

---

# A2A CHALLENGE: LA RACCOLTA RIFIUTI INTELLIGENTE

---

Tommaso Cattaneo<sup>1</sup>, Lorenzo Sasso<sup>1</sup>

<sup>1</sup>CdLM Data Science, Università degli Studi di Milano – Bicocca

Come rendere la raccolta rifiuti più efficiente e meno dispersiva? Questa è la domanda che si è posta il gruppo A2A, la più grande multi-utility italiana che, oltre a essere leader nella generazione, la distribuzione e la vendita di energia elettrica e gas nel capoluogo lombardo, svolge altre numerose attività, tra cui la raccolta e recupero dei rifiuti.

Proprio quest'ultima è affidata ad Amsa, società del gruppo A2A, che opera su tutto il territorio della città di Milano e in alcuni comuni della sua provincia.

---

## INTRODUZIONE

---

La raccolta e gestione dei rifiuti ricopre un ruolo fondamentale all'interno della gestione comunale e negli ultimi anni ha subito una forte evoluzione. Amsa, in collaborazione con Cefriel – Politecnico di Milano, a partire dal 2016, ha installato nella città i primi smart bin.

Smart bin è un cestino intelligente, in grado di rilevare il proprio stato di riempimento, la frequenza e l'orario di utilizzo attraverso una scansione costante effettuata da un sensore e successivo invio dei dati raccolti ad un software di back-end, tramite rete wireless low cost. Il cestino invia misurazioni relative al suo stato di riempimento (vuoto, pieno a metà o

completamente pieno) ed occlusione (occluso o non occluso)<sup>1</sup>.

Questo progetto nasce con l'intenzione di ottimizzare il processo di raccolta di rifiuti, riducendo l'impiego di risorse e mezzi, garantendo tempestività di intervento per le situazioni critiche. È stata quindi effettuata una cluster analysis sfruttando i dati forniti dai cestini.

---

## DESCRIZIONE DATASET

---

Il dataset fornito comprende le rilevazioni giornaliere di 3148 cestini distribuiti in 16 diverse aree di Milano. Ogni record fa riferimento ad una singola rilevazione e comprende le seguenti features:

- Bin serial: identificativo del cestino
- Bin level: livello di riempimento del cestino (1-4)
- Occluded: cestino non occluso (1) o occluso (2)
- Detected\_at: data e ora della rilevazione
- Latitudine
- Longitudine
- Indirizzo
- Area gestionale: area della città in cui è posizionato il cestino.

---

<sup>1</sup> <http://www.a2asmartcity.io/>

---

## VINCOLI PROGETTUALI

---

Il progetto è stato realizzato sulla base dei seguenti vincoli e variabili di progetto.

### Vincoli aziendali e contrattuali

- Ogni cestino dev'essere svuotato almeno una volta al giorno;
- Minimizzare il numero di visite per ogni cestino;
- Minimizzare il numero di km percorsi da ogni mezzo.

### Variabili di progetto

- Numero dei mezzi disponibili (20);
- Capienza illimitata di ogni mezzo;
- Tutti i mezzi partono dallo stesso punto (Via Olgettina, Milano);
- Ogni mezzo riesce a coprire tra i 150 e i 200 cestini in un turno.

---

## DATA PREPROCESSING

---

Per lo svolgimento del progetto si è utilizzata l'ultima versione della piattaforma Knime per poter avere accesso a tutte le estensioni e i nodi necessari per l'implementazione del workflow. Inoltre, all'interno di quest'ultimo si è utilizzato in alcuni casi il linguaggio di programmazione R.

Inizialmente, per permettere di lavorare con attributi contenenti date e orari, è stato convertito il campo "detected\_at" da stringa a dato di tipo *Date&Time*. Successivamente, è stato effettuato il *missing replacement* con cui si è deciso di rimuovere i records laddove potrebbero esserci valori mancanti in corrispondenza di attributi essenziali, quali "bin serial", "longitude" e "latitude"; per altre variabili, come "bin level" e "occluded", il valore mancante viene sostituito

con il valore massimo per fare in modo che, in mancanza di informazioni, vengano considerati come cestini pieni e/o occlusi. Per attributi di tipo stringa, come "indirizzo", "area gestionale", non viene effettuata alcuna modifica in quanto i dati possono essere dedotti da latitudine e longitudine.

---

## CLUSTERING

---

L'idea centrale per la risoluzione di questo problema si basa sulla clusterizzazione dei cestini sfruttando in tempo reale i dati trasmessi da quest'ultimi.

La clusterizzazione in tempo reale è stata pensata per poter minimizzare il numero di visite per ogni cestino ed evitare che durante un turno vengano svuotati/visitati anche i cestini non pieni.

