PART 1

Code

WITH cv\_by\_sensor AS (

-- Get CV counts near each sensor matching route\_id

SELECT

l.device\_id,

l.route\_id,

HOUR(cv.capturedtimestamp) as hour\_of\_day,

COUNT(DISTINCT cv.journeyid) as cv\_count -- Count unique journeys

FROM raghu\_wavetronix\_locations l

JOIN raghu\_cv\_data cv

ON cv.route\_id = l.route\_id -- Match routes

AND (

-- Calculate approx distance using latitude/longitude

POW(69.1 \* (l.Latitude - cv.latitude), 2) +

POW(69.1 \* COS(RADIANS(l.Latitude)) \* (l.Longitude - cv.longitude), 2)

) <= 4 -- Square of 2 mile radius

GROUP BY

l.device\_id,

l.route\_id,

HOUR(cv.capturedtimestamp)

),

hourly\_totals AS (

-- Get total CV count for each hour across all sensors

SELECT

hour\_of\_day,

SUM(cv\_count) as total\_cv\_count

FROM cv\_by\_sensor

GROUP BY hour\_of\_day

)

-- Combine individual sensor counts with hourly totals

SELECT

s.device\_id,

s.route\_id,

s.hour\_of\_day,

s.cv\_count as sensor\_cv\_count,

t.total\_cv\_count

FROM cv\_by\_sensor s

JOIN hourly\_totals t ON s.hour\_of\_day = t.hour\_of\_day

ORDER BY

s.hour\_of\_day,

s.device\_id;

Result

A screenshot of a computer

Description automatically generated

PART2

Code

WITH hourly\_wavetronix\_counts AS (

SELECT

REGEXP\_REPLACE(device\_id, '\_[0-9]+$', '') as device\_id, -- Remove lane number from device\_id

HOUR(cst\_time) as hour\_of\_day,

SUM(lane\_count) as total\_count -- Sum counts across all lanes

FROM raghu\_wavetronix\_data

GROUP BY

REGEXP\_REPLACE(device\_id, '\_[0-9]+$', ''),

HOUR(cst\_time)

)

SELECT

device\_id,

hour\_of\_day,

total\_count as wavetronix\_count,

SUM(total\_count) OVER (PARTITION BY hour\_of\_day) as total\_hourly\_vehicles

FROM hourly\_wavetronix\_counts

ORDER BY

hour\_of\_day,

device\_id;

Result

A screenshot of a computer

Description automatically generated

PART3

Code

-- First get CV counts per sensor per hour

WITH cv\_counts AS (

SELECT

l.device\_id,

l.route\_id,

HOUR(cv.capturedtimestamp) as hour\_of\_day,

COUNT(DISTINCT cv.journeyid) as cv\_count

FROM raghu\_wavetronix\_locations l

JOIN raghu\_cv\_data cv

ON cv.route\_id = l.route\_id

AND (

POW(69.1 \* (l.Latitude - cv.latitude), 2) +

POW(69.1 \* COS(RADIANS(l.Latitude)) \* (l.Longitude - cv.longitude), 2)

) <= 4

GROUP BY

l.device\_id,

l.route\_id,

HOUR(cv.capturedtimestamp)

),

-- Get Wavetronix counts per sensor per hour

wavetronix\_counts AS (

SELECT

REGEXP\_REPLACE(device\_id, '\_[0-9]+$', '') as device\_id,

HOUR(cst\_time) as hour\_of\_day,

SUM(lane\_count) as total\_count

FROM raghu\_wavetronix\_data

GROUP BY

REGEXP\_REPLACE(device\_id, '\_[0-9]+$', ''),

HOUR(cst\_time)

)

-- Calculate penetration rate

SELECT

c.device\_id,

c.hour\_of\_day,

c.cv\_count,

w.total\_count as wavetronix\_count,

CAST(c.cv\_count AS DOUBLE) / NULLIF(w.total\_count, 0) \* 100 as penetration\_rate

FROM cv\_counts c

JOIN wavetronix\_counts w

ON c.device\_id = w.device\_id

AND c.hour\_of\_day = w.hour\_of\_day

ORDER BY

c.hour\_of\_day,

c.device\_id;

Result

A screenshot of a computer

Description automatically generated

PART4

WITH cv\_counts AS (

SELECT

l.device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(cv.capturedtimestamp) as hour\_of\_day,

