

UNIVERSITÀ DEGLI STUDI DI PALERMO

Dipartimento di Scienze Economiche, Aziendali e Statistiche  
Master annuale di secondo livello in

**Data Science and Big Data Analytics**



# ANALISI E PULIZIA DI UN DATABASE SU BIGQUERY

---

Tesi di:  
Ludovica Tomaselli

Relatore:  
Prof. Marcello Chiodi  
Tutor aziendale:  
Dott.ssa Francesca Motisi

# INTRODUZIONE

---



SEDE  
TIROCINIO

Cloudtec



DURATA  
TIROCINIO

3 mesi



OGGETTO DEL  
TIROCINIO

Studio Google Cloud Platform  
Analisi e pulizia database

# BIGQUERY

The screenshot displays the Google Cloud Platform BigQuery interface. At the top, the navigation bar includes the Google Cloud Platform logo, the project name 'My Project 16084', and a search bar. Below the navigation bar, the left sidebar contains the 'Explorer' panel, which shows a search for 'public' and a list of datasets under 'bigquery-public-data'. The main panel is divided into two sections: the top section shows the query editor with a query that selects all data from the 'bigquery-public-data.covid19\_public\_forecasts.county\_14d' table, limited to 1000 rows. The bottom section shows the query results, which are displayed as a table with 10 columns: 'Riga', 'county\_fips\_code', 'county\_name', 'state\_name', 'forecast\_date', 'prediction\_date', 'new\_confirmed', 'cumulative\_confirmed', 'new\_confirmed\_7day\_rolling', and 'new\_dea'. The table contains 8 rows of data, showing information for various US counties and states.

Google Cloud Platform My Project 16084 Cerca Prodotti, risorse, documenti (/)

FUNZIONALITÀ E INFORMAZIONI SCORCIATOIA DISABILITA SCHEDE EDITOR

Explorer

public

Trovati 24 results. [Limita la ricerca ai progetti fissati.](#)

bigquery-public-data

covid19\_public\_forecasts

county\_14d

county\_14d\_historical

county\_14d\_historic...

county\_28d

county\_28d\_historical

county\_28d\_historic...

japan\_prefecture\_28d

japan\_prefecture\_28...

japan\_prefecture\_28...

state\_14d

state\_14d\_historical

state\_14d\_historical\_

state\_28d

state\_28d\_historical

state\_28d\_historical\_

EDITOR 2 COUNTY... QUERY N...3

ESEGUI SALVA CONDIVIDI PROGRAMMAZIONE ALTRO

Questa query elaborerà 13,1 MiB quando verrà eseguita.

1 SELECT \* FROM `bigquery-public-data.covid19\_public\_forecasts.county\_14d` LIMIT 1000

Località di elaborazione: US

Risultati delle query SALVA RISULTATI ESPLORA I DATI

Query completata (tempo trascorso: 0,3 sec, elaborati 302,9 kB)

Informazioni job Risultati JSON Dettagli esecuzione

Riga	county_fips_code	county_name	state_name	forecast_date	prediction_date	new_confirmed	cumulative_confirmed	new_confirmed_7day_rolling	new_dea
1	55001	Adams	Wisconsin	2022-02-05	2022-01-31	null	null	55.57142857142857	
2	48005	Angelina	Texas	2022-02-05	2022-01-31	null	null	58.42857142857143	
3	08009	Baca	Colorado	2022-02-05	2022-01-31	null	null	5.142857142857143	
4	20011	Bourbon	Kansas	2022-02-05	2022-01-31	null	null	29.428571428571427	
5	13039	Camden	Georgia	2022-02-05	2022-01-31	null	null	108.42857142857143	
6	42027	Centre	Pennsylvania	2022-02-05	2022-01-31	null	null	153.42857142857142	
7	22027	Claiborne	Louisiana	2022-02-05	2022-01-31	null	null	2.142857142857143	
8	17029	Coles	Illinois	2022-02-05	2022-01-31	null	null	106.71428571428571	

