

UNIVERSITA' DI MILANO-BICOCCA

PROGETTO DI DATA MANAGEMENT

**TRAFFICO E METEO A MILANO:
COSTRUZIONE E ANALISI DI UN DATASET**

GRAVINA GRETA, 881470
PUCCINELLI NICCOLÒ, 881395
SCATASSI MARCO, 883823



Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

- Costruzione di un dataset su meteo e traffico
- Milano: Cerchia dei Navigli + Cerchia dei Bastioni
- Totale: 24 vie
- Dal 28 dicembre 2021 al 10 gennaio 2022

DOMANDE DI RICERCA

Traffico e:

- Condizioni meteorologiche
- ZTL
- Giorno e ora
- Vie e circonvallazioni

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TRAFFICO

- TomTom API
- <https://developer.tomtom.com/traffic-api/api-explorer>
- 2500 chiamate giornaliere al servizio (4 chiavi)
- Output: JSON

METEO

- Openweather API
- <https://openweathermap.org/current>
- Nessun limite alle chiamate (una chiave)
- Output: JSON

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

METODOLOGIA

- API GET tramite coordinate geografiche
- Verifica delle coordinate per il traffico
- Una richiesta per ciascuna via ogni 10 minuti (traffico e meteo)
- Linguaggio Python
- *BlockingScheduler*, libreria *apscheduler*
- *get*, libreria *requests*
- *append* dei file JSON

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TRAFFICO

- Body della richiesta con i dati del traffico
- Header della richiesta con le informazioni temporali
- *append* dell'header dopo ogni body per identificare eventuali valori mancanti
- Un JSON per ogni via

METEO

- Body della richiesta con i dati del meteo
- Informazioni temporali già contenute nel campo **dt**
- Un JSON per ogni via

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TOTALE

48 file JSON:

- 24 per il traffico
 - 13 per la cerchia dei Bastioni
 - 11 per la cerchia dei Navigli
- 24 per il meteo
 - 13 per la cerchia dei Bastioni
 - 11 per la cerchia dei Navigli

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

DATA EXPLORATION AND DATA CLEANING

TRAFFICO E METEO

- Verifica della presenza di file non correttamente scaricati
- Conversione degli apici singoli (') in doppie virgolette (")
- Lettura delle stringhe in JSON

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TRAFFICO

- Conversione del campo **date** da UTC a CET
- Sostituzione campo **coordinates** con **street**
- Aggiunta dei campi:
 - **ringRoad;**
 - **roadType;**
 - **LTZ;**
 - **flowConditions**
- Eliminazione dei campi non utili all'analisi

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

METEO

- Conversione dei campi orari **sunrise**, **sunset**, e **dt** nello stesso formato usato per il campo **date** per il traffico
- Sostituzione campo **coordinates** con **street**
- Aggiunta del campo **ringRoad**
- Eliminazione dei campi non rilevanti o ridondanti
- Riorganizzazione della struttura dei JSON

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

DATA INTEGRATION

TRAFFICO E HEADER

- 3 dizionari per ogni circonvallazione:
 - Traffico;
 - Header traffico;
 - Meteo
- Integrazione traffico con header
- Unione circonvallazioni
- Risultato parziale:
 - Dizionario traffico (24 vie)
 - Dizionario meteo (24 vie)

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TRAFFICO E METEO

- Sincronizzazione temporale
- Diversi record mancanti
- Per ciascuna via, per ogni record del traffico si sceglie il record del meteo con l'orario di aggiornamento della stazione meteorologica più vicino al campo **date** del traffico
- Integrazione traffico con meteo (campo **weather**)

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

TEMPO

- Campo **dateTime** per ogni record contenente tutte le informazioni temporali:
 - **trafficDate**: Data di misurazione del traffico
 - **weatherDate**: Data di misurazione del meteo
 - **time**: Orario teorico di misurazione
 - **effectiveTrafficTime**: Orario effettivo di misurazione del traffico
 - **effectiveWeatherTime**: Orario effettivo di misurazione del meteo

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

MONGODB

Motivazioni principali:

- Formato JSON dei dati restituiti dalle API
- Struttura complessa per evitare un numero elevato di join
- Obiettivo: formulare query complesse a diversi livelli di dettaglio
- PyMongo, costruzione del database e struttura della collection

