

UNIVERSITÀ DEGLI STUDI DI  
MILANO-BICOCCA

ADVANCED MACHINE LEARNING  
FINAL PROJECT

---

Previsione della durata delle corse dei taxi con  
approccio basato su deep learning

---

*Authors:*

Pietro Colombo - 793679 - p.colombo45@campus.unimib.it

Carlo Radice - 807159 - c.radice@campus.unimib.it

Marco Fagioli - 808176 - m.fagioli2@campus.unimib.it

February 4, 2020



## Abstract

In questa relazione viene mostrato un approccio per la predizione della durata delle corse dei taxi nell'area di New York. I dati utilizzati sono derivanti dall'integrazione di tre datasets, contenenti features rilevanti per questo scopo. Per la parte di modellazione del problema, sono state studiate diverse architetture di reti neurali, ottimizzate tramite un'apposita libreria, per ottenere la configurazione migliore.

## 1 Introduzione

Uno dei problemi che affrontano le compagnie di taxi e le compagnie NCC (Noleggio Con Conducente) che operano in ambienti urbani è la previsione del tempo di percorrenza di una corsa lungo un dato tragitto.

Questo topic è d'interesse in quanto si può migliorare un servizio offrendo ai clienti una stima accurata del tempo necessario per il tragitto desiderato. La previsione può essere usata nel momento stesso in cui viene usufruito il servizio oppure anche tramite una applicazione nel momento della prenotazione della corsa.

Di conseguenza, lo scopo di questo progetto consiste nell'effettuare una predizione del tempo di percorrenza da un punto A ad un punto B sulla base degli attributi nativi del dataset messo a disposizione, con l'integrazione di features provenienti da altri dataset.

L'approccio utilizzato si è concentrato inizialmente nel realizzare una analisi esplorativa del dataset per poter conoscere e capire in dettaglio la sua struttura. Si è quindi integrato il dataset con altri due che contengono rispettivamente le condizioni meteo, del periodo delle rilevazioni, e i percorsi, ricavati da Open Street Map. Il dataset finale così ottenuto viene diviso in train e test, utilizzati successivamente dal modello di previsione basato su deep neural networks.

## 2 Datasets

### 2.1 Descrizione del dataset

Il dataset[1] grezzo utilizzato per questo progetto contiene i record delle corse dei tassisti della città di New York del 2016. Ogni riga rappresenta una corsa

di un taxi. Il dataset è formato da 1.458.644 tuple definite da 11 attributi, qui elencati:

- **id**: identificatore univoco di ogni corsa;
- **vendor\_id**: codice che identifica la compagnia di taxi;
- **pickup\_datetime**: data e ora in cui il tassametro è stato attivato;
- **dropoff\_datetime**: data e ora in cui il tassametro è stato disattivato;
- **passenger\_count**: numero di passeggeri nel veicolo (valore inserito dal conducente);
- **pickup\_longitude**: longitudine alla quale è stato attivato il tassametro;
- **pickup\_latitude**: latitudine alla quale è stato attivato il tassametro;
- **dropoff\_longitude**: longitudine alla quale è stato disattivato il tassametro;
- **dropoff\_latitude**: latitudine alla quale è stato disattivato il tassametro;
- **store\_and\_fwd\_flag**: questo flag indica se il record della corsa è stato conservato nella memoria del veicolo prima di essere inviato al server perché il veicolo era privo di connessione. I valori sono: Y = memorizzato e inviato; N = non salvato e inviato;
- **trip\_duration**: durata della corsa in secondi.

## 2.2 Analisi esplorativa

### 2.2.1 Valori nulli e consistenza dei dati

La prima operazione eseguita è stata una analisi del dataset per vedere la presenza o meno di valori nulli. Successivamente sono stati effettuati vari test per verificare la consistenza dei dati in input sui domini di riferimento per i diversi attributi:

- viene controllato se *pickup\_datetime* è sempre antecedente alla data *dropoff\_datetime*;

- viene controllato se  $trip\_duration$  è sempre maggiore di zero.