Righe per pagina: 100 1 - 100 di 1000 Prima pagina < > >| Ultima pagina

CRONOLOGIA PERSONALE CRONOLOGIA PROGETTO QUERY SALVATE

ID	INTEGER	NULLABLE
CREATED_DATE	DATETIME	NULLABLE
BARCODE	STRING	<b>REQUIRED</b>
AGENCY_ID	INTEGER	<b>REQUIRED</b>
AGENCY_NAME	STRING	NULLABLE
NOTE	STRING	NULLABLE
RECIPIENT_TYPE	STRING	NULLABLE
DOCUMENT_TYPE	STRING	NULLABLE
DOCUMENT_NUMBER	STRING	NULLABLE
PRODUCT_TYPE	STRING	<b>REQUIRED</b>
↓ STATE	RECORD	REPEATED
NAME	STRING	<b>REQUIRED</b>
SEGNCOD	STRING	NULLABLE
USER_ID	INTEGER	NULLABLE
USER_USERNAME	STRING	<b>REQUIRED</b>
CREATED_DATE	DATETIME	NULLABLE
DELIVERY_SEND_STATE	STRING	NULLABLE
DELIVERY_CREATED_DATE	DATETIME	NULLABLE
DELIVERY_LATITUDE	FLOAT	NULLABLE
DELIVERY_LONGITUDE	FLOAT	NULLABLE

# CARATTERISTICHE DATABASE

- ❖ Client:  
MySQL8
- ❖ Finalità:  
tracciabilità lettere
- ❖ Struttura:  
1 tabella annidata

## TABELLA PRINCIPALE

ID	Codice identificativo di ogni lettera ricevuta, meno affidabile del barcode dato che può essere mancante.
CREATED_DATE	Data di ricezione della lettera.
BARCODE	Si riferisce al barcode della lettera ed è il codice identificativo primario.
AGENCY_ID	Codice identificativo dell'agenzia che ha accettato la lettera.
AGENCY_NAME	Il nome dell'agenzia che ha accettato la lettera o pacco.
NOTE	Eventuali note.
RECIPIENT_TYPE	Ruolo di chi riceve la lettera (e.g. destinatario, parente, delegato, ente, etc).
DOCUMENT_TYPE	Tipo di documento mostrato alla ricezione (se necessario).
DOCUMENT_NUMBER	Codice di documento mostrato alla ricezione (se necessario).
PRODUCT_TYPE	Tipo di lettera (sdoc, parcella, raccomandata)

## TABELLA ANNIDATA

↓ STATE	
NAME	Identifica se la lettera è in accettazione o in consegna
SEGNCOD	Identifica la modalità di consegna o la ragione di una mancata consegna
USER_ID	Codice identificativo del postino
USER_USERNAME	Nome identificativo del postino
CREATED_DATE	Data del movimento indicato nella riga (accettazione o tentativo di consegna)
DELIVERY_SEND_STATE	Identifica se la lettera è stata inviata o se si è verificato un errore
DELIVERY_CREATED_DATE	Data in cui la lettera è stata consegnata
DELIVERY_LATITUDE	I due ambiti, latitudine e longitudine, si riferiscono alla posizione della consegna
DELIVERY_LONGITUDE	

## TABELLA PRINCIPALE

ID	Codice identificativo di ogni lettera ricevuta, meno affidabile del barcode dato che può essere mancante.
CREATED_DATE	Data di ricezione della lettera.
BARCODE	Si riferisce al barcode della lettera ed è il codice identificativo primario.
AGENCY_ID	Codice identificativo dell'agenzia che ha accettato la lettera.
AGENCY_NAME	Il nome dell'agenzia che ha accettato la lettera o pacco.
NOTE	Eventuali note.
RECIPIENT_TYPE	Ruolo di chi riceve la lettera (e.g. destinatario, parente, delegato, ente, etc).
DOCUMENT_TYPE	Tipo di documento mostrato alla ricezione (se necessario).
DOCUMENT_NUMBER	Codice di documento mostrato alla ricezione (se necessario).
PRODUCT_TYPE	Tipo di lettera (sdoc, parcella, raccomandata)

