========================================  
Autovacuum Configuration for Individual Table in PostgreSQL  
========================================  
  
Let's assume the table is already created as follows:  
  
CREATE TABLE emp2024 (eid INT PRIMARY KEY, ename VARCHAR(100),esal NUMERIC );  
  
To insert 10 million rows of data:  
  
INSERT INTO emp2024 (eid, ename, esal)  
SELECT gs, 'Employee ' || gs, (RANDOM() \* 50000 + 30000)::NUMERIC   
FROM generate\_series(1, 10000000) AS gs;   
  
ALTER table emp2024  
SET (  
autovacuum\_vacuum\_threshold = 10000, --> Trigger vacuum after 10,000 dead tuples  
autovacuum\_vacuum\_scale\_factor = 0.2, --> Trigger vacuum when 20% of the table rows are dead  
autovacuum\_analyze\_threshold = 5000, --> Trigger analyze after 5,000 changes  
autovacuum\_analyze\_scale\_factor = 0.1 --> Trigger analyze when 10% of the table rows change  
);  
  
The autovacuum\_vacuum\_threshold can be calculated based on the number of tuples in the table:  
  
Threshold = table\_row\_count \* autovacuum\_vacuum\_scale\_factor  
  
For example, if the table has 1 million rows and you want to trigger a vacuum when 10% of the rows are updated or deleted:  
  
SELECT reltuples AS row\_count FROM pg\_class WHERE relname = 'emp2024';  
  
Using the formula:  
  
Threshold = 1,000,000 \* 0.2 = 200,000  
  
Similarly, for auto-analyze, the autovacuum\_analyze\_threshold can be calculated based on the number of tuples:  
  
Threshold = table\_row\_count \* autovacuum\_analyze\_scale\_factor  
  
For example, if the table has 500,000 rows and you want to trigger an analyze when 5% of the rows are modified:  
  
SELECT reltuples AS row\_count FROM pg\_class WHERE relname = 'emp2024';  
  
Using the formula:  
  
Threshold = 500,000 \* 0.05 = 25,000  
  
If you want to disable autovacuum for a specific table (which is generally discouraged), you can use:  
  
ALTER TABLE emp2024 SET (autovacuum\_enabled = false);  
  
This completely disables autovacuum for that table.  
  
Best Practices for Autovacuum Settings:  
  
For large tables, you may want to make the autovacuum\_vacuum\_scale\_factor higher (e.g., 0.1 to 0.3), so that vacuum runs less frequently but processes more data when it does.  
  
For highly active tables, reduce the autovacuum\_vacuum\_scale\_factor (e.g., 0.05 to 0.1) to ensure autovacuum runs more often and prevents bloat.  
  
For read-heavy tables, you may want to focus more on autoanalyze settings rather than autovacuum since frequent updates and deletes are not expected.  
  
For a large table with 10 million rows, you may set:  
  
ALTER TABLE emp2024  
SET (  
autovacuum\_vacuum\_threshold = 200000, --> 2% of table rows  
autovacuum\_vacuum\_scale\_factor = 0.2, --> Trigger vacuum when 20% of rows are dead  
autovacuum\_analyze\_threshold = 50000, --> 0.5% of table rows  
autovacuum\_analyze\_scale\_factor = 0.05 --> Trigger analyze when 5% of rows change  
);  
  
In this case, autovacuum\_vacuum\_threshold would be based on 10 million rows and 200,000 dead tuples (10,000 \* 0.2), while autovacuum\_analyze\_threshold would be 50000 changes.