Da questi controlli preliminari si evince che il dataset non presenta al suo interno né valori nulli né inconsistenti.

### 2.2.2 Visualizzazione delle features

Le figure 1 e 2 rappresentano una visualizzazione geografica della posizione dei punti di partenza e arrivo all'intero dataset. Questi punti, nello specifico 1000 coppie di coordinate, sono stati scelti ad intervalli regolari all'interno del dataset come sottoinsieme campionario.



Figure 1: sottoinsieme delle coord. di partenza dei viaggi



Figure 2: sottoinsieme delle coord. di arrivo dei viaggi

### 2.2.3 Ricerca e rimozione delle coordinate outlier

Da una prima analisi emerge subito che alcune coordinate hanno valori di latitudine e longitudine non conformi ai valori di New York. Per poter trattare questi dati è stata creata una soglia che definisce quali siano le coordinate da eliminare. Si considerano come posizioni errate quelle in cui i valori sono:

- maggiori della media delle coordinate (lat/long) sommata a cinque volte la deviazione standard;
- minori della media delle coordinate (lat/long) meno cinque volte la deviazione standard.

I record che hanno le coordinate di questo tipo (1035 tuple) vengono rimossi dal dataset poiché si considerano come errori di localizzazione del veicolo. Le figure 3 e 4 mostrano la posizione dei punti outlier, esterni a New York.



Figure 3: outlier delle coordinate di partenza dei viaggi



Figure 4: outlier delle coordinate di arrivo dei viaggi

#### 2.2.4 Ricerca e rimozione delle corse outlier

Dopo aver effettuato una analisi delle coordinate è stato deciso di controllare anche i valori presenti per la durata delle corse. La figura 5 rappresenta un istogramma logaritmico sull'asse y, in cui si può vedere molto bene come la maggior parte delle corse siano di breve durata. Tuttavia, per come viene effettuato il binning con intervalli di egual ampiezza, si evidenzia la presenza di alcune corse di lunga durata che si estendono fino a 3.526.282 secondi (oltre 979 ore!). La figura 6 rappresenta le stesse corse tramite un istogramma logaritmico sull'asse x. In questo caso si nota meglio la distribuzione gaussiana della durata delle corse. Studiata, quindi, la distribuzione in base al tempo si è deciso di rimuovere tutte le 2068 corse con durata superiore alle 5 ore (18 000 secondi).

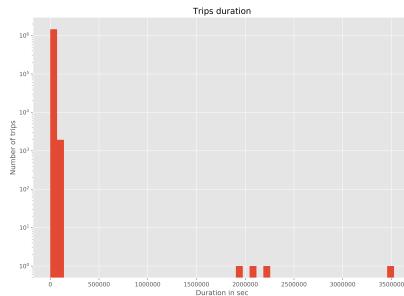


Figure 5: istogramma logaritmico sull'asse y, rappresentante la durata delle corse

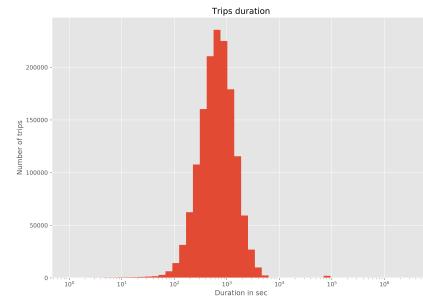


Figure 6: istogramma logaritmico sull'asse x, rappresentante la durata delle corse

## 2.2.5 Grafici esplicativi

L'analisi esplorativa ha portato alla creazione di diversi grafici che permettessero di visualizzare graficamente ed intuitivamente la correlazione tra diverse componenti in analisi. Di seguito ne vengono riportati alcuni, quelli aventi maggior contenuto informativo e interesse.