Per realizzare questo processo si suppone che:

1. La giornata sia suddivisa in più turni di lavoro.
2. Ci sia almeno una rilevazione sullo stato di ogni cestino tra un turno e quello successivo.

Dopo aver effettuato la fase di *preprocessing*, vengono estratti tutti i cestini dalla cui ultima rilevazione si evince siano pieni oppure occultati. L'ultima rilevazione di ogni cestino è stata estratta utilizzando il campo "detected\_at" con data e ora più recenti.

Il clustering utilizzato è di tipo partizionale, esclusivo e completo, in quanto:

- Ciascun cestino estratto, ad ogni turno, viene assegnato e svuotato da una sola squadra/mezzo.
  - Tutti i cestini estratti devono essere assegnati ad una squadra.
-

La clusterizzazione utilizzata sui cestini da svuotare si basa sulla loro posizione geografica, identificata dagli attributi latitudine e longitudine. Prima di operare, si effettua una normalizzazione di queste variabili.

Considerando che ogni cluster comprende i cestini che un camion deve svuotare, si procede a determinare il numero minimo di quelli necessari per garantire che durante il turno venga effettuata una vuotatura completa. Il numero minimo di mezzi necessari viene calcolato dividendo il numero di *smart bin* estratti per 150, per rispettare la variabile di progetto.

L'algoritmo utilizzato è il K-Means, che segue una procedura iterativa: inizialmente vengono scelte K osservazioni come centroidi. Successivamente, ogni osservazione viene assegnata al cluster del centroide più vicino secondo la distanza euclidea. Dopodiché, ad ogni iterazione, viene ricalcolato il centroide e ogni osservazione viene riassegnata nuovamente al cluster più vicino. L'algoritmo termina quando converge e quindi tra un'iterazione e quella successiva i centroidi restano immutati.

Al termine di questa fase ci sarà un numero di clusters che varierà tra 1 e 20, ognuno dei quali conterrà i cestini che ogni mezzo dovrà visitare in un turno e la sede di partenza e arrivo dei camion (via Olgettina).

Questa operazione di clustering andrà dunque effettuata prima dell'inizio di ogni turno di lavoro, in modo tale da visitare solo quei cestini che effettivamente necessitano di essere svuotati.

Durante ciascuna giornata lavorativa, nel corso di ogni turno, vengono memorizzati tutti i cestini che sono stati estratti per la vuotatura. Questo procedimento è utile per far sì che, prima dell'inizio dell'*ultimo* turno di lavoro, si abbia l'elenco dei cestini che non sono mai stati svuotati

nel corso della giornata. Questa operazione è necessaria per soddisfare il vincolo contrattuale tale per cui ogni cestino deve essere vuotato almeno una volta al giorno.

---

## IL PERCORSO DI VISITA

---

Per determinare il tragitto che ogni mezzo dovrà effettuare nel corso del turno, viene inizialmente calcolata la distanza geografica tra tutte le possibili coppie di cestini che si trovano all'interno di ogni cluster. La distanza è calcolata in km, utilizzando le coordinate di ogni cestino.

L'algoritmo utilizzato fa riferimento al problema del *Travelling salesman problem*, il quale cerca di rispondere alla domanda: "Dato un insieme di città e la distanza tra ognuna di esse, qual è il percorso più corto per visitarle tutte e ritornare al punto di partenza?"

È possibile porsi questa domanda anche nel nostro caso, considerando i cestini come fossero le città e la distanza tra di essi quella calcolata in precedenza.

Per far ciò viene utilizzata la libreria R "TSP" la quale rende disponibili sette algoritmi euristici differenti per la risoluzione del problema del commesso viaggiatore.

Questi algoritmi ci permettono di ottenere un'approssimazione della soluzione ottimale, ma risolvendo il problema in un tempo accettabile.

Per ogni cluster vengono quindi testati tutti gli algoritmi ed estratta la soluzione che tra tutte è risultata essere migliore.

Al termine di questa fase si avrà a disposizione per ogni cluster l'ordine con cui i cestini dovranno essere visitati.

---

---

## CLUSTER VALIDATION

---

Per valutare la bontà dei risultati, si sono considerati solamente indicatori interni, in quanto la metodologia esterna, o supervisionata, ha lo scopo di utilizzare algoritmi di clustering su dataset la cui struttura è già nota. La validazione è stata effettuata prendendo in considerazione tutti i cestini, confrontando i risultati ottenuti utilizzando l'algoritmo K-means con dati normalizzati e non.

Sono stati scartati fin da subito, senza effettuare la validazione, algoritmi di clustering gerarchico e a grafo, in quanto il loro costo computazionale è elevato e non adatto all'idea di effettuare una clusterizzazione in tempo reale.

Gli indici interni sono utili a valutare quanto gli elementi all'interno di un cluster siano simili tra loro e quanto sono distanti gli elementi tra cluster differenti.

