STREAMING DATA MANAGMENT AND TIME SERIES ANALSYS PROJECT

Biagio Spiezia, 920172

January 17, 2025

1 Introduction

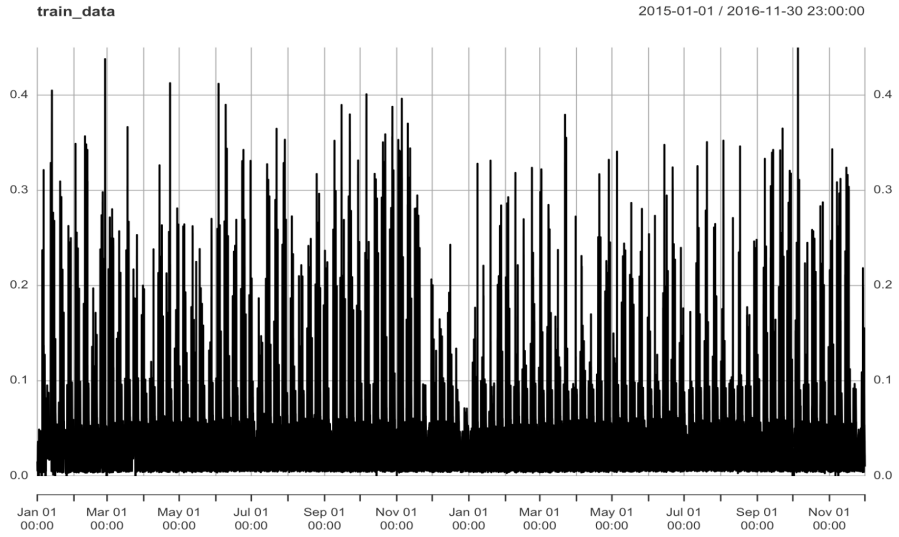
Scopo di questo progetto è la previsione di una serie storica oraria che rappresenti la quantità di traffico su un'autostrada situata in una città degli Stati Uniti. La serie storica analizzata copre il periodo che va dalla mezzanotte del 1° gennaio 2015 al 31 dicembre 2016, con l'eccezione dell'ultimo mese, per il quale i valori della serie non sono disponibili.

Per affrontare il problema della previsione sono stati utilizzati diversi approcci: modelli statistici come ARIMA (Autoregressive Integrated Moving Average) e UCM (Unobserved Components Model) ed un algoritmo di machine learning, il K-Nearest Neighbors (KNN). L'obiettivo è quello di confrontare le prestazioni di questi modelli per individuare il metodo più efficace che predica i dati mancanti. Per valutare le prestazioni di ciascun modello è stata utilizzata la metrica MAE (Mean Absolute Error), che permette di misurare l'accuratezza delle previsioni confrontandole con i valori reali. Il *Mean Absolute Error* (MAE) è una misura utilizzata per valutare la precisione di un modello di previsione. Esprime la media degli errori assoluti tra i valori osservati y_i e i valori previsti $\hat{y}_i$. È definito dalla formula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

dove n è il numero totale di osservazioni.

2 Data Exploration e Splitting dei dati



Prima di poter costruire un modello di previsione accurato, si è scelto di verificare le caratteristiche della serie storica. Nel progetto, si è proceduto a valutare la stazionarietà della serie utilizzando

l'Augmented Dickey-Fuller (ADF) test, calcolato considerando la serie storica nel suo complesso e suddividendola in intervalli giornalieri (24 ore), settimanali (24*7 ore) e mensili (24*30 ore). La statistica test ottenuta, risultata significativamente inferiore ai valori critici, consente di rifiutare l'ipotesi nulla secondo cui la serie contiene una radice unitaria. Pertanto, si può concludere che la serie è stazionaria e non è necessario applicare trasformazioni o calcolare differenze sui dati.

L'analisi dell'Autocorrelation Function (ACF) per i diversi intervalli temporali (giornaliero, settimanale e mensile) ha evidenziato un chiaro pattern: i valori di correlazione sono molto alti nei primi lag, tendono a diminuire progressivamente e poi risalgono.

Per addestrare i modelli,la serie storica è stata suddivisa in un train set e un validation set. L'83% delle osservazioni (circa 580 giorni, 83 settimane, 20 mesi) è stato assegnato al train set, mentre il restante 17% (circa 120 giorni, 16 settimane, 4 mesi) è stato utilizzato come validation set.

