the name that is left of the arrow.

In R, all environment objects are mutable, which means they can be assigned and reassigned multiple times

If at any point you are unsure what an R function does or what it needs to execute, you can always type ?<name of function> in the R console and it'll open the documentation in the Help pane

Depending on what delimiter (or value-separating character) is used, we can use read.csv()for comma-delimited files, read.delim() for tab-delimited files, or read.table() if we need to manually tell the function what delimiter is used.

When loading a package in R, do not use quotation marks

txt is the file path of the JSON file on our machine

import csv:

demo\_table <- read.csv(file='demo.csv',check.names=F,stringsAsFactors = F)

import JSON:

demo\_table <- read.csv(file='demo.csv',check.names=F,stringsAsFactors = F)

Unlike Python, R's index starts at 1. So, the third element would be index = 3.

filter\_table <- demo\_table2[demo\_table2$price > 10000,] # (The comma is necessary to subset by rows. Adding column(s) after the comma specifies the columns to select.)

Subset() : Return subsets of vectors, matrices or data frames which meet conditions.

The subset() function makes filtering and subsetting easier to read by assuming column names in the **subset** argument, which cuts down on statement length.

$s

The dplyr library contains a wide variety of functions that can be chained together to transform data quickly and easily.

The tidyverse package contains libraries such as dplyr, tidyr, and ggplot2. These packages work together to help simplify the process of creating transformed data columns, grouping data using factors, reshaping our two-dimensional data structures, and visualizing our results using plots.

Chain functions by using (%>%).

The summarize() function takes an additional argument, .groups. This allows you to control the the grouping of the result. The four possible values are:

* .groups = "drop\_last" drops the last grouping level (default)
* .groups = "drop" drops all grouping levels and returns a tibble
* .groups = "keep" preserves the grouping of the input
* .groups = "rowwise" turns each row into its own group

1. **ggplot function**—tells ggplot2 what variables to use
2. **geom function**—tells ggplot2 what plots to generate
3. **formatting or theme functions**—tells ggplot2 how to customize the plot

The ggplot() function only requires two arguments to declare the input data:

* **data**
* **mapping**

There are a number of optional aes() arguments to assign such as color, fill, shape, and size to customize the plots. We'll cover these optional assignments in this module.

Bar plots are used to visualize categorical data.

if we want to compare the number of vehicles from each manufacturer in the dataset, we can use dplyr's summarize() function to summarize the data, and ggplot2's geom\_col() to visualize the results:

The hjust argument tells ggplot that our rotated labels should be aligned horizontally to our tick marks.

Line plots are used to visualize the relationship between a categorical variable and a continuous numerical variable.

When creating the ggplot object for our line data, we need to set the categorical variable to the *x* value and our continuous variable to the *y* value within our aes() function. For example, if we want to compare the differences in average highway fuel economy (hwy) of Toyota vehicles as a function of the different cylinder sizes (cyl), our R code would look like the following:

In data science, we're often interested in the relationship between two quantitative variables in regards to other categorical variables (often referred to as grouping factors). By customizing our data points with aesthetic changes, we can add additional context to our scatter plots to help convey more information within a single visualization. Let's see this in action.

An alternative to xlabs() andylabs() is thelabs() function, which lets you customize your axis labels as well as any grouping variable labels.

Using the same dataset, create an additional visualization that uses City Fuel-Efficiency (MPG) to determine the size of the data point.

A picture containing diagram

Description automatically generated

Core statistical concepts:

* mathematical data types
* null and alternative hypothesis
* p-values and hypothesis testing
* t-test of the means
* correlation and linear regression tests
* comparing frequency distribution using chi-squared test

Types of Data

Graphical user interface, diagram

Description automatically generated

**Categorical Data**

**Categorical data** represents data characteristics or qualitative descriptions. Generally, categorical data is any data that is not measured, also known as qualitative data. Categorical data can be collected in the form of strings, true/false Boolean values, or even encoded numbers as categories (such as one for red, two for blue, three for green, etc.). Several statistical tests use categorical data to inform which groups to compare. Categorical data has three subtypes: dichotomous, ordinal, and nominal

## Dichotomous Data

**Dichotomous data** is collected from either one of two categories. For example, an online survey might collect member/non-member or demographic information. Dichotomous data can be collected in the form of true/false Boolean values, 0 or 1 binary values, or two strings. Later in the module, we'll use dichotomous data to help perform many of our comparative statistical tests.