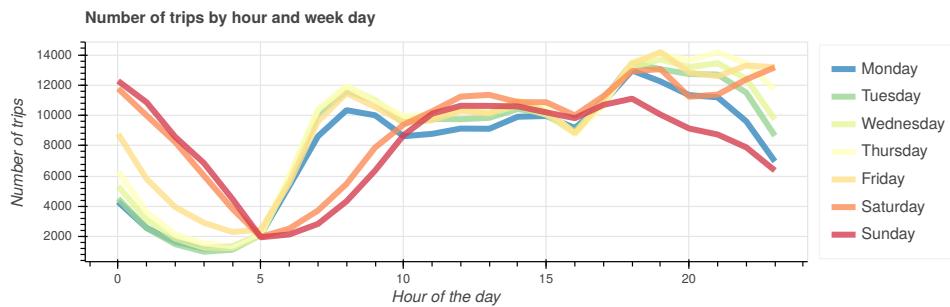


Figure 7: numero delle corse per ora e per giorno della settimana

La figura 7 mostra come l'utilizzo dei taxi durante il week-end sia maggiore nelle ore notturne mentre durante la settimana sia elevato alla mattina presto. Il dataset è quindi molto coerente con la situazione del mondo reale.

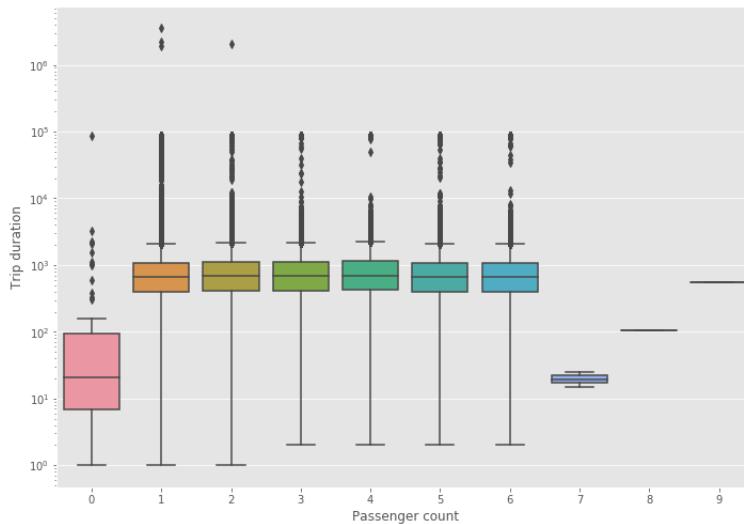


Figure 8: numero di passeggeri rispetto alla durata delle corse

La figura 8 mostra la distribuzione del numero di passeggeri nelle corse in relazione alla loro durata. Si può notare come i passeggeri da soli effettuino corse corte, molto probabilmente spostamenti di lavoro nella caotica metropoli, mentre, se il tempo sale, il numero di passeggeri aumenta.

## 2.3 Introduzione nuove feature

### 2.3.1 Clustering

Si è deciso di introdurre una nuova feature che potesse generalizzare i punti di pickup e dropoff etichettando la loro zona di riferimento. Le coordinate dei punti di partenza e di arrivo sono stati unite in un dataset su cui è stato applicato l'algoritmo K-means. Il modello è stato successivamente utilizzato per predire il cluster a cui appartengono i punti di pickup e dropoff per ogni record del dataset.

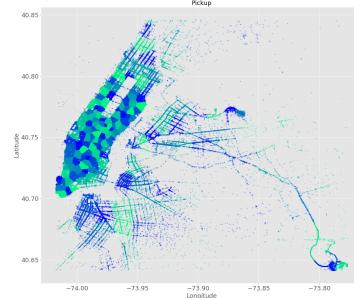


Figure 9: Cluster dei punti di partenza e arrivo

### 2.3.2 Aeroporti

Dai dati si può notare che è presente un elevato numero di viaggi che partono o arrivano in uno dei due aeroporti di NYC: JFK e LaGuardia. Vengono quindi aggiunte delle nuove colonne booleane che indicano se i punti di partenza e di arrivo sono nell'intorno di 2 km da queste località. Poiché gli aeroporti non sono situati nel centro città, è ragionevole assumere che le nuove features aggiunte nel dataset siano un buon indicatore per i viaggi di lunga percorrenza e durata.

