Cardinality Estimation in SQL Server - Quick Notes, Part 2

Hi Friends,

A few days back I penned down a few quick notes about Cardinality Estimation in SQL Server. You can read [Part 1 here](http://www.sqlmaestros.com/sql-server-cardinality-estimation-quick-notes-part-1/).

SQL Server Cardinality Estimation is a complicated topic until you understand the internals of SQL Server Statistics. Here are some quick notes that will answer some common questions that you generally have:

How does SQL Server Query Optimizer estimate a default cardinality? Be sure to check out [Part 1](http://www.sqlmaestros.com/sql-server-cardinality-estimation-quick-notes-part-1/)

Let’s see another example (turn ON Actual Execution Plan or press Ctrl + M)

useAdventureWorks2012

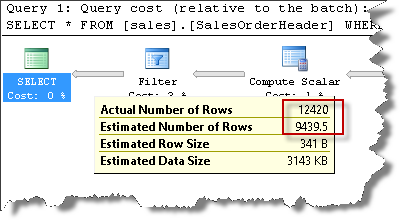
go

-- Turn On Actual Execution Plan

select\*fromsales.SalesOrderHeader

whereTaxAmt+TotalDue>2000

go



You can observe that the Estimated Number of Rows is 9439.5

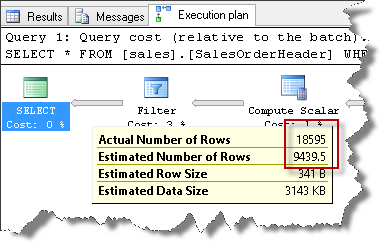
Let’s change the parameter value.

-- Turn On Actual Execution Plan

select\*fromsales.SalesOrderHeader

whereTaxAmt+TotalDue>472.3108

go



You again see that the Estimated Number of Rows is 9439.5

No matter what constant you specify, SQL Optimizer will always computer 9439.5

Why?

The optimizer cannot use the Statistics simply because of the preceding expression (TaxAmt + TotalDue). Even if there are statistics for TaxAmt and TotalDue columns, they are of no use because for SQL Server, it’s the left side expression value that is now considered, not the columns. So where does 9439.5 come from?

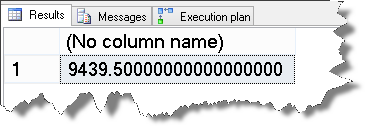
That value is 30% of the total number of rows. This is the default estimation done in case expressions are used.

Total number of rows are 31465.

30% of 31465 = 9439.5

selectcast((1.\*30/100)asdecimal(18,17))\*count(totaldue)

fromsales.SalesOrderHeader



That’s how the default estimation was computed in this case. Hope the quick note was helpful.