## Ordinal Data

**Ordinal data** has a ranked order. Although ordinal data has a sequence, we don't necessarily know the value between each ordinal data point. Data that is collected on a value scale (e.g., movie rankings, survey results, and the [Likert scale (Links to an external site.)](https://en.wikipedia.org/wiki/Likert_scale)) are common forms of ordinal data. Ordinal data combines the qualitative properties of labels to the quantitative properties of scale to allow for comparative analyses. Ordinal data is very popular with research and survey groups because it allows for quantitative analysis without the need of machinery and tools to obtain measurements.

## Nominal Data

**Nominal data** is data used as labels or names for other measures. Nominal data can be as individual as an identification number or can be as general as a list of three options. Unlike ordinal data, nominal data has no ranking. Therefore, nominal data is often used with a more quantitative data type to perform an analysis. Often nominal data will be transformed using a grouping function to decrease the complexity of the data.

## Numerical Data

Typically, **numerical data** is obtained by taking a measurement from an instrument (such as a ruler, measuring scale, sensor, etc.) or by counting. In statistics, numerical data is used to perform quantitative analysis that can produce the probability of an outcome or quantify the relationship between categories. Within numerical data there are two primary data types to consider: continuous and interval.

## Continuous Data

**Continuous data** can be subdivided infinitely. For example, if you want to describe the thickness of window glass, you could measure it in x number of centimeters, millimeters, nanometers, picometers, and so on. Continuous data is typically recorded with decimal places to match the precision of the measurement. Almost all statistical tests and models use continuous data to generate precise results.

## Interval Data

**Interval data** is spaced out evenly on a scale. Also known as integer data, interval data does not use decimal places and can't be subdivided. Interval data also can't be multiplied or divided. Because interval data is spaced out evenly, it can be grouped together or bucketed easily. For example, a set of integers 15, 4, 18, 10, 3, and 5 could be collected as a group that is less than 20. Due to this property, interval data can be treated as a numerical data type or transformed into a nominal data type.

Additionally, interval data can be generated through rounding continuous data at the cost of losing precision of the measurement. Therefore, interval data can be used by most statistical models as either a quantitative or qualitative variable, depending on the use case.

Chart, histogram

Description automatically generated

When numerical data is considered to be normally distributed, the probability of any data point follows the **68-95-99.7** rule, stating that 68.27%, 95.45%, and 99.73% (effectively 100%) of the values lie within one, two, and three standard deviations of the mean, respectively.

The **qualitative test for normality** is a visual assessment of the distribution of data, which looks for the characteristic bell curve shape across the distribution. In R, we would use ggplot2 to plot the distribution using the geom\_density() function.

The **quantitative test for normality** uses a statistical test to quantify the probability of whether or not the test data came from a normally distributed dataset.

If the p-value is greater than 0.05, the data is considered normally distributed.

Remember that most basic statistical tests assume an **approximate** **normal distribution**. Therefore, if our p-value is around 0.05 or more, we would say that our input data meets this assumption.

When data is skewed left, from the center of the distribution curve, there is a higher probability that extreme negative values exist within our dataset. When this occurs, the mean may no longer accurately reflect the central tendency of the data. Instead, we would use the median to describe the central tendency of the data. This skew is called negative skewed.

*When the mean is left skewed it would be best to use median to describe the central tendency of data*

When data is skewed right, from the center of the distribution curve, there is a higher probability that extreme positive values exist within our dataset. Once again, if this occurs, we would use the median to describe the central tendency of the data. This skew is called positive skewed.

*Right skewed gives the a longer tail on the right side. This tells us that there may be exaggerated positive outliers pulling the data to the right*

**Hypothesis testing**

In data science, we use statistical hypothesis testing to determine the probability of an event (or set of observations) under particular assumptions. In other words, we use statistical hypothesis testing to determine which of our hypotheses are most likely to be true

* The **null hypothesis** is also known as H0 and is generally the hypothesis that can be explained by random chance.
* The **alternate hypothesis** is also known as Ha and is generally the hypothesis that is influenced by non-random events.

## Hypothesis Testing in Five Steps

Regardless of the complexity of the dataset or the proposed question, hypothesis testing uses the same five steps:

1. Generate a null hypothesis, its corresponding alternate hypothesis, and the significance level.
2. Identify a statistical analysis to assess the truth of the null hypothesis.
3. Compute the p-value using statistical analysis.
4. Compare p-value to the significance level.
5. Reject (or fail to reject) the null hypothesis and generate the conclusion.