## TABELLA ANNIDATA

↓ STATE	
NAME	Identifica se la lettera è in accettazione o in consegna
SEGNCOD	Identifica la modalità di consegna o la ragione di una mancata consegna
USER_ID	Codice identificativo del postino
USER_USERNAME	Nome identificativo del postino
CREATED_DATE	Data del movimento indicato nella riga (accettazione o tentativo di consegna)
DELIVERY_SEND_STATE	Identifica se la lettera è stata inviata o se si è verificato un errore
DELIVERY_CREATED_DATE	Data in cui la lettera è stata consegnata
DELIVERY_LATITUDE	I due ambiti, latitudine e longitudine, si riferiscono alla posizione della consegna
DELIVERY_LONGITUDE	

```
THEN 1 ELSE 0 END) AS  
`database-dev-332416.datab  
T(state) AS s)
```

```
WHEN s.user_username = 'Phi'  
AND s.created_date BETWEEN '2021-  
AND '2021-10-10T00:00:00.000'  
AND s.delivery_latitude IS NOT NULL  
AND s.delivery_longitude IS NOT NULL  
THEN 1 ELSE 0 END) AS deliveries  
`database-dev-332416.database_prod.product`a  
T(state) AS s)
```

```
WHEN s.user_username = 'Phi'  
AND s.created_date BETWEEN '2021-09-01T00:00:00.000  
AND '2021-10-10T00:00:00.000'  
AND s.delivery_latitude IS NULL
```

# OBIETTIVO

VERIFICARE:

