ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE
CORSO DI LAUREA MAGISTRALE IN INGEGNERIA GESTIONALE

Tesi di METODI E MODELLI DI DATA ANALYTICS M

IL FORECASTING OPERATIVO NEL SETTORE BANCARIO

UN'APPLICAZIONE DI MACHINE LEARNING PER LA GESTIONE DEI CONSULENTI FINANZIARI: IL CASO FACILE.IT

CANDIDATO

Thanh Tri Riccardo Tran

RELATORE

Chiar.mo Prof. Ing. Andrea Borghesi

Anno Accademico 2022-2023

Sessione IV

ABSTRACT

In un'era caratterizzata dall'avanzamento tecnologico e dalla digitalizzazione dei servizi, le aziende operanti nel settore dei servizi finanziari online affrontano la sfida di gestire efficacemente un volume crescente di leads.

Questa tesi esplora l'impiego di modelli predittivi avanzati per prevedere il volume di leads generati dai siti web di Facile.it e Mutui.it, con l'obiettivo di ottimizzare l'allocazione delle risorse nel call center aziendale.

Attraverso un'analisi comparativa, vengono valutati due modelli principali: Prophet e SARIMAX, sia individualmente sia in combinazione con una suddivisione delle leads per fonte. Il lavoro dimostra che il modello SARIMAX, applicato con una suddivisione per fonte delle leads, supera in accuratezza il modello Prophet e il SARIMAX senza suddivisione, migliorando significativamente la previsione dei volumi di leads.

L'analisi evidenzia inoltre l'importanza di considerare le caratteristiche specifiche delle diverse fonti di leads e conferma che l'integrazione delle leads da mutui.it non compromette l'accuratezza dei modelli.

I risultati offrono spunti rilevanti per la pianificazione delle risorse umane nel call center, sottolineando l'efficacia degli approcci predittivi nel settore dei servizi finanziari online. La tesi propone infine possibili sviluppi futuri, tra cui l'esplorazione di tecniche di machine learning avanzate e l'implementazione di dashboard di monitoraggio predittive per il management, orientando la ricerca verso un utilizzo ancora più efficace dei dati nell'ottimizzazione delle operazioni aziendali.

INDICE

INTRODUZIONE

Nell'era moderna, l'analisi dei dati si è affermata come un pilastro fondamentale per la comprensione e la gestione di fenomeni complessi in vari settori, inclusi quelli finanziari e bancari. L'avvento di tecniche avanzate di data analytics e machine learning ha ulteriormente ampliato le potenzialità di queste discipline, permettendo di elaborare previsioni accurate e di ottimizzare processi decisionali basati su volumi di dati sempre più ampi e diversificati.

La presente tesi si focalizza sull'applicazione di metodi e modelli di data analytics avanzati, con particolare attenzione al contesto bancario e di broking. Il candidato, Thanh Tri Riccardo Tran, sotto la supervisione del Chiar.mo Prof. Ing. Andrea Borghesi, ha intrapreso un percorso di ricerca volto all'individuazione di strategie ottimali per la previsione di leads settimanali e per il dimensionamento della capacità settimanale richiesta in un call center specializzato in servizi di mutuo.

Il problema principale affrontato riguarda la capacità di prevedere il volume di leads - potenziali clienti interessati a servizi di mutuo - basandosi su dati storici e variabili esogene. Questo obiettivo viene perseguito attraverso l'impiego di modelli predittivi quali Prophet e SARIMAX, valutati e comparati in termini di performance predittiva. L'analisi esplorativa dei dati e l'applicazione di queste tecniche avanzate consentono di identificare pattern e tendenze all'interno del flusso di leads, fornendo indicazioni preziose per il dimensionamento ottimale del personale di call center, in modo da gestire in maniera lean le richieste dei potenziali clienti.[1]

La ricerca si inserisce in un contesto di crescente interesse verso l'applicazione di metodologie di data science nel settore bancario e finanziario, dove la capacità di elaborare previsioni accurate si traduce in vantaggi competitivi significativi. Attraverso l'analisi di un dataset complesso, che include informazioni relative alle fonti dei leads, alle finalità dei mutui richiesti e ai dispositivi utilizzati dagli

[1] Laureani, A., Antony, J., & Douglas, A. (2010). Lean six sigma in a call centre: a case study. International Journal of Productivity and Performance Management, 59(8), 757-768.

utenti, il lavoro si propone di contribuire alla letteratura esistente, offrendo spunti di riflessione e possibili direzioni future per la ricerca in questo ambito.

Il primo capitolo analizza il contesto specifico del mercato bancario e dei servizi di broking online, focalizzandosi sulle dinamiche competitive e sulle strategie di marketing nel settore dei mutui. Si discute di come le banche e i broker si posizionano nel mercato, delle strategie adottate per attrarre clienti e della gestione delle leads generate attraverso i canali digitali. Il capitolo fornisce un quadro comprensivo del contesto operativo e delle sfide che le aziende devono affrontare, collegando le tecniche di analisi e previsione discusse nei capitoli precedenti alle strategie di business.

Il secondo capitolo offre una panoramica dettagliata degli strumenti e delle tecniche di analisi dati utilizzati nel progetto, come l'analisi della correlazione, il test di chi quadro, l'analisi ANOVA, e l'Exploratory Data Analysis (EDA). Questo capitolo è necessario per la comprensione dei modelli di Machine Learning utilizzati, poiché derivanti dalle teorie statistiche sviluppate nel ventesimo secolo. La discussione include anche l'applicazione di queste tecniche al dataset specifico della tesi, fornendo un collegamento diretto tra teoria e pratica.

Il terzo capitolo si concentra sull'applicazione di modelli di machine learning, in particolare il modello ARIMA e Prophet, per la previsione delle leads settimanali e la capacità settimanale richiesta nel settore dei mutui. Attraverso l'uso di questi modelli, il capitolo illustra come l'analisi predittiva possa supportare le decisioni strategiche e operative, migliorando l'allocazione delle risorse e l'efficienza dei servizi. Vengono esplorate le specifiche tecniche di ciascun modello, il processo di addestramento e i risultati ottenuti, evidenziando come questi strumenti possano essere impiegati per anticipare le esigenze del mercato.

Il quarto capitolo presenta un approccio pratico alla modellazione e previsione delle leads settimanali, combinando le tecniche di analisi dati e machine learning introdotte nei capitoli precedenti. Attraverso l'analisi di un caso studio, il capitolo illustra come i modelli di previsione possano essere

utilizzati per ottimizzare la gestione delle leads e il dimensionamento del personale nei call center. Si discute della raccolta e pulizia dei dati, dell'analisi esplorativa e dell'addestramento dei modelli, fino alla valutazione dei risultati e alla stima della capacità operativa necessaria.

Capitolo 1 - IL CONTESTO BANCARIO E DI BROKING

Il mutuo

Il mutuo rappresenta uno strumento finanziario di debito, attraverso il quale un mutuatario ottiene risorse finanziarie con l'impegno di restituire il capitale, oltre agli interessi concordati, entro un periodo definito. La parola "mutuo" deriva dal latino "mutuum", che indica un prestito di consumo, sottolineando la natura di scambio e restituzione che caratterizza questo strumento.

La storia dei mutui affonda le radici in tempi antichi, con le prime forme di prestito e ipoteca documentate in civiltà come quella mesopotamica, greca e romana. Nel corso dei secoli, la pratica del mutuo si è evoluta, adattandosi alle esigenze economiche e sociali delle diverse epoche, fino a diventare uno dei pilastri del sistema finanziario moderno, particolarmente nel settore immobiliare.[2]

Il funzionamento dei mutui si basa su principi fondamentali di prestito e rimborso. Al momento della stipula del mutuo, mutuatario e mutuante concordano su importo del prestito, tasso di interesse, piano di ammortamento e durata del mutuo. Il tasso di interesse può essere fisso, rimanendo costante per tutta la durata del mutuo, o variabile, adattandosi secondo specifici indici di riferimento.

Il piano di ammortamento dettaglia il modo in cui il mutuatario rimborserà il mutuo, solitamente attraverso rate periodiche composte da una quota di capitale e una di interessi. La durata del mutuo può variare notevolmente, influenzando sia la dimensione della rata sia il costo totale degli interessi.

Il processo di mutuo vede la partecipazione di diversi attori, ognuno con un ruolo specifico:

- Mutuatario: l'individuo o l'entità che riceve il prestito e si impegna a rimborsarlo secondo le condizioni stabilite.

[2] V. Zamagni (1993) The Economic History of Italy 1860-1990. Clarendon Press

- Mutuante: l'istituzione finanziaria, solitamente una banca o un'entità di credito, che fornisce il prestito al mutuatario. Il mutuante detiene spesso una garanzia sul prestito, come l'ipoteca su un immobile.
- Intermediari: figure professionali come agenti immobiliari, notai, e consulenti finanziari, che facilitano la negoziazione, la stipula e la gestione del mutuo. Gli intermediari possono anche includere broker di mutui, che agiscono come mediatori tra mutuatari e mutuanti per trovare le condizioni di mutuo più vantaggiose. (Ad esempio: Facile.it è un intermediario)[3]

Tipologie di Mutui

Nel contesto dei mutui, il tasso d'interesse rappresenta un elemento cruciale, determinando l'importo degli interessi che il mutuatario dovrà corrispondere al mutuante nel corso del tempo. Si distinguono principalmente due tipologie di tasso d'interesse: fisso e variabile.

- Mutui a Tasso Fisso: Questi mutui caratterizzano un tasso di interesse che rimane invariato per tutta la durata del prestito. Tale stabilità fornisce al mutuatario la sicurezza di rate costanti, facilitando la pianificazione finanziaria a lungo termine senza rischi legati a fluttuazioni dei tassi di mercato.
- Mutui a Tasso Variabile: Al contrario, i mutui a tasso variabile prevedono un tasso d'interesse che può cambiare nel tempo in base a determinati indici di riferimento. Sebbene possano offrire tassi iniziali più bassi, comportano un maggiore rischio di variazioni delle rate, influenzando la capacità del mutuatario di assorbire aumenti futuri del costo del credito.
- Mutui a Rate Costanti (Ammortamento Francese): In questo schema di ammortamento, il mutuatario paga rate periodiche costanti composte da una parte di capitale e una di interessi,

[3] Dong, Y., Chung, M., Zhou, C., & Venkataraman, S. (2019). Banking on "Mobile Money": The Implications of Mobile Money Services on the Value Chain. Manufacturing & Service Operations Management, 21(2), 357-374.

con una progressiva riduzione della quota di interessi e un incremento della quota capitale nel tempo.

- Mutui a Rate Crescenti o Decrescenti: Questi prodotti prevedono un piano di rimborso in cui l'importo delle rate varia nel tempo. Le rate possono aumentare (rate crescenti) per adattarsi a previste capacità di reddito maggiori del mutuatario o diminuire (rate decrescenti) se si prevede una riduzione del reddito disponibile.
- Mutui con Opzione di Pagamento Solo degli Interessi: Questi mutui permettono al mutuatario di pagare solamente gli interessi su un prestito per un periodo iniziale, differendo il rimborso del capitale a un momento successivo. Tale opzione può alleggerire il carico finanziario a breve termine, ma può comportare un costo totale maggiore a lungo termine.
- Mutui Reversibili: Conosciuti anche come mutui vitalizi o equity release, permettono agli anziani proprietari di casa di ottenere liquidità dal valore del proprio immobile, mantenendone l'uso. Il rimborso del capitale e degli interessi maturati avviene generalmente alla vendita dell'immobile o alla morte del mutuatario.

Queste diverse tipologie di mutui rispondono alle varie esigenze dei mutuatari, offrendo soluzioni personalizzate in base alla capacità di reddito, alla tolleranza al rischio e agli obiettivi finanziari a lungo termine.[4]

Processo di Richiesta e Ottenimento del Mutuo

Il processo di richiesta e ottenimento di un mutuo inizia con la verifica dei criteri di eleggibilità stabiliti dalle istituzioni finanziarie. Questi criteri possono variare in funzione del profilo del richiedente e del tipo di mutuo richiesto, ma generalmente includono requisiti relativi all'età, al

[4] Liu, J. (2018). Bank Stability, Sovereign Debt and Derivatives. Eastern Economic Journal, 44(1), 174-176.

reddito, alla stabilità lavorativa e alla cittadinanza o residenza legale. La documentazione necessaria per la richiesta di un mutuo tipicamente comprende:

- Documenti di identità per verificare l'identità del richiedente.
- Documentazione finanziaria, come buste paga, dichiarazioni dei redditi e estratti conto, per attestare la capacità di rimborso.
- Documenti relativi all'immobile oggetto del mutuo, come la perizia di valutazione e il preliminare di vendita.

La valutazione del merito creditizio rappresenta un passaggio fondamentale nel processo di concessione del mutuo. Le istituzioni finanziarie utilizzano questa valutazione per determinare la probabilità che il mutuatario sia in grado di adempiere ai suoi obblighi di rimborso. Tale valutazione si basa su diversi fattori, tra cui:

- Storia creditizia del richiedente, valutata attraverso i report delle agenzie di credito, che riflettono il comportamento di rimborso di prestiti o debiti precedenti.
- Livello di indebitamento, ossia il rapporto tra le uscite mensili per debiti e il reddito del richiedente.
- Stabilità e adeguata entità del reddito, per assicurare la capacità di sostenere le rate del mutuo nel tempo.

Le garanzie e le assicurazioni giocano un ruolo cruciale nel mitigare il rischio di inadempienza per il mutuante e nel fornire protezione al mutuatario. Tra queste:

- Ipoteca: La più comune forma di garanzia, che concede al mutuante un diritto reale sull'immobile finanziato, permettendogli di rivendicare l'immobile in caso di inadempienza del mutuatario.

- Assicurazione sui Mutui: Alcune istituzioni richiedono una polizza assicurativa che copra il rischio di morte, invalidità permanente o perdita di lavoro del mutuatario, garantendo il rimborso del mutuo anche in casi di grave avversità.
- Fondo Garanzia Mutui: In alcuni contesti, esistono fondi di garanzia statali o privati che offrono copertura supplementare, facilitando l'accesso al credito per categorie di mutuatari considerati più rischiosi.

Piano di ammortamento del mutuo

Il piano di ammortamento di un mutuo descrive come verranno ripagati nel tempo il capitale prestato e gli interessi. Esso stabilisce la frequenza e l'importo delle rate, dettagliando la quota di capitale e la quota di interessi contenute in ciascuna rata. Esistono vari tipi di piani di ammortamento, tra cui:

- Ammortamento Francese: Le rate sono costanti nel tempo e composte da una parte di capitale e una di interessi, con una progressiva diminuzione degli interessi e aumento del capitale rimborsato ad ogni rata.
- Ammortamento Italiano: Caratterizzato da rate composte da una quota di interessi costante e una quota capitale crescente, questo piano prevede un onere finanziario decrescente nel tempo.
- Ammortamento Americano: Prevede il pagamento degli interessi a periodi regolari e la restituzione dell'intero capitale in un'unica soluzione alla scadenza del mutuo.

Il calcolo della rata di un mutuo si basa sulla formula dell'ammortamento francese, la più comune nei piani di ammortamento:

$$R = C \cdot \frac{i(1+i)^n}{(1+i)^n - 1}$$

Equazione 1 Calcolo della rata di un mutuo

Dove:

- R è l'importo della rata
- C è il capitale del mutuo
- i è il tasso di interesse periodico
- n è il numero totale di rate

Per esempio, considerando un mutuo di €100.000 a un TAEG del 3% da rimborsare in 20 anni (240 rate), la rata mensile R sarà di €554,00.

La durata del mutuo e il tasso di interesse hanno un impatto significativo sul piano di ammortamento e, di conseguenza, sul costo totale del mutuo.

Allungare la durata del mutuo riduce l'importo delle rate mensili ma aumenta il costo totale degli interessi pagati nel tempo. Viceversa, una durata più breve comporta rate più elevate ma un minor costo complessivo degli interessi.

Un tasso di interesse più elevato aumenta l'importo degli interessi da pagare su ogni rata, incrementando il costo totale del mutuo. Un tasso più basso riduce il carico degli interessi, rendendo il mutuo meno oneroso.

L'interazione tra durata e tasso di interesse è fondamentale nella scelta del piano di ammortamento più adatto alle proprie capacità finanziarie e agli obiettivi personali, richiedendo un'attenta valutazione delle proprie condizioni economiche e delle prospettive future.

Aspetti Legali e Fiscali dei Mutui

Le normative sui mutui variano significativamente da un paese all'altro, riflettendo differenze nelle pratiche bancarie, nei sistemi legali e nelle politiche di protezione dei consumatori. A livello internazionale, organizzazioni come il Comitato sulla Supervisione Bancaria do Basilea stabiliscono principi guida per la gestione del rischio nel settore dei mutui, influenzando le legislazioni nazionali.

Nel contesto dell'Unione Europea, direttive come la MCD (Mortgage Credit Directive) mirano ad armonizzare le norme sui crediti immobiliari, promuovendo trasparenza e diritti dei consumatori. Un focus specifico sul paese di interesse richiederebbe l'analisi delle leggi locali che regolamentano l'erogazione dei mutui, gli obblighi degli istituti di credito e la protezione dei mutuatari.

Le implicazioni fiscali legate ai mutui sono un altro aspetto cruciale da considerare. Per i mutuatari, gli interessi pagati sul mutuo possono talvolta essere deducibili dalle imposte sui redditi, a seconda della legislazione fiscale del paese. Questa possibilità riduce l'onere fiscale del mutuatario, influenzando indirettamente la convenienza del mutuo.

Per i mutuanti, i proventi derivanti dagli interessi sui mutui rappresentano reddito imponibile. Alcuni paesi prevedono regolamentazioni specifiche per l'accantonamento di riserve per rischi di credito, che possono avere implicazioni fiscali per le istituzioni finanziarie.[5]

La protezione dei consumatori nel settore dei mutui è una priorità per molti legislatori. Le leggi tendono a garantire che i mutuatari siano pienamente informati dei termini e delle condizioni del loro mutuo, compresi tasso di interesse, costi aggiuntivi, e conseguenze in caso di inadempienza.