Sono stati utilizzati due indici interni differenti: il coefficiente di Silhouette e di Dunn.

Il coefficiente di Silhouette è uguale a:

$$s_i = \frac{b_i - a_i}{\max(b_i, a_i)} \in [-1, 1]$$

dove

$a_i$  è la distanza media dell' $i$ -esima osservazione e le altre osservazioni appartenenti allo stesso cluster.

$b_i$  rappresenta il valore minimo della media delle distanze tra la  $i$ -esima osservazione e le osservazioni appartenenti a clusters diversi da quello a cui appartiene l' $i$ -esima osservazione.

Il coefficiente di Dunn risulta essere il rapporto tra la minima distanza tra cluster diversi e la massima distanza tra elementi dello stesso cluster. Assume valori compresi nell'intervallo  $[0, +\infty)$ .

Entrambi i coefficienti devono essere massimizzati.

Come si può osservare dalla tabella sottostante, il metodo migliore è quello che prevede l'utilizzo delle variabili geografiche normalizzate, in quanto a parità del coefficiente di Dunn, risulta più elevato il coefficiente di Silhouette.

	DATI NORMALIZZATI	DATI NON NORMALIZZATI
DUNN	0.0075	0.0075
SILHOUETTE	0.3922	0.3806

---

## SIMULAZIONE

---

Per chiarire al meglio l'idea sviluppata alla base di questo progetto, si fornisce una spiegazione dettagliata dei passaggi da effettuare. Il procedimento è diviso in due parti.

La *parte 1*, da svolgere prima dell'inizio di tutti i turni, prevede i seguenti passaggi:

1. Ottenere il file aggiornato delle rilevazioni dei cestini.
2. Eseguire il workflow knime. Per fare questa operazione è possibile sfruttare l'esecuzione di un workflow in batch mode. Con questa modalità un workflow può essere eseguito più velocemente direttamente da linea di comando senza l'utilizzo diretto della piattaforma Knime.

Attraverso l'esecuzione del comando riportato in [Appendice A](#) viene eseguito il workflow relativo alla *parte 1*, che

permette di ottenere un file contenente l'elenco ordinato di tutti i cestini da visitare suddivisi in base al relativo cluster (cestiniDaVisitare.xlsx). Inoltre, tutti i cestini visitati vengono inseriti in un altro file (elencoCestiniVisitati.csv) che tornerà utile nello svolgimento della seconda parte.

3. Realizzare tramite la piattaforma Tableau, utilizzando il file "cestiniDaVisitare.xlsx", una visualizzazione della clusterizzazione ([Appendice C](#)) e del percorso che ogni camion deve seguire. ([Appendice D](#)).

La *parte 2* è da svolgere subito dopo l'esecuzione della *parte 1*, ma solo in concomitanza dell'ultimo turno lavorativo della giornata.

Tramite il comando in [Appendice B](#), si esegue il workflow che permette di ottenere una clusterizzazione di tutti i cestini che durante la giornata non sono mai stati svuotati. Anche per questa parte è stato creato un file con l'elenco ordinato dei cestini da vuotare che, come in precedenza, permette di visualizzare graficamente il percorso che i camion devono seguire.

---

## CRITICITA' E LIMITI

---

L'utilizzo di una soluzione che opera in tempo reale potrebbe riscontrare alcune criticità; un'eventuale malfunzionamento della rete causerebbe un problema di trasmissione di informazioni del cestino, generando un'errata valutazione dello stato di quest'ultimo.

Inoltre, come già accennato in precedenza, anche i vincoli temporali ricoprono un ruolo importante. Per questo è stato fatto uso di algoritmi di visita che garantiscono un'esecuzione in tempi rapidi, a discapito di una soluzione non sempre ottimale.

Infine, non potendo gestire la dimensione dei cluster che vengono generati ad ogni iterazione, potrebbero verificarsi situazioni in cui vengono assegnati più di 200 cestini ad una squadra; in tal caso la soluzione potrebbe essere quella di assegnare due mezzi per la vuotatura dei cestini appartenenti a quel cluster.

---

## CONCLUSIONI

---

L'obiettivo della cluster analysis sviluppata in questo lavoro è stato quello di proporre una soluzione in grado di rendere il servizio di vuotatura il più efficiente possibile, rispettando i vincoli contrattuali e aziendali imposti.

Un'ottimizzazione da questo punto vista, porta ad una riduzione dei chilometri che un camion deve percorrere giornalmente; questo significa meno costi aziendali per il carburante e, nel lungo periodo, anche meno guasti e manutenzione ai mezzi. Ciò porterebbe anche a benefici ambientali, riducendo l'emissione di CO<sub>2</sub> dei camion.