## 2.4 Integrazione con altri datasets

Al fine di migliorare la successiva fase di apprendimento dei parametri si è considerato di unire al dataset originale altre due fonti di dati aggiungendo in questo modo diverse nuove features. Il primo[2] dataset contiene le condizioni atmosferiche di New York del periodo di acquisizione dei record, il secondo[3] mostra l'effettiva lunghezza dei tragitti tramite l'utilizzo di Open Street Map<sup>1</sup>.

### 2.4.1 Unione con “Weather data in New York city”

L'integrazione viene effettuata tramite un join sulla colonna *date*, ottenendo in output lo stesso numero di righe contenute nel dataset originale. Successivamente, vengono convertite le misure della temperatura da gradi Fahrenheit a gradi Celsius, le misure di profondità da pollici a centimetri e il valore T presente nelle colonne *precipitation*, *snow\_fall*, *snow\_depth* viene sostituito da valori numerici. Infine, vengono rinominate alcune colonne per renderle coerenti al tipo di nomenclatura utilizzato precedentemente.

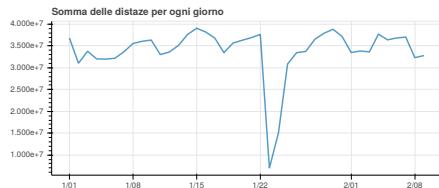


Figure 10: somma delle distanze per ogni giorno

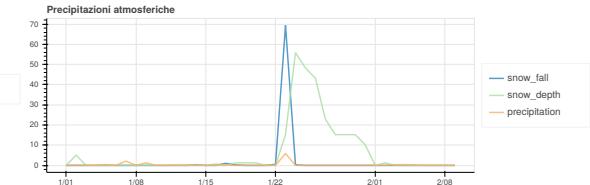


Figure 11: Precipitazioni atmosferiche

Confrontando il grafico in figura 10 che rappresenta le distanze totali nell'arco delle giornate, con il grafico in figura 11, che rappresenta le precipitazioni meteorologiche, si evidenzia una correlazione tra le condizioni meteo e le corse in taxi.

### 2.4.2 Unione con “New York City Taxi with OSRM”

L'integrazione di questi dati con il dataset creato al passo precedente è stata effettuata tramite un join sulla colonna *id*, ottenendo lo stesso numero di righe presenti allo step precedente.

<sup>1</sup><https://www.openstreetmap.org/about>

## 2.5 Pulizia del dataset

Dopo aver effettuato la seconda integrazione il dataset presenta un elevato numero di attributi di cui molti inutilizzati. Si è quindi deciso di effettuare una rimozione delle features derivanti dalle integrazioni ma che non aggiungono conoscenza. Successivamente, i dati relativi al dropoff vengono eliminati in quanto nella parte di testing non sono presenti.

## 3 Approccio metodologico

In questa sezione viene descritto l'approccio sviluppato per ottenere il modello della rete neurale in seguito utilizzato per la predizione e per la sottomissione dei risultati.

### 3.1 Preparazione dell'input

Per la parte di training le feature selezionate sono:

*total\_distance, total\_travel\_time, number\_of\_steps, maximum\_temperature, minimum\_temperature, average\_temperature, precipitation, snow\_fall, snow\_depth, vendor\_id, passenger\_count, store\_and\_fwd\_flag, pickup\_cluster, dropoff\_cluster, JFK\_start, JFK\_end, guardia\_start, guardia\_end, pickup\_hour, pickup\_minute, pickup\_day\_week.*

Per poter valutare e migliorare le prestazioni del modello, il dataset è stato processato in modo da essere suddiviso in due sottoinsiemi disgiunti. Questa parte viene descritta in modo esaustivo nel capitolo successivo.