Diritti quali la possibilità di rimborso anticipato, la protezione contro le pratiche di vendita ingannevoli, e il diritto a un trattamento equo in caso di difficoltà finanziarie sono spesso codificati nella legislazione. Alcuni paesi hanno introdotto misure specifiche per prevenire gli sfratti forzati, fornendo ai mutuatari in difficoltà vie alternative come la ristrutturazione del debito.[6]

EURIBOR e IRS

Il Swap su Tasso di Interesse (IRS, Interest Rate Swap) e l'Euro Interbank Offered Rate (EURIBOR) sono strumenti finanziari e tassi di riferimento cruciali nei mercati dei mutui e dei prodotti finanziari

[5] Stockenstrand, A.-K., & Nilsson, F. (Eds.). (2017). Bank Regulation. Routledge.
[6] Levine, R., Lin, C., Wang, Z., & Xie, W. (2017). Bank Liquidity, Credit Supply and the Environment.

in Europa. L'IRS è un contratto finanziario attraverso il quale due parti si scambiano flussi di interesse su un capitale nozionale, generalmente con uno che paga un tasso fisso e riceve un tasso variabile, e viceversa.[7] L'EURIBOR, d'altra parte, rappresenta il tasso medio a cui le banche europee offrono di prestare fondi non garantiti ad altre banche nel mercato interbancario euro.[8]

Il Tasso Annuo Effettivo Globale (TAEG) rappresenta il costo totale del credito per il mutuatario, espresso in percentuale annuale del totale del credito. L'influenza di IRS ed EURIBOR sul TAEG si manifesta attraverso la loro incidenza sui tassi di interesse applicati ai mutui, in particolare quelli a tasso variabile.

Anche se i mutui a tasso fisso sembrano non essere direttamente influenzati dal variare dei tassi di mercato, in realtà, il tasso fisso offerto riflette le aspettative future del mercato, delle quali l'IRS è un indicatore principale. Pertanto, variazioni significative nelle curve degli IRS possono influenzare le condizioni di offerta dei mutui a tasso fisso, incidendo indirettamente sul TAEG.

L'EURIBOR è spesso utilizzato come indice di riferimento per la determinazione del tasso di interesse applicabile ai mutui a tasso variabile. Variazioni nell'EURIBOR si riflettono direttamente sulle rate di interesse dei mutui, influenzando il TAEG. Un aumento dell'EURIBOR comporta un incremento del tasso di interesse applicabile, elevando il TAEG, e viceversa.

Per mitigare l'incertezza e il rischio associato alla volatilità di IRS ed EURIBOR, sia mutuatari che istituti di credito possono adottare diverse strategie.

[7] Witzany, J. (2020). Interest Rate Derivatives. In Derivatives: Theory and Practice of Trading, Valuation, and Risk Management (pp. 43-75). Cham: Springer International Publishing.

[8] Golitsis, P., Bellos, S. K., Fassas, A. P., & Demiralay, S. (2021). The Spillover Effect of Euribor on Southeastern European Economies: A Global VAR Approach. Journal of East-West Business, 27(1), 57-91.

I mutuatari possono stipulare contratti IRS per fissare il costo del finanziamento, proteggendosi dal rischio di aumento dei tassi di interesse. Questo permette una maggiore prevedibilità delle spese future, influenzando positivamente la gestione del TAEG.

Alcune istituzioni finanziarie offrono la possibilità di rinegoziare le condizioni del mutuo in risposta alle fluttuazioni di EURIBOR, consentendo ai mutuatari di adattarsi a condizioni di mercato più favorevoli.

I tassi IRS e EURIBOR influenzano i tassi d'interesse, che influenza a sua volta il costo del mutuo al cliente finale e i margini di profitto delle banche. Se i tassi salgono, sale il TAEG, così come i costi di un mutuo e i profitti delle banche. A fianco viene riportata l'oscillazione del TAEG medio di un segmento dei mutui erogati dal 2019 al 2024, un periodo particolare a causa della pandemia COVID-19, che ha prima visto un ribasso netto dei tassi di interesse tra il 2020 e il 2023, per poi avere un aumento esponenziale.

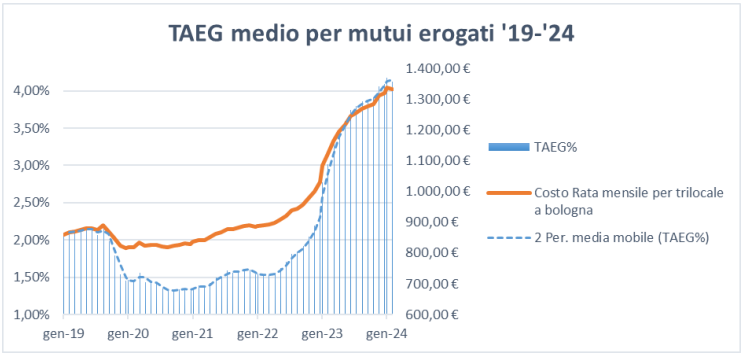


Figura 1 Andamento del TAEG dal 2019 al 2024

Interazioni tra il mercato dei mutui e il mercato immobiliare

Il mercato dei mutui e il mercato immobiliare sono strettamente interconnessi, con dinamiche che si influenzano reciprocamente in modo significativo. L'accessibilità ai mutui gioca un ruolo cruciale nella determinazione della domanda di immobili, influenzando così i prezzi delle case. Quando le condizioni di credito sono favorevoli, con tassi di interesse bassi e requisiti di eleggibilità meno stringenti, si registra un incremento nella capacità di acquisto, che può portare a un aumento della domanda di immobili e, di conseguenza, a un rialzo dei prezzi. Al contrario, il restringimento del credito tende a ridurre la domanda e a moderare l'incremento dei prezzi immobiliari.

I tassi di interesse rappresentano uno degli strumenti più influenti attraverso i quali le politiche monetarie agiscono sul mercato immobiliare. Una diminuzione dei tassi di interesse rende il costo dei mutui meno oneroso, stimolando la domanda di immobili e contribuendo all'aumento dei prezzi. Inversamente, un aumento dei tassi di interesse rende il finanziamento più costoso, limitando la domanda e esercitando pressione al ribasso sui prezzi delle case. Questa dinamica evidenzia come variazioni anche modeste dei tassi di interesse possano avere impatti significativi sul mercato immobiliare.

La storia economica offre numerosi esempi di come l'interazione tra mercato dei mutui e mercato immobiliare possa portare a bolle speculative e crisi finanziarie. Un caso emblematico è rappresentato dalla crisi finanziaria del 2007-2008, originata in parte dall'eccessiva facilità di accesso al credito immobiliare negli Stati Uniti, che ha alimentato una bolla speculativa nel mercato delle case. L'esplosione della bolla ha avuto ripercussioni devastanti sull'economia globale, evidenziando i rischi di una regolamentazione inadeguata del settore dei mutui e la necessità di meccanismi di vigilanza più efficaci per prevenire l'accumulo di squilibri finanziari.

Un altro esempio storico è la bolla immobiliare giapponese degli anni '80, che ha visto un aumento esponenziale dei prezzi immobiliari, alimentato da politiche di credito eccessivamente permissive.[9] La successiva correzione dei prezzi ha portato a una prolungata stagnazione economica, dimostrando le profonde conseguenze che le dinamiche del mercato immobiliare possono avere sull'intero sistema economico.

Questi casi storici sottolineano l'importanza di monitorare attentamente l'interazione tra mercato dei mutui e mercato immobiliare, al fine di identificare tempestivamente segnali di sovrapprezzo e prevenire le conseguenze negative associate alle bolle speculative. La comprensione delle dinamiche

[9] Hu, Y., & Oxley, L. (2018). Bubble contagion: Evidence from Japan's asset price bubble of the 1980-90s. Journal of the Japanese and International Economies, 50, 89-95.

di questi mercati è fondamentale per la formulazione di politiche economiche volte a mantenere la stabilità finanziaria e promuovere uno sviluppo sostenibile.

Concorrenza tra banche e tra broker

In questo contesto bancario si distinguono due tipi di concorrenza. La concorrenza fra banche diverse e la concorrenza tra i diversi broker di clienti per le banche.

La concorrenza tra le banche retail si manifesta in diversi ambiti[10], che riguardano la qualità, il costo e l'innovazione dei prodotti e dei servizi bancari offerti alla clientela di massa. Tra i prodotti più rilevanti per la concorrenza bancaria ci sono i mutui, che implicano un impegno finanziario di lunga durata e di elevato importo per i clienti, e una fonte di reddito e di rischio per le banche. Per confrontare le diverse offerte di mutui, si devono considerare i diversi aspetti.

Il tasso di interesse, che può essere fisso o variabile, e che determina la rata e il costo totale del mutuo. Il tasso di interesse dipende da indici di mercato, come l'EURIBOR per i mutui a tasso variabile e l'IRS per i mutui a tasso fisso.

La durata del mutuo, che può variare da pochi anni a decenni, e che influisce sul livello della rata e sul rapporto tra interessi e capitale. In generale, maggiore è la durata, minore è la rata, ma maggiore è il costo complessivo del mutuo.

L'importo del mutuo, che dipende dal valore dell'immobile e dalla capacità di rimborso del cliente, e che determina il rapporto tra il mutuo e il valore dell'immobile (LTV). In generale, minore è l'importo, minore è il rischio per la banca e migliore è il tasso offerto al cliente. L'offerta al cliente

[10] Kanas, A., Hassan Al-Tamimi, H. A., Albaity, M., & Mallek, R. S. (2019). Bank competition, stability, and intervention quality. Journal of Financial Stability, 43, 100-710.

può essere variegata, dipendentemente da fattori come il tipo di casa da acquistare (green/non green), l'età (under/over 36), e così via.[11]

Le spese e le commissioni, che sono i costi aggiuntivi legati all'erogazione e alla gestione del mutuo, come le spese di istruttoria, di perizia, di assicurazione e di surroga. Queste spese incidono sul Tasso Annuo Effettivo Globale (TAEG), che è l'indicatore che misura il costo effettivo del mutuo, comprensivo di interessi e spese.

La flessibilità e la personalizzazione, che sono le possibilità di modificare le condizioni del mutuo nel corso del tempo, in base alle esigenze del cliente, come la variazione della rata, la sospensione del pagamento, il cambio del tasso o la surroga. Queste opzioni possono rendere il mutuo più adatto alle esigenze del cliente, ma possono comportare dei costi aggiuntivi.

I broker (online) di mutui hanno un tipo di competizione diversa. Definiamo i broker di mutui in Italia come tali piattaforme che permettono una comparazione di mutui di varie banche all'utente che si collega, in modo gratuito, che viene poi contattato dal call center della suddetta azienda per procedere con l'avvio di un mutuo assistito. Si individuano le seguenti caratteristiche su cui valutare i broker:

La qualità del servizio offerto, comprendendo l'abilità del call center di convertire, la bontà della piattaforma online e quanti leads riesce a generare e i tempi di gestione delle leads. La qualità e la dimensione di un broker di mutui sono teoricamente proporzionali.

Un altro fattore di valutazione sono le partnership del broker con le banche, che corrispondono ai loro "fornitori". Più banche un broker ha a disposizione, maggiore è la sua offerta[12] e più clienti riceverà,

[11] Vives, X. (2019). Competition and stability in modern banking: A post-crisis perspective. International Journal of Industrial Organization, 64, 55-69.

[12] Mi, B., & Han, L. (2018). Banking market concentration and syndicated loan prices. Review of Quantitative Finance and Accounting, 54(1), 1-28.

anche considerando la grande quantità di richiedenti di mutuo che si affidano alla loro banca correntista.

La quota di mercato detenuta dal broker, considerando l'intero mercato del broking di mutui. Questo è correlato al numero di clienti gestiti dall'azienda, specialmente se sono clienti che riescono a ottenere il mutuo.

Infine, nonostante le piattaforme di broking di mutui siano nate online, la presenza fisica sul territorio italiano con i "stores fisici" rappresenta un tema importante che porta veridicità al brand e maggior fiducia da parte dei clienti e potenziali clienti.

Innovazioni nel Settore dei Mutui

L'avvento della blockchain e l'evoluzione dei mutui digitali rappresentano due delle innovazioni più significative nel settore dei mutui.[13] La blockchain, con la sua capacità di garantire trasparenza, sicurezza e immutabilità delle transazioni, promette di rivoluzionare il processo di concessione dei mutui. L'impiego di questa tecnologia può semplificare e velocizzare le procedure, riducendo i costi associati e migliorando l'efficienza operativa. I mutui digitali, dall'altra parte, offrono un'esperienza utente semplificata, consentendo ai mutuatari di gestire l'intero processo online, dalla richiesta all'erogazione, fino alla gestione del mutuo.

Parallelamente all'innovazione tecnologica, si assiste a un crescente interesse verso i mutui sostenibili e il green financing. Questi prodotti finanziari sono progettati per supportare investimenti in efficienza energetica, energie rinnovabili e progetti immobiliari che rispettano elevati standard ambientali. Attraverso incentivi come tassi di interesse ridotti o condizioni di finanziamento

[13] Mullan, J., Bradley, L., Loane, S., Estelami, H., & Laukkanen, T. (2017). Bank adoption of mobile banking: stakeholder perspective. International Journal of Bank Marketing, 35(7), 1154-1174. https://doi.org/10.1108/IJBM-09-2015-0145

favorevoli, i mutui sostenibili mirano a promuovere pratiche di costruzione e ristrutturazione che abbiano un impatto positivo sull'ambiente.[14]

Le innovazioni tecnologiche e la spinta verso la sostenibilità stanno delineando le tendenze future nel settore dei mutui. Si prevede che la digitalizzazione continuerà a essere un fattore chiave, con l'adozione di intelligenza artificiale e machine learning per personalizzare l'offerta di prodotti finanziari e migliorare la valutazione del rischio. Allo stesso tempo, l'attenzione verso l'impatto ambientale e sociale dei finanziamenti immobiliari guiderà lo sviluppo di nuovi prodotti e servizi che rispondano alle esigenze di un mercato sempre più consapevole.

Queste tendenze non solo modelleranno il futuro del settore dei mutui[15] ma avranno anche un impatto significativo sul mercato immobiliare nel suo complesso, influenzando le decisioni di investimento e le strategie di sviluppo urbano. L'innovazione nel settore dei mutui, quindi, rappresenta non solo un'opportunità per migliorare l'efficienza e l'accessibilità del credito ma anche un veicolo per promuovere pratiche di sviluppo sostenibile.[16]

Comparativa del mutui nei diversi paesi

Le pratiche di mutuo variano notevolmente tra i diversi paesi, influenzate da fattori culturali, economici e regolamentari. Mentre in alcuni paesi prevale l'utilizzo di mutui a tasso fisso per offrire maggiore sicurezza ai mutuatari, in altri, come alcune economie europee, i mutui a tasso variabile sono più comuni, riflettendo una maggiore tolleranza al rischio da parte dei consumatori e un diverso

[14] Esposito, L., Mastromatteo, G., Molocchi, A., Brambilla, P. C., Carvalho, M. L., Girardi, P., Marmiroli, B., & Mela, G. (2022). Green mortgages, EU taxonomy and environment risk weighted assets: A key link for the transition. Sustainability, 14(3), 1633.

[15] Fuster, A., Plosser, M., Schnabl, P., & Vickery, J. (2019). The role of technology in mortgage lending. The Review of Financial Studies, 32(5), 1854-1899. https://doi.org/10.1093/rfs/hhz018

[16] Litvinova, S. A., Ivanova, O. B., Chernobay, O. S., & Zarubina, V. R. (2023). Green mortgage as the key to energy-efficient and resource-saving real estate. In A. K. Bahl, S. Batra, & S. K. Dhameja (Eds.), Advances in entrepreneurial economics and sustainable development: Climate-smart innovation (pp. 269-284). Springer.

approccio delle politiche monetarie. Queste differenze si riflettono anche nella durata dei mutui, nei requisiti di capitale proprio e nelle opzioni di ammortamento disponibili per i mutuatari.

Le politiche monetarie e fiscali hanno un impatto diretto sui mercati dei mutui e immobiliari. Un ambiente di bassi tassi di interesse, per esempio, tende a stimolare la domanda di mutui e, di conseguenza, i prezzi delle case. Al contrario, misure fiscali come deduzioni fiscali sugli interessi del mutuo o tasse sulla proprietà possono influenzare la convenienza e l'attrattiva degli investimenti immobiliari. La relazione tra capitale, rischio e efficienza delle banche, come evidenziato nello studio sulle banche commerciali vietnamite, sottolinea l'importanza di un equilibrio tra regolamentazione e stimoli al mercato per promuovere la stabilità finanziaria e l'efficienza operativa.

L'analisi dei casi storici e delle pratiche attuali offre preziose lezioni su come bilanciare crescita e stabilità nei mercati dei mutui e immobiliari. L'importanza di una regolamentazione prudente e di meccanismi di supervisione adeguati è evidenziata dalla necessità di prevenire comportamenti di rischio eccessivo e di garantire la solidità del sistema bancario. La promozione dell'efficienza bancaria, attraverso l'adozione di tecnologie avanzate e la diversificazione dei servizi, emerge come strategia chiave per migliorare la performance e ridurre il rischio di credito. Inoltre, l'incremento del capitale minimo richiesto, come suggerito dalle normative internazionali, può contribuire a rafforzare la resilienza delle banche alle fluttuazioni economiche.[17]

Business Model e Processo di creazione valore di un intermediario di mutui

Nel mercato degli intermediari di mutui, il valore viene generato se il cliente effettua un processo completo, dalla comparazione iniziale sul sito, fino all’erogazione della banca del mutuo.

[17] Le, T. (2016). Bank risk, capitalisation and technical efficiency in Vietnamese banking. SSRN Electronic Journal.

Figura 2 Processo di creazione valore per i Broker di mutui

A valle di questo processo, se conclusosi con successo, l'azienda riceve una commissione.

Ci sono 3 tipi di commissioni che l'azienda può ricevere. Ad esempio:

- 1% per mutui d'acquisto
- 0,55% per mutui surroga
- 1,3% per mutui d'acquisto con gestione due diligence

Nel caso d'acquisto due diligence l'azienda prende carico delle operazioni di verifica al posto della banca, risultando in un processo più snello per il partner e più remunerativo per l'azienda.

Le tempistiche tra la quarta e la quinta fase sono tendenzialmente di 4-6 mesi. Questo risulta in una difficoltà aggiunta del business dettato dalla dilazione del pagamento delle commissioni dalla banca e, di conseguenza, di ricavi per l'azienda.[18]

L'azienda riceve denaro dalle commissioni verso i suoi fornitori. Ad esempio, se un cliente di Facile.it ottiene un mutuo di 100.000€ dalla banca, l'1% del importo totale (€1.000) viene pagato dalla banca a Facile.it come commissione.

Tramite un'analisi con Business Model Canvas si può inquadrare il modello di business di un broker che si occupa dei mutui.

[18] Robles-Garcia, C. (2019). Competition and incentives in mortgage markets: The role of brokers. Unpublished working paper.