- ❖ Bontà dei dati
- ❖ Tracciabilità movimenti

# ANALISI CONOSCITIVA

---



1.261

Numero  
postini



361

Giorni  
di attività



16.677.672

Totale barcode  
gestiti



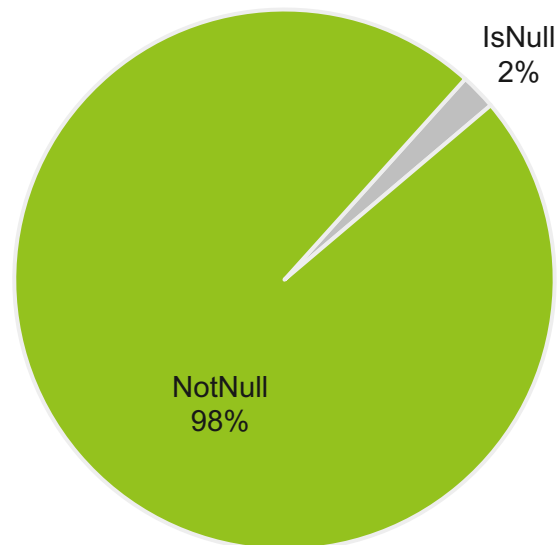
# ANALISI DATI MANCANTI


---

- ❖ Latitudine e longitudine
- ❖ created\_date
- ❖ state.created\_date

total	NotNull	IsNull
17672087	17290203	381884

Null_percent	NotNull_percent
2.16	97.84



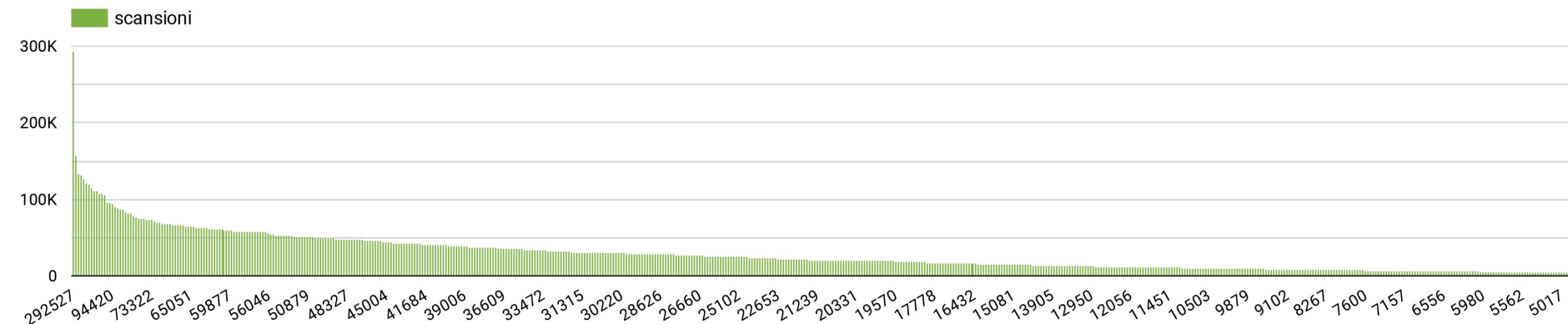


```
SELECT total, NotNull, IsNull, ROUND(IsNull * 100.0 / total,2) AS Null_percent, ROUND(NotNull *
100.0 / total,2) AS NotNull_percent
FROM (SELECT
    (SELECT COUNT(*) AS deliveries
     FROM `databasedatabase-dev-332416.databasedatabase_prod.product`as p,
     UNNEST(state) AS s)
    AS total,
    (SELECT SUM
     (CASE WHEN s.delivery_latitude IS NOT NULL
      AND s.delivery_longitude IS NOT NULL
      THEN 1 ELSE 0 END) AS deliveries
     FROM `databasedatabase-dev-332416.databasedatabase_prod.product`as p,
     UNNEST(state) AS s)
    AS NotNull,
    (SELECT SUM
     (CASE WHEN s.delivery_latitude IS NULL
      AND s.delivery_longitude IS NULL
      THEN 1 ELSE 0 END) AS deliveries
     FROM `databasedatabase-dev-332416.databasedatabase_prod.product`as p,
     UNNEST(state) AS s)
    AS IsNull);
```

