Applying Analytical Patterns

Day 2 Lab
Data Engineering Design Patterns at Meta - Funnel Accounting

Transcript:

for the day two lab so we're going to be
44:40
almost exclusively working with this events table again um this is the table we're going to be working with and uh
44:47
initially what we want to do is I want to figure
44:52
out for every person who goes to my sign up page how many of them
45:01
actually sign up so one of the ways that I can show you uh so there's essentially
45:07
two uh URLs here that will determine this so if I say we URL in and then I
45:17
say sign up and I say comma um AP back SL API V1
45:25
users okay so you see here are okay and you see how there's some some of these are uh API V1 users
45:32
and some of them are signup so essentially what these are are doing here what we're going to be doing here
45:38
is we're going to create we're going to create a funnel say I go to Zack wilson. Tech great and then I go to sign up all
45:45
right so here's the sign up page and if I actually sign up what will happen when I hit this submit button is behind the
45:52
scenes you'll see in my uh in my code here there is this uh where is it um API
45:59
B1 user yeah so you see right here this create user route it used to be called
46:05
users in the old data that I made a change and like that's why where it's users because like this is data from six
46:10
months ago and then I changed it to user because user is actually correct when you're designing endpoints uh like all
46:18

your entities should be based on uh like the actual user um object or it should be singular objects right so anyways the
46:27
idea here is you'll you'll see some of these are actually going to be like kind of connected where people are going to
46:32
go to that page so you see here this user actually did it right this person was they they visited the signup page
46:39
and then they created an account that's literally what what happened for this user so what we're going to do is we're
46:48
trying to figure out what percent of users who got to the signup
46:54
page actually signed up that is our plan here today and uh this
47:01
is going to be um more tricky than you would think and we're going to do it with no window functions we're going to
47:08
do this with a self join so if you guys want to follow along the queries are already in uh pie charm in this applying
47:15
analytical patterns uh you'll see we'll have this um funnel analysis query that
47:21
we're going to do so one of the things I know about this is like my logger and uh this data set is actually a little bit
47:27
buggy in terms of uh like duplicates so what we want to do first is we want to
47:32
get rid of those duplicates so it's say uh duped events as and then in this case I'm
47:40
going to just take this and pop them in here then remove this and then we also
47:45
have null user ID so we're going to say where user ID um um is not null and then what we want is we got
47:52
user ID we want a URL and then there's event time and then we can also get date of event
48:00
time and this is as event date uh I'll explain why we need this in a second
48:05
because we essentially want it to be like we we care about the time stamp but we also care about the date so then the
48:12

last thing we want to do here is I want to say Group by right Group by user ID URL event time and date event time so
48:19
this is going to give us our first set of data right so we can say like select star from deduped events so let's just
48:26
look at this real quick oh I forgot to buy so if we do this bada bing bada boom
48:34
we have all of our data it's now duped because of that group bu great so now
48:40
what we want is we can also filter because we only care about two events in this case we care about sign up and uh
48:48
like visiting the sign up page and actually signing up so we're going to say URL in in this case we're going to
48:54
say back SL signup and back slash um API V1
49:01
users so those are our two events that we care about and we can filter everything else out for now so that like
49:07
it also will make the query a lot faster because it kind of shrinks the data set down so now what we want to do is we
49:15
want to say okay did this user ever make uh a sign
49:22
who visited a sign up page did they ever sign up after they visited sign up page
49:27
because we don't want it to be before because technically you could like log out like you could sign up and then log out and then go visit the sign up page
49:34
again just like what I did I did that literally in this lab today and we don't want and just because I create I I did
49:40
sign up earlier we don't want that new sign up to like have that attribution so
49:45
in this case what we want to do is we essentially want to join the table on
49:51
itself so that we can have the sign up event and the uh and the the visiting
49:57
the signup page and the actual signup event happen on the same row on the same
50:03
record right so in this case we're going to say uh we're going to say dded events D1 and then we're going to say join