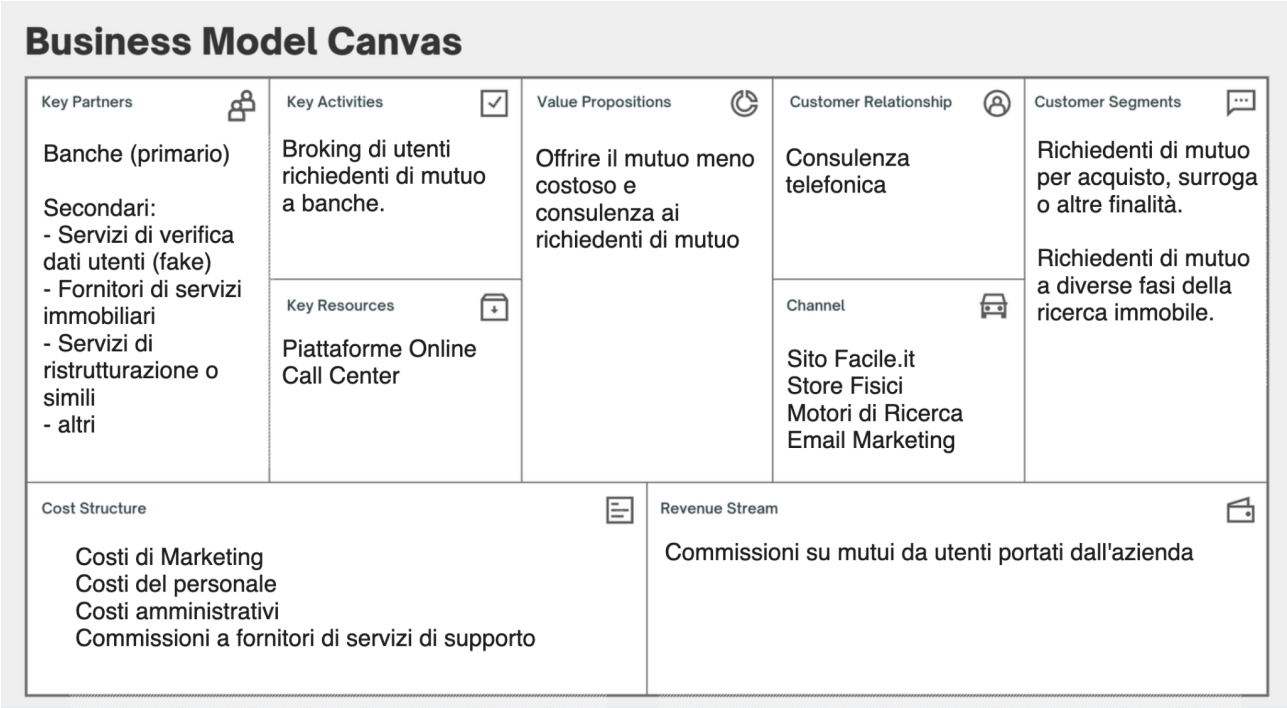


Figura 3 Business Model Canvas dei broker di mutui

Confronto tra Lead Parziali e Complete e differenza nella gestione dei call center

Nel contesto di Facile.it, la distinzione tra lead parziali e completi si rivela un fattore cruciale nel determinare le strategie di marketing e di vendita. I lead completi, avendo fornito un set completo di informazioni, mostrano un grado di coinvolgimento più elevato nel processo di acquisizione del mutuo. Questo comportamento è in linea con il modello di engagement del cliente discusso da Puccinelli et al. (2010), che evidenzia come un maggiore coinvolgimento con un marchio o un prodotto aumenti la probabilità di una decisione di acquisto. Un esempio pratico di questo fenomeno può essere trovato nello studio di Lemon e Verhoef (2016), che analizza il comportamento dei clienti nel settore bancario. Lo studio ha rilevato che i clienti che interagiscono attivamente con la banca attraverso più canali hanno maggiori probabilità di approfondire il loro rapporto con l'istituto, evidenziando il legame tra coinvolgimento e fedeltà.

I lead parziali, d'altra parte, rappresentano una sfida maggiore. Essi hanno fornito informazioni limitate e richiedono un approccio diversificato per la conversione. La loro gestione richiede tecniche

di nurturing e sviluppo delle relazioni, come illustrato da Gummesson (2008), che enfatizza l'importanza di costruire e mantenere relazioni di lungo termine con i clienti. Un esempio pertinente è stato esplorato da Godin (2007) nel suo lavoro sui lead freddi nel marketing digitale. Godin sostiene che i lead freddi, simili ai lead parziali di Facile.it, possono essere gradualmente 'riscaldati' attraverso comunicazioni mirate e contenuti di valore, trasformandosi infine in clienti fidati.[19]

La differenza nella conversione di lead caldi (completi) e lead freddi (parziali) è una questione centrale nella gestione efficace del funnel di vendita. I lead caldi sono già ben avviati nel percorso di acquisto e richiedono un approccio più diretto e incentrato sulla chiusura. Tuttavia, i lead freddi non devono essere trascurati, poiché rappresentano un potenziale insediato. Un approccio efficace per la conversione di questi lead è descritto nel lavoro di Kotler e Armstrong (2010), che suggerisce l'uso di tecniche di marketing inbound per attrarre e coinvolgere i lead freddi, fornendo loro informazioni utili e pertinenti e costruendo una relazione che può, col tempo, portare alla conversione.

Le fonti dei lead, in particolare il passaggio da team Verifica Dati a team Consulenti, svolgono un ruolo vitale nella pipeline di vendita di Facile.it.

Nell'ecosistema di Facile.it, la dinamica tra il team Verifica Dati e il team Consulenti si rivela cruciale nella gestione e nel trattamento dei lead. La maggior parte dei lead iniziali viene gestita dal team Verifica Dati, che si occupa di un volume significativo di richieste, principalmente concentrate sui lead parziali. Questi lead, caratterizzati da un livello di impegno iniziale più basso, rappresentano per l'azienda una sfida unica in termini di conversione e gestione delle risorse.

Il team Verifica Dati agisce come un filtro iniziale nel processo di vendita, selezionando i lead più promettenti tra il grande volume di richieste parziali. Questa selezione è basata su diversi criteri, come l'interesse dimostrato dal cliente, la completezza delle informazioni fornite e potenziali indicatori di

[19] Godin, S. (2007). Permission Marketing: Turning Strangers into Friends and Friends into Customers. New York, NY: Simon & Schuster.

qualità del lead. Circa l'8% di questi lead parziali, che mostrano segnali di maggiore impegno o avanzano nel processo di decisione, viene poi trasferito al team Consulenti. Questo passaggio rappresenta un momento critico nella pipeline di vendita, poiché segna la transizione di un lead da una fase di basso coinvolgimento a una di alto coinvolgimento.

La Conversion Rate Verifica Dati/Parziali, che traccia il tasso di successo di questo trasferimento, è una metrica fondamentale per valutare l'efficacia del team Verifica Dati nel qualificare i lead.[20] Un tasso di conversione elevato indica che il team Verifica Dati sta efficacemente identificando e passando i lead più promettenti ai Consulenti. Questi lead selezionati sono generalmente più informati, più interessati e quindi hanno una probabilità più alta di concludere il processo di acquisizione del mutuo. Hughes e Reynolds (2019) sottolineano l'importanza di monitorare tali metriche per valutare l'efficienza dei processi di vendita e marketing. I lead gestiti dai consulenti, essendo più qualificati e filtrati, tendono a mostrare tassi di conversione più elevati, riflettendo la teoria di Kaushik (2010) sulla necessità di ottimizzare le risorse di marketing sui segmenti più produttivi.

Il ruolo dei Consulenti, quindi, diventa quello di affinare ulteriormente la relazione con questi lead selezionati, utilizzando un approccio più personalizzato e mirato. I Consulenti lavorano per comprendere meglio le esigenze specifiche di ogni cliente, guidandoli attraverso le opzioni di mutuo disponibili e offrendo soluzioni su misura per le loro circostanze. Questo processo non solo migliora l'esperienza del cliente, ma aumenta anche le probabilità di concludere con successo la vendita.

[20] Li, Dongfang, Baotian Hu, Qingcai Chen, Xiao Wang, Quanchang Qi, Liubin Wang, and Haishan Liu. "Attentive capsule network for click-through rate and conversion rate prediction in online advertising." Knowledge-based systems 211 (2021): 106522.

Capitolo 2 - STRUMENTI DI ANALISI DATI

Il 90% dei dati mondiali è stato creato negli ultimi due anni. La crescita della quantità di dati disponibile al giorno d'oggi è stata esponenziale e, grazie all'avanzamento tecnologico, ad oggi si ha a disposizione la potenza di calcolo per analizzare queste grandi quantità di dati.

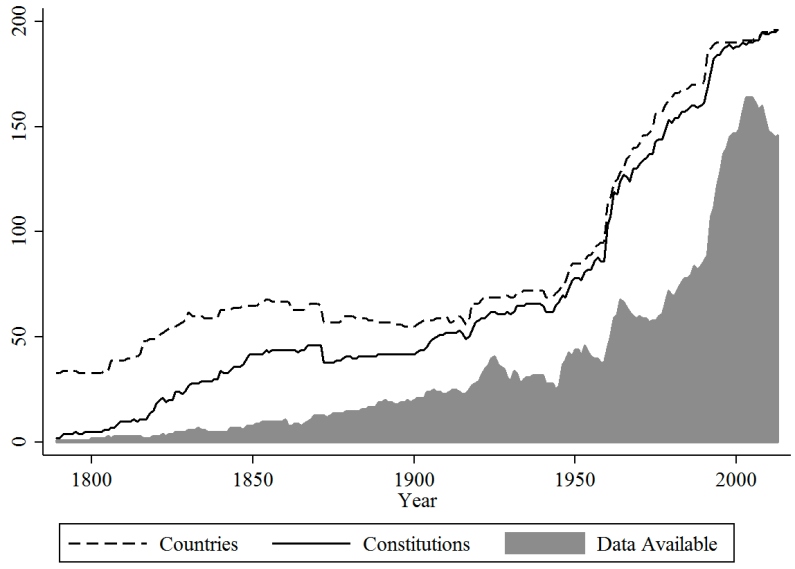


Figura 4 Evoluzione dei dati disponibili nel tempo

La maggior parte dei metodi per analizzare i dati, incluso il Machine Learning, proviene dalla statistica. Questo capitolo ha infatti l'obiettivo di presentare l'evoluzione nell'analisi dei dati nel ventesimo secolo, con i rispettivi autori, e di spiegare la funzione, l'innovazione e l'utilizzo concreto di ogni tecnica.

Analisi della correlazione di Pearson

La correlazione di Pearson è stata sviluppata dal matematico Karl Pearson alla fine del XIX secolo, intorno agli anni 1890. Karl Pearson (1857-1936) è stato un matematico e statistico britannico, noto come uno dei fondatori della moderna statistica. Le sue contribuzioni includono lo sviluppo di coefficienti di correlazione e la formulazione del test del chi quadrato. Questo metodo statistico è utilizzato per misurare il grado di relazione lineare tra due variabili.

La formula della correlazione di Pearson è la seguente:

Equazione 2 Formula di Pearson

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

Analisi del chi quadro

Pearson ha presentato la teoria del chi quadrato nel suo elaborato "On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling". Questo articolo, pubblicato nel 1900, stabilì le basi per il test del chi quadrato. A seguito, G. Udny Yule ampliò le sue teorie nel suo libro "An Introduction to the Theory of Statistics", uno dei primi libri di testo sulla statistica moderna.

L'innovazione principale della tecnica del chi quadrato sta nel fornire un metodo quantitativo per testare l'ipotesi di indipendenza tra categorie in una tabella di contingenza. Prima del chi quadrato, l'analisi delle tabelle di frequenza era più soggettiva e meno rigorosa. Il chi quadrato ha permesso di valutare statisticamente se le differenze osservate nelle frequenze potessero essere dovute al caso o se fossero statisticamente significative.

L'esperimento dei dadi di Weldon è un caso classico nella storia della statistica, utilizzato da Karl Pearson per illustrare l'applicazione del test del chi quadrato.

L'esperimento di Weldon si concentrava sul lancio di 12 dadi a sei facce ripetuto 26.306 volte. Weldon aveva poi confrontato la frequenza di uscita del 5 o 6 attesa con quella effettiva, misurando quindi la deviazione.

Tabella 1 Frequenze dell'esperimento di Pearson

No. of Dice in Cast with 5 or 6 Points.	Observed Frequency, m'.	Theoretical Frequency, m.	Deviation, e.
0	185	203	− 18
1	1149	1217	− 68
2	3265	3345	− 80
3	5475	5576	−101
4	6114	6273	−159
5	5194	5018	+176
6	3067	2927	+140
7	1331	1254	+ 77
8	403	392	+ 11
9	105	87	+ 18
10	14	13	+ 1
11	4	1	+ 3
12	0	0	0
	26306	26306	

Osservando le frequenze si nota immediatamente che il 5 o il 6 sono usciti più spesso per 4 volte in un lancio, seguito da 3 e da 5 volte. Dallo studio della deviazione, però, si nota che rispetto alle frequenze attese, la frequenza di 5 e 6 volte è occorsa significativamente più volte, mentre la frequenza di 3 e 4 volte è occorsa significativamente meno volte (significativamente = valore assoluto della deviazione maggiore di 100).

Questo esperimento ha permesso di valutare se le frequenze osservate di ciascun numero differivano in modo significativo da quelle attese in un lancio equo di dadi. In un dado equo, ci si aspetterebbe che ogni numero compaia con la stessa frequenza. Pearson ha elevato alla quadra la deviazione calcolata e diviso per la frequenza attesa. Ciò ha permesso di calcolare la probabilità che le deviazioni osservate dalle frequenze attese potessero essere dovute al caso, o se invece indicassero una qualche forma di bias nei dadi.

Tabella 2 Deviazioni dell'esperimento di Pearson

Group.	e^2.	e^2/m.	Group.	e^2.	e^2/m.
0	324	1·59606	7	5929	4·72807
1	4624	3·79951	8	121	0·30903
2	6400	1·91330	9	324	3·72414
3	10201	1·82945	10	1	0·07346
4	25281	4·03013	11	9	9·00000
5	30976	6·17298	12	0	·00000
6	19600	6·69628	Total...	...	43·87241

Da notare che la deviazione inizia e continua a essere positiva alla frequenza di 5 in poi, un indizio che Karl Pearson andrà a sviluppare trovando che l'esperimento aveva un bias rivolto verso le frequenze più alte di uscita di 5 o 6.

L'importanza di questo studio non stava solo nel risultato specifico relativo ai dadi, ma nel metodo utilizzato: l'applicazione del test del chi quadrato forniva uno strumento statistico oggettivo per valutare l'equità e la casualità in esperimenti simili. Questo metodo è diventato uno strumento fondamentale nella statistica, utilizzato in una vasta gamma di applicazioni per testare l'ipotesi di indipendenza e l'equità in dati categorici.

Al tempo di Pearson, gli strumenti per l'analisi statistica erano molto più rudimentali rispetto a oggi. Si basavano principalmente su calcoli manuali e tavole matematiche. Le calcolatrici meccaniche, come le macchine di somma e le prime forme di calcolatrici, erano disponibili, ma erano limitate nelle loro capacità. Non esistevano computer o software statistici.

Prima del chi quadrato, questo tipo di analisi sarebbe stato effettuato semplicemente confrontando le frequenze o le percentuali, senza un metodo per testare la significatività statistica delle differenze osservate.

Infine, se la numerosità del totale degli eventi è compresa tra 40 e 200 va aggiunta la correzione di Yates al test del chi quadrato, ovvero la sottrazione al valore assoluto della deviazione di 0,5:

Equazione 3 Test del chi quadrato

$$x^2 = \sum^{k} \frac{(|F.Osservata - F.Attesa| - 0{,}5)^2}{Frequenza\ Attesa}$$

Analisi ANOVA

Nel 1925 Sir Ronald Aylmer Fisher, un eminente statistico e genetista britannico, scrive "Statistical Methods for Research Workers", introducendo una tecnica rivoluzionaria nell'analisi statistica, ovvero l'analisi della varianza (ANOVA), che ha teorizzato per confrontare medie tra diversi gruppi e per gestire la complessità derivante da sperimentazioni con più variabili.

Nell'ambito della statistica, l'analisi della varianza (ANOVA), che sta per "Analisi della Varianza", è un metodo utilizzato per confrontare le medie di una variabile quantitativa, nota come variabile dipendente, tra diverse categorie di una o più variabili indipendenti, sia quantitative sia qualitative. Questo tipo di analisi presuppone la normalità e l'omogeneità delle varianze (omoschedasticità) della variabile dipendente.

Quando l'analisi coinvolge più variabili dipendenti in relazione a uno o più fattori indipendenti, si parla di MANOVA, acronimo di "Multivariate Analysis of Variance". Anche in questo contesto, è fondamentale che le variabili dipendenti rispettino le condizioni di normalità e omoschedasticità.

I diversi stati o categorie delle variabili indipendenti sono definiti come livelli. In scenari con un solo fattore, questi livelli corrispondono ai vari trattamenti. L'obiettivo principale di tali analisi è determinare se i trattamenti influenzano significativamente la media della variabile dipendente o le medie di più variabili dipendenti.

Quando si lavora con due o più fattori, si analizzano i valori della variabile dipendente (o delle variabili dipendenti) in diverse combinazioni di livelli, identificate come trattamenti. In questi casi, l'interesse si sposta sull'effetto di ciascun fattore e sulla loro possibile interazione. L'assenza di interazione tra i fattori è indicata quando l'effetto complessivo di un trattamento sulla variabile

dipendente è semplicemente la somma degli effetti di ciascun fattore. Al contrario, se l'effetto complessivo differisce significativamente dalla somma degli effetti dei singoli fattori, si conclude che esiste un'interazione tra i fattori.

In ogni caso, l'analisi ANOVA o MANOVA procede scomponendo la varianza totale della variabile dipendente in varianza spiegata, attribuibile ai fattori, e varianza residua, cioè la varianza all'interno dei trattamenti. Questa scomposizione permette di valutare l'impatto di ciascun fattore e delle loro interazioni sulle variabili in studio.

L'innovazione principale dell'ANOVA stava nella sua capacità di suddividere la varianza osservata in componenti attribuibili a diverse fonti di variazione. Fu emblematica l'applicazione della tecnica ANOVA alla sperimentazione agricola e biologica, permettendo di separare e valutare l'influenza di differenti fattori su un esperimento, ovvero l'esperimento sulle rese del grano condotto da Fisher nel 1926 presso la Rothamsted Experimental Station in Inghilterra. Fisher voleva studiare l'effetto di due fattori sperimentali: la varietà di grano e il tipo di fertilizzante applicato.

Immaginiamo, ad esempio, di voler analizzare l'efficacia di tre tipi di fertilizzante (identificati come a1, a2 e a3) su terreni con quattro diverse caratteristiche di composizione (etichettate come b1, b2, b3 e b4). Se l'obiettivo primario è esaminare l'effetto dei fertilizzanti, tralasciando le variazioni dovute alle specifiche qualità dei suoli, si potrebbe applicare ciascuno dei tre fertilizzanti a tutti e quattro i tipi di suolo. In questo scenario, le diverse composizioni del suolo rappresentano un fattore secondario nell'esperimento, il cui impatto non è l'oggetto principale dell'analisi.