COUNT(DISTINCT cv.journeyid) as cv\_count,

COUNT(\*) as total\_trajectory\_points -- Total CV trajectory points

FROM raghu\_wavetronix\_locations l

JOIN raghu\_cv\_data cv

ON cv.route\_id = l.route\_id

AND (

POW(69.1 \* (l.Latitude - cv.latitude), 2) +

POW(69.1 \* COS(RADIANS(l.Latitude)) \* (l.Longitude - cv.longitude), 2)

) <= 4 -- Square of 2 mile radius

GROUP BY

l.device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(cv.capturedtimestamp)

)

SELECT

device\_id,

route\_id,

Latitude,

Longitude,

hour\_of\_day,

cv\_count as unique\_vehicles,

total\_trajectory\_points as trajectory\_count,

SUM(cv\_count) OVER (PARTITION BY hour\_of\_day) as total\_hourly\_vehicles,

SUM(total\_trajectory\_points) OVER (PARTITION BY hour\_of\_day) as total\_hourly\_trajectories

FROM cv\_counts

ORDER BY

hour\_of\_day,

device\_id;

Result

A screenshot of a computer

Description automatically generated

PART5

Code

WITH hourly\_wavetronix\_counts AS (

SELECT

REGEXP\_REPLACE(w.device\_id, '\_[0-9]+$', '') as device\_id, -- Remove lane number

HOUR(w.cst\_time) as hour\_of\_day,

SUM(w.lane\_count) as total\_count -- Sum all lane counts

FROM raghu\_wavetronix\_data w

GROUP BY

REGEXP\_REPLACE(w.device\_id, '\_[0-9]+$', ''),

HOUR(w.cst\_time)

)

SELECT

w.device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

w.hour\_of\_day,

w.total\_count as wavetronix\_count,

SUM(w.total\_count) OVER (PARTITION BY w.hour\_of\_day) as total\_hourly\_vehicles

FROM hourly\_wavetronix\_counts w

JOIN raghu\_wavetronix\_locations l ON w.device\_id = l.device\_id

ORDER BY

w.hour\_of\_day,

w.device\_id;

Result

A screenshot of a computer

Description automatically generated

PART6

Code

WITH cv\_counts AS (

SELECT

l.device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(cv.capturedtimestamp) as hour\_of\_day,

COUNT(DISTINCT cv.journeyid) as cv\_count,

COUNT(\*) as trajectory\_count

FROM raghu\_wavetronix\_locations l

JOIN raghu\_cv\_data cv

ON cv.route\_id = l.route\_id

AND (

POW(69.1 \* (l.Latitude - cv.latitude), 2) +

POW(69.1 \* COS(RADIANS(l.Latitude)) \* (l.Longitude - cv.longitude), 2)

) <= 4

GROUP BY

l.device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(cv.capturedtimestamp)

),

wavetronix\_counts AS (

SELECT

REGEXP\_REPLACE(w.device\_id, '\_[0-9]+$', '') as device\_id,

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(w.cst\_time) as hour\_of\_day,

SUM(w.lane\_count) as total\_count

FROM raghu\_wavetronix\_data w

JOIN raghu\_wavetronix\_locations l

ON REGEXP\_REPLACE(w.device\_id, '\_[0-9]+$', '') = l.device\_id

GROUP BY

REGEXP\_REPLACE(w.device\_id, '\_[0-9]+$', ''),

l.route\_id,

l.Latitude,

l.Longitude,

HOUR(w.cst\_time)

)

SELECT

w.device\_id,

w.route\_id,

w.Latitude,

w.Longitude,

w.hour\_of\_day,

COALESCE(c.cv\_count, 0) as cv\_count,

COALESCE(c.trajectory\_count, 0) as trajectory\_count,

w.total\_count as wavetronix\_count,

CASE

WHEN w.total\_count > 0 THEN

ROUND(CAST(COALESCE(c.cv\_count, 0) AS DOUBLE) / w.total\_count \* 100, 2)

ELSE 0

END as penetration\_rate,

ROUND(AVG(CASE

WHEN w.total\_count > 0 THEN

CAST(COALESCE(c.cv\_count, 0) AS DOUBLE) / w.total\_count \* 100

ELSE 0

END) OVER (PARTITION BY w.hour\_of\_day), 2) as avg\_hourly\_penetration

FROM wavetronix\_counts w

LEFT JOIN cv\_counts c

ON w.device\_id = c.device\_id

AND w.hour\_of\_day = c.hour\_of\_day

ORDER BY

w.hour\_of\_day,

penetration\_rate DESC;

Result

A screenshot of a computer

Description automatically generated