

How to Calculate Average with OR Conditions Using Excel's AVERAGEIFS Function

Authored by
stats writer

February 15, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Calculate Average with OR Conditions Using Excel's AVERAGEIFS Function*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=130877>

Excel: How to Use the AVERAGEIFS Function with an OR Condition

In the realm of **data analysis**, **Microsoft Excel** remains a cornerstone tool for professionals across various industries. One of its most powerful capabilities is the ability to perform conditional calculations, allowing users to extract meaningful insights from vast datasets. Among these tools, the **AVERAGEIFS** function is frequently utilized to determine the mean value of a specific range based on multiple criteria. However, a common challenge arises when users need to apply **Boolean logic** that deviates from the standard parameters of the function.

By default, the **AVERAGEIFS** function is designed to operate using **AND** logic. This means that for a cell to be included in the final average, it must satisfy every single condition specified within the formula. While this is ideal for narrowing down data to highly specific subsets, it poses a significant limitation when the objective is to include data that meets any one of several different criteria. For instance, if you wish to find the average performance of multiple departments or categories simultaneously, the standard **AVERAGEIFS** approach will fail to deliver the desired result because it searches for rows that match all criteria at once.

To overcome this limitation and implement **OR** logic within your **spreadsheet** calculations, a more flexible structural approach is required. Instead of relying solely on the built-in multi-criteria function, users can leverage the power of an **array formula**. This method combines the standard **AVERAGE** function with a nested **IF** statement, utilizing mathematical addition to represent the "OR" condition. This sophisticated technique ensures that your data processing remains both accurate and comprehensive, capturing all relevant data points that meet at least one of your specified requirements.

Understanding the Syntax for OR Logic in Excel

Transitioning from simple conditional functions to **array programming** logic requires a clear understanding of the underlying **syntax**. In Excel, when you add two logical tests together, the application treats the operation as an **OR** condition. If the first test is true (resulting in 1) or the second test is true (resulting in 1), the sum is greater than zero, triggering the **IF** statement to include the corresponding value in the calculation. This is a fundamental concept in **computational logic** that allows for much more granular control over how **data sets** are aggregated.

The specific formula structure used to achieve this is as follows:

```
=AVERAGE(IF((A2:A11="Guard")+(A2:A11="Forward"),B2:B11))
```

In this expression, the **IF** function evaluates two separate conditions within the range **A2:A11**. By

using the plus sign (+) between the conditions, we instruct Excel to identify any row where the value is either "Guard" **OR** "Forward." When a match is found, the corresponding numerical value from the range **B2:B11** is passed to the **AVERAGE** function. This bypasses the restrictive nature of the standard **AVERAGEIFS** function, providing a robust solution for complex **statistical** queries.

It is important to note that in older versions of **Microsoft Excel**, such as Excel 2016 or 2019, you may need to press **Ctrl+Shift+Enter** to activate this as a legacy **array formula**. However, in modern versions like **Office 365** or Excel 2021, the **dynamic array** engine handles this calculation automatically. This evolution in software design has made it significantly easier for users to implement advanced **Boolean logic** without deep technical expertise in formula nesting.

A Practical Example: Analyzing Basketball Performance

To illustrate the utility of this method in a real-world scenario, consider a **database** tracking the performance of basketball players. In such a dataset, you might have columns for player names, their specific positions, and the points they scored during a season. Often, a coach or analyst might want to determine the average scoring output for a group of positions, such as all "Guards" and "Forwards" combined, to compare backcourt and frontcourt efficiency.

Suppose we have the following dataset in Excel that shows the position and points scored by various basketball players:

	A	B	C	D	E	F
1	Position	Points				
2	Guard	22				
3	Guard	14				
4	Forward	17				
5	Center	30				
6	Forward	35				
7	Center	18				
8	Guard	11				
9	Forward	19				
10	Forward	24				
11	Center	16				
12						
13						
14						
15						
16						

In the image above, the data is organized into two primary columns: **Position** (Column A) and **Points** (Column B). The goal is to isolate the points scored by players who occupy specific roles while ignoring others, such as those labeled "Center." Using a standard **AVERAGEIFS** function here would be impossible if we wanted to check for two different positions in the same column, as a single cell cannot be both a "Guard" and a "Forward" simultaneously.

By applying our specialized **OR** logic formula, we can efficiently aggregate the data for both categories. This approach is particularly useful when dealing with large **information systems** where manual filtering would be too time-consuming or prone to human error. By automating the process through a formula, the **data integrity** is maintained, and the results are updated instantly if any underlying player statistics change.

Implementing the Formula for Accurate Results

To execute this calculation, you would enter the formula into a target cell where you want the average to appear. This requires selecting the appropriate ranges that correspond to your specific **business analytics** needs. The precision of the range selection is vital; ensuring that the criteria range and the average range are of identical length is a prerequisite for any successful **algorithm** in Excel.