# ANALISI DATI MANCANTI

# VALORI FUORI SCALA

## TOTALE SCANSIONI PER POSTINO



SCANSIONI  
GIORNALIERE  
PER OGNI  
POSTINO

TUTTE  
SCANSIONI  
SINGOLO  
POSTINO

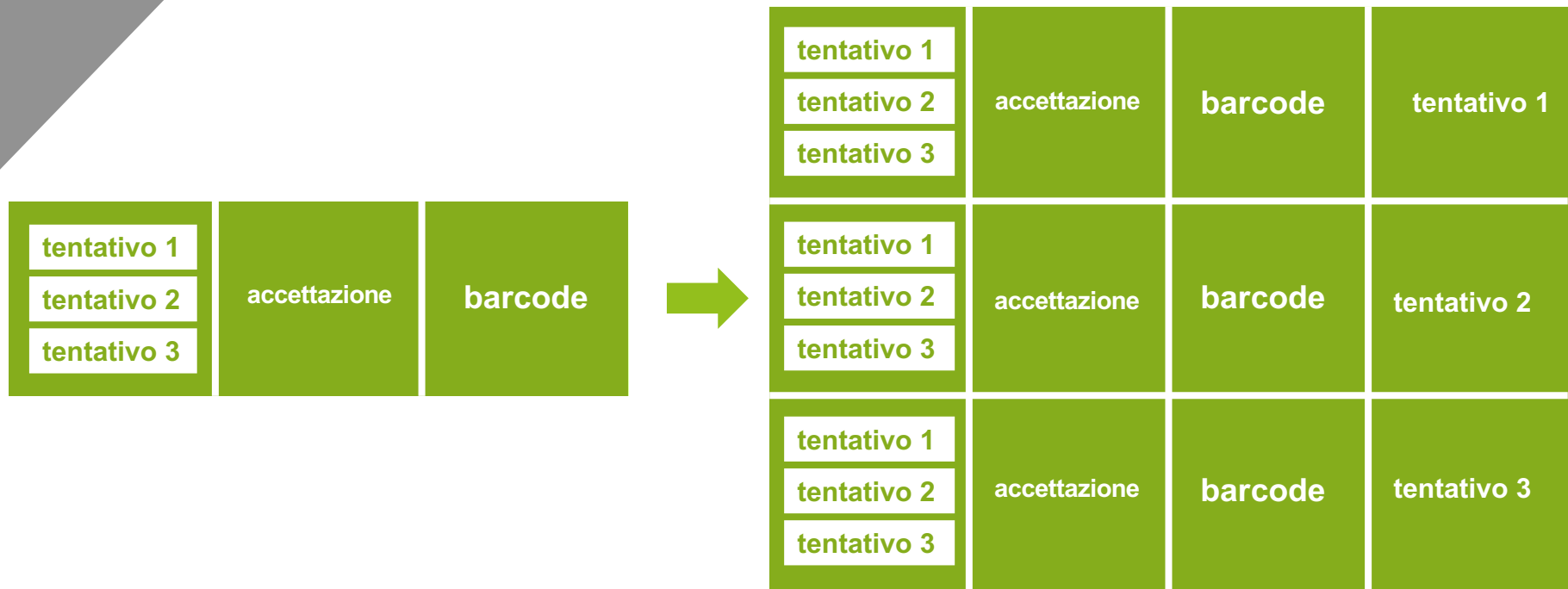
SCANSIONI  
MENSILI E  
SETTIMANALI


# IL PROBLEMA DELL'UNNEST

---

tentativo 1	accettazione	barcode
tentativo 2		
tentativo 3		

# IL PROBLEMA DELL'UNNEST





```
SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
FROM
(
    SELECT  s.created_date, s.user_username
    FROM `database`database-dev-332416.database`database_prod.product`AS p,
    UNNEST(state) AS s
    WHERE s.name = '_CONSEGNA'
    GROUP by s.created_date, s.user_username
) AS res
GROUP BY giorno, postino
ORDER BY scansioni DESC;
```

