**Statistics cheatsheet**

**Introduction**

The introduction of a statistical summary to Adventure Works’ Sales report ignited a spark of excitement. The management team recognized the significance of applying statistics to Adventure Work’s datasets and agreed to explore the full capabilities of statistical tools. As you prepare to adapt to these needs, you decide to explore the statistical capabilities of Microsoft Power BI. Power BI unlocks the full potential of statistics, providing a diverse toolbox of statistical functions, crafted to cater to the specific needs of today’s businesses.

This reading aims to provide you with a list of statistical functions in Power BI to better understand the functionality of each statistical function, a sample dataset is provided below. By observing the results of each function with the sample data, you will be able to grasp the practical application of each DAX function and how they can be used in real-world scenarios.

Sample sales dataset:

| **ItemPrice** | **ItemPrice\_with Text** | **QuantitySold** |
| --- | --- | --- |
| 120 | 120 | 5 |
| 140 | True | 7 |
| 180 | Random Text | 3 |
| 130 | 130 | 6 |
| 200 | 180 | 4 |
| 240 | False | 9 |
| 160 | 280 | 2 |
| 190 |  | 8 |

For each DAX function seen in the below reading, you will find its expression, a clear definition, and an example of how it works with the provided sample dataset.

**Average and median functions**

**Average** and **median** functions are essential in almost any dataset containing numerical columns. For instance, in an online store, you can use these functions to evaluate average and median order values, providing insights into customer spending habits. Similarly, as a teacher assessing student test scores, calculating average and median scores offers a quick overview of class performance while identifying outliers.

**Average functions**

**1. AVERAGE(<column>):** The average value in the given column is calculated by summing all values and dividing it by the number of data points.

Example: **AVERAGE([ItemPrice])** For example, if the **ItemPrice** column contained 8 values, the formula would be calculated as: (120 + 140 + 180 + 130 + 200 + 240 + 160 + 190) / 8 = 170

**2. AVERAGEA(<column>)**: The average value in the given column is calculated, handling non-numeric data types based on specific rules.

**The rules are as follows:** - Values that evaluate to TRUE count as 1.

- Values that evaluate to FALSE count as 0 (zero).

- Values that contain non-numeric text count as 0 (zero).

- Empty text ("") counts as 0 (zero).

Example: **AVERAGEA([ItemPrice\_withText])**

For example, if the **ItemPrice\_withText** column contained 8 values, with non-numeric values being counted as 0, the formula would be calculated as:(120 + 1 + 0 + 130 + 180 + 0 + 280 + 1) / 8 = 89

**3. AVERAGEX(<table>, <expression>)**: The average of the expression in the specified table is calculated. It evaluates the expression for each row in the table and uses the average of the results.

Example: **AVERAGEX(Sales, [ItemPrice] \* [QuantitySold])** For example, if the **ItemPrice** column and the **QuantitySold** column each contained 8 values, the formula would be calculated as: ((120 \* 5) + (140 \* 7) + (180 \* 3) + (130 \* 6) + (200 \* 4) + (240 \* 9) + (160 \* 2) + (190 \* 8)) / 8 = 962.50

**Median function**

**1. MEDIAN(<column>)**: Returns the median value of a numeric column.

Example: **MEDIAN(Sales[ItemPrice])** For example, if the **ItemPrice** column in the **Sales** table contained 2 values, the formula would be calculated as: (160 + 180) / 2 = 170

**Variance and standard deviation functions**

Variance and standard deviation are both measures of how spread out or dispersed a set of data points is, playing a critical role when data variability needs to be measured. In the context of financial portfolio management, computing variance and standard deviation of stock returns assists in gauging the riskiness of investments. In manufacturing, monitoring paint thickness in a car plant involves determining the variance and standard deviation of measurements to highlight process inconsistencies. The calculation of these two functions differs slightly depending on whether the dataset represents the entire population of data points or just a sample from it. In DAX, this differentiation is denoted by a P for population and an S, for a sample in their formulas. Using the appropriate variance or standard deviation formula is essential to ensure accurate and meaningful analysis based on the nature of the data available.

**Variance and standard deviation using a sample population**

Assuming the column on which the variance/deviation will be calculated refers to a sample population:

**1. VAR.S(<column>)**: Returns the variance of a column containing a sample population.

Example: **VAR.S(Sales[ItemPrice])** = 1.629

**2. STDEV.S(<column>)**: Returns the standard deviation of a column containing a sample population.

Example: **STDEV.S(Sales[ItemPrice])**= 40

**Variance and standard deviation using the entire population**

Assuming the column on which the variance/deviation will be calculated contains the entire population of data points:

**1. VAR.P(<column>)**: Returns the variance of a column, assuming that the column refers to the entire population.

Example: **VAR.P(Sales[ItemPrice])** = 1.425

**2. STDEV.P(<column>)**: Returns the standard deviation of a column containing an entire population.

Example: **STDEV.P(Sales[ItemPrice])** = 38

**Count, min and max functions**

Count, min and max are versatile functions with various applications in any data analysis project. For example, in customer support, counting resolved tickets, finding minimum resolution times, and identifying maximum customer satisfaction ratings, showcase individual agent achievements. Similarly, in retail inventory management, counting items, finding minimum stock levels, and identifying maximum sales volumes contribute to optimizing inventory levels.

**Count functions**

**1. COUNT(<column>)**: Counts the total number of rows, containing any or no value.

Example: **COUNT([ItemPrice\_withText])** = 8

**2. COUNTBLANK(<column>)**: Counts the number of blank or empty rows in a column.

Example: **COUNTBLANK([ItemPrice\_withText])**= 1

**3. DISTINCTCOUNT(<column>)**: Counts the number of distinct values in the column.

Example: **DISTINCTCOUNT([ItemPrice\_withText])** = 8

**Max function**

**1. MAX(<column>)**:Returns the largest numeric value or largest string in a column. Ignores logical values.

Example **MAX([ItemPrice\_withText])** = “TRUE”

**Note:** As the MAX function ignores logical values, TRUE and FALSE in the table are interpreted as texts. Texts are considered “larger” than numbers in DAX alphabetically, so the largest value in the column is TRUE, as it’s the last text in alphabetical order.

**Min function**

**1. MIN(<column)**: Returns the smallest numeric value or shortest string in a column. Ignores logical values.

Example MIN**([ItemPrice\_withText])** = “”

**Note:** As the MIN function also ignores logical values, same as the MAX function, the sequence of blank values, numbers and texts is followed in ascending order. A blank value in the sample column is considered as the minimum value.

**Combination and permutation functions**

Combination and permutation functions in statistics calculate the number of combinations for a given number of items. The difference between the two functions is that in a combination function, the output order of items is not important, while in a permutation function, the output order of the items is significant. For an easy example, let’s calculate the odds of winning a lottery ticket, where you have to guess 6 numbers out of 49, using these two functions in Power BI.

**Combination function**

**1. COMBIN(n,k**): Returns the number of ways **k** items can be selected from **n**, where the order does not matter.

Example: **COMBIN(49,6)** = 13.983.816

**Permutation function**

**1. PERMUT(n,k):** Returns the number of ways **k** items can be selected from **n**, where the order does matter.

Example: **PERMUT(49,6)** = 10.068.347.520

**Conclusion**

This statistical cheat sheet covered the most used statistical functions in Power BI, leveraging the capabilities of the DAX language. It provided a comprehensive range of functions, from simple calculations like averages and medians to more complex ones like standard deviation, and probability functions, calculating the odds of winning a lottery ticket. By using the practical examples with the provided sample dataset in this reading, you are now able to identify, explain and use the different statistical functions available in DAX.