Future Connect Media

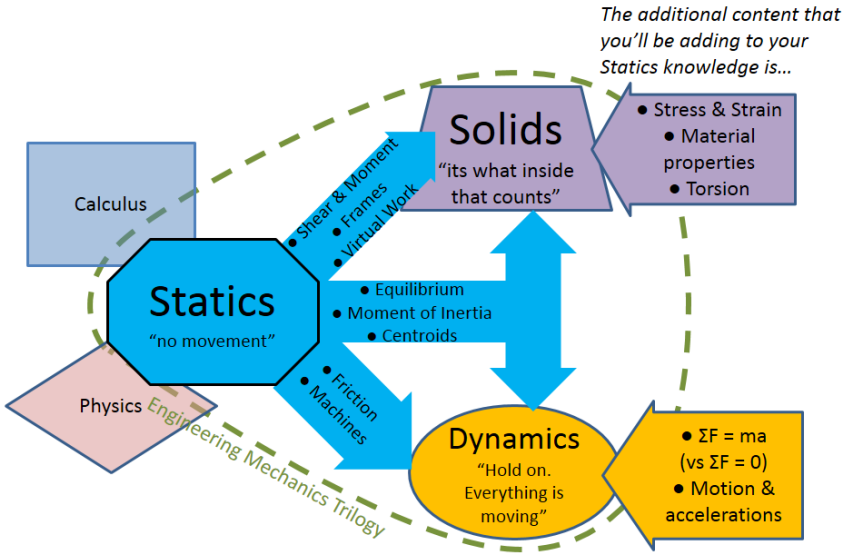
R Statistics

Part of Future Connect Media's IT Course

By Abdullah Hashmi

Statistics Introduction

Statistics is a branch of **mathematics** and a **fundamental** tool for **data analysis** and **decision-making**. It involves the collection, analysis, interpretation, and presentation of data. Statistics is used in a wide range of fields, including **science**, **business**, **economics**, **social sciences**, and more.



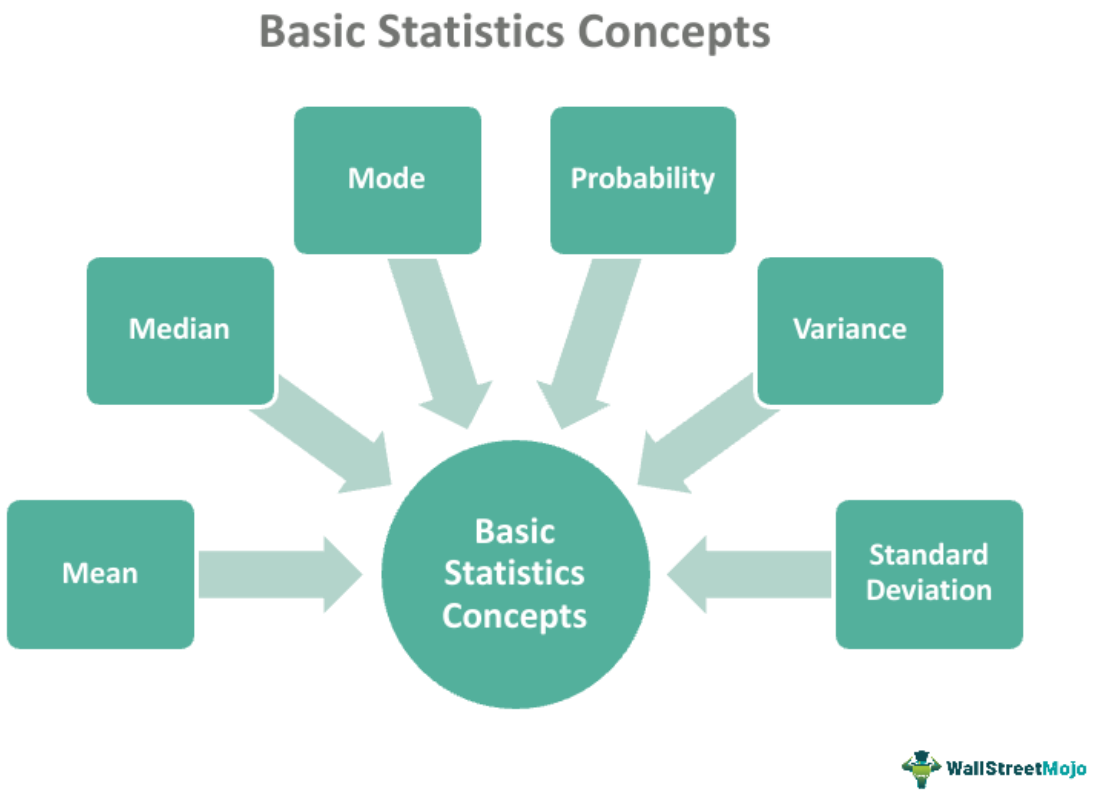
Statistics Introduction

Some basic statistical numbers include:

- Mean, median and mode
- Minimum and maximum value
- Percentiles
- Variance and Standard Deviation
- Covariance and Correlation
- Probability distributions

Statistics Introduction

The R language was developed by two statisticians. It has many built-in functionalities, in addition to libraries for the exact purpose of statistical analysis.