3 Modelli ARIMA

3.1 Primo Approccio: Auto-ARIMA

Come primo tentativo di individuare il miglior modello, è stata utilizzata la funzione auto.arima per stimare automaticamente il miglior modello. I parametri p e q sono stati inizialmente settati basandosi sull'analisi dell'ACF e PACF, analisi che ha permesso di formulare ipotesi sui valori più appropriati.La funzione non ha suggerito alcuna stagionalità evidente.

3.2 Approccio Manuale: Grid Search

Successivamente, è stato implementato un approccio manuale tramite Grid Search per individuare i migliori parametri del modello, includendo una stagionalità giornaliera (periodo 24) e considerando anche l'inclusione di sinusoidi per catturare le componenti periodiche.

Il loop di ricerca ha identificato i due modelli migliori in :

- Senza sinusoidi: SARIMA(1,0,0) (2,0,1).
- Con sinusoidi: SARIMA(3,0,0) (1,0,1).

3.3 Regressori Esterni: Festività

A ciascuno dei modelli scelti sono stati aggiunti regressori esterni che rappresentano le festività negli Stati Uniti. Inoltre, è stato considerato un modello basato su Prophet, che include solo le festività come regressori e non utilizza sinusoidi.

3.4 Estensione a Stagionalità Settimanale

I modelli selezionati sono stati testati anche con una periodicità settimanale (168 ore), per valutare se la stagionalità, su scala settimanale, potesse migliorare le prestazioni del modello.

3.5 Separazione in 24 serie oraria

Perseguendo l'obiettivo di catturare al meglio la stagionalità oraria, è stato implementato un approccio che divide la serie storica in sottoserie orarie. Ogni sottoserie rappresenta i valori di una specifica ora del giorno. La procedura utilizzata è stata la seguente:

- Divisione della Serie: La serie storica è stata suddivisa in 24 sottoserie, ciascuna contenente i dati di una specifica ora.
- Modellazione: Per ogni sottoserie, è stato applicato un modello ARIMA con differenza stagionale (lag 24), attraverso la funzione auto.arima
- Previsione e Aggregazione: Le previsioni di ciascun modello sono state combinate per generare la previsione complessiva della serie storica.

3.6 Separazione serie in settimane

Un ulteriore approccio è stato quello di modellare la stagionalità settimanale (168 ore), suddividendo la serie storica in 168 sottoserie.

- Creazione delle Sottoserie: La serie è stata suddivisa in sottoserie settimanali.
- Modellazione: È stato applicato un modello ARIMA per ciascuna delle 168 sottoserie, senza considerare la stagionalità interna.
- Previsione e Aggregazione: Le previsioni per ciascuna sottoserie sono state assemblate per ottenere la previsione complessiva

3.7 Analisi dei risultati

I risultati raggiunti dai modelli, calcolato utilizzando il validation set, sono riportati nella successiva tabella:

Modello	**MAE**
Baseline (MAE)	0.029
SARIMA(1,0,0) $(2,0,1)_{24} senzasinusoidi$	0.017
SARIMA(3,0,0) $(1,0,1)_{24} consinusoidi$	0.024
SARIMA(1,0,0) $(2,0,1)_{24} confestività$	0.017
SARIMA (3,0,0) $(1,0,1)_{24} confestività - sinusoidi$	0.024
SARIMA (1,0,0) $(2,0,1)_{168} confestività$	0.014
SARIMA (3,0,0) $(1,0,1)_{168} confestività - sinusoidi$	0.013
ARIMA separato per ore	0.046
ARIMA separato per settimane	0.037