50:10
duped events D2 and then in this case uh we have on all right and then in this
50:15
case we're going to say uh obviously the first one is D1 user ID equals D2 user ID that's like the obvious one and we
50:23
also want to put in and D1 do event dat equals D2 do event date that's going to
50:29
be uh probably another way that you would want to look at this uh this is an interesting one where it's like okay do
50:34
we measure conversion over uh a specific time of like we only count the visit if
50:41
it happens within the same day uh I think that makes this an this analytics a little bit easier because otherwise
50:47
like if you just join on user ID and then you have this last condition right there's one more condition here which is
50:53
and D2 do event time is greater than D1 event time so what this will do like
51:01
let's not do select star but let's do um we'll say D1 user ID and we'll say uh D1
51:08
URL D1 or D2 URL D1 event time D2 event
51:14
time so let's let's run this query this query is going to be a little bit slower
51:19
but oh not not too bad so um oh interesting uh okay so we have one
51:27
more here though right we need we need the join to also include uh like see the
51:33
the URLs are matching here and we don't want them to match like and uh so that
51:39
means this this person signed up like twice or something they must have like double clicked the button or they
51:44
updated their account or something like that so what we want to do is we want to add one more condition in here we're
51:50
going to say D1 URL does not equal D2 URL because that's um that's a mess and
51:58
uh so now this will give us probably better okay but the problem here is oh
52:04

there's actually one more here because we have the uh we we only want the D1 we want this to D1 we only want to care as
52:11
the sign up page visits and you see like we have both now so uh I think that would be in the wear Cloud though so we
52:17
can say where D1 URL equals back signup and D2L equals API V1 users and we can
52:26
actually get rid of this does not equal condition because like this essentially will make it so like the wear CLA is
52:31
going to do that in Big Data we might want to keep that because it will filter out more of the data but um in this case
52:38
uh we probably don't need it okay there we go so now we have
52:45
a we have a visit that turns into a sign up that is essentially what's happening
52:51
here and you'll see some of these events are pretty close right this visit like these are not that close but they're
52:58
like you know like this is what 16 minutes apart like so obviously there's
53:03
probably uh um like you would want to have like an attribution window of that
53:09
sign up visit like because you could like imagine you could visit the signup page leave and then come back later like
53:16
uh um that you come back at like a um like a day later and we would still get
53:23
the attribution here right and but that's fine so we have all of these sign up events that are are here but you'll
53:30
know that like some of these events actually do not have like they're actually isn't uh
53:38
um uh like a created event because they can visit the sign up page and abandon
53:43
so that's where this one this this wear Clause is actually buggy right because this right now is filtering down to only
53:50
the people who have converted so we don't want just the people who converted but like and that was the whole point
53:56

here was to show you okay these are the people who converted and this is what's going on so what we want to do is we
54:03
essentially want to create another uh we're going to call this like self joined as and then this will be uh this
54:11
query here this will give us the uh kind of self drawing query but we're going to
54:16
remove this second wear condition because we don't care about that we will do that in the um uh in the next in the
54:24
next query here that will show us like how is going to work so we say select star from self joined you'll see in this
54:30
case now there's going to be all these other things that can happen right um
54:35
because oh wait yeah yeah that that should be fine so you'll see sometimes
54:40
there'll be uh a user who has like just a sign up or they will have like another
54:46
record here on the other side that will um potentially be um this is interesting
54:53
why are these all like the same because okay so this person visited sign up like 18 hours apart that's essentially what's
55:00
going on this person like refreshed sign up a lot before they actually like created but that's fine because uh like
55:07
we would expect that to be the case but what we want to do here is we want to aggregate along users essentially so we
55:13
can say uh so I have user ID and then uh it doesn't matter how many like uh
55:19
joined events here we have because obviously there's going to be a lot so in this case we have user ID and then uh
55:25
obviously the URL is just going to be sign up so we don't need to have that but we can have a a oh we need to change
55:32
this one uh we need to change this one as destination URL because otherwise we have see have URL and URL so we need to
55:39
change the second one so what we want to do is we want to say count we can say case when uh destination URL um equals