Per fare ciò, ha usato un disegno sperimentale a blocchi randomizzati, in cui ha diviso il campo in 24 blocchi e ha assegnato casualmente a ogni blocco una delle sei varietà di grano e una delle quattro quantità di fertilizzante. Poi ha misurato la resa del grano per ogni blocco e ha applicato l'ANOVA a due vie per analizzare i dati.

Tabella 3 Tabella a due vie ANOVA

Source of variation	df	Mean square				
		Plant height	Number of leaves	Number of nodes	Number of lateral branches	Internode length
Replication	2	77.35^{*}	10660.5^{**}	8.44^{*}	13.08^{**}	0.04^{ns}
Treatment	5	221.89^{**}	1571.56^{**}	0.41^{ns}	16.82^{**}	1.11^{*}
Error	10	14.38	267.71	1.31	1.42	0.19
C.V. (%)		6.06	17.05	9.98	6.39	7.28

^{ns}, *, **: non-significant or significant at $P \leq 0.01$ and $P \leq 0.05$, respectively.

Ha poi confrontato i p-value con il livello di significatività scelto per verificare se i fattori o l'interazione avevano un effetto significativo sulla resa del grano. Ha trovato che sia il primo fattore (varietà di grano) che il secondo fattore (quantità di fertilizzante) erano significativi, ma non l'interazione. Questo significa che la resa del grano dipendeva dalla varietà che dal fertilizzante, ma che la varietà di grano non dipendeva dalla quantità di fertilizzante. Questo esperimento è considerato uno dei primi e più importanti esempi di applicazione dell'ANOVA in agricoltura.

L'approccio teorizzato da Fisher ha permesso agli scienziati di testare l'effetto di vari fattori in esperimenti complessi, un miglioramento significativo rispetto ai metodi precedenti che non permettevano analisi così dettagliate. Infatti, prima dell'introduzione dell'ANOVA, le tecniche di analisi statistica erano più limitate e meno sofisticate. Si faceva affidamento su metodi più semplici per il confronto di medie e proporzioni, spesso senza la capacità di gestire efficacemente la variazione all'interno e tra i gruppi.

L'analisi della varianza (ANOVA) viene utilizzata oggi in diversi ambiti scientifici e di ricerca per confrontare e valutare le differenze tra i gruppi. È particolarmente utile in esperimenti dove si vogliono confrontare più di due gruppi o trattamenti, ed è facilmente implementata con strumenti di analisi dati come R o STATA, tramite una funzione "anova()".

Per eseguire un'analisi ANOVA con una tabella a due vie, è necessario disporre di dati raccolti per almeno due variabili indipendenti categoriali e una variabile dipendente continua. I dati devono essere suddivisi in gruppi basati su queste variabili indipendenti.

L'output principale da cercare in un'analisi ANOVA è il valore di F, che si calcola dividendo la varianza tra i gruppi (che misura quanto i diversi gruppi differiscono l'uno dall'altro) per la varianza all'interno dei gruppi (che misura quanto i dati all'interno di ciascun gruppo si discostano dalla loro media di gruppo). Un valore di F elevato suggerisce che le differenze tra i gruppi sono maggiori rispetto alle differenze all'interno dei gruppi, indicando che le variabili indipendenti hanno un effetto significativo sulla variabile dipendente. Inoltre, si osservano i valori di p per valutare la significatività statistica delle differenze tra i gruppi.

L’analisi esplorativa dei dati (Exploratory Data Analysis - EDA) e il test di Tukey

Anche l'analisi esplorativa dei dati (EDA) è, ad oggi, una pratica fondamentale per la scienza dei dati, che consiste nell'esaminare i dati tramite valori, statistiche e visualizzazioni, al fine di comprenderne le caratteristiche e le relazioni. L'EDA è stata introdotta da Tukey come complemento all'analisi confermativa, con l'obiettivo di adottare un'attitudine flessibile e lasciare che i dati suggeriscano le decisioni di modellazione statistica. L'EDA si integra quindi con l'analisi confermativa e il processo di modellazione statistica, formando un ciclo iterativo di esplorazione e verifica delle ipotesi.

Gli obiettivi principali dell'EDA sono due: la profilazione e la scoperta . La profilazione consiste nel capire cosa contengono i dati e valutarne la qualità, identificando eventuali errori, valori mancanti, outlier o anomalie . La scoperta consiste nel ricavare nuove intuizioni dai dati, individuando pattern, trend, correlazioni o differenze significative . Sebbene la letteratura sull'EDA enfatizzi la scoperta, la profilazione è un'attività che tutti gli analisti svolgono in tutte le loro analisi, mentre la scoperta si verifica solo in analisi aperte, che gli analisti effettuano meno frequentemente .

Prima dell'introduzione dell'Exploratory Data Analysis (EDA) da parte di John W. Tukey, le tecniche comunemente utilizzate per l'analisi dei dati erano principalmente orientate alla statistica inferenziale e ai test di ipotesi. Queste metodologie richiedevano che il ricercatore formulasse una o più ipotesi specifiche da testare attraverso l'uso di tecniche statistiche standardizzate, come test t, analisi della varianza (ANOVA), e la regressione lineare. L'obiettivo era confermare o respingere le ipotesi formulate sulla base di modelli statistici.

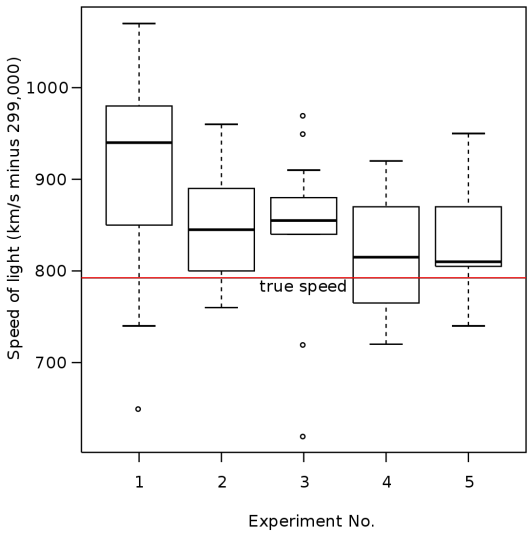
La novità introdotta da Tukey con l'EDA era l'enfasi sulla scoperta e l'esplorazione dei dati senza presupposti iniziali. Al contrario degli approcci tradizionali, l'EDA non iniziava con ipotesi specifiche, ma piuttosto invitava gli analisti a esaminare i dati in modo aperto e investigativo, utilizzando tecniche visive e metodi semplici per identificare pattern, tendenze e anomalie nei dati.

John W. Tukey, l'autore di "Exploratory Data Analysis", era un rinomato statistico americano. Ha dato contributi significativi al campo della statistica ed è meglio conosciuto per aver sviluppato il concetto di analisi esplorativa dei dati (EDA). L'innovazione di Tukey nell'EDA era la sua enfasi sull'uso di tecniche semplici per scoprire modelli e anomalie nei dati senza fare ipotesi preliminari. Questo approccio contrastava con i metodi statistici più tradizionali che si concentravano sulla conferma delle ipotesi esistenti.

L'EDA, come concepito da Tukey, si basava su metodi visivi e aritmetica diretta, consentendo agli analisti di esplorare liberamente i dati e identificare modelli interessanti. Questo rappresentava un significativo cambiamento rispetto agli approcci più rigidi di verifica delle ipotesi prevalenti all'epoca. Le tecniche principali spiegate nel suo elaborato del 1977 erano:

1. **Box Plot (Diagrammi a Scatola e Baffi):** Tukey ha introdotto il box plot come uno strumento per visualizzare la distribuzione dei dati. Questo grafico mostra la mediana, i quartili e gli eventuali outlier. È uno strumento semplice ma potente per identificare rapidamente le tendenze centrali, la dispersione e le anomalie nei dati.

2. **Stem-and-Leaf Plot (Grafico a Stelo e Foglia):** Questo è un altro strumento grafico utilizzato in EDA per visualizzare la distribuzione dei dati. È simile a un istogramma, ma conserva i dati effettivi, permettendo un'analisi più dettagliata. Permette agli analisti di vedere sia la forma generale della distribuzione dei dati sia i valori specifici.
3. **Scatter Plot (Grafico a Dispersione):** Tukey ha enfatizzato l'uso dei grafici a dispersione per esaminare le relazioni tra coppie di variabili. Questi grafici possono rivelare correlazioni, tendenze e gruppi di dati.
4. **Analisi dei Residui:** L'EDA di Tukey includeva anche l'analisi dei residui nei modelli statistici. Questo coinvolgeva l'esame dei residui (le differenze tra i valori osservati e quelli previsti da un modello) per identificare modelli non catturati dal modello stesso.



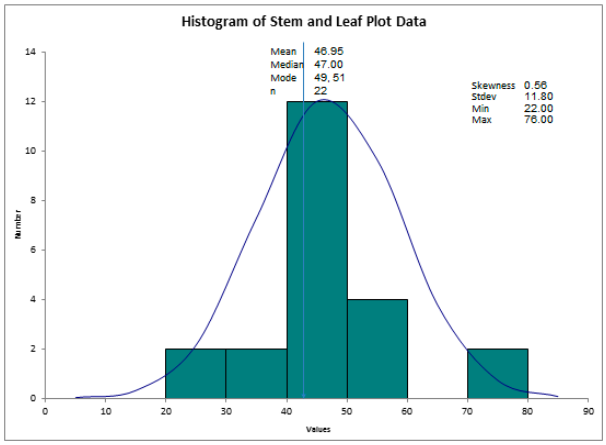


Figura 5 Grafico Stem and Leaf

Figura 6 Grafico Box Plot

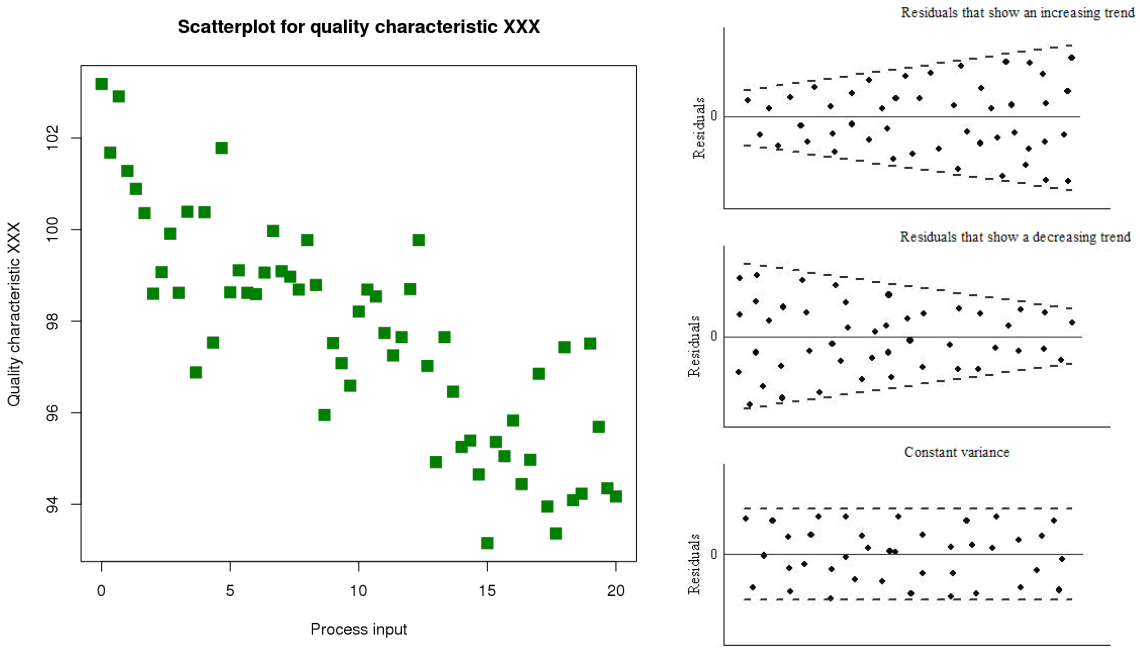


Figura 7 Analisi dei residui

Figura 8 Grafico Scatter

Queste tecniche dimostrano come l'EDA, secondo Tukey, fosse focalizzata sull'uso di strumenti visivi semplici e intuitivi per esplorare e comprendere i dati, piuttosto che su complessi calcoli statistici o test di ipotesi formali. Questo approccio ha permesso agli analisti di avvicinarsi ai dati in modo più flessibile e creativo, consentendo loro di scoprire informazioni (denominate in gergo “insights”) che potrebbero essere stati trascurati con metodi più tradizionali.

Nel 1977 la tecnologia era principalmente costituita da strumenti statistici di base, come carta e penna, calcolatrici e le prime tecnologie informatiche. I computer di quell'epoca erano meno potenti rispetto agli standard odierni, e gran parte dell'analisi dei dati veniva eseguita manualmente o con assistenza computazionale limitata. Questo contesto rende ancora più rilevante l'enfasi di Tukey su tecniche semplici e visivamente guidate, poiché forniva modi efficaci per analizzare i dati senza richiedere risorse computazionali avanzate.

Stato dell'arte dell'EDA

La EDA si avvale di diverse tecniche e strumenti per facilitare l'esplorazione dei dati e la generazione di intuizioni.[21] Tra le tecniche più usate e innovative, possiamo citare:

- I metodi interattivi, che permettono agli analisti di manipolare e modificare i dati e le visualizzazioni in tempo reale, adattandoli alle proprie domande e ipotesi. Alcuni esempi di strumenti interattivi sono Tableau, Vega-Lite e R Shiny.
- I metodi visivi, che sfruttano la capacità del cervello umano di percepire e interpretare le informazioni grafiche, facilitando la scoperta di pattern, trend e anomalie nei dati. Alcuni esempi di strumenti visivi sono ggplot2 , D3.js e matplotlib .
- I metodi automatici, che applicano algoritmi di apprendimento automatico o statistica ai dati, per estrarre informazioni rilevanti, come cluster, associazioni, classificazioni o predizioni. Alcuni esempi di strumenti automatici sono scikit-learn , TensorFlow e Weka .
- I metodi guidati, che forniscono suggerimenti o raccomandazioni agli analisti, basandosi su criteri di qualità, rilevanza o interesse, per aiutarli a scegliere i dati, le visualizzazioni o le analisi più appropriate. Alcuni esempi di strumenti guidati sono SeeDB , Data Voyager e Zenvisage .

La EDA si basa anche su teorie dell'inferenza grafica e statistica, che forniscono un quadro concettuale e metodologico per l'analisi dei dati. Tra le teorie più influenti, possiamo citare:

- Il framework bayesiano, che considera i dati come una fonte di evidenza per aggiornare le credenze sugli eventi o i parametri di interesse, tramite il teorema di Bayes. Il framework bayesiano permette di incorporare la conoscenza a priori, di quantificare l'incertezza e di confrontare modelli alternativi .

[21] Sahoo, K., Samal, A. K., Pramanik, J., & Pani, S. K. (2019). Exploratory data analysis using Python. *International Journal of Innovative Technology and Exploring Engineering*, *8*(12), 4727-4735.

- L'inferenza grafica, che propone di usare le visualizzazioni come strumenti per formulare e testare ipotesi sui dati, tramite il confronto con distribuzioni di riferimento, come il modello nullo o il modello predittivo. L'inferenza grafica si basa su principi di percezione visiva, come la discriminabilità e la comparabilità .

Exploratory Data Analysis con Pandas

L'Exploratory Data Analysis (EDA) rappresenta un passaggio fondamentale nel processo di analisi dei dati, in cui l'utilizzo di Pandas, una libreria Python, emerge come un metodo imprescindibile. L'EDA è cruciale per identificare pattern, comprendere le relazioni tra le caratteristiche e rilevare anomalie nei dati. Una componente essenziale dell'EDA è la visualizzazione, che consente di ottenere una visione più ampia e comprensiva del dataset.

L'EDA inizia con l'esame e la preparazione iniziale dei dati. Funzioni di Pandas come "info" e "describe" forniscono una panoramica generale, mostrando conteggi, medie, deviazioni standard, valori minimi e massimi. Questa fase iniziale è vitale per una prima comprensione delle caratteristiche dei dati. La gestione dei valori mancanti, un aspetto chiave della pulizia dei dati, è gestita attraverso funzioni specifiche come "isnull" e "sum", permettendo di affrontare uno dei problemi più comuni nell'analisi dei dati.

L'analisi delle correlazioni tra diverse variabili è un'altra tecnica fondamentale in EDA. Utilizzando la funzione ".corr()" di Pandas, è possibile esplorare le relazioni tra le colonne, scoprendo correlazioni forti o deboli. La visualizzazione di queste correlazioni, ad esempio attraverso mappe di calore, trasforma i dati complessi in informazioni facilmente interpretabili.

La trasposizione dei data frame per migliorare la visualizzazione grafica dei dati è un altro aspetto importante. La creazione di grafici specifici, come i box plot, è essenziale per identificare gli outlier, evidenziando così anomalie o particolarità nei dati.

Comprendere i tipi di dati in Pandas è fondamentale per l'EDA, specialmente nella selezione e nel filtraggio delle colonne in base ai loro tipi. Questo sottolinea l'importanza dell'EDA come primo passo cruciale nel processo di analisi dei dati, poiché pone le basi per analisi più dettagliate e prepara i dati per tecniche più avanzate, come il machine learning.

L'EDA con Pandas non è soltanto un processo iterativo e flessibile, adattabile agli obiettivi specifici dell'analisi, ma rappresenta anche una competenza chiave nel campo della Data Analytics. Con l'avanzamento delle tecnologie e delle tecniche di analisi dei dati, l'EDA è destinato a evolversi, integrando metodi avanzati di machine learning e intelligenza artificiale. L'a

umento dell'accessibilità a dataset sempre più grandi e complessi richiederà metodologie di EDA più sofisticate e scalabili.

L'EDA non è solo un insieme di tecniche, ma una filosofia di approccio ai dati che enfatizza l'importanza della curiosità, dell'esplorazione e dell'apertura mentale nell'analisi dei dati. In un mondo sempre più guidato dai dati, la capacità di condurre un'EDA efficace sarà sempre più preziosa, consentendo ai professionisti di adattarsi rapidamente ai cambiamenti e alle evoluzioni nel campo della Data Analytics.

Matrice di correlazione (Correlation Matrix)

Una matrice di correlazione è uno strumento statistico molto utile nelle analisi dei dati, specialmente nell'esplorazione e nella comprensione delle relazioni tra variabili in un dataset.

La matrice di correlazione mostra come ogni variabile nel dataset è correlata a tutte le altre variabili. Questo è particolarmente utile per identificare potenziali relazioni lineari tra coppie di variabili. Un coefficiente di correlazione vicino a 1 indica una forte correlazione positiva (cioè, quando una variabile aumenta, anche l'altra tende ad aumentare), mentre un coefficiente vicino a -1 indica una

forte correlazione negativa (quando una variabile aumenta, l'altra tende a diminuire). Un coefficiente vicino a 0 suggerisce che non c'è una forte correlazione lineare.

