

Data Science - Data Prep with SQL - Quick Reference

DATASET PROFILING

Volume	SELECT COUNT (*) FROM t;
Velocity	SELECT t.date1, COUNT(*) FROM t GROUP by t.date1 ORDER BY t.date1 desc;
Attribute Selection	SELECT attr1, attr2, attr3, attr4 FROM t;
Incomplete Records	SELECT * FROM t WHERE t.attr1 IS NULL AND t.attr2 IS NULL;

VALIDATE ATTRIBUTES

Domain	SELECT DISTINCT (attr1) FROM t;
Missing Values	SELECT * FROM t WHERE t.attr1 IS NULL ;
Range	SELECT MIN (attr1), MAX (attr1), AVG (attr1) FROM t;
Data Type	SELECT * FROM information_schema.columns WHERE table_name = 't';
Outliers (95% confidence)	WITH dev_cte AS (SELECT STDDEV (attr1) sdev FROM t) SELECT attr1, attr2 FROM t CROSS JOIN dev_cte c WHERE t.attr1 > c.sdev * 2;
Distribution	SELECT attr1, WIDTH_BUCKET (attr1,100,500,5) FROM t;

STANDARDIZE ATTRIBUTES

Data Types	SELECT CAST (attr1 AS DATE), CAST (attr2 AS INT) FROM t;
Patterns	SELECT CASE WHEN attr1 = ..., REPLACE (attr2,'Street','St') FROM t;
Formatting	SELECT UPPER (attr1), REPLACE (attr2,'-' ,',') FROM t;
Scaling	SELECT attr1, attr2/(MAX (attr2) OVER (PARTITION BY attr1)) FROM t;

CREATE INTERFACE

Create view	CREATE VIEW AS SELECT...
-------------	---------------------------------

CLEAN ATTRIBUTES

Outliers (Quantitative)	SELECT CASE WHEN attr1 < 0 THEN 0 WHEN attr1 > 1000 THEN 1000 ELSE attr1 END as attr1 FROM t;
Missing Values (At Random)	SELECT COALESCE (attr1,AVG(attr1) OVER ()), COALESCE (attr1,'Unknown') FROM t;
Missing Values (Not at Random)	SELECT COALESCE (attr1,0) FROM t;
Incorrect Values	SELECT REPLACE (attr1,'bad','good') FROM t;

DERIVE ATTRIBUTES

Buckets\Binning	SELECT attr1, CASE WHEN attr1 <= 50 THEN 'bin1' WHEN attr1 > 50 THEN 'bin2' ELSE 'bin3' END as attr1_bin FROM t;
Date Parts	SELECT DAYOFMONTH (date1), MONTHOFYEAER (date1) FROM t;
Date Difference	SELECT DATEDIFF (date1,date2) FROM t;
Last Period	SELECT DATEADD (year,-1,date1) FROM t;
Dummy Encoding (One Hot)	SELECT attr1, CASE WHEN attr1 = 'Male' THEN 1 ELSE 0 as male_gender FROM t;

COMBINE DATASETS

Join Horizontally (Full Match)	SELECT t1.attr1, t2.attr2 FROM t1 INNER JOIN t2 ON t1.ID = t2.ID;
Join Horizontally (Optional Match)	SELECT t1.attr1, t2.attr2 FROM t1 LEFT JOIN t2 ON t1.ID = t2.ID;
Union Vertically (Deduplicate)	SELECT attr1, attr2 FROM t1 UNION SELECT attr1, attr2 FROM t2
Union Vertically (No Deduplicate)	SELECT attr1, attr2 FROM t1 UNION ALL SELECT attr1, attr2 FROM t2

SPLIT DATASETS

Simple Filter	SELECT attr1, attr2 FROM t WHERE attr1 IS NOT NULL;
Filter Based on Aggregation	SELECT attr1, SUM(attr2) FROM t GROUP BY attr1 HAVING SUM(attr2) > 10;
Sampling (Random)	SELECT attr1, ROW_NUMBER () OVER (ORDER BY RANDOM()) as random FROM t;
Sampling (Non-Random)	SELECT attr1, NTILE (4) OVER (ORDER BY date()) as quartile FROM t;