The final report

Question_1 : No, we don't have enough information to calculate metrics like 30-day view-binary, we still need the test_start_date data for each item.

2. Reformat the Data

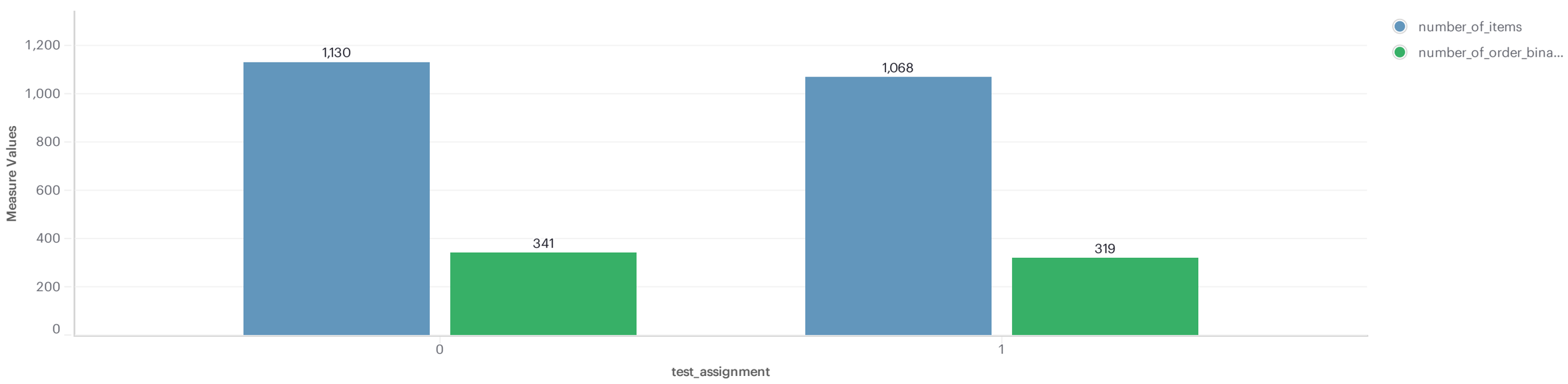
	item_id	test_start_date	test_number	test_assignment
1	3748	2020-01-01	test_a	0
2	316	2020-01-01	test_a	0
3	104	2020-01-01	test_a	1
4	2871	2020-01-01	test_a	0
5	2812	2020-01-01	test_a	1

This table only shows the first 1,000 rows.

3. Compute Order Binary

	test_assignment	number_of_items	number_of_order_binary_30d
1	0	1130	341
2	1	1068	319

