https://tech.instacart.com/3-million-instacart-orders-open-sourced-d40d29ead6f2
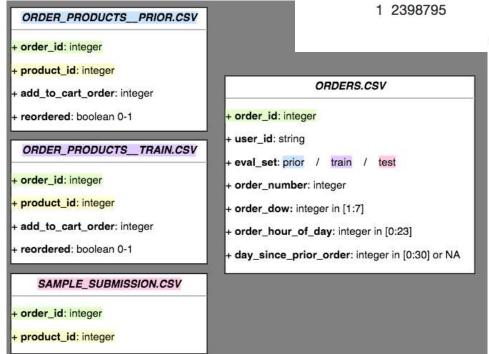
- 3,4 Mio Bestellungen
- 32 Mio Bestellpositionen

user id	order id	order number	order dow	order hour of day	add to cart order	product id	product name
1	2539329	1	2	8	1	196	Soda
1	2539329	1	2	8	2	14084	Organic Unsweetened Vanilla Almond Milk
1	2539329	1	2	8	3	12427	Original Beef Jerky
1	2539329	1	2	8	4	26088	Aged White Cheddar Popcorn
1	2539329	1	2	8	5	26405	XL Pick-A-Size Paper Towel Rolls
1	2398795	2	3	7	1	196	Soda
1	2398795	2	3	7	2	10258	Pistachios
1	2398795	2	3	7	3	12427	Original Beef Jerky
1	2398795	2	3	7	4	13176	Bag of Organic Bananas
1	2398795	2	3	7	5	26088	Aged White Cheddar Popcorn
1	2398795	2	3	7	6	13032	Cinnamon Toast Crunch

Note that this dataset includes orders from many different retailers and is a

ORDER_PRODUCTS__PRIOR.CSV

+ order_id: integer
+ product_id: integer
+ add_to_cart_order: integer
+ reordered: boolean 0-1

ORDER_PRODUCTS__TRAIN.CSV

+ order_id: integer
+ product_id: integer
+ add_to_cart_order: integer
+ reordered: boolean 0-1

SAMPLE_SUBMISSION.CSV

+ order_id: integer
+ product_id: integer

ORDERS.CSV

+ order_id: integer
+ user_id: string
+ eval_set: prior / train / test
+ order_number: integer
+ order_dow: integer in [1:7]
+ order_hour_of_day: integer in [0:23]
+ day_since_prior_order: integer in [0:30] or NA

```
create view prior_orders_of_test_users as
select prior_orders.user_id, order_number, order_id
from
    (select user_id, order_number, order_id
    from adbkt.orders
    where eval_set = 'prior') prior_orders
 inner join
    (select distinct user_id from adbkt.orders where eval_set = 'test') test_user_ids
 on prior_orders.user_id = test_user_ids.user_id;


create view prior_orders_3_of_test_users as
select order_id, user_id, order_number
from
   (select order_id, user_id, order_number,
     rank() over (partition by user_id order by order_number desc) as rank
   from prior_orders_of_test_users) x
where rank <= 3;


create view user_products as
select user_id, product_id
from prior_orders_3_of_test_users o3
       inner join adbkt.order_products__prior opp on o3.order_id =
opp.order_id
group by user_id, product_id
having avg(reordered) >= 0.5;


create view testorders_products as
select order_id, product_id
from adbkt.orders o inner join user_products up on o.user_id =
up.user_id
where eval_set='test' ;
```

```
create view  vsubmission as                                    select *
select s.order_id, string_agg(product_id::varchar(10), ' ')    from vsubmission
from adbkt.submission s                                        order by order_id;
    left join testorders_products tp on s.order_id=tp.order_id
group by s.order_id;
```

*	ORDER_ID	STRING_AGG(PRODUCT_ID,' ')
1	17	13107 21709 38777 21463 47766
2	34	13176 44663 39475 43504 21137 47766 47792 2596 16083
3	137	38689 28934 29594 44422 23794 2326 25890 5134 41787 24852
4	182	29079 16973 28800 5479 8021 47672 13629 33000 41149 32109 11520 39275 9337 15252
5	257	37646 25659 29837 24838 4605 35734 1025 45013 24852 49235 13870 38693 27966 28476 12456 39475 .
6	313	30391 12779 21903 46906 45007 28535 47626 13198 34969
7	353	35561 40688
8	386	36695 2326 6046 1158 24852 8174 42265 26497 29987 28985 27156 47766 17652 45066 35221 30450 40
9	414	19006 33320 31040 7503 44292 20119 27845 3599 20564 31215 7539 21376 14947 46616 21709 4472 35.
10	418	32360 13702 47766 41950 45364 5262 40268 12036 38694 1503 27086 30489
11	437	45106 27966 42265 47144 22825 4589 10132 47209 5769 1463 30233 26790 26209 365 13176
12	452	17948 30388 43504 36313 28666 47209 24964 27966 10036 27553 23644 12020 48395 44632 5262 6046 .
13	474	47966
14	492	24933 21137 46667 13702 45603 21903 21174 8424 40396 24499 20119 33232 43961 20542 14833 27323
15	497	1821 36216 30047 27275 31064 21501