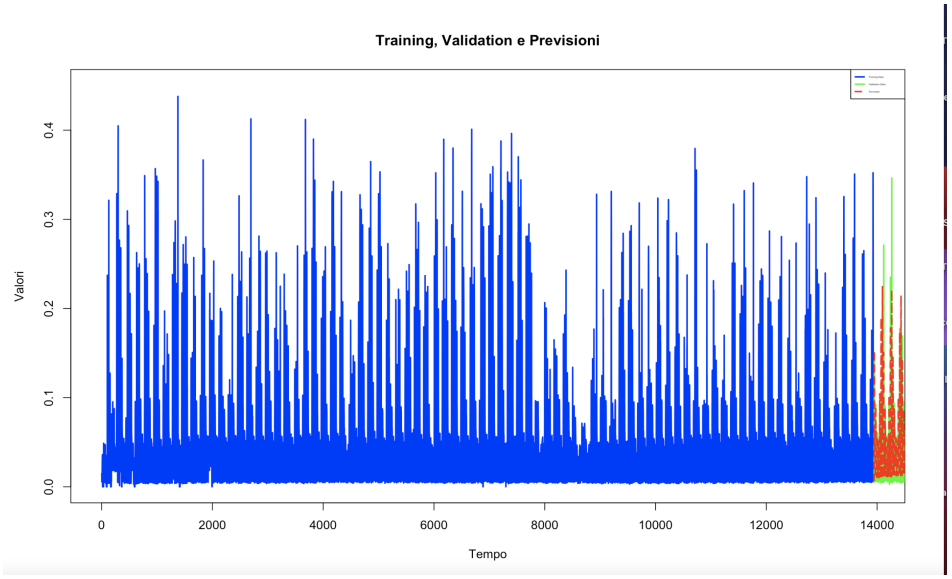
Table 1: Risultati del MAE per i diversi modelli

In fase di selezione tra i diversi modelli SARIMA proposti, ai residui di ciascun modello è stato applicato il test di Ljung-Box per verificare se essi possano essere considerati White Noise. Questo test consente di valutare se i residui sono privi di correlazione seriale, un requisito fondamentale per un modello ben specificato.

Per il modello SARIMA (1,0,0)(2,0,1), il test di Ljung-Box ha restituito un p-value molto basso, indicando che i residui non mostrano una correlazione significativa e possono essere considerati white noise. Questo suggerisce che il modello è adeguato per descrivere la serie storica e cattura efficacemente le sue componenti strutturali.

Al contrario, per il modello SARIMA (3,0,0)(1,0,1), il test di Ljung-Box ha restituito un p-value elevato, portando ad accettare l'ipotesi nulla, secondo la quale i residui non sono white noise. Questo implica che il modello non è in grado di rappresentare completamente la struttura della serie storica, lasciando delle correlazioni non spiegate nei residui.

Sulla base di queste considerazioni, si è preferito il modello SARIMA (1,0,0)(2,0,1) -con festività - perchè soddisfa il requisito di produzione di residui assimilabili a white noise, garantendo così una maggiore affidabilità nelle previsioni.



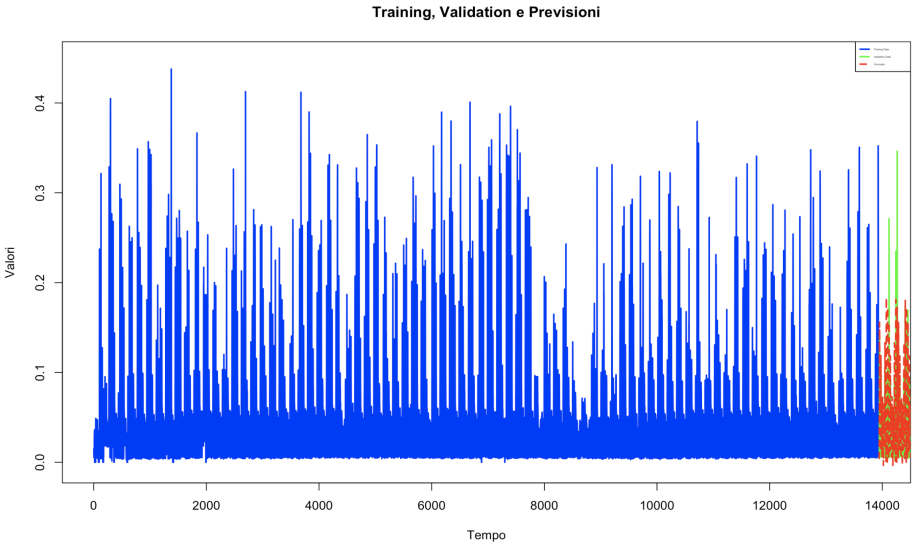
Blu=Train set,Verde=Validation set,Rosso=Previsioni

4 Modelli UCM

Il modello Unobserved Components Model (UCM) è utilizzato per catturare le componenti latenti della serie storica, come trend, stagionalità e effetti regressori.

Per modellare il trend, è stata utilizzata una struttura a due livelli di differenziazione (SSMtrend(2)), in grado di rappresentare sia un trend lineare sia variazioni più complesse nel tempo. I parametri di variazione (Q) sono stati inizializzati in funzione della varianza della serie storica.