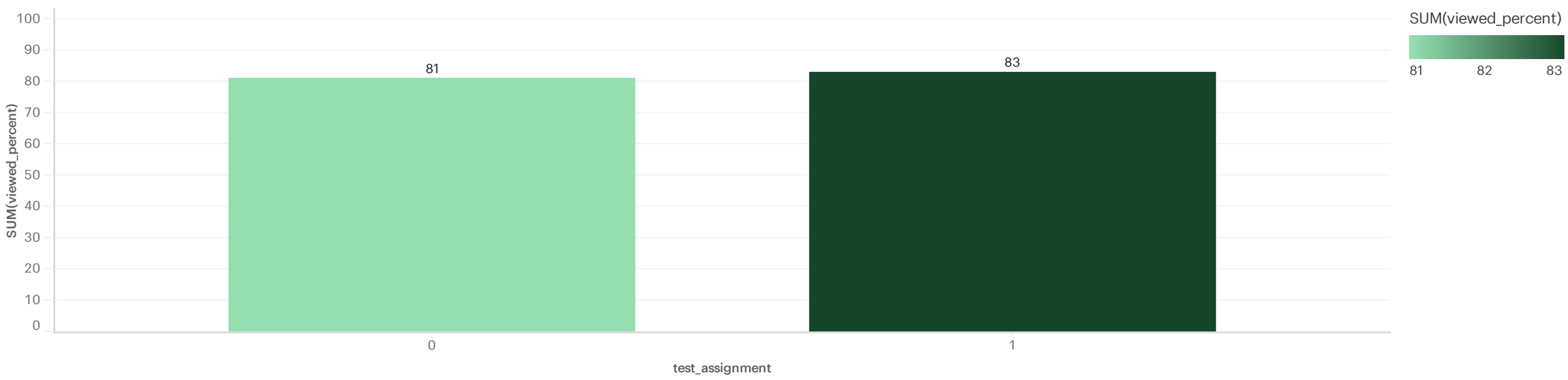
Number of Items Ordered by Test Group



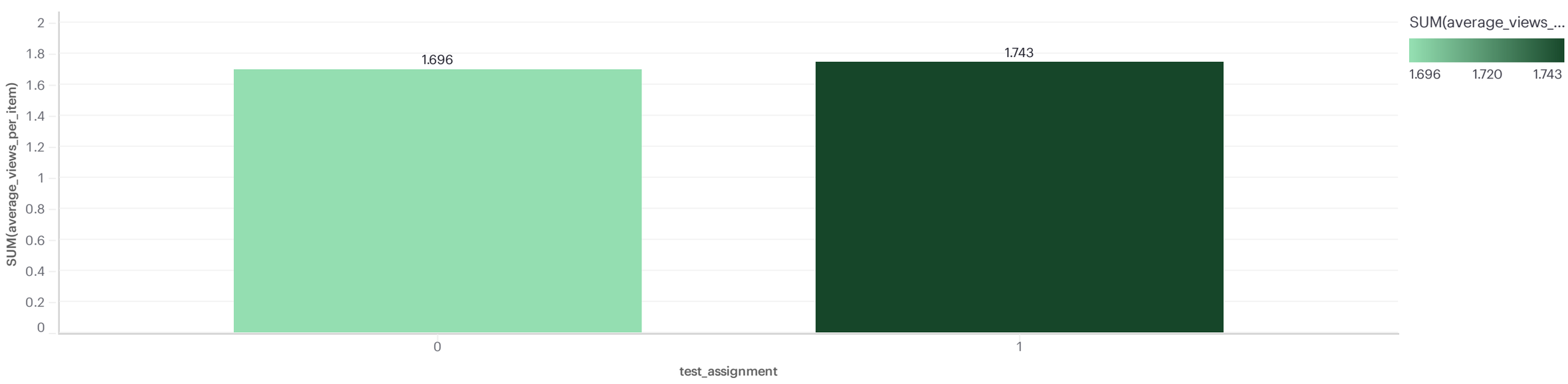
4. Compute View Item Metrics

	test_assignment	items	viewed_items	viewed_percent	views	average_views_per_item
1	0	1130	918	81	1916	1.69557522124
2	1	1068	890	83	1862	1.74344569288

