-- JOIN customers and transaction with customer\_id

-- JOIN transaction and products with product\_id

SELECT

DISTINCT ON

(customers.state) state,

products.product\_name product\_name,

COUNT(products.product\_name) product\_count

FROM

customers

JOIN

transactions

ON

transactions.customer\_id = customers.customer\_id

JOIN

products

ON

transactions.product\_id = products.product\_id

GROUP BY 1,2

ORDER BY

customers.state, 3 DESC;

WITH

customerrank

AS (

SELECT DISTINCT

customers.name,

customers.state,

SUM(transactions.transact\_amt) transact\_amt,

rank() OVER (PARTITION BY state ORDER BY SUM(transactions.transact\_amt) DESC )

FROM

customers

JOIN

transactions

ON

transactions.customer\_id = customers.customer\_id

GROUP BY 1,2

ORDER BY

customers.state

)

SELECT

name,

state,

rank

FROM

customerrank

GROUP BY 1,2,3

HAVING rank <= 5

ORDER BY state, rank;

-- No product shows up, cuz no product is there

-- within 30 days of the latest date in the table

SELECT

products.product\_name,

COUNT(products.product\_name) AS product\_count

FROM

customers

JOIN

transactions

ON

transactions.customer\_id = customers.customer\_id

JOIN

products

ON

transactions.product\_id = products.product\_id

WHERE customers.email LIKE '%@gmail%'

-- count ones successful sold

AND transactions.payment\_success = 1

-- 30 days from the latest date in the table

AND transactions.transact\_at >= TIMESTAMP '2017-04-13 21:02:17' - INTERVAL '30 days'

GROUP BY products.product\_name, customers.email

ORDER BY product\_count DESC

LIMIT 5;

Assuming the two groups are similar in every aspect because randomly picked. The key metric would be the conversion rate (e.g. product purchase, sign up etc.). I would also collect, duration of the visit, number of clicks, age, geolocation, date, time of the day, ask for their experience and any metric that might help. The key metric will tell us version success. Other metrics could tell us about their experience, for example multiple click and long duration might indicate confusion and result in negative experience.

Mean and standard deviation would be calculated for the versions. The difference in mean determined which version was successful but the difference must be statistically significant. To determine that a t-test is conducted, giving as a t-value. Larger the t-value, the larger the difference between the means. How large is large enough? to determine that we calculate the probability of getting that t-value, known as p-value, for which the threshold is preselected (usually 95% interval, higher for less error).

This would be a Canary deployment, similar to Blue/Green deployment, in which both versions are kept but in Blue/Green deployment when the new version is ready some of the old users and all the new users are directed to the new version. In the Canary deployment, new version users are picked based on experiments and gradually expanded. New users are directed toward new or old version based on the experiments, therefore, offering the new version to a subset of the customer in perpetuity.

This is not a rolling deployment, in which the old version is gradually decommissioned therefore everyone gets the new version in "perpetuity" eventually.