Obiettivo

Fonti

Data acquisition

Data preparation

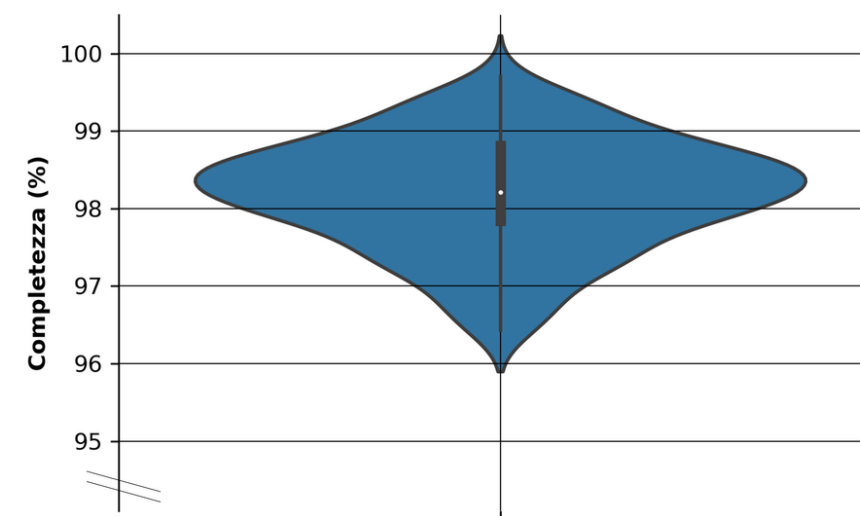
Data storage

Data quality

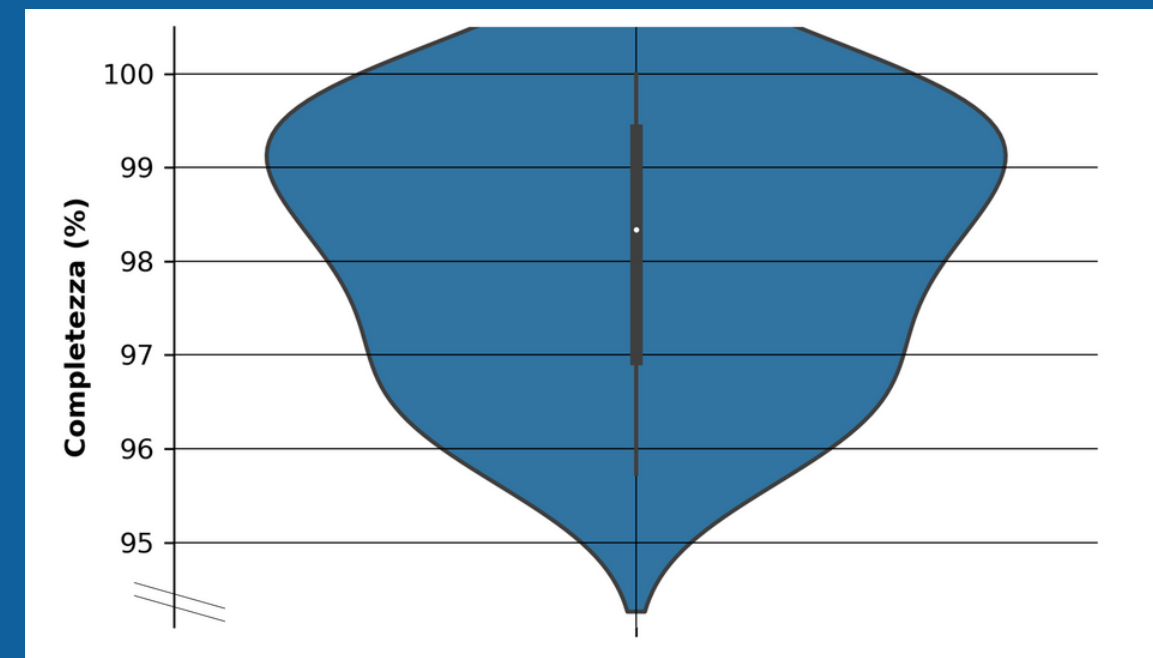
Conclusioni

DIMENSIONI DI QUALITÀ

- **Consistenza:** standardizzazione dei dati
- **Completezza:** ipotesi di mondo aperto, *object completeness*: documenti mancanti per numero di ore e numero di vie.
- Ex: 14:50:00 (99.7%) e 20:50:00 (96.43%)
- Ex: Via de Amicis (100%) e Bastioni Porta Volta (95.73%)