Test group and baseline group: percent of items have been viewed



Test group and baseline group: average views per item



5. Compute the lift and p-value
    1. Ordered items metric: The group with test_assignment 1 had an improvement of -1%, the p-value is 0.88, which is not small enough compares to the threshold of significance. So there is no statistically significant improvement to the number of ordered items as a result of the test treatment.
    2. Viewed items metric: The group with test_assignment 1 had an improvement of 2.6%, the p-value is 0.2, which is not small enough compares to the threshold of significance. So there is no statistically significant improvement to the number of viewed items as a result of the test treatment.

```sql
--Reformat the final_assignments_qa to look like the final_assignments table,
select *
from(
SELECT
  item_id,'2020-01-01'  as test_start_date,
  case
    when test_a=0 or test_a =1 then 'test_a'
    END AS test_number,
  case
    when test_a=0 or test_a =1 then test_a
    END AS test_assignment

FROM
  dsv1069.final_assignments_qa
union
SELECT
  item_id,'2020-01-01' as test_start_date,
  case
    when test_b=0 or test_b =1 then 'test_b'
    END AS test_number,
  case
    when test_b=0 or test_b =1 then test_b


    END AS test_assignment

FROM
  dsv1069.final_assignments_qa
union
SELECT
  item_id,'2020-01-01' as test_start_date,
  case
    when test_c=0 or test_c =1 then 'test_c'
    END AS test_number,
  case
    when test_c=0 or test_c =1 then test_c


    END AS test_assignment

FROM
  dsv1069.final_assignments_qa
union
SELECT
  item_id,'2020-01-01' as test_start_date,
  case
    when test_d=0 or test_d =1 then 'test_d'
    END AS test_number,
  case
    when test_d=0 or test_d =1 then test_d


    END AS test_assignment

FROM
  dsv1069.final_assignments_qa
```

```
-- Returns first 100 rows from dsv1069.orders
select test_assignment,count(*) as number_of_items,sum(case when order_binary_30d


from
(select ass.item_id,ass.test_assignment,
max(case
when date_part('day',created_at-test_start_date)<=30 and ord.created_at > ass.test
else 0
end) as order_binary_30d
from dsv1069.final_assignments as ass left join dsv1069.orders as ord on ass.item_
--and ass.item_id = 3313
where ass.test_number='item_test_2'
GROUP BY
  ass.test_assignment,
  ass.item_id
) as foo
group by test_assignment

-- Use this table to
-- compute order_binary for the 30 day window after the test_start_date
-- for the test named item_test_2
```

```
-- Use this table to
-- compute view_binary for the 30 day window after the test_start_date
-- for the test named item_test_2
SELECT
test_assignment,
COUNT(item_id) AS items,
SUM(view_binary_30d) AS viewed_items,
CAST(100*SUM(view_binary_30d)/COUNT(item_id) AS FLOAT) AS viewed_percent,
SUM(views) AS views,
SUM(views)/COUNT(item_id) AS average_views_per_item
FROM
(
 SELECT
   ass.test_assignment,
   ass.item_id,
   MAX(CASE WHEN events.event_time > ass.test_start_date and DATE_PART('day', ever
   sum(CASE WHEN events.event_time > ass.test_start_date and DATE_PART('day', ever
   FROM
    dsv1069.final_assignments ass

  LEFT JOIN
    (SELECT       event_time,
      event_id,    cast(parameter_value as int) as item_id
    from   dsv1069.events
        WHERE
      event_name = 'view_item'
    AND
      parameter_name = 'item_id'
) events

  ON
    ass.item_id = events.item_id

  WHERE
    ass.test_number= 'item_test_2'
    --and DATE_PART('day', events.event_time - ass.test_start_date ) <= 30
    --and events.event_time >= ass.test_start_date
  GROUP BY
    ass.test_assignment,
    ass.item_id
) item_level
GROUP BY
 test_assignment
```

```
--Use the https://thumbtack.github.io/abba/demo/abba.html to compute the lifts in
select count(*),test_assignment,order_binary_30d
from
(select ass.item_id,ass.test_assignment,
max(case
when date_part('day',created_at-test_start_date)<=30 and ord.created_at > ass.test
else 0
end) as order_binary_30d
from dsv1069.final_assignments as ass left join dsv1069.orders as ord on ass.item_
--and ass.item_id = 3313
where ass.test_number='item_test_2'
GROUP BY
  ass.test_assignment,
  ass.item_id
) as order_binary_table
group by test_assignment,order_binary_30d
-- p_value: 0.63
--lift:-2.8% - 4.6%(0.89%)

/*
select count(*),test_assignment,view_binary
from
(select ass.item_id,ass.test_assignment,ass.test_number,
case
when date_part('day',event_time-test_start_date)<30 then 1
else 0
end as view_binary
from dsv1069.final_assignments as ass left join dsv1069.view_item_events as view o
where test_number= 'item_test_2') as view_binary_table
group by test_assignment,view_binary
*/
-- p_value: < 0.0001

--lift:3.3% - 6.4% (4.8%)
```

```
select count(*),test_assignment,view_binary
from
(select ass.item_id,ass.test_assignment,ass.test_number,
case
when date_part('day',event_time-test_start_date)<30 then 1
else 0
end as view_binary
from dsv1069.final_assignments as ass left join dsv1069.view_item_events as view c
where test_number= 'item_test_2') as view_binary_table
group by test_assignment,view_binary

-- p_value: < 0.0001

--lift:3.3% - 6.4% (4.8%)
```