**H0 : PH = 0.5**

**Ha : PH ≠ 0.5**

**Where PH represents the probability of flipping heads.**

We must quantify our statistical results using probability

The **p-value**, or probability value, tells us the likelihood that we would see similar results if we tested our data again, if the null hypothesis is true. Therefore, we use the p-value to provide quantitative evidence as to which of our hypotheses are true.

To determine which hypothesis is most likely to be true, we compare the p-value against a significance level. A **significance level** (also denoted as alpha or ɑ) is a predetermined cutoff for our hypothesis test. When designing our hypothesis, we would determine the significance level based on the importance of our findings.

In most cases, a significance level of 0.05 is sufficient, but if our hypotheses are being used for critical decision-making (such as the performance of a drug or the durability of a helmet), we might want to use smaller cutoffs such as 0.01 or 0.001. Regardless of what significance level we select, we want to predetermine our cutoff prior to computing the p-value as to not introduce bias into our results. Refer to the following chart:

Table

Description automatically generated

If our hypotheses and statistical test are both two-tailed, use the statistical test p-value as is.

If our hypotheses are one-tailed, but our statistical test is two-tailed, divide the statistical test p-value by 2.

Once we have determined the significance level and the calculated p-value, we can complete our statistical analysis. If our calculated p-value is smaller than our significance level, we would state that there is sufficient statistical evidence that our null hypothesis is not true, and therefore we would reject our null hypothesis. Alternatively, if our calculated p-value is larger than our significance level, we would state that we do not have sufficient evidence to reject our null hypothesis, and therefore we fail to reject our null hypothesis.

***We must approximate truth using a subset of data.***

Assessing Error:

* **Type I error** (also known as a **false positive** error)—an error in which we reject the null hypothesis when it is actually true. In other words, the observations and measurements used in our statistical test should have been attributed to random chance, but we attributed them to something else.
* **Type 1 can be mitigated by:** A **type I error** can be limited by making your significance level smaller. A smaller significance level makes it harder to accidentally reject the null hypothesis when the data was truly random. This is also why our significance level (alpha or ɑ) is sometimes referred to as our false positive rate.
* **Type II error** (also known as a **false negative** error)—an error in which we fail to reject the null hypothesis when it is actually false. In other words, our analysis demonstrates that the observations were due to random chance, but they were not. The observations and measurements used in our statistical test failed to reflect an external force or influence to our problem.

**Random sampling** is a technique in data science in which every subject or data point has an equal chance of being included in the sample.

In R:

In most cases we will want to use the sample\_n() function to select sample data from a two-dimensional data object.

When it comes to our one-sample t-test there are five assumptions about our input data:

1. The input data is numerical and continuous. This is because we are testing the distribution of two datasets.
2. The sample data was selected randomly from its population data.
3. The input data is considered to be normally distributed.
4. The sample size is reasonably large. Generally speaking, this means that the sample data distribution should be similar to its population data distribution.
5. The variance of the input data should be very similar.

There are a number of metrics produced from the t.test()function, but for now we will only concern ourselves with the calculated p-value. Assuming our significance level was the common 0.05 percent, our p-value is above our significance level. Therefore, we do not have sufficient evidence to reject the null hypothesis, and we would state that the two means are statistically similar.

Two sample T-Test:

The second main form of the t-Test is a two-sample t-Test. Instead of testing whether a sample mean is statistically different from its population mean, the two-sample t-Test determines whether the means of two samples are statistically different. In other words, a two-sample t-Test is used to test the following hypotheses:

* H0 : There is **no statistical difference** between the two observed sample means.
* Ha : There is **a statistical difference** between the two observed sample means.

In R, we use the same t.test() function to calculate both a one-sample t-Test and two-sample t-Test. However, the two-sample t-Test arguments are slightly different:

* ***x*** is the first numeric vector of sample data.
* ***y*** is the second numeric vector of sample data.
* **alternative** tells the t.test() function if the hypothesis is one-sided (one-tailed) or two-sided (two-tailed). The options for the alternative argument are "two.sided,""less," or "greater." By default, the t.test() function assumes a two-sided t-Test.

