5 Main Steps of Data Analysis

1. Define the questions that you want to answer through the analysis

- clearly define the problem
- make hypothesis or research questions
- identify what types of data you will need and where it'll come from

2. Collect the Data

- primary sources: internal data that the company already has (eg: CRM software, email marketing tools etc.)
- secondary sources: public/external data (eg: government, Google Trends, WHO etc.)

3. Clean the Data

- remove duplicates
- handle anomalies and missing data

4. Analyze the Data

- regression analysis
- cluster analysis
- time-series analysis

5. Interpret and share the results

- visualization

Terms

- ETL = Extract, Transform, Load
- Dax = Data analyst expressions
- Copilot = Feature in Excel, like ChatGPT powered by Microsoft

Notes

1. Concatenate values from 2 string columns:

=A2&B2 if A2='x' and B2='y', then this will produce "xy"

=A2&"-"&B2 adds a hyphen (-) between the two cell values

- the quotation marks are necessary to put text characters between two values

2. Range = collection of cells

Functions

1. Logical

a. IF(condition, returned_value_if_true, returned_value_if_false)

b. AND(condition1, condition2)

c. OR(condition1, condition2)

d. IFS(condition1, value1_if_cond1_is_true, condition2, value2_if_cond2_is_true)

2. Math

a. COUNT(B:B) - counts all non-blank NUMERICAL cells of the given range (column B)

b. COUNTA(B:B) - counts all non-blank cells of the given range

c. COUNTIF(B:B, $A2) - counts how many values of B is equal to the value of A2

d. COUNTIFS(range1, condition1, range2, condition2)

- counts how many cells of range1 match condition1 & how many cells of range 2 match condition2

e. SUM(B:B)

f. SUMIF(range_to_assess, condition_to_assess, range_to_add)

g. SUMIFS(range_to_add, range1_to_assess, condition1_to_assess, range2_to_assess, condition2_to_assess)
h. AVERAGE(B:B)
i. AVERAGEIF
j. AVERAGEIFS
k. MIN(B:B)
l. MAX(B:B)

3. Statistical

Median

It is the number in the middle of the sorted set (smallest to largest)

Median is better than average for statistics because average is affected by the outliers in the dataset

Standard Deviation (σ)

= how much the data point deviates from the standard (mean)

= how spread out the data is

If SD is low, that means the data is closely clustered around the mean/average

If SD is high, that means the data is dispersed over a wider range of values

SD is used to understand if a specific data point is standard and expected, or unusual and unexpected

A data point's distance from the mean can be measured by the number of standard deviations (i.e. if it is above or below the mean)

1σ = 68% = 68% of data fall within 1 SD of the mean = 34.1% on either side of the median

2σ = 95% = (13.6+34.1)% on either side of the median

3σ = 99.7%

SD is used when the distribution of data is approximately normal (like a bell curve)

SD calculation:

Population	Sample
$\sigma = \sqrt{\frac{\Sigma(x_i-\mu)^2}{n}}$	$S = \sqrt{\frac{\Sigma(x_i-\bar{x})^2}{n-1}}$
μ - Population Average xi - Individual Population Value n - Total Number of Population	x - Sample Average x_i - Individual Population Value n - Total Number of Sample

Quartile

Quartile 0 = 0% = 0th percentile = MINIMUM

Quartile 1 = 25% = 25th percentile = 1st Quartile = first 25% of the sorted data or the histogram?

Quartile 2 = 50% = 50th percentile = 2nd Quartile = MEDIAN

Quartile 3 = 75% = 75th percentile = 3rd Quartile

Quartile 4 = 100% = 100th percentile = 4th Quartile = MAXIMUM

[0 and 4 won't work for QUARTILE.EXC()]

Interquartile Range = Q3-Q1

a. MEDIAN(B:B)
b. STDEV.P(B:B) - standard deviation of the population
c. STDEV.S(B:B) - standard deviation of a sample
d. QUARTILE.INC(range, quartile_value)
    i. eg: QUARTILE.INC(A:A, 2) = 2nd quartile (median)

ii. Inclusive (percentile range of 0 to 1 inclusive)

iii. Considers the median

iv. Used often for odd-numbered sample size

e. QUARTILE.EXC(range, quartile_value)

i. Exclusive (percentile range of 0 to 1 exclusive)

ii. Doesn't consider the median

iii. Used for even-numbered sample size

iv. 0 and 4 won't work for the 2nd argument

f. RANK(value_to_rank, entire_range, 0)

i. 0 = descending order, 1 = ascending order, this is optional argument

ii. Ranks the given value out of all values in the given range

iii. Returns the rank of a value in a list of values . The rank of a value is its size relative to other values in a list. (If you were to sort the list, the rank of the number would be its position.)

g. MODE(A:A)

i. The most frequently occurring value in the given range

4. Arrays

a. =A1:A5 - copies a range of values from column A to an empty column

- You cannot delete a value of the array without deleting the whole array from A

b. A1:A5 * B1:B5 -> multiples values of A and B, row by row (A1*B1, A2*B2,...,Ax*Bx)

c. UNIQUE(A2:A3000) - Gets all the unique values in the specified range

- Ctrl + Shift + Down -> shortcut to get to the bottom of a column while selecting the range

d. SORT(A2#) - sorts the entire array in R2 in alphabetical order

e. SUMPRODUCT(A1:A5, B1:B5) - row-wise multiplication and then sum of those products

- It can also return the sum of all values in ONE array : SUMPRODUCT(A2#)

f. TEXT(value, format_text)

- eg: TEXT(A1, "mmm") -> converts a date-time value into its 3-letter month name, like "Apr"
- TEXT(A1, "mmmm") -> converts the date-time into the full name of the month, like "April"
- Similarly:
  - 'ddd' -> day (eg: Wed)
  - 'dddd' -> full day name (eg: Wednesday)
  - 'yy' -> year, shorter (eg: 24)
  - 'yyyy' -> year, full (eg: 2024)
- TEXT(A2:A3000, "mmmm") -> converts a date to a month from the array of date values
- --W2# -> converts an array of boolean values into 0 or 1
  - One Minus sign can make it negative (-1). That's why we have put two minus signs
  - W2# -> the array that starts from W2

5. Lookup

- **VLOOKUP**(value_you_are_looking_for, table_array_where_you_are_looking, column_index_number)
  - Vertical lookup
  - Column index number : if you start counting from the leftmost column of the referenced table array, this is the number of the column where you will find the value you are looking for
  - eg: VLOOKUP(T5, M1:P3000, 3) - The table is from M to P. So, if we want to look for the value T5 in the column O, we have to provide the index number 3, because M=1, N=2, O=3.
  - Table/Column must be sorted in ascending order by default. If not, change or add the 4th parameter to FALSE
    eg: VLOOKUP(T5, M1:P3000, 3, FALSE)
    FALSE will look for an exact match, TRUE will look for an approximate match
  - VLOOKUP only provides the first instance of matching value/criteria. That's why XLOOKUP is better
- HLOOKUP(value_you_are_looking_for, table_array_where_you_are_looking, row_index_number)
  - Horizontal lookup
  - For horizontally oriented tables
  - eg: HLOOKUP("Jan", B1:M7, 2, FALSE)
- XLOOKUP()

6. Text
7. Date & Time
8. V