55:49
API V1 users then one end and this is going to be as um as
55:57
converted and oh wait a minute because this is but
56:03
this this is probably going to double count I want to make sure like because we only want to count each user once like whether or not they converted okay
56:10
so I think oh yeah see this is going to double count so what we need to do here is we need to throw we need to slap a
56:16
distinct on there so that uh it's just like a a Boolean right so we can see uh
56:22
that like it's actually there or uh you can actually no that's not the to do it distin that's like super ugly we want to
56:28
actually do Max right Max is going to that'll give it what we want you then we
56:34
want to put else zero here so we can have the so now this means that that
56:39
user converted and these other users didn't convert even but these other users did visit the signup page key
56:46
thing to remember here so we now have our uh users who convert it and the ones
56:53
so but we can also have uh another column here right which is going to be so we have the converted column so this
57:00
is essentially our next like this is like user level as so this is going to give us our users who have converted so
57:07
now we have user level and then we want to do one more aggregation here though right so if we say uh select and then we
57:13
can say a count one comma sum converted right we from user level this is going
57:22
to give us one big old number here right and so you'll see in this case we
57:29
have uh our number here of we have 145 counts of uh users and then the sum here
57:37
is 49 so we can also do a division here right and we can say like cast as real
57:45
so this is going to be our like overall conversion rate and but keeping in mind not this conversion rate is different

57:52
than the one in uh in here cuz in here the difference is is I
57:59
um okay because in this one the difference is is like we do it where it's at the um it's at the URL level
58:07
instead of at the um the user level so the essentially what this means is
58:14
33.7% of all the users who ever visited
58:19
the signup page will convert that's essentially what this means right
58:25
because we have all the users here and then that like and that was the conversion rate so this is going to be
58:30
as percent converted and so this is going to be one way to do it um I want
58:36
to compare this though because I think that this this is going to be very interesting to y'all so if we throw in
58:42
uh URL here and then instead we group on uh both username and URL but in this
58:48
case we also need to put a count so count one is this is as number um of hits so because just because the
58:56
conversion rate is that good where it's this high right that doesn't necessarily mean that that's the level that you
59:03
would expect from each individual uh like web traffic right because some
59:09
users might have visited like the sign up page 10 times and then they finally uh converted on the last go right and so
59:16
that that that's different that's like the per hit as opposed to the per user conversion and so in this case uh you'll
59:23
see where what we can do here is we actually want to uh stomp out this line 21 so that we can just have like
59:30
wherever they're coming from so in this case we're gonna have URL and then we're going to have some number of hits uh
59:37
number of hits uh and then we want to group by
59:43
URL and so now this it should give us a a fairly different picture like how is

59:50
that oh yeah because we need to like stomp out that too we got to have like all the hits CU now this should give us
59:56
a very different picture okay there we go so oh yeah that makes sense that API
1:00:02
users uh convert to themselves very well okay so what I want to show here though
1:00:08
is um this is not quite right because this is we want to not divide by count
1:00:14
one we want to divide by number of hits because count one actually doesn't give us very much so this is going to be uh
1:00:21
let's put some converted outside here as num converted and this going to be M as
1:00:27
num hits and so we want to put a having clause in here because there's going to
1:00:32
be a lot of garbage you see all just like marijuana. PHP or just like people like hit my website and they try to go
1:00:39
to random paths all the time it's like super obnoxious so what we want to do here is we want to put like a having
1:00:45
here we can say having um some uh number of hits greater than maybe like let's
1:00:51
put like 500 so that like we don't have like all these like stupid hits in here then we can run this
1:00:58
query okay there we go so that no that didn't
1:01:05
run there we go see it's it's a lot slower now because I didn't filter out anything there we go so now we have our
1:01:12
like we have about 110 records and you'll see that like some pages are going to convert better than other Pages
1:01:18
like which makes a lot of sense that you would expect that so in this case we can
1:01:24
sort and and okay so now this is going to give us a lot clearer picture of who
1:01:30
converted where right and this these numbers seem like a lot clearer and a lot better so you'll notice uh that some
1:01:39
of these Pages uh like we have like essentially like a a 3% like conversion
1:01:45