Inoltre, poiché è stata scelta la libreria Keras<sup>2</sup>, l'input necessita di una normalizzazione preliminare.

### 3.2 Definizione del modello

#### 3.2.1 Modelli iniziali

Inizialmente, in modo molto naïve si è deciso di provare vari modelli scegliendo dapprima i parametri in modo pseudo casuale e cercando, ad ogni iterazione successiva, di migliorare l'architettura della rete.

Questa opzione, benchè desse risultati abbastanza buoni nel breve periodo,

---

<sup>2</sup><https://keras.io/>

è stata ritenuta non essere la soluzione ottimale. Si è deciso, quindi, di utilizzare un tool di tuning dei parametri, quale Hyperas<sup>3</sup>.

### 3.2.2 Tuning dei parametri con Hyperas

Si è pensato di creare diverse architetture di rete i cui parametri sono stati fatti variare per poter ottenere il modello di rete migliore. Ci si è concentrati principalmente nel trovare il numero ottimale di neuroni per i singoli layer, oltre alla dimensione del batch, il numero di epoche e all'ottimizzatore.

### 3.2.3 Modello finale

Grazie alle prove effettuate con Hyperas si è arrivati a progettare una architettura che contiene 4 layer densi composti da 128, 32, 64, e 1 neuroni rispettivamente per un totale di 9121 parametri allenabili. Inoltre l'ottimizzatore, tra i differenti candidati scelti, che ha avuto i migliori risultati è *adam*.

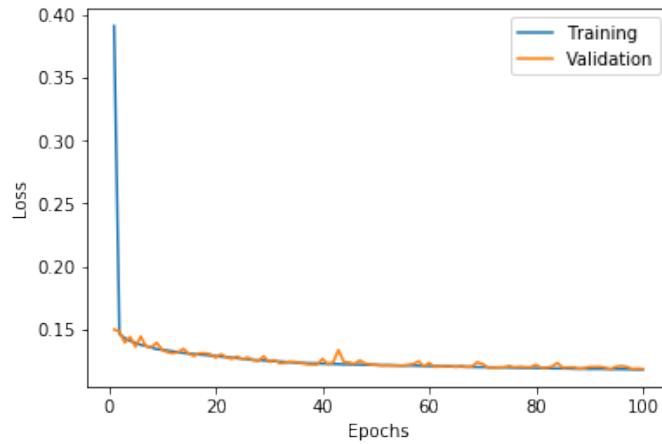


Figure 12: Andamento della funzione loss

La figura 12 mostra il comportamento della funzione *loss* nell'addestramento del modello, sia per la parte di training che quella di validation, durante tutte le iterazioni delle epoche.

---

<sup>3</sup><https://github.com/maxpumperla/hyperas>

## 4 Risultati e Valutazioni

### 4.1 Selezione della metrica

La metrica di riferimento utilizzata per la valutazione dei risultati ottenuti dal modello è la stessa usata per la competizione su Kaggle, ovvero la RMSLE (Root Mean Squared Logarithmic Error) (Formula 1). Questa metrica è resistente all'effetto degli outliers.

La sua formula è la seguente:

$$\epsilon = \sqrt{\frac{1}{n} \sum_{i=1}^n (\log(p_i + 1) - \log(a_i + 1))^2} \quad (1)$$

### 4.2 Validazione

Per poter misurare le performance del modello si è scelto di utilizzare il validation set. Il dataset in input viene perciò diviso in due parti: la prima utilizzata per istruire il modello, la rimanente per effettuare la previsione. Di quest'ultima, infatti, si è a conoscenza del reale valore del tempo di percorrenza che può quindi essere usato come riferimento per il valore ottenuto dalla rete.

