-- Fee impact analysis: Correlation between fees and activity metrics

WITH fee\_activity\_correlation AS (

-- Get fee data

SELECT

DATE\_TRUNC('month', block\_timestamp) AS month,

AVG(tx\_fee) AS avg\_fee\_eth,

COUNT(\*) AS total\_transactions,

COUNT(DISTINCT from\_address) AS unique\_users

FROM ETHEREUM.CORE.FACT\_TRANSACTIONS

WHERE block\_timestamp >= '2023-01-01'

AND block\_timestamp < '2025-09-01'

AND tx\_fee IS NOT NULL

AND tx\_fee > 0

AND tx\_fee < 10

GROUP BY 1

),

eth\_prices AS (

SELECT

DATE\_TRUNC('month', hour) AS month,

AVG(price) AS avg\_eth\_price\_usd

FROM ETHEREUM.PRICE.EZ\_PRICES\_HOURLY

WHERE symbol = 'ETH'

AND hour >= '2023-01-01'

AND hour < '2025-09-01'

AND price BETWEEN 100 AND 10000

GROUP BY 1

),

dex\_activity AS (

SELECT

DATE\_TRUNC('month', block\_timestamp) AS month,

COUNT(\*) AS dex\_transactions,

COUNT(DISTINCT origin\_from\_address) AS dex\_users,

SUM(COALESCE(amount\_in\_usd, 0)) AS dex\_volume\_usd

FROM ETHEREUM.DEFI.EZ\_DEX\_SWAPS

WHERE block\_timestamp >= '2023-01-01'

AND block\_timestamp < '2025-09-01'

AND amount\_in\_usd BETWEEN 0.1 AND 1e9

AND amount\_in\_usd IS NOT NULL

GROUP BY 1

),

combined\_metrics AS (

SELECT

f.month,

f.avg\_fee\_eth,

f.avg\_fee\_eth \* p.avg\_eth\_price\_usd AS avg\_fee\_usd,

f.total\_transactions,

f.unique\_users,

d.dex\_transactions,

d.dex\_users,

d.dex\_volume\_usd / 1e9 AS dex\_volume\_billions,

-- Calculate fee categories

CASE

WHEN f.avg\_fee\_eth \* p.avg\_eth\_price\_usd < 1 THEN 'Ultra Low (<$1)'

WHEN f.avg\_fee\_eth \* p.avg\_eth\_price\_usd < 3 THEN 'Low ($1-$3)'

WHEN f.avg\_fee\_eth \* p.avg\_eth\_price\_usd < 7 THEN 'Medium ($3-$7)'

WHEN f.avg\_fee\_eth \* p.avg\_eth\_price\_usd < 15 THEN 'High ($7-$15)'

ELSE 'Very High (>$15)'

END AS fee\_category,

-- Activity intensity metrics

f.total\_transactions / 1000000.0 AS transactions\_millions,

f.unique\_users / 1000000.0 AS users\_millions

FROM fee\_activity\_correlation f

JOIN eth\_prices p ON f.month = p.month

LEFT JOIN dex\_activity d ON f.month = d.month

)

SELECT

month,

avg\_fee\_usd,

fee\_category,

total\_transactions,

unique\_users,

transactions\_millions,

users\_millions,

dex\_transactions,

dex\_users,

dex\_volume\_billions,

-- Month-over-month changes

LAG(avg\_fee\_usd, 1) OVER (ORDER BY month) AS prev\_month\_fee,

LAG(transactions\_millions, 1) OVER (ORDER BY month) AS prev\_month\_tx\_millions,

-- Calculate correlations (simplified)

(avg\_fee\_usd - LAG(avg\_fee\_usd, 1) OVER (ORDER BY month)) AS fee\_change,

(transactions\_millions - LAG(transactions\_millions, 1) OVER (ORDER BY month)) AS tx\_change\_millions

FROM combined\_metrics

ORDER BY month