rate here I think there's going to be a oh there is a divisor problem here though because this converted is like a
1:01:53
one-time event whereas uh the other one is knocked right the other
1:01:58
one is going to be the the all the times that they visited the website so like even if I so you know I signed up but
1:02:05
then if I visited the sign up page again I'm not going to sign up again but that's still going to count in the denominator so that's why these numbers
1:02:12
now look a lot worse you saw how like in the previous example we had like at the user level and the conversion rate
1:02:18
looked like 30% and now it's like 3% but that's because of like all of the like
1:02:24
there's a lot of bots and a lot of other things that are in here that are going to be uh really dragging this number
1:02:30
down quite a bit um and so that's going to be like probably the big thing that I
1:02:36
would say that's definitely happening here um but uh this is the idea but you
1:02:41
can see how like oh yeah if they actually signed up and like then that's going to be where you get this conversion here I'm I'm curious like if
1:02:48
we change this to a sum does what what does that number look like
1:02:56
okay so okay this this is probably actually more accurate yeah this I think this is going to be more accurate on
1:03:02
like the uh like what I would say is the numbers because obviously if they signed up they signed up but like in this case
1:03:08
why these numbers are different right that's a that's an interesting thing and the reason for that is based on the
1:03:14
self-join logic right and because like uh the sign up doesn't uh isn't preceded
1:03:21
or isn't followed by a signup or what that means is that like half of these is like a sign up is followed by an edit
1:03:27
and so that's essentially what's happening here but this this conversion rate seems like more accurate for what

1:03:33
you would see in like normal web traffic right so it's like 2. 2.7% or so that's kind of the
1:03:41
perspective that I would expect uh when I'm uh creating my funnels and doing my
1:03:46
traffic is it's like okay they went to the signup page and now they're over here and it's now 2.7% which I think is
1:03:54
totally reasonable so and you'll see that like some of these Creator pages also have like a
1:03:59
decent percent because like they can edit as well and so this is kind of the idea that like and but you'll see some
1:04:06
of these Pages down here are terrible like so it's like the about page so if someone lands on the about page the odds
1:04:12
that they sign up is very low it's like 0 2% and that yeah these numbers like I
1:04:17
I I I trust these these seem like pretty accurate now so anyways that's the idea
1:04:24
behind funnel analysis right right the idea here is you have two events right and you have like you give this as like
1:04:30
fact data in a lot of cases right where you have two events that like you want to see like this one came after this one
1:04:37
and uh how often does that happen versus other things happening and that's what we're going to that's essentially what's
1:04:42
going on here when we're going through this lab together that like now I know that like I probably need to increase
1:04:49
this number because this 2.7% is not good enough like I think that more people should uh be giving me their
1:04:55
email and conver I probably need to optimize things but obviously like this data only goes through like this is only up through
1:05:01
like March like I think this is only like the first month that I launched my startup I don't have data after that so
1:05:09
uh that's that can also be a little gotcha um for this so um now what I want
1:05:15

to talk about is I have another one here around grouping sets so what I'm going to do is I'm going to just grab this
1:05:22
real quick going to grab this just to start with and then um we can go and work from it from there okay so if I
1:05:29
say select star from events augmented like you can pull this query in from the repo uh this query now has uh just a
1:05:38
couple things so what I'm trying to do now is just look at my website events and try to see okay what event is um or
1:05:47
or what type of device is the most common device that hits my website and I
1:05:54
think that we'll be able to do a lot of really cool stuff with this as well so we have all the all this device
1:06:00
information here so you could imagine that what we could do uh if we were
1:06:06
doing this like kind of in a naive approach we could just say OS type
1:06:13
device type browser type and then we want to say count one and then we can
1:06:20
say Group by OS type device type browser type right and let me just like format
1:06:25
that real quick so this query is obviously going to run and then we can see oh we have all of our data here and
1:06:32
you'll see that like the most common one for me is Android on generic smartphone on Chrome mobile so that's uh the most
1:06:41
common one but like how what about all like I want to know about just Android
1:06:47
traffic not like Android generic smart home Chrome mobile traffic I just care about Android what if I care about
1:06:52
Android but I also care about other Cuts here so what we want to do is we're going to introduce A New Concept here
1:06:59
called grouping sets so if we go in here we can say Group by grouping sets and
1:07:05
the first grouping set like so you'll see if I don't put everything in the grouping set this query will fail
1:07:12
because you have to put all of the all of them in there at least once otherwise
1:07:18