Ore



Vie

Obiettivo

Fonti

Data acquisition

Data preparation

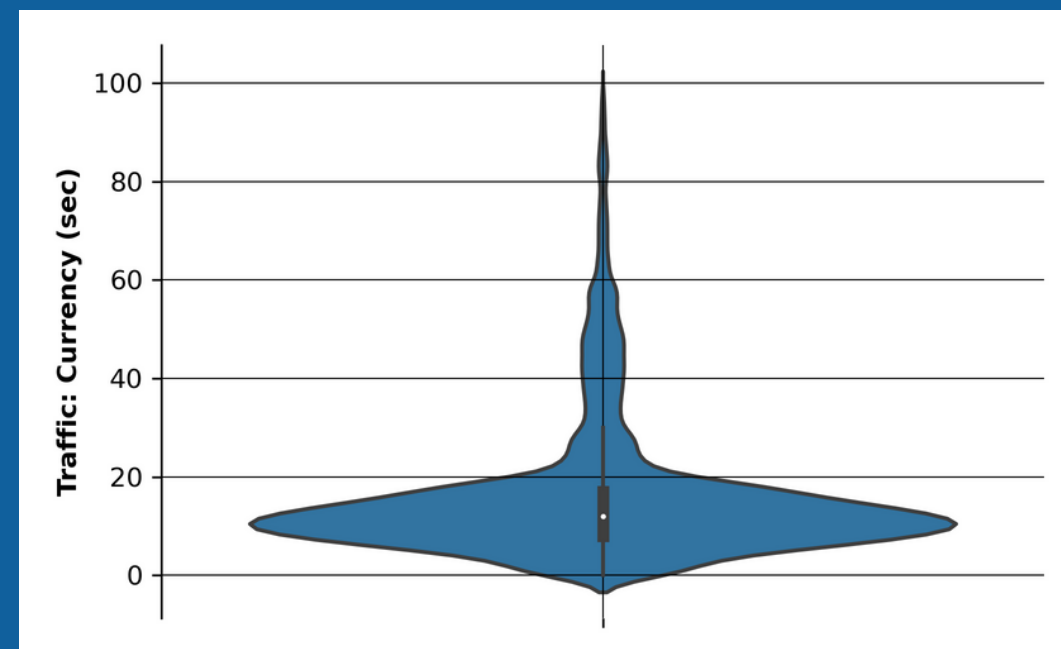
Data storage

Data quality

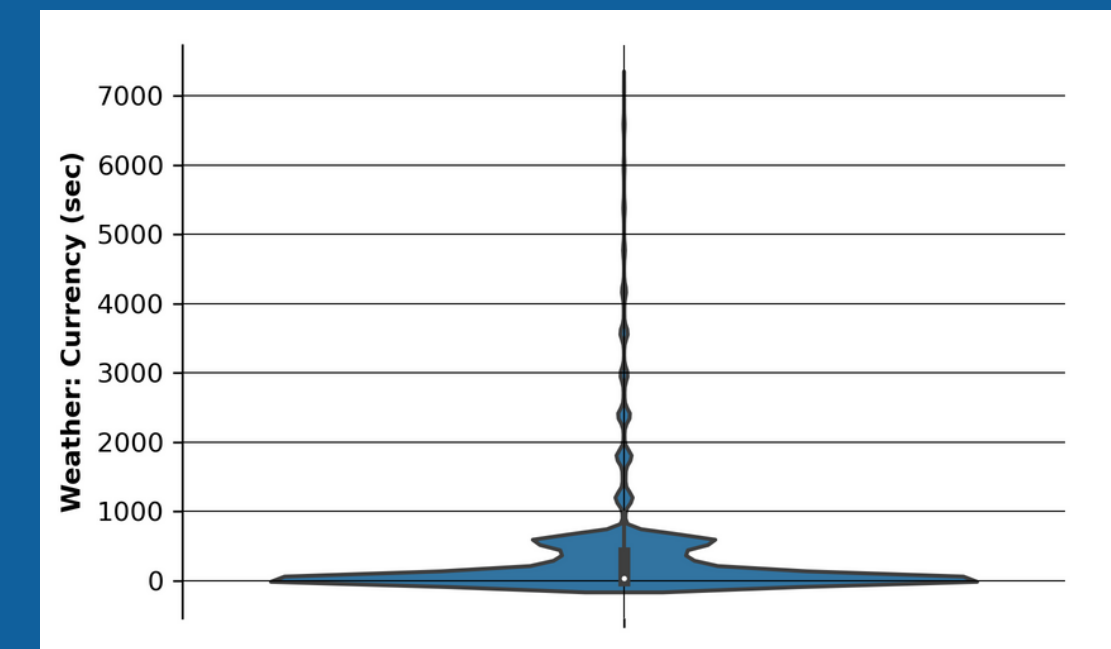
Conclusioni

DIMENSIONI DI QUALITÀ

- **Currency:** differenza in secondi tra tempo ideale di misurazione (*time* di *DateTime*) e tempo effettivo di arrivo della risposta (*effectiveTrafficTime* e *effectiveWeatherTime*)
- Variabilità del tempo di aggiornamento delle stazioni meteorologiche



Traffico



Meteo

Obiettivo

Fonti

Data acquisition

Data preparation

Data storage

Data quality

Conclusioni

CONCLUSIONI E SVILUPPI FUTURI

Sintesi del lavoro svolto

Utilizzo del dataset per:

- Analisi e predizione della viabilità milanese secondo diverse variabili (meteo, giorno, ...)
- Verifica del funzionamento di eventuali misure predisposte dal Comune di Milano (e.g. ZTL)

GRAZIE PER L'ATTENZIONE

GRAVINA GRETA
PUCCINELLI NICCOLÒ
SCATASSI MARCO