Per questo elaborato è stata utilizzata una scala di correlazione definita in questo modo:

- 0.9 a 1.0 (-0.9 a -1.0): Correlazione molto alta o molto forte.
- 0.7 a 0.9 (-0.7 a -0.9): Correlazione alta o forte.
- 0.5 a 0.7 (-0.5 a -0.7): Correlazione moderata.
- 0.3 a 0.5 (-0.3 a -0.5): Correlazione bassa o debole.
- 0 a 0.3 (0 a -0.3): Correlazione molto bassa, molto debole o nessuna correlazione.

In contesti come il machine learning, la selezione delle caratteristiche (feature selection) è cruciale per costruire modelli efficienti e performanti. La matrice di correlazione può aiutare a identificare e rimuovere le variabili altamente correlate tra loro, riducendo la dimensionalità dei dati senza perdere informazioni importanti. Questo processo è noto come riduzione della multicollinearità.

Prima di approfondire analisi complesse o costruire modelli predittivi, una matrice di correlazione fornisce una visione d'insieme delle relazioni tra variabili. Questo può aiutare a formulare ipotesi iniziali o a decidere quali variabili potrebbero meritare ulteriori indagini.

Spesso, le matrici di correlazione vengono accompagnate da heatmaps o altre forme di visualizzazione che rendono immediatamente evidente la forza e la direzione delle correlazioni tra le variabili. Queste visualizzazioni possono essere particolarmente utili durante le presentazioni o i report per comunicare le scoperte chiave a un pubblico.

Anche se la correlazione non implica causalità, identificare le variabili che sono fortemente correlate può suggerire aree per ulteriori studi causali. Per esempio, se due variabili sono fortemente correlate, potrebbe valere la pena indagare se esiste una relazione causale tra di loro attraverso studi o esperimenti più dettagliati.

Test di Dickey-Fuller per le serie temporali

Si consideri una serie temporale come una sequenza di dati raccolti in momenti successivi nel tempo. Ad esempio, la temperatura registrata ogni giorno in una città è una serie temporale. Mettendo questi dati in un grafico, con il tempo sull'asse orizzontale e la temperatura sull'asse verticale, si ottiene una rappresentazione visiva della serie temporale.

Una serie temporale è definita "stazionaria" se le sue proprietà statistiche, come la media e la varianza, rimangono costanti nel tempo. Questo significa che non importa in quale punto della serie si guardi, le caratteristiche generali (come la temperatura media nell'esempio) sono più o meno le stesse. Una serie temporale non stazionaria, al contrario, mostra tendenze o modelli che cambiano nel tempo, come un aumento costante della temperatura ogni anno.

Questo test è un metodo per controllare se una serie temporale è stazionaria o meno. In particolare, verifica la presenza di quello che si chiama "radice unitaria", un indicatore di non stazionarietà.

In termini semplificati, il test considera un modello matematico per la serie temporale e verifica se esiste una tendenza nel tempo che indica non stazionarietà. La formula di base del test Dickey-Fuller è:

Equazione 4 Formula del test Dickey Fuller

$$\Delta y_t = (\rho - 1) y_{t-1} + u_t = \delta y_{t-1} + u_t$$

Per questo elaborato è interessante considerare anche il test con costante e trend di tempo deterministico:

Equazione 5 Test Dickey Fuller con costante e trend deterministico

$$\Delta y_t = a_0 + a_1 t + \delta y_{t-1} + u_t$$

Dove:

- y_t è il valore della serie temporale al tempo t

- ρ è un coefficiente che corrisponde alla relazione tra il valore attuale della serie temporale e quello precedente
- y_{t-1} è il valore della serie temporale al tempo t-1
- $\Delta\, y_t$ è la differenza tra valori consecutivi della serie temporale, cioè $y_t - y_{t-1}$
- u_t è il termine di errore, cioè fattori casuali che non possono essere previsti
- a_0 è il termine costante
- $a_1 t$ è la tendenza

Alla fine del test di Dickey-Fuller il focus principale è sul valore stimato del coefficiente ρ. Questo valore è cruciale per determinare se la serie temporale esaminata è stazionaria o meno. Se $\rho = 1$ allora è presente una radice unitaria e la serie temporale non è stazionaria. Viceversa se il valore è statisticamente e significativamente diverso da 1 (e idealmente più piccolo) allora la serie sarà la stazionaria.

Stazionarietà

Una serie temporale stazionaria è quella le cui proprietà statistiche, come la media, la varianza, l'autocorrelazione, ecc., rimangono costanti nel tempo. Ci sono due tipi di stazionarietà:

- Stazionarietà forte: è un processo stocastico la cui distribuzione di probabilità congiunta incondizionata non cambia quando viene spostata nel tempo. Di conseguenza, parametri come la media e la varianza non cambiano nel tempo.
- Stazionarietà debole: è un processo in cui la media, la varianza, l'autocorrelazione sono costanti nel tempo.

Nell'ambito delle serie temporali, le caratteristiche osservate sono dipendenti dal tempo, ovvero le caratteristiche sono una funzione del tempo. I parametri statistici mutano i loro valori nel corso del tempo poiché vengono estratti direttamente dalle caratteristiche analizzate. Se la serie temporale

analizzata non è stazionaria, si verifica che le previsioni si discostano dai valori originali, aumentando l'errore di previsione. Questo accade perché non siamo in grado di prevedere le variazioni di questi parametri statistici, essendo essi funzioni del tempo.

Rendere una serie temporale stazionaria permette, in un certo senso, di annullare gli effetti che i parametri statistici possono avere sulle previsioni. Per verificare la stazionarietà di una serie temporale, ci avvaliamo del Test di Dickey Fuller Aumentato:

- Ipotesi Nulla: Presuppone che la serie temporale non sia stazionaria.
- Ipotesi Alternativa: Se l'ipotesi nulla viene rifiutata, allora la serie temporale è considerata stazionaria.

I risultati del Test di Dickey Fuller Aumentato includono:

- Statistiche del Test
- Valore-p
- #Lag Utilizzati
- Numero di Osservazioni Utilizzate
- Valore Critico (1%)
- Valore Critico (5%)
- Valore Critico (10%)

Per rifiutare l'Ipotesi Nulla e accettare che la serie temporale sia stazionaria, sono necessari due criteri:

1. Il Valore Critico (5%) deve essere maggiore delle Statistiche del Test (Critical Value (5%) > Test Statistic)
2. Il Valore-p deve essere inferiore a 0,05. (p-value < 0,05)

Questi criteri ci permettono di determinare con maggiore precisione la stazionarietà di una serie temporale, fondamentale per realizzare previsioni accurate e affidabili nel tempo.

```
def test_stationarity(timeseries):
    #Determing rolling statistics
    MA = timeseries.rolling(window=12).mean() #12 perché sono le
mensilità
    MSTD = timeseries.rolling(window=12).std()

    #Plot rolling statistics:
    plt.figure(figsize=(15,5))
    orig = plt.plot(timeseries, color='blue',label='Original')
    mean = plt.plot(MA, color='red', label='Rolling Mean')
    std = plt.plot(MSTD, color='black', label = 'Rolling Std')
    plt.legend(loc='best')
    plt.title('Rolling Mean & Standard Deviation')
    plt.show(block=False)

    #Perform Dickey-Fuller test:
    print('Results of Dickey-Fuller Test:')
    dftest = adfuller(timeseries, autolag='AIC')
    dfoutput = pd.Series(dftest[0:4], index=['Test Statistic','p-
value','#Lags Used','Number of Observations Used'])
    for key,value in dftest[4].items():
        dfoutput['Critical Value (%s)'%key] = value
    print(dfoutput)
```

Facendo un test ADF di esempio con un dataset sample, si ottengono e commentano i seguenti risultati:

Tabella 4 Tabella del test ADF

Output	Valore	Risultato Null Hypothesis	Commento
Test Statistic	0,82	Errore	> Critical Value (5%)
p-value	0,99	Errore	> 0.05
#Lags Used	13,00	Ok	
N°Observations Used	130,00		
Critical Value (1%)	-3,48		
Critical Value (5%)	-2,88		
Critical Value (10%)	-2,58		

L’ipotesi nulla è fallita dato che le sue due condizioni non sono state soddisfatte. Si conclude che la serie non è stazionaria e va resa tale. La stazionarietà è importante poiché le serie non stazionarie che dipendono dal tempo hanno troppi parametri da considerare quando si modella la serie temporale.

Il metodo diff() su python può facilmente convertire una serie non stazionaria in una serie stazionaria. “Diff” sta per differencing, cioè la sottrazione del valore t-1 al valore t+1.

Ecco il confronto del test ADF tra una serie non stazionaria da una serie stazionarizzata tramite il metodo diff():

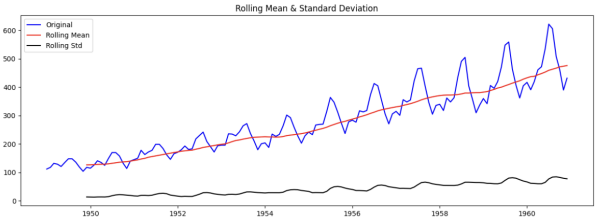


Figura 10 Serie non stazionaria

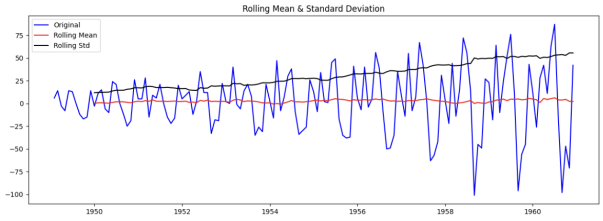


Figura 9 Serie stazionarizzata

Dove per la serie stazionaria si può notare un valore di media tendente allo zero e una deviazione che cresce costante nel tempo.

Autoregressione

Il concetto di autoregressione (AR), che rappresenta la correlazione tra un'osservazione e i suoi ritardi (osservazioni precedenti)[22], è stato introdotto e sviluppato da diversi statistici nel corso del tempo. Tuttavia, uno dei principali contributi a questo concetto viene attribuito a Sir George Udny Yule.

George Udny Yule, un statistico britannico, è spesso citato come una figura chiave nell'introduzione del modello autoregressivo. Il suo lavoro più influente in questo campo fu pubblicato nel 1927, quando presentò l'idea di modelli autoregressivi mentre studiava le serie temporali e le loro correlazioni seriali. La sua ricerca ha gettato le basi per la comprensione moderna dell'autocorrelazione nelle serie temporali e ha influenzato lo sviluppo di modelli statistici più avanzati, inclusi quelli che costituiscono la base per l'analisi ARIMA.

Uno dei suoi principali interessi era l'analisi delle serie temporali relative al numero di macchie solari, un argomento di grande interesse per gli astronomi e i climatologi dell'epoca. Yule ha esplorato l'idea che i valori in una serie temporale potrebbero dipendere da valori precedenti nella stessa serie, un concetto fondamentale per l'autoregressione. Questo approccio è stato rivoluzionario perché ha permesso di analizzare e modellare fenomeni complessi in cui i dati presentano dipendenze interne. Da allora, il concetto di autoregressione è diventato un pilastro fondamentale nell'analisi delle serie temporali e nel campo della statistica in generale.

Nel suo studio sulle macchie solari, Yule cercava di indagare sulla presenza di periodicità in serie temporali "perturbate", con particolare riferimento ai numeri di macchie solari di Wolfer. Le macchie solari sono fenomeni astronomici che esibiscono variazioni cicliche, in particolare un noto ciclo di

[22]Buxton, E., Kriz, K., Cremeens, M., & Jay, K. (2019, December). An auto regressive deep learning model for sales tax forecasting from multiple short time series. In 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA) (pp. 1359-1364). IEEE.

circa 11 anni. Yule si concentrò sull'analisi di queste variazioni per esplorare i metodi statistici nelle serie temporali.

Serie Temporali

Una serie temporale è caratterizzata da quattro principali componenti, ciascuno dei quali influisce sul comportamento complessivo dei dati nel tempo:

- Trend: Questo elemento rappresenta la tendenza di fondo o la direzione generale che i dati assumono nel lungo periodo. Un esempio classico di trend è l'incremento progressivo della temperatura media globale anno dopo anno, un fenomeno riconducibile agli effetti del riscaldamento globale.
- Stagionalità: Si riferisce ai pattern che si ripetono a intervalli regolari nel breve termine e che sono spesso associati alle stagioni dell'anno. Ad esempio, le vendite di maglioni che tendono ad aumentare specificamente durante i mesi invernali sono un chiaro indicatore di stagionalità.
- Variazioni Cicliche: Queste sono variazioni che si verificano su cicli più lunghi di un anno e non sono necessariamente legate alle stagioni. Un tipico esempio può essere osservato nei cicli economici, quali i trimestri commerciali (Q1, Q2, Q3 e Q4), che influenzano l'andamento di molte attività economiche.
- Irregolarità: Questo componente comprende le fluttuazioni imprevedibili o casuali che possono verificarsi nei dati. Eventi come terremoti o alluvioni sono esempi di irregolarità che introducono variazioni improvvise e non ripetitive nella serie temporale.

Per condurre un'analisi accurata di una serie temporale, è fondamentale riconoscere e, ove possibile, isolare questi componenti. Ciò consente di purificare i dati da queste influenze, stabilendo una base più solida per effettuare previsioni affidabili sul comportamento futuro della serie.

Per visualizzare una serie temporale si può definire un metodo personalizzato:

```
def tsplot(y, lags=None, figsize=(12, 7), style='bmh'):
    if not isinstance(y, pd.Series):
        y = pd.Series(y)

    with plt.style.context(style):
        fig = plt.figure(figsize=figsize)
        layout = (2, 2)
        ts_ax = plt.subplot2grid(layout, (0, 0), colspan=2)
        acf_ax = plt.subplot2grid(layout, (1, 0))
        pacf_ax = plt.subplot2grid(layout, (1, 1))

        y.plot(ax=ts_ax)
        p_value = sm.tsa.stattools.adfuller(y)[1]
        ts_ax.set_title('Time Series Analysis Plots\n Dickey-
Fuller: p={0:.5f}'.format(p_value))
        smt.graphics.plot_acf(y, lags=lags, ax=acf_ax)
        smt.graphics.plot_pacf(y, lags=lags, ax=pacf_ax)
        plt.tight_layout()
```

Capitolo 3 - MODELLI DI MACHINE LEARNING

Modello ARIMA

ARIMA è un modello statistico che serve a prevedere il futuro di una serie storica, cioè di una sequenza di dati che cambiano nel tempo. Per esempio, se si vuole sapere quante vendite farà un negozio nei prossimi mesi, si può usare un modello ARIMA basato sulle vendite passate.

ARIMA significa AutoRegressivo Integrato Media Mobile. Questi sono tre elementi che compongono il modello:

- AutoRegressivo: (AR) significa che il modello usa i valori passati della serie storica per prevedere i valori futuri. Ad esempio, se le vendite di gennaio dipendono da quelle di dicembre e di novembre, il modello è autoregressivo. "p" indica il numero di termini autoregressivi. Questa componente modella l'influenza che i valori passati hanno sul valore corrente.[23]

  Equazione 6 Parte Autoregressiva del modello

  - $Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_p Y_{t-\text{è}} + \in_1$

- Integrato: (I) significa che il modello trasforma la serie storica in modo da renderla più stabile e regolare. Ad esempio, se le vendite hanno una tendenza crescente o oscillano in modo ciclico, il modello le differenzia per eliminare questi effetti. "d" rappresenta l'ordine di differenziazione necessario a rendere la serie temporale stazionaria, ossia a stabilizzare la media della serie nel tempo.[24]
- Media Mobile: (MA) significa che il modello usa anche gli errori di previsione passati per correggere i valori futuri. Ad esempio, se le vendite di gennaio sono state più alte di quelle

[23] Maslim, M., & Arinanda, K. (2020). Motorcycle parts sales forecasting using auto-Regressive Integrated moving average model. International Journal of Computer Theory and Engineering,, 12(1), 28-31.
[24] Castiglione, J., Astroza, R., Azam, S. E., & Linzell, D. (2020). Auto-regressive model based input and parameter estimation for nonlinear finite element models. Mechanical Systems and Signal Processing, 143, 106779.

previste, il modello usa questo errore per aumentare le previsioni di febbraio. "q" è il numero di errori di previsione arretrati inclusi nell'equazione di previsione.[25]

Equazione 7 Parte a media mobile del modello

- $Y_1 = \alpha + \epsilon_1 + \Phi_1 \epsilon_{t-1} + \Phi_2 \epsilon_{t-2} + \ldots + \Phi_q \epsilon_{t-q}$

Perciò il modello ARIMA avrà come formula combinata tra la parte autoregressiva e quella a media mobile: $Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_p Y_{t-p} \epsilon_1 + \Phi_1 \epsilon_{t-1} + \Phi_2 \epsilon_{t-2} + \ldots + \Phi_q \epsilon_{t-q}$

Equazione 8 Somma della parte autoregressiva e a media mobile del modello

Semplificando a parole: Predicted Yt = Constant + Linear combination Lags of Y (up to p lags) + Linear combination of Lagged forecast errors (up to q lags)

Il concetto di "lags" si riferisce a valori precedenti nella stessa serie temporale, ovvero Yt-1, Yt-2, etc. Nel contesto delle serie temporali, considerare i "lags" significa guardare indietro nel tempo a valori precedenti per aiutare a prevedere i valori futuri.

In un modello ARIMA, i "lags" di Y rappresentano la parte "AR" (autoregressiva), mentre i lags degli errori di previsione rappresentano la parte "MA" (media mobile). Il numero di lags utilizzati (indicati come p per la parte AR e q per la parte MA) sono parametri chiave del modello e devono essere scelti con cura durante la fase di modellazione.

Il modello ARIMA ha quindi tre parametri che indicano quanto sono importanti questi elementi: p, d e q.

- p: corrisponde al valore di ritardo in cui il grafico della **Funzione di Autocorrelazione Parziale** (PACF) scende a zero o sotto lo zero per la prima volta. (Bisogna fare attenzione

[25] Darjani, N., & Omranpour, H. (2022). Comprehensive Learning Polynomial Auto-Regressive Model based on Optimization with Application of Time Series Forecasting. International Journal of Industrial Electronics Control and Optimization, 5(1), 43-50.

che esiste l'istante 0 nel grafico) Questo indica il numero di termini autoregressivi da includere nel modello.