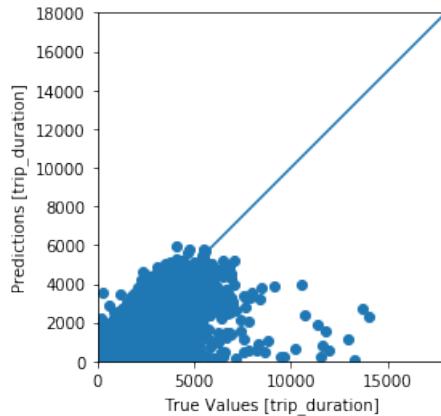


Figure 13: Interpolazione tra i valori reali dei tempi e quelli predetti

La figura 13 mostra la relazione esistente tra i valori reali e quelli predetti per il tempo di percorrenza. Un'previsione corretta, o che si avvicini molto

al vero valore, è identificata nel grafico da un coppia di coordinate molto simili tra di loro. Un buon andamento in figura è perciò rappresentato da una grande concentrazione di punti lungo la retta  $y = x$ .

Le performance ottenute dalla rete sul validation set secondo la metrica RMSLE è di *0.3430*

### 4.3 Previsione sul testset e feedback Kaggle

La rete addestrata è stata infine utilizzata sull'insieme di test per effettuare le previsioni. In questo caso non è presente il valore reale del tempo di percorrenza e pertanto non è possibile calcolare direttamente le performance del modello.

Per avere una valutazione dei valori ottenuti dalla rete viene creato l'apposito file con i risultati e caricato sulla pagina Kaggle della competizione.

La valutazione ottenuta, secondo la metrica descritta precedentemente, è di: *0.43055* la quale ci permette di collocarci nella prima metà della classifica, il miglior team ha ottenuto uno score di *0.28976*.

## 5 Discussione

L'obiettivo del progetto è stato raggiunto, la predizione del tempo stimato per un viaggio da un dato punto A ad un punto B viene effettuata tramite il modello proposto. Le performance della rete sono buone e soddisfacenti, garantendo una stima accettabile sul tempo di percorrenza.

Un aspetto negativo è sicuramente la bassa numerosità dei viaggi con durata superiore a 5 ore, i viaggi più lunghi, sui quali la previsione risulta essere meno precisa. Un possibile miglioramento, come sviluppo futuro, è lavorare con un nuovo dataset, o comunque una aggiunta di nuovi dati, che possa contenere una maggior numerosità per questi elementi.

Un'ulteriore sviluppo futuro potrebbe consistere nell'integrazione di dati in real time sulle condizioni della strada, come ad esempio lavori e strade chiuse, così da poter effettuare delle previsioni più accurate in base al traffico e alle condizioni stradali.

## 6 Conclusioni

In questo documento, viene presentato un approccio su reti neurali per la previsione della durata di un percorso in taxi nella città di New York, che implica innumerevoli variabili per la sua previsione.

L'impiego delle reti neurali ha permesso di completare il task di previsione garantendo una buona affidabilità dei risultati. Possibili sviluppi futuri potrebbero tenere in considerazione l'introduzione di ulteriori dati per arricchire l'insieme dei viaggi lunghi (che comprendono i viaggi con una durata maggiore di 5 ore), oltre all'aggiunta di nuovi esempi e attributi. Questo nuovo dataset, contenente informazioni più realistiche, si potrebbe utilizzare per ottenere performance migliori.

Il lavoro svolto per questo progetto ha raggiunto l'obiettivo prefissato, potendo così essere utilizzato come parte integrante per migliorare il servizio offerto dalle compagnie di taxi e da quelle NCC.

## References

- [1] Kaggle, “New york city taxi trip duration,” 2017, dataset ottenuto da kaggle, <https://www.kaggle.com/c/nyc-taxi-trip-duration>.
- [2] M. Waegemakers, “Weather data in new york city,” 2016, dataset ottenuto da kaggle, <https://www.kaggle.com/mathijs/weather-data-in-new-york-city-2016>.
- [3] O. Leo, “New york city taxi with osrm,” 2017, dataset ottenuto da kaggle, <https://www.kaggle.com/oscarleo/new-york-city-taxi-with-osrm>.