# SOLUZIONE UNNEST

# PULIZIA DEI DATI

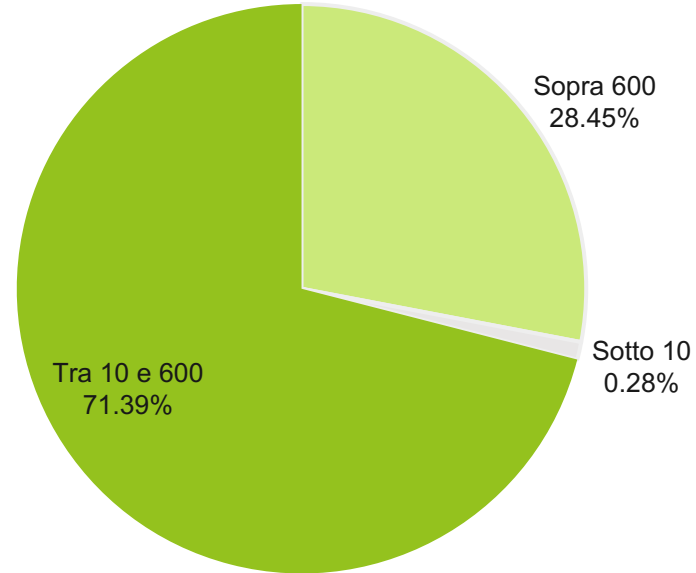
```

SELECT totale, Sopra_600, Sotto_10 , Tra_10_e_600, ROUND(Sopra_600 * 100.0 /
totale,2) AS Sopra_600_percent, ROUND(Sotto_10 * 100.0 / totale,2) AS
Sotto_10_percent, ROUND(Tra_10_e_600 * 100.0 / totale,2) AS Tra_10_e_600_percent
FROM (SELECT
(
SELECT COUNT(test.scansioni)
FROM
(SELECT giorno, scansioni, postino
FROM
(SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
FROM
(
SELECT s.created_date, s.user_username
FROM `database-dev-332416.database_prod.product` AS p,
UNNEST(state) AS s
WHERE s.name = 'CONSEGNA'
GROUP by s.created_date, s.user_username
) AS res
GROUP BY giorno, postino
ORDER BY scansioni DESC)) as test
)
AS totale,

(
SELECT COUNT(test.scansioni)
FROM
(SELECT giorno, scansioni, postino
FROM
(SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
FROM
(
SELECT s.created_date, s.user_username
FROM `database-dev-332416.database_prod.product` AS p,
UNNEST(state) AS s
WHERE s.name = 'CONSEGNA'
GROUP by s.created_date, s.user_username
) AS res
GROUP BY giorno, postino
ORDER BY scansioni DESC)
WHERE scansioni >= 600) as test
)
AS Sopra_600,

(
SELECT COUNT(test.scansioni)
FROM
(SELECT giorno, scansioni, postino
FROM
(SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
FROM
(
SELECT s.created_date, s.user_username
FROM `database-dev-332416.database_prod.product` AS p,
UNNEST(state) AS s
WHERE s.name = 'CONSEGNA'
GROUP by s.created_date, s.user_username
) AS res
GROUP BY giorno, postino
ORDER BY scansioni DESC)
WHERE scansioni <= 10) as test
)
AS Sotto_10,


(
SELECT COUNT(test.scansioni)
FROM
(SELECT giorno, scansioni, postino
FROM
(SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
FROM
(
SELECT s.created_date, s.user_username
FROM `database-dev-332416.database_prod.product` AS p,
UNNEST(state) AS s
WHERE s.name = 'CONSEGNA'
GROUP by s.created_date, s.user_username
) AS res
GROUP BY giorno, postino
ORDER BY scansioni DESC)
WHERE scansioni BETWEEN 10 AND 600) as test
)
AS Tra_10_e_600);
    
```



Totale	Sopra_600	Sotto_10	Tra_10_e_600
14930960	4247380	41123	10659787

Sopra_600_percent	Sotto_10_percent	Tra_10_e_600_percent
28.45	0.28	71.39

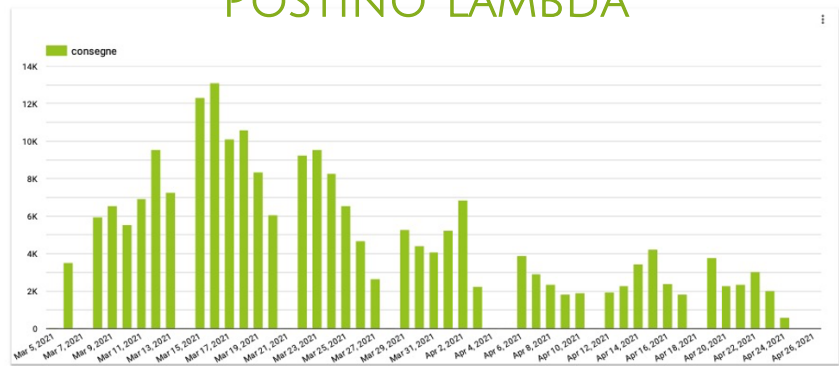


```
SELECT COUNT(test.scansioni)
      FROM
      (SELECT giorno, scansioni, postino
      FROM
      (SELECT DATE(res.created_date) AS giorno , COUNT(res.created_date) AS
scansioni, res.user_username AS postino
      FROM
      (   SELECT  s.created_date, s.user_username
      FROM `databasedatabase-dev-332416.databasedatabase_prod.product` AS p,
      UNNEST(state) AS s
      WHERE s.name = `CONSEGNA`
      GROUP by s.created_date, s.user_username
      ) AS res
      GROUP BY giorno, postino
      ORDER BY scansioni DESC)
      WHERE scansioni BETWEEN 10 AND 600) as test
```