Data Set

Future Connect Media

A data set, also known as a dataset, is a **collection of data** that is organized in **a structured manner.** Data sets can come in various forms and can be used for a wide range of purposes, such as **statistical analysis**, **research**, **machine learning**, and more.

Obs	vehicle	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
1	Volvo 14	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2
2	Toyota C	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
3	Datsun 7	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
4	Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
5	Merc 240	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
6	Porsche	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
7	Fiat X1-	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
8	Honda Ci	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
9	Lotus Eu	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
10	Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1

Data set

There is a popular built-in data set in R called "**mtcars**" (Motor Trend Car Road Tests), which is retrieved from the 1974 Motor Trend US Magazine.

Example:

```
# Print the mtcars data set
mtcars
```

```
> mtcars
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D         24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230          22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C         17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Merc 450SE        16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL        17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 450SLC       15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Fiat 128          32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic       30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla    33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona     21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Dodge Challenger  15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin       15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28        13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Pontiac Firebird  19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Fiat X1-9         27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2     26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa      30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ford Pantera L    15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Ferrari Dino      19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Maserati Bora     15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Volvo 142E        21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

Get Information:

Use the **dim()** function to find the dimensions of the data set, and

the **names()** function to view the names of the variables:

Example:

```
Data_Cars <- mtcars # create a variable of the mtcars
data set for better organization

# Use dim() to find the dimension of the data set
dim(Data_Cars)

# Use names() to find the names of the variables from
the data set
names(Data_Cars)
```

Data set

Use the **rownames()** function to get the name of each row in the first column, which is the name of each car:

Example:

Data_Cars <- mtcars

rownames(Data_Cars)

```
> rownames(Data_Cars)
 [1] "Mazda RX4"           "Mazda RX4 Wag"       "Datsun 710"
 [4] "Hornet 4 Drive"      "Hornet Sportabout"   "Valiant"
 [7] "Duster 360"          "Merc 240D"           "Merc 230"
[10] "Merc 280"            "Merc 280C"           "Merc 450SE"
[13] "Merc 450SL"          "Merc 450SLC"         "Cadillac Fleetwood"
[16] "Lincoln Continental" "Chrysler Imperial"   "Fiat 128"
[19] "Honda Civic"         "Toyota Corolla"      "Toyota Corona"
[22] "Dodge Challenger"    "AMC Javelin"         "Camaro Z28"
[25] "Pontiac Firebird"    "Fiat X1-9"           "Porsche 914-2"
[28] "Lotus Europa"        "Ford Pantera L"      "Ferrari Dino"
[31] "Maserati Bora"       "Volvo 142E"
>
```

Max & Min

Future Connect Media

You learned from the R Math chapter that R has several built-in math functions. For example, the **min()** and **max()** functions can be used to find the lowest or highest value in a set:

Example:
Find the largest and smallest value of the variable hp (horsepower).

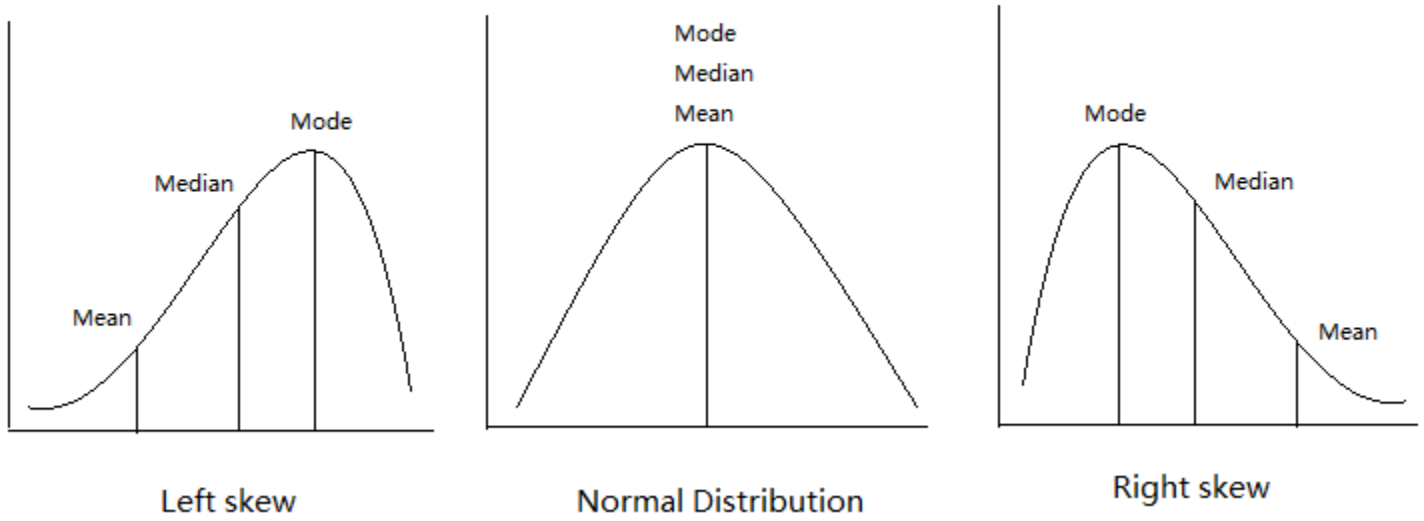
```
Data_Cars <- mtcars

