

Part 1.5 Grouping Data

Like many restaurants and retailers, Starbucks offers a variety of discounts and promotions to help entice customers and boost their business.

But how can they tell which promotions are working?

This table contains a record of every time an offer has been sent to a customer (an “Offer”) or used for a purchase (a “Transaction”).

Table with Event, Offer ID, Revenue (\$)

Can you tell from the table which offer types are most effective? Clearly not. As is typical for large datasets, it's very difficult to get a sense of the data and the relationships in the data without doing something else.

We also can't make a useful visualization directly from the table either. The data just isn't in the right format.

Here is a more useful-looking table of total revenue from each promotion.

Table with Offer ID, Revenue

How did we get from the original table to this?

Added up the revenue associated with each type of offer.

We made this table from the original data by **grouping** rows by their Offer ID, and **summing** the revenues. Every good data analysis program has this functionality.

Reproduce the table above by grouping and summing.

To reproduce the table above we need to group by Offer ID and then sum.

Notice that our grouped table has 5 rows, corresponding to the 5 different Offer IDs. But that's not the only column we can group by.

How many rows would you expect if we group by “Event”?

Here's what we get when we group by “Event” and then sum.

Table with Event, Revenue

There are only two rows, because there are only two types of Event, and the sum value for “Offer” is 0 because there is no revenue from these Events.

Produce a table that counts the number of each type of Event.

To count the number of each Event we should group by Event and then count.

Now we have an idea of how grouping works, let's use it to find out which offer has been most successful.

Grouping and Filtering

Table with Offer ID, Revenue

We saw earlier that of all Starbucks' offers, *2off10* brought in the most revenue in total. But we also want our offers to encourage customers to spend more individually.

Which command would help us find the average spend per transaction?

Table with Offer ID, Revenue/mean

When we group by Offer ID and take the mean we get surprisingly small numbers. How can the average spend be only 8.81 for the offer "5 off when you spend 20"?

We forgot to remove the Offer rows.

All the Offer rows in the table had a revenue of 0. Before we can find the mean revenue, we need to filter for only Transaction rows.

Filter for Transactions only and then find the mean revenue for each Offer ID.

Table with Event, Revenue, Offer ID

Filter Event == Transaction, Group by == Mean

To find the mean transaction amount we should filter for Event = Transaction, then group by Offer ID and get the mean.

We can see now that *5off20* brings in the most Revenue per Transaction. Do you think that means it is the most effective offer?

It depends what we mean by effective.

You have to spend at least \$20 to unlock this offer, so it's not surprising that the average spend is high.

What other measure of a promotion's success could be helpful?

Both Transactions per Offer and Revenue per Offer seem like a relevant measure of a promotion's success. We'll explore each of them before we make our final recommendation.

Drawing Conclusions

To measure the success of each Starbucks promotion, we need to know how many times it was sent out in the first place.

Produce a table showing the number of Offers for each Offer ID.

Table with Event, Revenue, Offer ID

Filter on Event == Offer, Group by Offer ID == Count

To find the number of Offers for each Offer ID we should filter for Event = Offer, group by Offer ID and then count.

To see what proportion of Offers resulted in Transactions, we will also need to count the number of Transactions.

Produce a table showing the number of Transactions for each Offer ID .

Table with Row, Event, Revenue, Offer ID

Filter on Event == Transaction, Group by Offer ID == Count

To find the number of Transactions for each Offer ID we should filter for Event = Transaction, group by Offer ID and then count.

The table below has both the columns you just worked out as well as the total revenue we found earlier.

Table with Offer ID, Offers, Transactions, Revenue

Make a new column showing the Revenue per Offer for each of the Offer IDs.

To create the Revenue per Offer column we should divide revenue by number of offers. Selecting "Revenue / Offers" will apply this formula to each row of the table.

Figure Showing Revenue per Offer

It looks like 5off20 is the most effective by this metric, though at \$5 per transaction it's fairly expensive for Starbucks. Let's look at Transactions per Offer — the portion of offers that actually get used.

We'll start by constructing the new column "Transactions/ Offer"

Figure Showing Revenue per Offer

5off20 doesn't look so good on this metric — less than half of the 5off20 offers sent to customers are ever used.

Show Both Figures

Which two offers rank in the top three on **both** Transactions per Offer and Revenue per Offer?

2off10 ranks second on both metrics, while 3off7 is best for Transactions per Offer and third best for Revenue per Offer.

3off7 and 2off10 do a similar job of bringing in customers and revenue. But the cost of offering these discounts for the cafe is not the same.

Which would you recommend as the most cost effective offer?

2 off when you spend 10.

2off10 is a smaller discount than *3off7* both in dollars (*2vs.3*) and as a percentage of the sale (20% vs. 43%)