it's not going to work so you can have that first like and this is essentially the finest grain which is our the old
1:07:24
data but then we can also put in what if we put in like browser type and we can put in OS type and we can put in device
1:07:31
type so these are three other uh grains that I want to put in and you'll see now
1:07:38
if we do this okay there we go so now you'll see
1:07:44
um we have uh like if we sort on count here okay you see these nulles see how
1:07:51
there's nulles here now so those nulles are because of grouping set so that's because those are ignored so
1:07:58
what that means is this is the browser type uh column that's what that means
1:08:03
and so a lot of times what people do is they're going to put coals on these columns so that the the data doesn't
1:08:10
look so weird and then this is going to be our uh and they coales it with overall in parenthesis to show like hey
1:08:18
this is our overall colum and it's kind it's similar to like you know like a grand total or like a subtotal if y'all
1:08:24
are like familiar with like Excel that's like uh where I would say this uh kind
1:08:29
of has a similar vibe to it right and so this is going to give us our you'll see
1:08:35
now if we run this query that's just going to get rid of those nulles and then we'll have
1:08:40
a can scroll up here and you'll see we have um we have a better idea here where
1:08:47
looks like a Chrome mobile is going to be our biggest one but I don't know if that's true we say order by um count one
1:08:55
descent ending I don't think that that's right
1:09:01
okay there we go okay so it looks like uh actually other is the most common and
1:09:07
I think that's actually from uh Bots like Google bot and stuff like that so
1:09:12

and then after that we have like iPhone and then like we have IOS at the highest level then we have iPhone then Chrome
1:09:20
then Android right those all make sense right then generic smartphone love that
1:09:25
I love that that's the name of it and then there's that Chrome mobile hit and obviously it looks like these ones like
1:09:31
Android generic smartphone and Chrome mobile these are like almost the same right you see how these dimensional cuts
1:09:36
are like almost the same so there's like a small number of values in these two spots that are different but what you
1:09:42
can see is like if you use grouping sets you get a lot of really powerful information here so one of the one of
1:09:48
the things I want to talk about though real quick is there's actually another thing here called grouping um I want to say this has is it
1:09:57
it's like this I want to say you use that I will uh let's uh see real quick
1:10:03
if that gives it okay cool so okay so what happens here is is if the so this
1:10:11
is how we determine if the grouping set includes or ignores uh each of the
1:10:18
things here right so let me let me show you more like if we add more of these groupings in here so if we say grouping
1:10:25
of of we'll say a device type and grouping of browser type these
1:10:32
are different right and um but like I think it'll make more sense when you see
1:10:37
like all of them together so okay so we have this overall other and you'll see
1:10:43
uh in this case the grouping is not device type but it uh what what does
1:10:49
means is device type is uh set on and then the other two are off that's what
1:10:55
this means right because you see how it's like overall here so in this case that means that grouping the the device
1:11:02
type grouping is there but the other ones are not so and you'll see in some of these cases
1:11:08

like for this Android generic smartphone in this case that means it's all three so what we want to do is we want to
1:11:16
essentially build a a new column for this table that uses these groupings so
1:11:21
we can say case when uh OS type equals zero than um OS type and
1:11:31
then uh that's going to give us our first um that will give us like that
1:11:36
that first aggregation right to understand like how all these uh work together right
1:11:43
um so there there's essentially like the way you want to do this is you want to do this in a smart way where it is kind
1:11:50
of it looks at like what your choices are here um so in this case we we know
1:11:55
that if uh they are all zero then uh like let's do that one
1:12:01
first so we're going to say case when grouping type equals zero and grouping uh device type equals zero and grouping
1:12:10
um browser type equals zero then this is our uh so in this case we have OS type
1:12:16
device type browser type so like at Facebook what we did was we kind of like split it out where like in between each
1:12:23
uh column name we put a double underscore to in signify that that's the aggregation level so then uh like then
1:12:31
the rest of them are just single singular right so then we can say when grouping uh browser type equals zero
1:12:39
then browser type and then when grouping uh device
1:12:46
type equals zero then device type and when grouping um OS
1:12:53
type equals zero then OS type and then this is and we can call
1:12:59
this as aggregation level so now if we run
1:13:07
this you'll see we have this nice um aggregation level column so what I want
1:13:12
to do is I'm just going to uh get rid of these top level groupings here and um
1:13:18