max(Data_Cars$hp)
min(Data_Cars$hp)
```

```
> Data_Cars <- mtcars
>
> max(Data_Cars$hp)
[1] 335
> min(Data_Cars$hp)
[1] 52
>
```

Mean, Median, & Mode

In statistics, there are often three values that interests us:
•Mean - The average value
•Median - The middle value
•Mode - The most common value



Mean

To calculate the average value **(mean)** of a variable from the **mtcars** data set, find the **sum** of all **values**, and divide the sum by the **number** of **values**.

- Sorted observation of wt (weight)

1.513	1.615	1.835	1.935	2.140	2.200	2.320	2.465
2.620	2.770	2.780	2.875	3.150	3.170	3.190	3.215
3.435	3.440	3.440	3.440	3.460	3.520	3.570	3.570
3.730	3.780	3.840	3.845	4.070	5.250	5.345	5.424

Mean

Luckily for us, the mean() function in R can do it for you:

Example:

- Find the average weight (wt) of a car:

Data_Cars <- mtcars

mean(Data_Cars$wt)

```
> Data_Cars <- mtcars
>
> mean(Data_Cars$wt)
[1] 3.21725
>
```

Median

The median value is the value in the middle, after you have sorted all the values.
If we take a look at the values of the **wt** variable (from the **mtcars** data set), we will see that there are two **numbers** in the middle:

- Sorted observation of wt (weight)

1.513	1.615	1.835	1.935	2.140	2.200	2.320	2.465
2.620	2.770	2.780	2.875	3.150	3.170	3.190	3.215
3.435	3.440	3.440	3.440	3.460	3.520	3.570	3.570
3.730	3.780	3.840	3.845	4.070	5.250	5.345	5.424

Median

Example:

Find the mid point value of

weight (wt):

Data_Cars <- mtcars

median(Data_Cars$wt)

```
> Data_Cars <- mtcars
>
> median(Data_Cars$wt)
[1] 3.325
>
```

Mode

The mode value is the value that appears the most number of times.
R does not have a function to calculate the mode. However, we can create our own function to find it.

If we take a look at the values of the **wt** variable (from the **mtcars** data set), we will see that the numbers **3.440** are often shown:

- Sorted observation of wt (weight)

1.513	1.615	1.835	1.935	2.140	2.200	2.320	2.465
2.620	2.770	2.780	2.875	3.150	3.170	3.190	3.215
3.435	3.440	3.440	3.440	3.460	3.520	3.570	3.570
3.730	3.780	3.840	3.845	4.070	5.250	5.345	5.424

Mode

Future Connect Media

Example:

Data_Cars <- mtcars

names(sort(-table(Data_Cars$wt)))[1]

```
> names(sort(-table(Data_Cars$wt)))[1]
[1] "3.44"
>
```

Percentiles

Future Connect Media

Percentiles are used in statistics to give you a number that describes the value that a given percent of the values are lower than.

If we take a look at the values of the **wt** (weight) variable from the **mtcars** data set:

- Observation of wt (weight)

1.513	1.615	1.835	1.935	2.140	2.200	2.320	2.465
2.620	2.770	2.780	2.875	3.150	3.170	3.190	3.215
3.435	3.440	3.440	3.440	3.460	3.520	3.570	3.570
3.730	3.780	3.840	3.845	4.070	5.250	5.345	5.424

Percentiles

What is the 75. percentile of the weight of the cars? The answer is 3.61 or 3 610 lbs, meaning that 75% or the cars weight 3 610 lbs or less:

Example:

```
Data_Cars <- mtcars

# c() specifies which percentile you want
quantile(Data_Cars$wt, c(0.75))
```

```
> Data_Cars <- mtcars
>
> # c() specifies which percentile you want
> quantile(Data_Cars$wt, c(0.75))
   75%
3.61
>
```

Quartiles:

Quartiles are data divided into four parts, when sorted in an ascending order:

1. The value of the first quartile cuts off the first 25% of the data
2. The value of the second quartile cuts off the first 50% of the data
3. The value of the third quartile cuts off the first 75% of the data
4. The value of the fourth quartile cuts off the 100% of the data

- Use the quantile() function to get the quartiles.

Percentiles

If you run the **quantile()** function without specifying the **c()** parameter, you will get the percentiles of 0, 25, 50, 75 and 100:

Example:
Data_Cars <- mtcars

quantile(Data_Cars$wt)

```
> Data_Cars <- mtcars
>
> quantile(Data_Cars$wt)
     0%     25%     50%     75%    100%
1.51300 2.58125 3.32500 3.61000 5.42400
>
```

Thank you

Future Connect Training Institute

Website: https://www.fctraining.co.uk/