**Pair T-Test:**

 two-sample t-tests are flexible and can be used for another purpose: to compare two samples, each from a different population. This is known as a **pair t-test**, because we pair observations in one dataset with observations in another. We use the pair t-test when:

* Comparing measurements on the same subjects across a single span of time (e.g., fuel efficiency before and after an oil change)
* Comparing different methods of measurement (e.g., testing tire pressure using two different tire pressure gauges)

The required arguments are slightly different from the unpaired two-sample t-test:

* **x** is the first numeric vector of sample data.
* **y** is the second numeric vector of sample data.
* **paired** tells the t.test() function to perform a paired t-test. This value must be set to TRUE.
* **alternative** tells the t.test() function if the hypothesis is one-sided (one-tailed) or two-sided (two-tailed). The options for the alternative argument are "two.sided," "less," or "greater." By default, the t.test() function assumes a two-sided t-test.

**ANOVA TEST:**

When dealing with large real-world numerical data, we're often interested in comparing the means across more than two samples or groups. The most straightforward way to do this is to use the **analysis of variance (ANOVA) test,** which is used to compare the means of a continuous numerical variable across a number of groups (or factors in R).

two different types of ANOVA tests:

* A **one-way ANOVA** is used to test the means of a single dependent variable across a single independent variable with multiple groups. (e.g., fuel efficiency of different cars based on vehicle class).
* A **two-way ANOVA** does the same thing, but for two different independent variables (e.g., vehicle braking distance based on weather conditions and transmission type).

Additionally, both ANOVA tests have assumptions about the input data that must be validated prior to using the statistical test:

1. The dependent variable is numerical and continuous, and the independent variables are categorical.
2. The dependent variable is considered to be normally distributed.
3. The variance among each group should be very similar

**Correlation analysis** is a statistical technique that identifies how strongly (or weakly) two variables are related.

Correlation is quantified by calculating a **correlation coefficient**, and the most common correlation coefficient is the Pearson correlation coefficient.

The **Pearson correlation coefficient** is denoted as "r" in mathematics and is used to quantify a linear relationship between two numeric variables. The Pearson correlation coefficient ranges between -1 and 1, depending on the direction of the linear relationship.

The following image is an example of an **ideal positive correlation** where r = 1. When two variables are positively correlated, they move in the same direction. In other words, when the variable on the x-axis increases, the variable on the y-axis increases as well:

Chart, scatter chart

Description automatically generated

The following image is an example of an **ideal negative correlation** where r = -1. When two variables are negatively correlated, they move in opposite directions. In other words, when the variable on the x-axis increases, the variable on the y-axis decreases.

Chart, scatter chart

Description automatically generated

The following image is an example of two variables with **no correlation** where r ≈ 0. When two variables are not correlated, their values are completely independent between one another.

**Chart, scatter chart

Description automatically generated**

For real-world data, it can be very difficult to determine if two variables are correlated, so we must use the Pearson correlation coefficient to calculate the correlation strength. Refer to the table below.

Table

Description automatically generated

To use the cor() function to perform a correlation analysis between two numeric variables, we need to provide the following arguments:

x is the first variable, which would be plotted on the x-axis.

y is the second variable, which would be plotted on the y-axis

 A **correlation matrix** is a lookup table where the variable names of a data frame are stored as rows and columns, and the intersection of each variable is the corresponding Pearson correlation coefficient. We can use the cor() function to produce a correlation matrix by providing a matrix of numeric vectors.

**linear regression** is a statistical model that is used to predict a continuous dependent variable based on one or more independent variables fitted to the equation of a line.

Diagram

Description automatically generated

One common form of categorical data is **frequency data**, where we record how often something was observed within a single variable.

The **chi-squared test** is used to compare the distribution of frequencies across two groups and tests the following hypotheses:

H0 : There **is no difference** in frequency distribution between both groups.

Ha : There **is a difference** in frequency distribution between both groups

**A/B testing** is a randomized controlled experiment that uses a control (unchanged) and experimental (changed) group to test potential changes using a success metric. A/B testing is used to test whether or not the distribution of the success metric increases in the experiment group instead of the control group; we would not want to make changes to the product that would cause a decrease in the success metric.

Df

* If the success metric is **numerical** and the **sample size is small**, a **z-score summary** **statistic** can be sufficient to compare the mean and variability of both groups.
* If the success metric is **numerical** and the **sample size is large**, a **two-sample t-test** should be used to compare the distribution of both groups.
* If the success metric is **categorical**, you may use a **chi-squared test** to compare the distribution of categorical values between both groups.