now we have our aggregation level and this we're going to say is as a number of
1:13:23
hits so we have this table and what I want to do is I want to call this we're going to create this as a table we're
1:13:29
going to call this create table uh this is going to be a device um hits
1:13:34
dashboard as so we're just going to create this table real quick and then we're going to run
1:13:41
this okay so now we have our table so if I say in this case I say um select star
1:13:46
from device hits dashboard where aggregation level equals device type
1:13:52
like this you'll see okay now we have uh we can see okay
1:13:58
here's our aggregation level device type and then we can get all the hits and then you can also say like OS type and
1:14:04
then you see how this is so powerful because now you can like just you don't
1:14:10
have to query the event data and it's just pre-aggregated and like you can now have uh like your dashboards can just be
1:14:16
powered based off of wear conditions and not based off group ey conditions so um
1:14:22
and then obviously you have the the big ith one right the this one OS type
1:14:27
device type like this guy and so you got to teach your analysts about the double underscore kind of uh me mechanism for
1:14:35
that aggregation level but then you can see like okay this is uh like like the
1:14:42
the cut that is the most common is going to be these three together if if we actually include all the
1:14:49
dimensions so um that's essentially what how this
1:14:55
Works um I kind of want to go over how this is different from like how this is
1:15:00
different from like uh like how grouping sets versus uh roll up and all this
1:15:07
works uh before that though I want to uh move this query into I want to move this
1:15:12
query as this quer is better it's a lot better than this
1:15:18

one so that and you could put other things in here right and in the old
1:15:24
query I had like some like sign up conversion rate stuff that was in there but like those numbers weren't even real
1:15:29
the really we have like this number of hits is going to be the big uh thing that we're going to want to look at so
1:15:36
if we change this right if I just change this real quick and I say root by Cube
1:15:41
and then I put OS type device type and um browser type in here and then uh got
1:15:49
to get rid of this create table real quick and then comment this out so if we just changes the cube real quick
1:15:58
you'll see that um sometimes we're going to have an aggregation level that's not there because this is like the overall
1:16:04
aggregation level and um then you'll see all these other ones that are going to are going to show up here that are based
1:16:11
on different uh versions right but the thing is is like with Cube you have
1:16:17
other combinations that you need to worry about you have the double combinations like OS type and browser
1:16:22
and um browser and device type right and you have those like the double combinations that you got to worry about
1:16:29
so um some of these aggregation levels are going to be wrong right you see like this one here like this this one is not
1:16:35
device type because it has Linux and other this one's actually uh device type and Os type together uh that's how that
1:16:43
that one works so I think there's another like I want to say that there's also one called grouping ID is there is
1:16:50
that oh no it's I think grouping that's in Presto I think that's like a slightly different way to do it but so you'll see
1:16:57
with Cube one of the big things you get is you get that overall right you get this like overall one that's like
1:17:03

essentially like grouping on nothing and you get that like like the the million hits or whatever that's in this data set
1:17:09
and you just have all the records and so that you can see how like cube is pretty
1:17:15
awesome so I just want to compare that with rollup so that y'all can kind of see very vividly like what's going on
1:17:21
here so in this case with uh with rollup
1:17:26
okay so okay rollup still does okay interesting rollup does still give you the um uh the overall uh aggregation but
1:17:35
then it also gives you uh then you'll see that for the rest of the time though that OS type is never overall it's
1:17:43
always it's always present but then you'll see that uh device type can be gone and browser type
1:17:50
can be gone but sometimes they're in there so this is what I was talking about the hierarchy right so OS type so
1:17:56
rollup gives you like the grand total which is nice but then OS type after that will always be in and it won't go
1:18:02
overall and then device type can be going overall and browser type can as well uh where but like but then uh when
1:18:10
you'll see like if a device um or if browser type is overall I mean no if
1:18:16
device type is overall browser type has to be overall you'll notice that that's going to always be the case because that
1:18:22
it's a hierarchy right so um it's kind of like that whole country state city kind of hierarchy and so um in this case
1:18:30
the um this would be like the state and that would uh that's that's essentially what's going on with that data so that
1:18:38
is essentially how you do um fancy like
1:18:43
grouping operations in postgress congrats on finishing the day two lab in the course I hope you enjoyed it if you
1:18:50

like this Channel and like this boot camp make sure to like comment and subscribe and share this content with

1:18:55

your friends I'm so happy that you're taking this time to invest in your knowledge and getting better at data see

1:19:00

you at the next one

English (auto-generated)

All

From Data with Zach

Data analysis

Software Engineering

Computer Science

Related

For you

Recently uploaded

Watched

Learning