- d: Indica quante volte è stato necessario usare il metodo .diff() sulla serie temporale per renderla stazionaria.
- q: È il valore di ritardo in cui il grafico della **Funzione di Autocorrelazione** (ACF) entra nell'intervallo di confidenza (segnalato dal bordo azzurro nel grafico) per la prima volta, suggerendo il numero di errori di previsione arretrati da includere. (Anche qui bisogna fare attenzione che a conteggiare le istanze si parte da 0)[26]

Ad esempio con un dataset sample, proiettando i grafici di Autocorrelazione e Autocorrelazione Parziale (PACF e ACF) della serie temporale stazionarizzata (dopo aver applicato un solo .diff()) si ottiene:

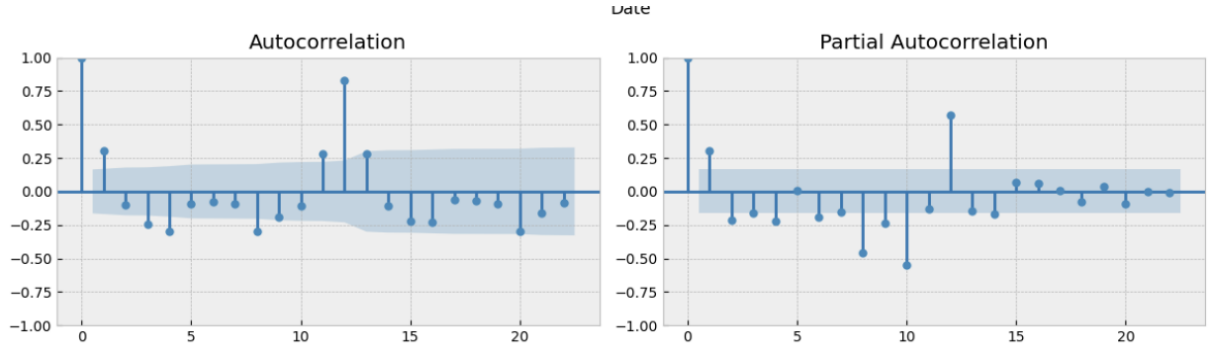


Figura 11 Confronto tra i grafici di autocorrelazione e correlazione parziale

Da queste funzioni, traiamo che i valori di p, d e q sono: (p,d,q) = (2,1,2).

Per costruire un modello ARIMA, bisogna scegliere i valori di p, d e q che si adattano meglio alla serie storica che si vuole prevedere. Su python esiste la funzione ARIMA(df.value, order = (x,y,z) in

[26] Rahimi, Z., Shafri, H. Z. M., & Norman, M. (2018). A GNSS-based weather forecasting approach using nonlinear auto regressive approach with exogenous input (NARX). Journal of Atmospheric and Solar-Terrestrial Physics, 178, 74-84.

cui x,y,z sono rispettivamente il numero di ritardi autoregressivi (AR), differenziazioni (I) e termini di media mobile (MA) che l'utente imposta.

L'applicazione di questi criteri nella selezione dell'ordine del modello ARIMA permette di strutturare un modello che si adatta efficacemente alle dinamiche specifiche della serie temporale in esame, migliorando così l'affidabilità delle previsioni generate.

La selezione manuale dei parametri può risultare onerosa e soggetta a errore, specialmente in presenza di serie temporali complesse. Di conseguenza, si ricorre spesso a metodi di selezione automatica, come l'uso di AUTOARIMA, che attraverso un processo iterativo esplora diverse combinazioni di p,d,q basandosi su criteri informativi come l'Akaike Information Criterion (AIC) o il Bayesian Information Criterion (BIC). Questi criteri valutano la qualità del fit del modello penalizzando l'aggiunta di parametri, al fine di prevenire il rischio di overfitting.[27]

La configurazione ottimale dei parametri p,d,q è determinante per la capacità del modello ARIMA di catturare accuratamente le dinamiche sottostanti la serie temporale. Un numero eccessivo di termini autoregressivi o di media mobile può portare a un modello sovradimensionato che, sebbene si adatti bene ai dati di addestramento, potrebbe non generalizzare efficacemente su nuovi dati. Al contrario, un modello sottodimensionato potrebbe non essere in grado di catturare le strutture complesse dei dati, risultando in previsioni imprecise.[28]

La scelta accurata di p,d,q richiede quindi un equilibrio tra complessità del modello e capacità di adattamento ai dati. Tecniche di validazione incrociata e la valutazione delle prestazioni del modello

[27] Pan, X., Qin, P., Li, Y., Xue, H., & Chen, W. (2024). Synthesizing coherent story with auto-regressive latent diffusion models. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 2920-2930).

[28] Khan, R. U., Hussain, S. M., Haq, A. U., Asif, M., Yousaf, M., Zafar, A., ... & Malghani, M. A. (2021, December). Forecasting Time Series COVID-19 Statistical Data with Auto-Regressive Integrated Moving Average and Box-Jenkins' Models. In 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP) (pp. 353-358). IEEE.

su set di dati di test forniscono un feedback essenziale per la verifica dell'adeguatezza dei parametri scelti.

I coefficienti del modello sono calcolati automaticamente da Python quando si esegue il metodo .fit() sul modello ARIMA. Questo processo implica un'ottimizzazione matematica per trovare i valori dei coefficienti che meglio si adattano ai tuoi dati di serie temporali. L'obiettivo è minimizzare una funzione di perdita, tipicamente la somma dei quadrati degli errori (SSE), attraverso metodi come il Maximum Likelihood Estimation (MLE) o altri algoritmi di ottimizzazione.[29]

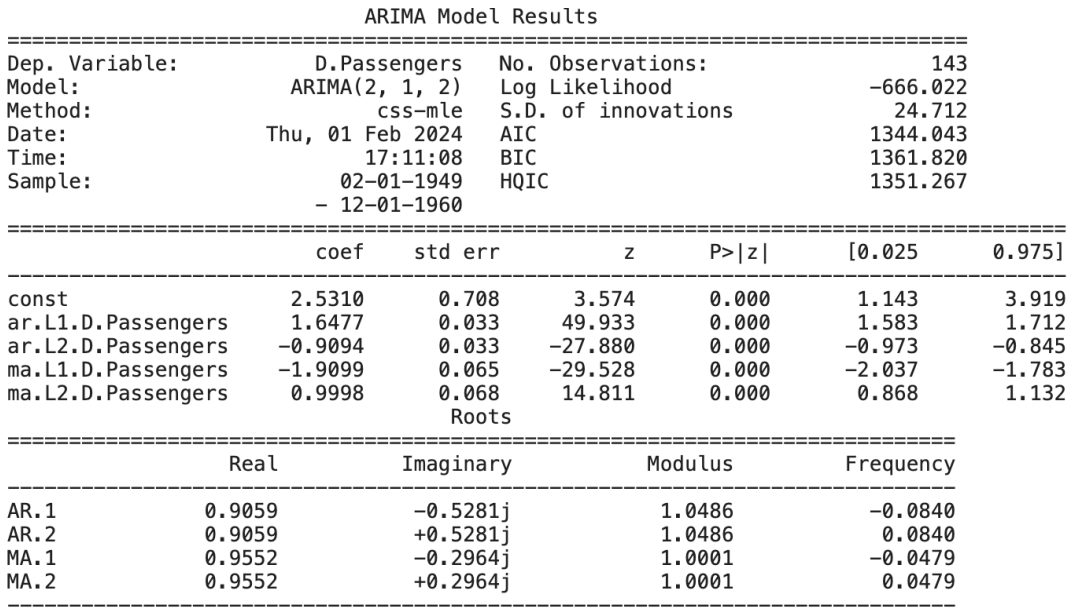
Esempio:

```
model = ARIMA(data['Passengers'],order = (2,1,2))
model_fit = model.fit()
print(model_fit.summary())
```

In questo caso data[‘Passengers’] si riferisce al dataset chiamato data alla colonna ‘Passengers’.

Questo metodo costruirà una variabile model e una variabile model_fit. La variabile model è la definizione iniziale e non adattata del modello ARIMA basata sui parametri scelti, mentre model_fit è il risultato dell'adattamento di quel modello ai dati, contenente il modello ottimizzato e dettagli diagnostici sull'adattamento.[30]

[29] Li, M., & Liu, X. (2020). Maximum likelihood least squares based iterative estimation for a class of bilinear systems using the data filtering technique. International Journal of Control, Automation and Systems, 18(6), 1581-1592.
[30] Fattah, J., Ezzine, L., Aman, Z., El Moussami, H., & Lachhab, A. (2018). Forecasting of demand using ARIMA model. International Journal of Engineering Business Management, 10, 1847979018808673.



```
                            ARIMA Model Results
==============================================================================
Dep. Variable:           D.Passengers   No. Observations:                  143
Model:                ARIMA(2, 1, 2)   Log Likelihood                -666.022
Method:                       css-mle   S.D. of innovations             24.712
Date:                Thu, 01 Feb 2024   AIC                           1344.043
Time:                        17:11:08   BIC                           1361.820
Sample:                  02-01-1949   HQIC                          1351.267
                       - 12-01-1960
==========================================================================================
                             coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------
const                     2.5310      0.708      3.574      0.000       1.143       3.919
ar.L1.D.Passengers        1.6477      0.033     49.933      0.000       1.583       1.712
ar.L2.D.Passengers       -0.9094      0.033    -27.880      0.000      -0.973      -0.845
ma.L1.D.Passengers       -1.9099      0.065    -29.528      0.000      -2.037      -1.783
ma.L2.D.Passengers        0.9998      0.068     14.811      0.000       0.868       1.132
                                    Roots
=============================================================================
                  Real          Imaginary           Modulus         Frequency
-----------------------------------------------------------------------------
AR.1            0.9059           -0.5281j            1.0486           -0.0840
AR.2            0.9059           +0.5281j            1.0486            0.0840
MA.1            0.9552           -0.2964j            1.0001           -0.0479
MA.2            0.9552           +0.2964j            1.0001            0.0479
-----------------------------------------------------------------------------
```

Figura 12 Risultati del modello ARIMA

Dopo aver adattato il modello ARIMA, si procede con l'In-Sample Forecasting.

In Sample Forecasting

Uno degli approcci fondamentali nell'analisi previsionale delle serie temporali è il Forecasting In-Sample.[31] Questo metodo prevede la generazione di valori previsti per i punti dati esistenti all'interno del dataset della serie temporale. Tale approccio si presenta analogo al formato di addestramento e test utilizzato nei problemi di regressione o classificazione, dove l'insieme dei dati viene suddiviso in due subset distinti: i dati di addestramento e i dati di test.

Per implementare il forecasting In-Sample, si divide il dataset in un insieme di addestramento e un insieme di test. Nell'esempio, gli ultimi 30 elementi (equivalenti ad altrettanti mesi) del dataset vengono riservati per il test, mentre il resto dei dati viene utilizzato per l'addestramento del modello.

La metodologia specifica impiegata in questo tipo di forecasting è quella del rolling forecast. Questo approccio prevede la previsione o la stima di un singolo valore alla volta. Una volta ottenuto il valore

[31] Lee, Y. K., Mammen, E., Nielsen, J. P., & Park, B. P. (2018). In-sample forecasting: A brief review and new algorithms. ALEA-Latin American Journal of Probability and Mathematical Statistics, 15, 875-895.

previsto, questo viene a sua volta utilizzato per aggiornare il modello prima di procedere alla previsione del valore successivo. Questa tecnica permette di adattare continuamente il modello alle dinamiche della serie temporale, migliorando potenzialmente l'accuratezza delle previsioni successive.

Per fare l'in-sample forecasting ci si avvale dei seguenti passaggi, descritti con commento iniziale:

```
# Definisce la dimensione del set di dati di addestramento
escludendo gli ultimi 30 dati per il test
size = int(len(data) - 30)

# Divide i dati in set di addestramento e test basandosi sulla
dimensione calcolata
train, test = data['Passengers'][0:size],
data['Passengers'][size:len(data)]

# Stampa l'intestazione per indicare l'inizio del report sul
modello ARIMA per previsioni interne al campione
print('\t ARIMA MODEL : In- Sample Forecasting \n')

# Inizializza la lista 'history' con i dati di addestramento per
utilizzarli nel modello ARIMA
history = [x for x in train]

# Inizializza una lista vuota per raccogliere le previsioni
predictions = []

# Itera per ogni punto nel set di test
for t in range(len(test)):

    # Crea e adatta il modello ARIMA ai dati storici
    model = ARIMA(history, order=(2,1,2))
    model_fit = model.fit(disp = 0)  # 'disp=0' nasconde l'output
del processo di ottimizzazione

    # Prevede il prossimo valore
    output = model_fit.forecast()
    yhat = output[0]  # Prende la previsione dal risultato del
metodo forecast()
    predictions.append(float(yhat))  # Aggiunge la previsione
alla lista delle previsioni

    # Ottiene il valore osservato dal set di test per il
confronto
```

```
    obs = test[t]
    history.append(obs)  # Aggiunge l'osservazione alla lista
'history' per il futuro adattamento del modello

    # Stampa la previsione e il valore osservato
    print('predicted = %f, expected = %f' % (yhat, obs))

# Definisce la dimensione del set di dati di addestramento
escludendo gli ultimi 30 dati per il test
size = int(len(data) - 30)

# Divide i dati in set di addestramento e test basandosi sulla
dimensione calcolata
train, test = data['Passengers'][0:size],
data['Passengers'][size:len(data)]

# Stampa l'intestazione per indicare l'inizio del report sul
modello ARIMA per previsioni interne al campione
print('\t ARIMA MODEL : In- Sample Forecasting \n')

# Inizializza la lista 'history' con i dati di addestramento per
utilizzarli nel modello ARIMA
history = [x for x in train]

# Inizializza una lista vuota per raccogliere le previsioni
predictions = []

# Itera per ogni punto nel set di test
for t in range(len(test)):

    # Crea e adatta il modello ARIMA ai dati storici
    model = ARIMA(history, order=(2,1,2))
    model_fit = model.fit(disp = 0)  # 'disp=0' nasconde l'output
del processo di ottimizzazione

    # Prevede il prossimo valore
    output = model_fit.forecast()
    yhat = output[0]  # Prende la previsione dal risultato del
metodo forecast()
    predictions.append(float(yhat))  # Aggiunge la previsione
alla lista delle previsioni

    # Ottiene il valore osservato dal set di test per il
confronto
    obs = test[t]
    history.append(obs)  # Aggiunge l'osservazione alla lista
'history' per il futuro adattamento del modello
```

```
    # Stampa la previsione e il valore osservato
    print('predicted = %f, expected = %f' % (yhat, obs))
```

Questo codice implementa una previsione passo-passo (step-ahead forecasting) con il modello ARIMA, dove il modello viene adattato a ogni nuovo punto dati man mano che diventa disponibile.[32]

L'output restituisce:

```
ARIMA MODEL : In- Sample Forecasting

predicted = 433.264961, expected = 491.000000
predicted = 478.354780, expected = 505.000000
predicted = 474.554046, expected = 404.000000
predicted = 367.686905, expected = 359.000000
predicted = 386.044942, expected = 310.000000
predicted = 300.551746, expected = 337.000000
predicted = 342.709374, expected = 360.000000
predicted = 374.434590, expected = 342.000000
predicted = 368.418730, expected = 406.000000
predicted = 427.293724, expected = 396.000000
predicted = 416.580521, expected = 420.000000
predicted = 431.952319, expected = 472.000000
predicted = 465.574556, expected = 548.000000
predicted = 516.133943, expected = 559.000000
predicted = 522.643316, expected = 463.000000
predicted = 407.122886, expected = 407.000000
predicted = 367.581870, expected = 362.000000
predicted = 349.941716, expected = 405.000000
predicted = 415.817567, expected = 417.000000
predicted = 443.408002, expected = 391.000000
predicted = 432.877307, expected = 419.000000
predicted = 467.788432, expected = 461.000000
predicted = 505.289436, expected = 472.000000
predicted = 505.208451, expected = 535.000000
predicted = 548.677941, expected = 622.000000
predicted = 603.229240, expected = 606.000000
predicted = 560.780687, expected = 508.000000
predicted = 458.394782, expected = 461.000000
predicted = 419.507946, expected = 390.000000
predicted = 373.834760, expected = 432.000000
```

[32] Duan, J., & Kashima, H. (2021). Learning to rank for multi-step ahead time-series forecasting. IEEE Access, 9, 49372-49386.

Dopodiché si procede con il confronto tra il grafico dei dati reali con i dati degli ultimi 30 mesi predetti dal modello:

```
# Crea una serie pandas con le previsioni e imposta l'indice come quello del set di test
predictions_series = pd.Series(predictions, index = test.index)

# Imposta la figura e gli assi per il plot con dimensioni specificate
fig,ax = plt.subplots(nrows = 1, ncols = 1, figsize = (15,5))

# Seleziona la prima e unica cella di subplot per il grafico
plt.subplot(1, 1, 1)

# Disegna il grafico della serie originale dei passeggeri con l'etichetta per i valori attesi
plt.plot(data['Passengers'], label = 'Expected Values')

# Aggiunge al grafico la serie delle previsioni con l'etichetta per i valori previsti
plt.plot(predictions_series, label = 'Predicted Values');

# Mostra la legenda nell'angolo in alto a sinistra
plt.legend(loc="upper left")

# Visualizza il grafico
plt.show()
```

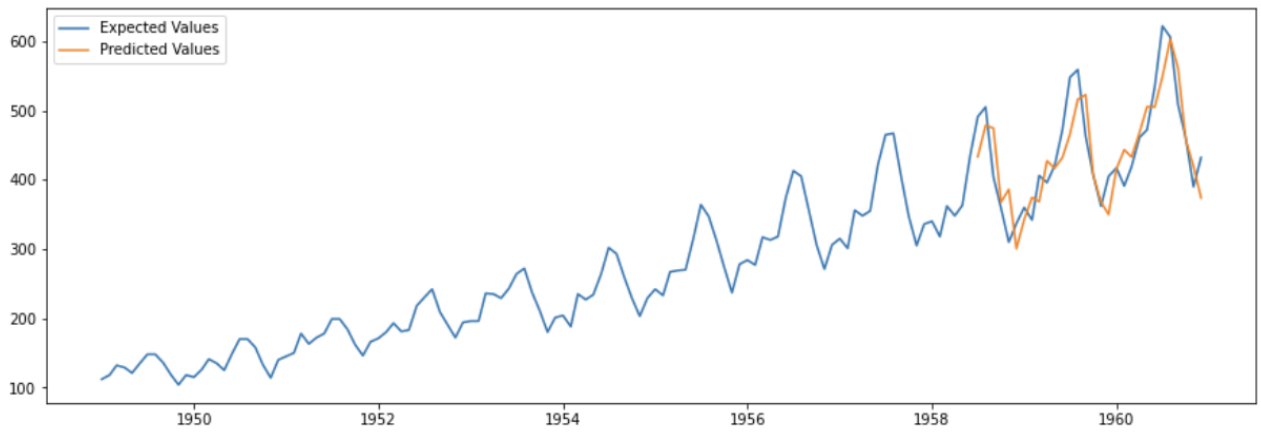


Figura 13 Esempio del modello ARIMA in funzione. Predizione del numero di passeggeri e confronto con dati reali

I valori predetti e reali sembrano di ordine di grandezza simile, tuttavia la precisione del modello si valuta calcolando l'errore quadratico medio nel Test RMSE.[33]

```
error = np.sqrt(mean_squared_error(test,predictions))
print('Test RMSE: %.4f' % error)
```

Il test restituisce un RMSE = 42,5167, che è abbastanza alto. Questo valore sarà poi confrontabile al RMSE del modello SARIMA.

