-- #1 Return a list of users who blocked another user after connecting for at least 90 days.

-- Show user\_id and receiver\_id.

-- Return a pair of users who once connected for at least 90 days then blocked.

SELECT u.user\_id,

o.receiver\_id

FROM

(

-- Before a user can block another user, the users must be connected. Create two

-- sub-queries - one with blocked events and another with connected events.

-- Use inner join on user\_ids and receiver\_ids.

(SELECT \*

FROM connections

WHERE action = 'Blocked') u

JOIN

(SELECT \*

FROM connections

WHERE action = 'Connected') o

ON u.user\_id = o.user\_id AND u.receiver\_id = o.receiver\_id

)

WHERE (u.dates - o.dates) >= 90;

-- #2 - For each user, what is the proportion of each action? Note that the receiver\_id

-- can appear in multiple actions per user, only regard the latest action performed

-- when calculating the distribution.

-- Assign event order by user\_id and receiver\_id to filter on the latest event

WITH friendship\_status AS (

SELECT \*,

ROW\_NUMBER() OVER(PARTITION BY user\_id, receiver\_id ORDER BY dates DESC) AS event\_order

FROM connections

),

-- Filter on the latest event per user\_id and receiver\_id pair

latest\_friendship\_status AS (

SELECT \*

FROM friendship\_status

WHERE event\_order = 1

),

-- Create a dummy variable column per action type

status\_dummy\_variables AS (

SELECT \*,

CASE WHEN action = 'Sent' THEN 1 ELSE 0 END AS sent,

CASE WHEN action = 'Received' THEN 1 ELSE 0 END AS received,

CASE WHEN action = 'Connected' THEN 1 ELSE 0 END AS connected,

CASE WHEN action = 'Blocked' THEN 1 ELSE 0 END AS blocked

FROM latest\_friendship\_status

),

-- For each action column, divide by event order to get proportion of action types per user.

distribution AS (

SELECT user\_id,

SUM(sent) / SUM(event\_order) AS sent,

SUM(received) / SUM(event\_order) AS received,

SUM(connected) / SUM(event\_order) AS connected,

SUM(blocked) / SUM(event\_order) AS blocked

FROM status\_dummy\_variables

GROUP BY user\_id

)

SELECT \* FROM distribution;