## APPENDICE A

```
Microsoft Windows [Versione 10.0.18362.592]
(c) 2019 Microsoft Corporation. Tutti i diritti sono riservati.

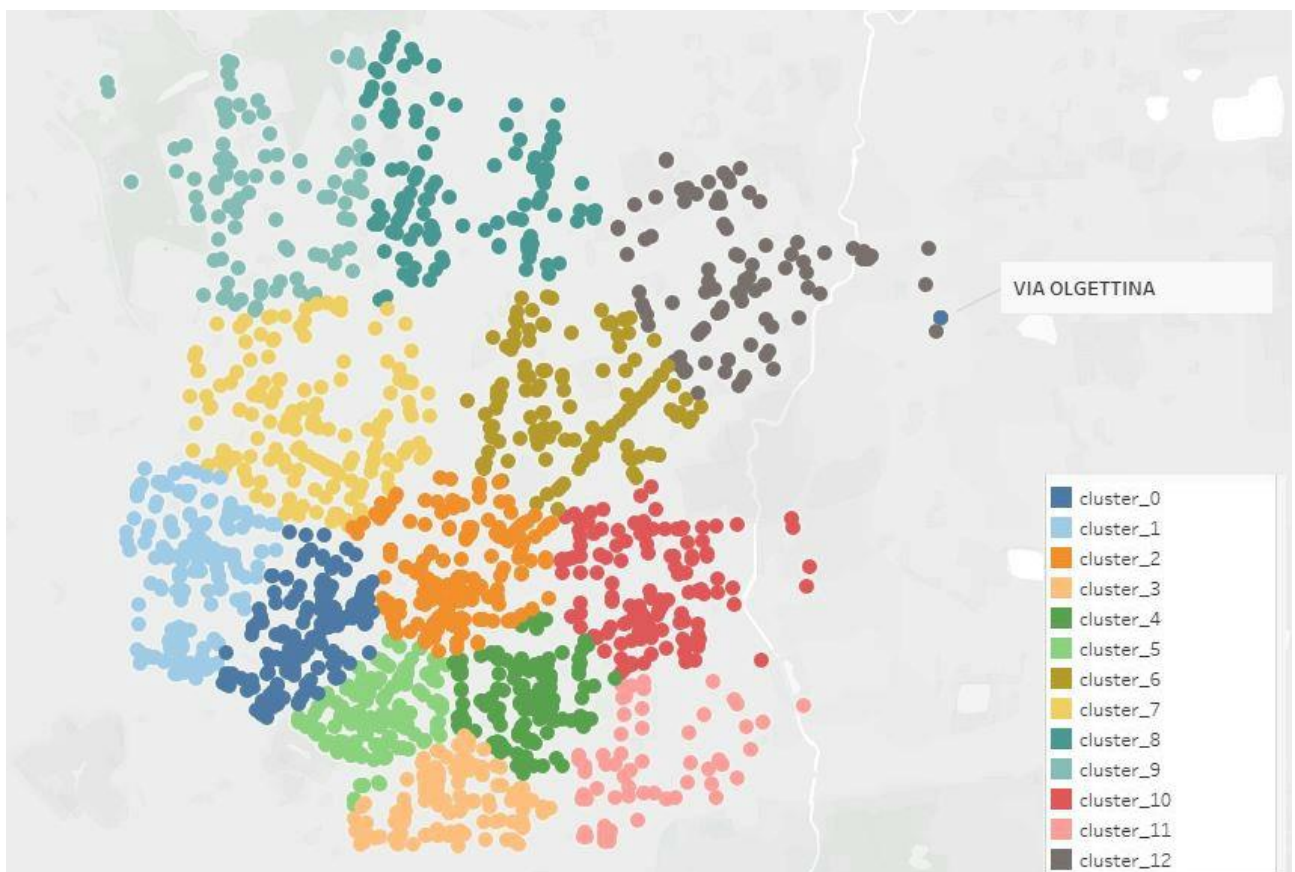
C:\Users\hp>
C:\Users\hp>"C:\Program Files\KNIME1\knime.exe" -nosplash -reset -consoleLog -application org.knime.product.KNIME_BATCH_
APPLICATION -workflowDir="C:\Users\hp\knime-workspace\Dynamicmerge\Dynamicmerge\parte1"
```

## APPENDICE B

```
Microsoft Windows [Versione 10.0.18362.592]
(c) 2019 Microsoft Corporation. Tutti i diritti sono riservati.

C:\Users\hp>"C:\Program Files\KNIME1\knime.exe" -nosplash -reset -consoleLog -application org.knime.product.KNIME_BATCH_
APPLICATION -workflowDir="C:\Users\hp\knime-workspace\Dynamicmerge\Dynamicmerge\parte2"
```

## APPENDICE C





APPENDICE D