Out of Sample Forecasting

Una tecnica complementare al forecasting in-sample, particolarmente rilevante per l'analisi previsionale, è il Forecasting Out-of-Sample.[34] Questo metodo si concentra sulla previsione di valori futuri non ancora osservati, estendendo l'indice temporale della serie oltre i dati disponibili. Il cuore di questo approccio consiste nell'impegnare i valori dell'indice temporale futuro per generare le previsioni, basandosi sul modello sviluppato con i dati storici.

Per attuare efficacemente il forecasting Out-of-Sample, si procede con la creazione di un nuovo dataframe che incorpora i valori futuri dell'indice temporale, mantenendo la stessa struttura di colonne del dataframe utilizzato per l'addestramento del modello. In pratica, si cerca di creare un ponte tra i dati passati e reali con i dati futuri, sconosciuti.

L'approccio utilizzato per il forecasting Out-of-Sample è il rolling method, simile a quello impiegato per il forecasting In-Sample.

[33] Le, T. T., Pham, B. T., Ly, H. B., Shirzadi, A., & Le, L. M. (2020). Development of 48-hour precipitation forecasting model using nonlinear autoregressive neural network. In CIGOS 2019, Innovation for Sustainable Infrastructure: Proceedings of the 5th International Conference on Geotechnics, Civil Engineering Works and Structures (pp. 1191-1196). Springer Singapore.

[34] Chu, B., & Qureshi, S. (2023). Comparing out-of-sample performance of machine learning methods to forecast US GDP growth. Computational Economics, 62(4), 1567-1609.

Gestione di Serie temporali non stazionarie

La stazionarietà è un presupposto fondamentale nella modellazione delle serie temporali, particolarmente nel contesto dei modelli ARIMA.[35] Una serie temporale si definisce stazionaria se le sue proprietà statistiche, come la media e la varianza, rimangono costanti nel tempo. Tuttavia, nella pratica, molte serie temporali presentano caratteristiche di non-stazionarietà, quali tendenze e stagionalità, che possono compromettere l'efficacia dei modelli predittivi. Questo capitolo esplora le strategie per gestire la non-stazionarietà nelle serie temporali, enfatizzando l'importanza di trasformazioni e tecniche di differenziazione.

Il primo passo nel trattamento delle serie temporali non stazionarie consiste nell'identificazione della loro natura non-stazionaria. Tecniche visive, come il grafico della serie temporale, possono offrire indicazioni preliminari. Tuttavia, test statistici, come il test aumentato di Dickey-Fuller (ADF), il test KPSS (Kwiatkowski-Phillips-Schmidt-Shin) e il test di Phillips-Perron (PP), forniscono conferme quantitative della presenza di radici unitarie, indicando non-stazionarietà.[36]

Una volta identificata la non-stazionarietà, è possibile applicare trasformazioni per stabilizzare la media e ridurre le variazioni nella varianza. Le trasformazioni comuni includono:

- Differenziazione: Consiste nel sottrarre il valore corrente della serie dal suo valore precedente. La differenziazione di primo grado spesso elimina le tendenze, mentre la differenziazione stagionale può gestire la stagionalità.
- Trasformazioni Logaritmiche e di Potenza: Utili per stabilizzare la varianza nei dati. La trasformazione logaritmica è particolarmente efficace per serie temporali che mostrano una crescita esponenziale.

[35] Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice (2nd ed.). OTexts. https://otexts.com/fpp2/

[36] Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: forecasting and control (5th ed.). Wiley.

- Metodi Box-Cox: Una generalizzazione delle trasformazioni logaritmiche e di potenza che può essere utilizzata per stabilizzare la varianza attraverso la scelta di un parametro di trasformazione ottimale.

Dopo l'applicazione di trasformazioni, è cruciale verificare nuovamente la stazionarietà della serie trasformata. Il ricorso ai test statistici precedentemente menzionati consente di confermare l'efficacia delle trasformazioni applicate. Inoltre, l'analisi delle funzioni di autocorrelazione (ACF)[37] e autocorrelazione parziale (PACF)[38] delle serie trasformate può offrire ulteriori evidenze della stazionarietà.

La gestione delle serie temporali non stazionarie attraverso trasformazioni appropriate e tecniche di differenziazione è un aspetto critico nella modellazione predittiva. Queste tecniche non solo facilitano il soddisfacimento dei presupposti di stazionarietà richiesti dai modelli ARIMA ma migliorano anche l'accuratezza e la generalizzabilità delle previsioni. Pertanto, l'abilità nell'identificare e trattare la non-stazionarietà si rivela essenziale per gli analisti di serie temporali che mirano a costruire modelli efficaci e affidabili.

Decomposizione Stagionale, Riduzione della Dimensionalità e Clustering di Serie Temporali

Nell'analisi delle serie temporali, è fondamentale distinguere tra le varie componenti che influenzano i dati, come la tendenza, la stagionalità e il rumore. La decomposizione stagionale è un metodo efficace per isolare e comprendere queste componenti, facilitando una migliore interpretazione e previsione dei dati. Parallelamente, la riduzione della dimensionalità e il clustering di serie temporali

[37] Maes, M. A., Breitung, K., & Dann, M. R. (2021, May). At issue: the Gaussian autocorrelation function. In International Probabilistic Workshop (pp. 191-203). Cham: Springer International Publishing.
[38] Yakubu, U. A., & Saputra, M. P. A. (2022). Time series model analysis using autocorrelation function (acf) and partial autocorrelation function (pacf) for e-wallet transactions during a pandemic. International Journal of Global Operations Research, 3(3), 80-85.

emergono come strumenti potenti per gestire la complessità e scoprire pattern intrinseci nei dati multivariati.[39]

La decomposizione stagionale si occupa di scomporre una serie temporale nelle sue componenti fondamentali: trend, stagionalità e residuo.[40] Il trend mostra la direzione di lungo termine dei dati, la stagionalità riflette le fluttuazioni periodiche, e il residuo comprende l'irregolarità non spiegata dalle prime due componenti. Metodi comuni per la decomposizione includono il modello additivo, dove si assume che le componenti si sommano linearmente, e il modello moltiplicativo, adatto per dati in cui la variazione stagionale aumenta proporzionalmente al livello della serie. Questa distinzione è cruciale per le serie temporali economiche, climatiche e di traffico, dove la stagionalità gioca un ruolo predominante.

Il processo di decomposizione non solo aiuta a visualizzare le componenti sottostanti dei dati,[41] ma fornisce anche una base solida per la modellazione predittiva, permettendo di adattare modelli specifici per ogni componente. Ad esempio, una volta isolata la stagionalità, si possono utilizzare modelli ARIMA stagionali per prevedere la serie temporale con maggiore precisione.

La riduzione della dimensionalità, in particolare attraverso l'analisi delle componenti principali (PCA), è fondamentale nell'analisi di serie temporali multivariate. La PCA trasforma un insieme di variabili correlate in un numero minore di variabili non correlate chiamate componenti principali. Questo metodo è particolarmente utile per identificare i pattern sottostanti nei dati, ridurre il rumore e semplificare i modelli successivi.

---

[39] Basatnia, N., Hossein, S. A., Rodrigo-Comino, J., Khaledian, Y., Brevik, E. C., Aitkenhead-Peterson, J., & Natesan, U. (2018). Assessment of temporal and spatial water quality in international Gomishan Lagoon, Iran, using multivariate analysis. Environmental Monitoring and Assessment, 190, 1-17.

[40] Hyndman, R. J., Wang, E., & Laptev, N. (2020). Large-scale unusual time series detection. IEEE Transactions on Knowledge and Data Engineering, 32(3), 569-579. https://doi.org/10.1109/TKDE.2018.2876855

[41] Bandara, K., Hyndman, R. J., & Bergmeir, C. (2021). MSTL: A seasonal-trend decomposition algorithm for time series with multiple seasonal patterns. arXiv preprint arXiv:2107.13462.

Nel contesto delle serie temporali, la PCA può essere impiegata per condensare l'informazione contenuta in molteplici serie parallele in un insieme ridotto di serie sintetiche che catturano la maggior parte della varianza. Questo approccio non solo facilita l'interpretazione dei dati complessi ma può anche migliorare l'efficienza computazionale dei modelli predittivi, riducendo il rischio di overfitting.

Il clustering di serie temporali rappresenta un altro strumento potente per l'analisi di dati complessi. Questa tecnica raggruppa serie temporali simili basandosi sulla loro forma, tendenza o schemi stagionali, senza richiedere la conoscenza a priori delle etichette di gruppo. Metodi comuni includono il clustering gerarchico, k-means e DBSCAN, ciascuno con i propri vantaggi nella gestione di vari tipi e dimensioni di dati.[42]

Il clustering facilita l'analisi comparativa delle serie temporali, permettendo di identificare comportamenti tipici o anomalie tra gruppi simili. Inoltre, questa tecnica può essere utilizzata per segmentare i dati in insiemi più omogenei, su cui applicare modelli predittivi specifici. Ad esempio, in ambito finanziario, il clustering può distinguere tra diversi tipi di comportamenti di mercato, mentre in meteorologia può identificare pattern climatici simili.

La decomposizione stagionale, la riduzione della dimensionalità e il clustering di serie temporali sono strumenti essenziali nell'arsenale dell'analista di dati. Queste tecniche non solo forniscono una comprensione più profonda delle dinamiche sottostanti ma anche migliorano l'accuratezza e l'efficienza dei modelli predittivi. Attraverso la decomposizione, è possibile isolare e modellare separatamente le componenti di tendenza e stagionalità. La PCA offre un metodo per semplificare e focalizzare l'analisi su aspetti significativi dei dati multivariati. Infine, il clustering rivela strutture e relazioni nascoste, guidando verso approcci di modellazione più mirati e informati. L'adozione

[42] Aghabozorgi, S., Shirkhorshidi, A. S., & Wah, T. Y. (2019). Time-series clustering - A decade review. Information Systems, 80, 16-38. https://doi.org/10.1016/j.is.2018.09.003

combinata di queste tecniche apre nuove prospettive nell'analisi delle serie temporali, spingendo i confini della previsione e dell'interpretazione dei dati.[43]

Modello AUTO ARIMA

Il modello AUTOARIMA, implementato nella libreria Python pmdarima, è uno strumento progettato per automatizzare il processo di identificazione dei migliori parametri per un modello ARIMA (Autoregressive Integrated Moving Average). Questo modello è ampiamente utilizzato nella previsione di serie temporali, sfruttando valori passati della serie stessa per prevederne il futuro. L'approccio AUTOARIMA mira a semplificare la selezione del modello ARIMA ottimale, attraverso la ricerca dei parametri più adatti senza la necessità di un'intensa configurazione manuale.

Il funzionamento di AUTOARIMA si basa sulla ricerca dei parametri ideali per il modello ARIMA, includendo i termini autoregressivi (p), i termini di differenziazione (d) per rendere la serie stazionaria, e i termini della media mobile (q). Si avvale di criteri informativi come AIC, BIC, HQIC o OOB per valutare e selezionare il modello migliore. Supporta sia la modellazione stagionale che non stagionale, regolando automaticamente il grado di differenziazione e integrando test per la stazionarietà e la stagionalità come il test KPSS o il test OCSB.

Sebbene AUTOARIMA non si basi direttamente sulla programmazione lineare, il suo processo di ottimizzazione dei parametri del modello può essere visto come un problema di ottimizzazione. L'obiettivo è minimizzare criteri di informazione o massimizzare la probabilità, selezionando la combinazione di parametri che offre le migliori prestazioni previsionali. Questo processo coinvolge l'esplorazione di vari modelli e la selezione di quello che fornisce il miglior equilibrio tra adattamento ai dati e complessità del modello, per evitare il sovradimensionamento.[44]

[43] Li, Y., Wang, N., Shi, J., Liu, J., & Hou, B. (2020). A survey on deep learning for time series data. Information Fusion, 63, 147-161. https://doi.org/10.1016/j.inffus.2020.05.012

[44] https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/

AUTOARIMA esegue una serie di calcoli per determinare i parametri ottimali, inclusi test per la stazionarietà e la stagionalità, e utilizza algoritmi di ottimizzazione come il Limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) per trovare i parametri ottimali del modello. Può eseguire una ricerca esaustiva o stepwise dei parametri p, d, q (e P, D, Q stagionali) entro limiti specificati dall'utente, valutando ciascuna combinazione in base al criterio informativo selezionato.[45][46]

AUTOARIMA facilita significativamente il processo di modellazione delle serie temporali, automatizzando la selezione dei parametri ARIMA e riducendo il tempo e l'expertise richiesti per costruire modelli accurati e affidabili. La sua relazione con l'ottimizzazione si manifesta nel processo di selezione del modello, che mira a trovare la configurazione ottimale che bilancia bene l'adattamento dei dati e la complessità del modello.

Di seguito vengono riportati alcuni dei comandi più utilizzati del gruppo ARIMA - AUTO ARIMA

Tabella 5 Comandi della libreria ARIMA

Comando	Descrizione
auto_arima()	auto_arima automatizza il processo di selezione del miglior modello ARIMA basandosi su vari criteri informativi (come AIC, BIC, ecc.). Esplora diverse combinazioni di parametri (p, d, q) e (P, D, Q) per i modelli stagionali, selezionando il modello che migliora la performance previsionale.
ARIMA()	Questa classe consente di specificare manualmente i parametri di un modello ARIMA e di adattarlo ai dati. È utile quando si hanno già informazioni sui parametri p, d, q che si vogliono utilizzare.

[45] https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.auto_arima.html

[46] Gupta, S., & Sharma, D. (2022). Prediction of COVID-19 spread in world using pandemic dataset with application of auto ARIMA and SIR models. International Journal of Critical Infrastructures, 18(2), 148-158.

plot_diagnostics()	Dopo aver adattato un modello ARIMA, questo comando può essere utilizzato per generare una serie di grafici diagnostici per valutare la qualità dell'adattamento del modello. Questi grafici includono il grafico dei residui, il grafico dell'istogramma dei residui con la curva di densità sovrapposta, il correlogramma dei residui e il grafico Q-Q.
seasonal_decompose()	Fornisce un'analisi dei componenti stagionali di una serie temporale, decomponendola in trend, stagionalità e residuo. È utile per comprendere meglio la struttura dei dati prima di modellarli.
ADFTest(), KPSS(), PPTest()	Questi comandi implementano test per la stazionarietà della serie temporale, un passaggio importante nella verifica delle ipotesi di base per la modellazione ARIMA.
ndiffs() e nsdiffs()	Queste funzioni aiutano a determinare il numero ottimale di differenziazioni necessarie per rendere una serie temporale stazionaria, rispettivamente per la differenziazione non stagionale e stagionale.
stepwise_fit()	Questo metodo è spesso utilizzato internamente dalla funzione auto_arima, ma può essere invocato anche direttamente per controllare più finemente il processo di selezione del modello stepwise.
plot_acf() e plot_pacf()	Funzioni per il plotting delle funzioni di autocorrelazione (ACF) e autocorrelazione parziale (PACF), utili per l'identificazione dei parametri p e q del modello ARIMA. Sono trattati nel precedente sotto capitolo.

Modello SARIMAX

Il modello SARIMAX (Seasonal Autoregressive Integrated Moving Average with eXogenous variables) è una generalizzazione del modello ARIMA che incorpora sia la stagionalità che variabili esogene nel processo di previsione delle serie temporali. Questo modello è particolarmente utile

quando i dati mostrano schemi stagionali o quando sono influenzati da fattori esterni oltre alle loro dinamiche interne.[47]

Il modello SARIMAX è utilizzato per modellare e prevedere serie temporali che presentano sia una stagionalità che l'influenza di variabili esogene. Può catturare la struttura autoregressiva, la differenziazione per rendere la serie stazionaria, l'effetto di medie mobili, e incorporare l'effetto di variabili esterne sulle serie temporali.[48]

Il modello combina componenti AR (Autoregressive), I (Integrated), MA (Moving Average) con componenti stagionali (S) e variabili esogene (X). I parametri stagionali (P, D, Q) funzionano in modo simile ai parametri non stagionali ma si applicano a cicli stagionali della serie temporale, mentre le variabili esogene vengono utilizzate per modellare l'effetto di fattori esterni sulla variabile di interesse

.

La principale differenza tra SARIMAX e ARIMA sta nella capacità del SARIMAX di modellare la stagionalità e incorporare variabili esogene. Mentre ARIMA si concentra su trend e stagionalità interni alla serie temporale, SARIMAX aggiunge la dimensione delle influenze esterne e dei modelli stagionali, rendendolo più versatile per una gamma più ampia di applicazioni.

Il modello SARIMAX estende la formula ARIMA includendo termini per gestire la stagionalità e le variabili esogene.[49] La configurazione del modello richiede la specificazione degli ordini non stagionali (p, d, q), degli ordini stagionali (P, D, Q), e della lunghezza del ciclo stagionale (s). Il modello calcola quindi i coefficienti per questi termini attraverso la stima dei minimi quadrati o altri

[47] Nontapa, C., Kesamoon, C., Kaewhawong, N., & Intrapaiboon, P. (2020). A new time series forecasting using decomposition method with SARIMAX model. In Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part V 27 (pp. 743-751). Springer International Publishing.

[48] https://www.statsmodels.org/dev/examples/notebooks/generated/statespace_sarimax_stata.html

[49] Elamin, N., & Fukushige, M. (2018). Modeling and forecasting hourly electricity demand by SARIMAX with interactions. Energy, 165, 257-268.

metodi di ottimizzazione per adattarsi al meglio ai dati storici. Questo processo include l'analisi delle autocorrelazioni e delle autocorrelazioni parziali, la differenziazione per ottenere la stazionarietà, e l'inclusione delle variabili esogene per migliorare la precisione delle previsioni.[50]

In pratica, SARIMAX offre un quadro robusto e flessibile per affrontare le sfide comuni nella previsione delle serie temporali, in particolare quando si tratta di dati con pattern stagionali pronunciati o quando si desidera valutare l'impatto di fattori esterni sulle previsioni.