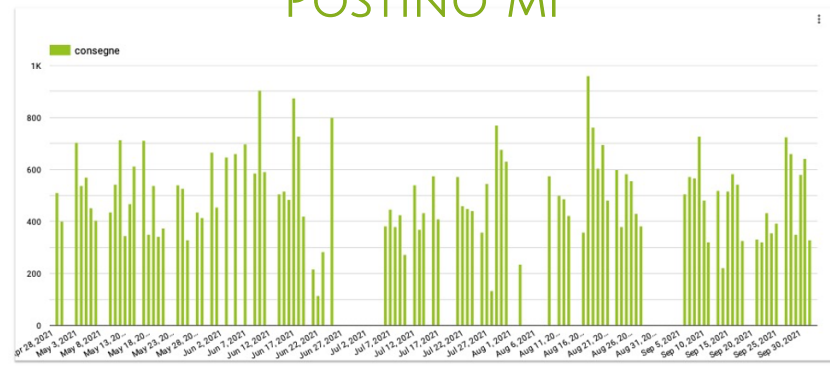
# STRALCIO CODICE



## POSTINO LAMBDA

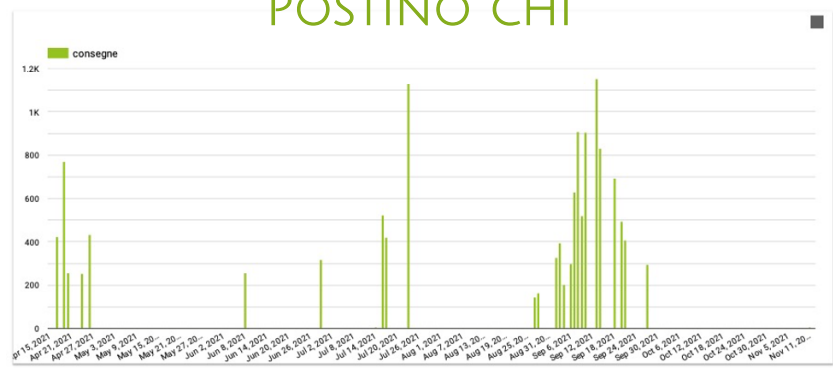


## POSTINO MI

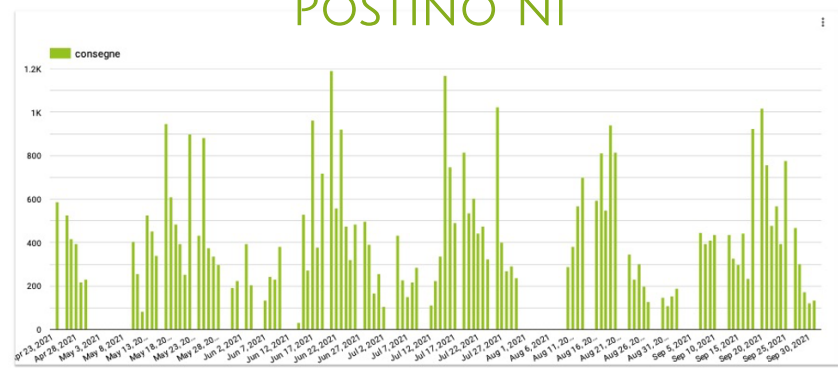


# SCANSIONI GIORNALIERE

## POSTINO CHI



## POSTINO NI





**Mappatura  
consegne**

## SVILUPPI FUTURI

- ❖ Estrazione percorsi
- ❖ Generazione percorsi ottimali
- ❖ Calcolo tempistiche
- ❖ Analisi aree più attive
- ❖ Analisi trend crescita
- ❖ Comparazione crescita economica



GRAZIE

---