La stagionalità giornaliera è stata modellata con una struttura "dummy" (SSMseasonal(24,"dummy")), che cattura le variazioni orarie ripetute giornalmente. Per la stagionalità settimanale, è stata adottata una struttura trigonometrica SSMseasonal(168,"trig",harmonics=1:16), in grado di rappresentare pattern settimanali complessi. Nel secondo modello sono stati aggiunti regressori che rappresentano la presenza di festività negli Stati Uniti, basandosi sull'ipotesi che il traffico, durante le festività, segua pattern differenti rispetto ai giorni normali. Tuttavia l'aggiunta delle festività non ha migliorato le previsioni del nostro modello ed è stato preferito il modello precedente.

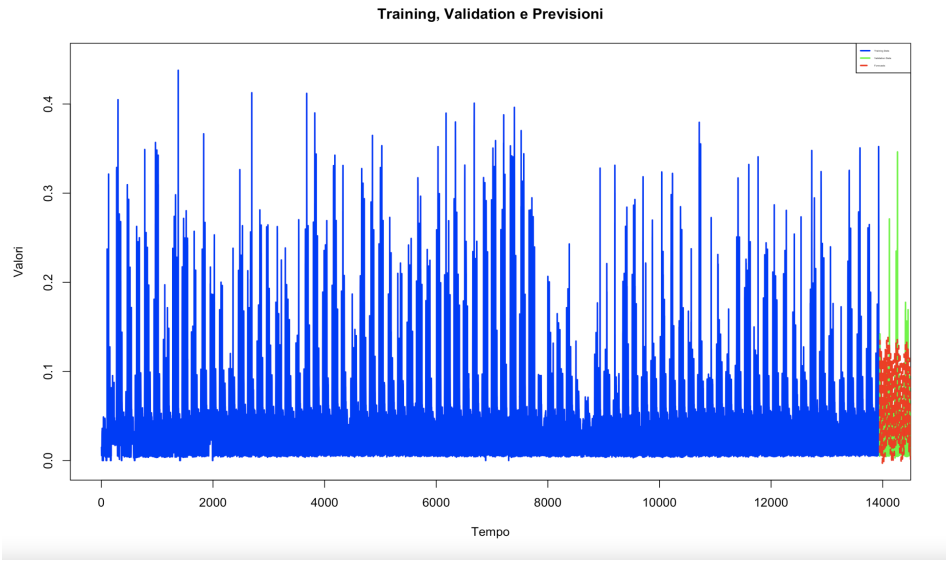


Blu=Train-set,Verde=Validation set,Rosso=Previsioni

5 Algoritmo Machine Learning: Il KNN

L'algoritmo K-Nearest Neighbors (KNN) è un metodo di machine learning supervisionato, utilizzato per ragioni di di classificazione e regressione. Si basa sull'idea che un'osservazione possa essere prevista considerando i k punti più vicini nel dataset di training. La vicinanza è misurata da metriche come la distanza euclidea. In regressione, la previsione è calcolata come la media (o un'altra aggregazione) dei valori target dei k vicini, mentre nella classificazione si utilizza la classe più frequente tra i vicini.

Per il modello KNN (K-Nearest Neighbors) si è proceduto con la creazione di un dataset laggato, per sfruttare le osservazioni precedenti della serie temporale. Questo approccio consente di modellare la dipendenza temporale dei dati. Per trovare il valore ottimale di k (numero di vicini), è stata utilizzata una validazione incrociata con 5 fold. Il parametro k è stato ricercato in un intervallo da 1 a 10, utilizzando l'opzione train control che permette di inserire una griglia per la ricerca del parametro ottimale. Il miglior valore di k trovato è stato utilizzato per addestrare il modello finale. Nonostante l'ottimizzazione, il modello KNN ha mostrato prestazioni molto inferiori rispetto agli altri modelli utilizzati (ARIMA e UCM). In particolare: il valore del MAE ha raggiunto livelli superiori agli altri, pari a circa 392, rendendo il modello meno accurato per questo specifico problema. Questo comportamento può essere attribuito alla natura complessa della serie temporale e alla limitata capacità del KNN di catturare le dinamiche temporali, e stagionali, presenti nei dati.



Blu=Train set,Verde=Validation set,Rosso=Previsioni

6 Notazione finale

Ciascun modello scelto viene poi riaddestrato utilizzando l'intero dataset disponibile (train + validation) per sfruttare al meglio tutte le informazioni storiche. Successivamente, verranno generate e salvate le previsioni per i successivi 744 giorni.