We can use the following formula to do so:

```
=AVERAGE(IF((A2:A11="Guard")+ (A2:A11="Forward"),B2:B11))
```

The following screenshot shows how to use this formula in practice:

	A	B	C	D	E	F
1	Position	Points		Avg. Points for Guards or Forwards		
2	Guard	22		20.28571429		
3	Guard	14				
4	Forward	17				
5	Center	30				
6	Forward	35				
7	Center	18				
8	Guard	11				
9	Forward	19				
10	Forward	24				
11	Center	16				
12						
13						
14						
15						

Once the formula is applied, Excel evaluates each row in the **Position** column. If it finds "Guard," it marks the row as true. If it finds "Forward," it also marks the row as true. The **IF** function then collects all the point values from the **Points** column for these "true" rows. Finally, the **AVERAGE** function calculates the arithmetic mean of that collected subset. This multi-step logical process is performed almost instantaneously, providing a seamless user experience.

As demonstrated in the example, the resulting average points scored by players with a position of Guard or Forward is **20.2857**. This single value represents a weighted insight into the performance of specific player groups, allowing for better strategic decision-making. Whether you are managing sports statistics, financial records, or inventory data, the ability to merge criteria using **OR** logic is an essential skill for any **data scientist** or administrative professional.

Manual Verification and Data Validation

In any **quantitative research** or data reporting task, it is considered best practice to verify your results through manual calculation or alternative methods. This ensures that the **logic error** risk is minimized and that the formula is behaving exactly as intended. Validation is the cornerstone of **quality assurance** in data management, providing confidence in the reports presented to stakeholders.

We can verify this is correct by manually calculating the average of all points values for players who are Guards or Forwards. By looking at the raw data, we identify the following scores for the relevant players: 22, 14, 17, 35, 11, 19, and 24. Summing these values gives us a total of 142

points scored across the seven players who met our **OR** criteria.

Average Points: $(22 + 14 + 17 + 35 + 11 + 19 + 24) / 7 = \mathbf{20.2857}$

This manual calculation perfectly matches the value derived from our **expression**. Such consistency confirms that the addition of logical arrays effectively mimics the **OR** operator. Understanding this manual process also helps users troubleshoot their formulas if the output seems unexpected, as they can trace the calculation steps back to the source data.

Advanced Considerations for Logical Operations

While the combination of **AVERAGE** and **IF** is highly effective, it is important to consider the broader context of **functional programming** within Excel. For instance, if you have more than two criteria, you can continue adding segments to the formula, such as ``(range="A")+(range="B")+(range="C")``. This scalability makes the **array formula** approach superior to many other workarounds, such as creating multiple helper columns or using complex **Pivot Tables** for simple averages.

Furthermore, users should be aware of how **Excel** handles empty cells or text within the numerical range. The **AVERAGE** function is designed to ignore non-numeric values, but the **IF** statement must be carefully constructed to avoid returning errors. Utilizing **error handling** functions like **IFERROR** can further refine your **software engine** within the spreadsheet, ensuring that your dashboard remains clean and professional even when data is missing.

Another alternative for those using the latest versions of **Microsoft 365** is the **FILTER** function. The **FILTER** function can be used to create a dynamic list of values that meet the **OR** criteria, which can then be wrapped in a standard **AVERAGE** function. This modern approach is often more intuitive for new users and reflects the current direction of **cloud-based** spreadsheet technology, where **reactive programming** models are becoming the standard.

Conclusion and Further Learning

Mastering the **AVERAGEIFS** function with **OR** logic is a significant milestone in becoming an advanced **power user** of Excel. By moving beyond the default **AND** constraints, you unlock the ability to perform more nuanced **data analysis** and generate reports that truly reflect the complexity of real-world variables. The techniques discussed here--specifically the use of array formulas and **Boolean addition**--are foundational skills that apply to many other functions within the software.

As you continue to develop your skills, it is beneficial to explore how these logical principles apply to other aggregation functions like **SUMIFS**, **COUNTIFS**, and **MAXIFS**. Often, the same logic used

to calculate an average can be adapted to find the total sum or the maximum value within a filtered dataset. This versatility is what makes **Microsoft Office** such a dominant force in the professional world, providing a unified environment for diverse **computational** tasks.

The following tutorials explain how to perform other common operations in Excel, providing further opportunities to enhance your **digital literacy** and technical proficiency. By consistently practicing these methods and staying updated with official documentation, you will be well-equipped to handle any data challenge that comes your way.

Advanced logical functions for data filtering.

Creating dynamic dashboards with array formulas.

Best practices for data validation and error checking.

Optimizing spreadsheet performance for large datasets.

ARABPSYCHOLOGY.COM