Di seguito vengono riportati alcuni dei comandi più utilizzati di SARIMAX:

Tabella 6 Comandi della libreria SARIMAX

Comando	Descrizione
SARIMAX()	Questo è il costruttore del modello SARIMAX in statsmodels. Accetta parametri per specificare il modello, inclusi l'ordine ARIMA (p, d, q), l'ordine stagionale (P, D, Q, s), e le variabili esogene
fit()	Una volta specificato il modello SARIMAX, il metodo fit() viene utilizzato per adattare il modello ai dati. Questo metodo implementa la stima dei parametri del modello basata sui dati storici forniti
predict()	Dopo aver adattato il modello, predict() può essere utilizzato per generare previsioni future basate sul modello adattato. Il metodo accetta argomenti per specificare l'intervallo di tempo per cui si desidera generare previsioni
forecast()	Simile a predict(), il metodo forecast() è utilizzato per fare previsioni sul futuro immediato a partire dall'ultimo punto temporale noto nel dataset. Può essere particolarmente utile per fare previsioni passo dopo passo in applicazioni di forecasting in tempo reale
get_prediction() e get_forecast()	Questi metodi forniscono previsioni insieme a intervalli di confidenza. Sono utili per ottenere

[50] McHugh, C., Coleman, S., Kerr, D., & McGlynn, D. (2019, December). Forecasting day-ahead electricity prices with a SARIMAX model. In 2019 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1523-1529). IEEE.

	una stima dell'incertezza associata alle previsioni del modello
summary()	Fornisce un riepilogo dei risultati dell'adattamento del modello, inclusi i valori dei parametri stimati, le statistiche di adattamento e le informazioni sui test diagnostici. È utile per valutare la bontà di adattamento del modello e per la selezione del modello
aic, bic, hqic	Queste sono proprietà dell'oggetto risultato che forniscono valori per i criteri di informazione Akaike (AIC), Bayesian Information Criterion (BIC), e Hannan-Quinn Information Criterion (HQIC), utilizzati per confrontare la bontà di adattamento tra modelli diversi

Modello Prophet (Meta)

Nell'ambito dell'analisi predittiva, l'utilizzo di modelli avanzati quali Prophet, sviluppato da Facebook, permette la modellazione di serie temporali con un'efficacia notevole. Prophet si distingue per la sua capacità di gestire le serie temporali con tendenze non lineari, stagionalità annuale, settimanale e giornaliera, e giorni festivi, adattandosi pertanto a una vasta gamma di scenari applicativi.[51]

Con Prophet si procede alla previsione di dati di serie temporali attraverso un modello che adatta tendenze non lineari incorporando stagionalità annuale, settimanale e giornaliera, oltre agli effetti delle festività. Si adatta particolarmente bene a serie temporali con marcata stagionalità e con diversi

[51] Huang, Y. T., Bai, Y. L., Yu, Q. H., Ding, L., & Ma, Y. J. (2022). Application of a hybrid model based on the Prophet model, ICEEMDAN and multi-model optimization error correction in metal price prediction. Resources Policy, 79, 102969.

anni di dati storici. Prophet dimostra robustezza nei confronti di dati mancanti e variazioni di tendenza, gestendo generalmente bene anche i valori anomali.[52]

Di default, Prophet impiega un modello lineare per le sue previsioni. Quando si prevede la crescita, generalmente esiste un punto massimo raggiungibile, quale la dimensione totale del mercato o la popolazione totale. Questo limite è noto come capacità portante e le previsioni tendono a saturare una volta raggiunto questo punto.

Per identificare i punti di cambiamento, si specificano inizialmente numerosi potenziali punti di cambiamento dove è consentito modificare il tasso di crescita. Si applica quindi una priorità sparsa alle magnitudini di queste variazioni di tasso (equivalente alla regolarizzazione L1), il che significa che, sebbene Prophet consideri molti possibili punti di variazione, ne utilizza effettivamente il minor numero possibile. Di default, i punti di cambiamento vengono inferiti solo per l'80% iniziale della serie temporale, per permettere una proiezione adeguata della tendenza in avanti e per evitare il sovradattamento delle fluttuazioni verso la fine della serie. Questo parametro di default è adatto a molte situazioni, ma può essere modificato attraverso l'argomento changepoint_range. Se necessario, anziché affidarsi alla rilevazione automatica dei punti di cambiamento, è possibile specificarli manualmente con l'argomento changepoints, consentendo variazioni di pendenza solo in questi punti, con la stessa regolarizzazione sparsa di prima.

Prophet permette di effettuare previsioni utilizzando un modello di tendenza di crescita logistica, specificando una capacità portante. Di default, si adattano stagionalità additive, ovvero l'effetto della stagionalità viene sommato alla tendenza per ottenere la previsione.

[52] Armando, E., & Craparotta, G. (2019). A Meta-Model for Fashion Retail Category Sales Forecasting. In Business Models and ICT Technologies for the Fashion Supply Chain: Proceedings of IT4Fashion 2017 and IT4Fashion 2018 7 (pp. 79-93). Springer International Publishing.

Inoltre, Prophet include strumenti per la validazione incrociata delle serie temporali per misurare l'errore di previsione utilizzando dati storici. Questo processo si realizza selezionando punti di taglio nella storia e adattando il modello con i dati disponibili fino a quel punto. Si possono poi confrontare i valori previsti con quelli reali.[53]

Per modellare festività o eventi ricorrenti, si deve creare un dataframe dedicato. Questo deve avere due colonne (holiday e ds) e una riga per ogni occorrenza dell'evento, includendo tutte le occorrenze passate e future. Se tali eventi non si ripeteranno in futuro, Prophet li modellerà senza includerli nelle previsioni future.

Le stagionalità vengono stimate utilizzando una somma parziale di Fourier.[54] Per includere variazioni periodiche dovute alle festività, le serie di Fourier funzionano come regressori aggiuntivi nel modello. Queste serie, composte da funzioni trigonometriche, permettono di approssimare qualsiasi funzione periodica. Così facendo, è possibile catturare sia le festività fisse che quelle mobili, avendo un impatto significativo sulle serie storiche. Il numero di termini da includere nella serie di Fourier si determina attraverso un criterio basato sull'errore quadratico medio (MSE) tra previsioni e dati osservati, aggiungendo una penalità di regolarizzazione L2 per evitare sovradattamento, equilibrando così complessità e accuratezza del modello.

Di seguito vengono riportati alcuni dei comandi più utilizzati di Prophet:

Tabella 7 Comandi della libreria prophet

Comando	Descrizione
Prophet()	Questo è il costruttore per creare un'istanza del modello Prophet. Accetta vari parametri

[53] Vartholomaios, A., Karlos, S., Kouloumpris, E., & Tsoumakas, G. (2021, September). Short-term renewable energy forecasting in greece using prophet decomposition and tree-based ensembles. In International Conference on Database and Expert Systems Applications (pp. 227-238). Cham: Springer International Publishing.
[54] Taylor, S. J., & Letham, B. (2017). Forecasting at scale. PeerJ Preprints.

	opzionali che consentono di personalizzare il modello, come la flessibilità del trend e la stagionalità
fit()	Metodo per adattare il modello ai dati storici. I dati devono essere un DataFrame di pandas con due colonne: ds (la colonna della data) e y (la colonna della metrica da prevedere)
predict()	Dopo aver adattato il modello, predict() viene utilizzato per fare previsioni future. Il metodo richiede un DataFrame contenente una colonna ds con le date per cui si desidera fare previsioni. Restituisce un DataFrame con le previsioni e componenti come trend, stagionalità e festività
make_future_dataframe()	Questo metodo aiuta a generare un DataFrame contenente date future per le quali si desidera fare previsioni. È possibile specificare il numero di periodi futuri e la frequenza (ad esempio, giorni o mesi)
plot()	Metodo per visualizzare il risultato delle previsioni di Prophet. Mostra i dati storici, la previsione e gli intervalli di incertezza sia per il trend che per la componente stagionale
plot_components()	Fornisce una scomposizione grafica delle componenti del modello, come trend, stagionalità annuale, settimanale e l'effetto delle festività
add_seasonality()	Permette di aggiungere stagionalità personalizzate al modello, oltre a quelle annuali, settimanali e giornaliere predefinite. È utile quando i dati presentano cicli stagionali non standard
add_country_holidays()	Metodo per includere automaticamente le festività di un determinato paese come componenti regressori del modello. Aiuta a gestire gli effetti delle festività sulle previsioni
add_regressor()	Consente di aggiungere variabili esogene al modello, che possono migliorare la qualità delle previsioni includendo informazioni aggiuntive note in anticipo

Capitolo 4 - MODELLO PER LA PREVISIONE DI LEADS SETTIMANALI E CAPACITY SETTIMANALE RICHIESTA. IL CASO FACILE.IT

Raccolta dei dati e costruzione del dataset

La struttura del dataset di Facile.it fornisce una ricchezza di informazioni cruciali per comprendere le dinamiche dei lead nel settore dei servizi finanziari. Di seguito viene riportato un comprensivo dei dati a disposizione.

Tabella 8 Descrizione del Dataset

DATO	NUMEROSITA' DATASET (SU 24 MESI)	DESCRIZIONE
leads	2,8 Milioni	Numero di leads totale in tale data
googlecpc	1,5 Milioni	Leads provenienti da Google Ads (Adwords)
googleorganic	345.000	Leads provenienti da ricerche organiche
CRM	344.000	Leads provenienti da campagne su CRM (Es. Email Marketing)
fbcpc	310.000	Leads provenienti da facebook ads
direct	277.000	Leads provenienti da utenti che hanno cercato l'url del sito direttamente
altricpc	52.000	Leads provenienti da altre campagne paid
referral	12.000	Leads provenienti da altri referral
altriorganic	10.000	Leads provenienti da altre fonti organiche
acqprimacasa	1,4 Milioni	Leads che richiedono mutuo per l'acquisto di prima casa

surroga	420.000	Leads che richiedono la surroga di un mutuo esistente
acqsecondacasa	112.000	Leads che richiedono mutuo perl'acquisto di seconda casa
ristrutturazione	66.000	Leads che richiedono mutuo per ristrutturazione
liquidità	32.000	Leads che richiedono mutuo per denaro su conto corrente
acqristrutt	22.000	Leads che richiedono mutuo per acquisto e ristrutturazione
leadsdestkop	520.000	Leads che effettuano la richiesta da tablet o pc
leadsmobile	2,32 Milioni	Leads che effettuano la richiesta da mobile
complete	318.000	Leads che compilano sia il primo che il secondo form
parziali	2,52 Milioni	Leads che compilano solo il primo form

Pulizia e Preparazione dei Dati

La pulizia e la preparazione dei dati sono passaggi fondamentali in qualsiasi processo di analisi dei dati. Come affermato da Witten et al. (2016), la qualità dei dati è critica per l'affidabilità dei risultati ottenuti. L'imputazione dei valori mancanti, la correzione degli errori e la standardizzazione dei formati sono tutti aspetti cruciali in questo processo.[55]

La standardizzazione dei formati è altrettanto importante. Date e valori monetari devono essere uniformati per garantire la coerenza nell'analisi. Ad esempio, le date possono essere convertite in un

[55] Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques (4th ed.). Burlington, MA: Morgan Kaufmann.

formato standard 'YYYY-MM-DD', mentre i valori monetari possono essere arrotondati al centesimo più vicino per evitare discrepanze dovute a differenze di arrotondamento.

Innanzitutto è stato impostato l'orizzonte temporale.

```
prediction_window = 28 #testing window (4 weeks of data)
forecast_window = 7 # forecasting window (1 week)
window = prediction_window + forecast_window #prediction +
forecasting window
```

Il prediction window setta il test set, ciò significa che tutto ciò che è indietro di 28 giorni sarà il training set. In questo caso si setta anche la forecast window, cioè i giorni nel futuro di cui il modello prevederà le leads.

Subito dopo si effettuano le indicizzazioni sulle date, cambiando anche il nome delle colonne in un formato comprensibile al modello. (data -> ds, leads -> y)

```
df_p = df['leads'].reset_index().copy()
df_p = df_p.rename(columns={'data': 'ds', 'leads': 'y'})
df_p.set_index('ds', inplace=True)


# Ordina il DataFrame per l'indice corretto
df_p.sort_index(inplace=True)
df_p.head()

# Primo DataFrame: 'googlecpc'
df_1 = df['googlecpc'].reset_index().copy()
df_1 = df_1.rename(columns={'data': 'ds', 'googlecpc': 'y'})
df_1.set_index('ds', inplace=True)
df_1.sort_index(inplace=True)
df_1.head()
```

In questo pezzo del codice si mostra solo il setup per il data frame originale e quello di google cpc per accorciare, ma naturalmente ci sono le altre 7 fonti di dati. In generale, ciò che si vuole fare in

questa fase è preparare due tipi di dataset: quello che verrà dato a prophet senza suddivisione per fonte, e quello per diviso per ogni fonte che verrà sommato interamente.

Exploratory Data Analysis

L'analisi esplorativa dei dati viene fatta per acquisire una maggior chiarezza del dataset.

Rispetto alle analisi portate avanti in questo sotto-capitolo, si è deciso di analizzare la correlazione fra 4 divisioni di categorie di leads:

1. Divisione per fonte (Google Ads, Organico, CRM/Email Marketing, etc.)
2. Divisione per finalità (Acquisto prima casa, surroga, acquisto seconda casa, etc.)
3. Divisione per dispositivo utilizzato (Destkop[56], Mobile)
4. Divisione per leads complete e parziali[57]

Si può fare immediatamente una prima analisi rispetto alla fonte da cui provengono le leads. Si denota immediatamente una maggioranza di leads provenienti da Google Ads, in molti dei casi rappresentando oltre il 50% delle leads totali.

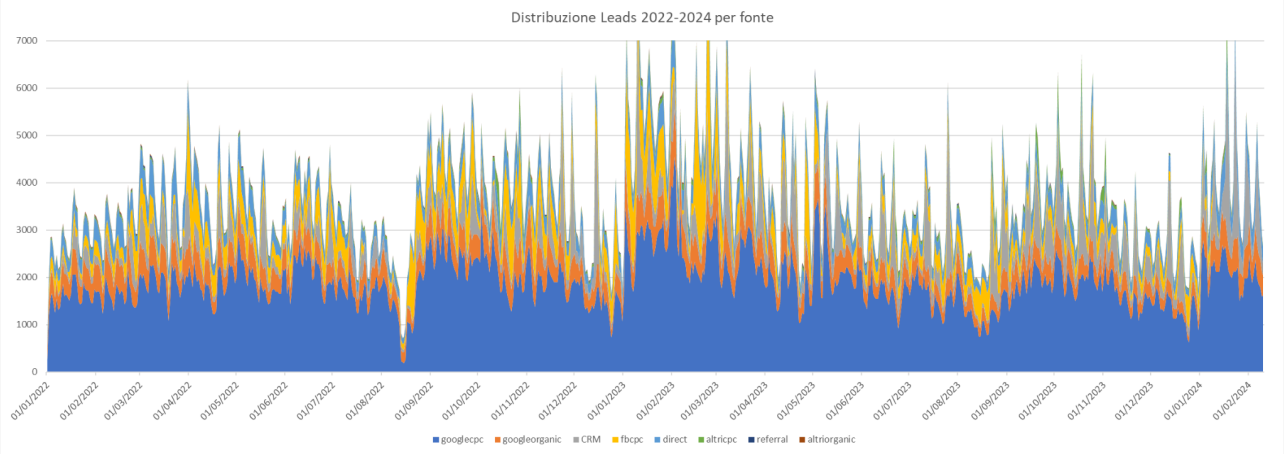


Figura 14 Provenienza delle leads divise per fonte

[56] Per destkop si intende utenti da tablet e pc

[57] Per la definizione di leads complete e parziali si rimanda al capitolo "Il contesto bancario"

In ordine di numerosità si hanno Google Ads, Google Organic e CRM. Per misurare l'impatto che hanno queste tre fonti, si è calcolata l'incidenza combinata di questa "top 3 di fonti":

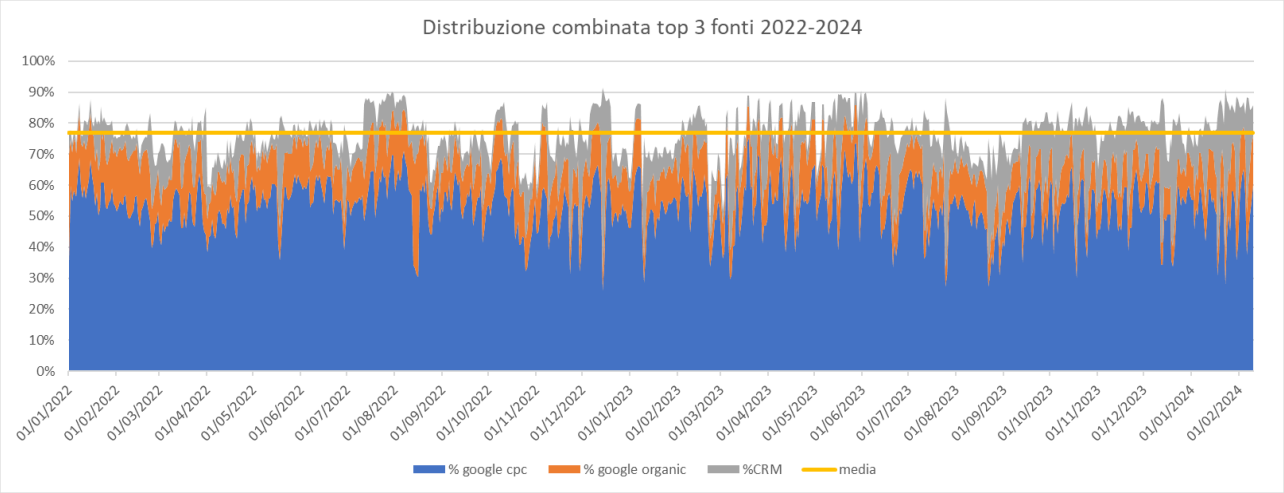
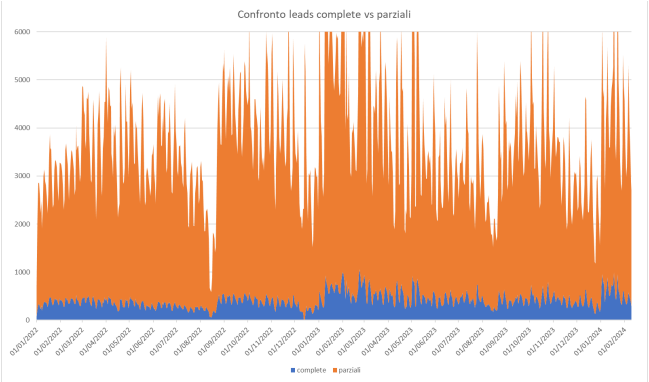


Figura 15 Numerosità delle leads provenienti dalle prime 3 fonti in percentuale

Si ottengono delle informazioni interessanti. La media della sommatoria di queste tre fonti nell'arco degli ultimi due anni è del 77% (linea gialla). Si può concludere che i canali di acquisizione clienti sono concentrati, e non frammentati. Questo porta l'azienda ad avere un'entrata di clienti stabile nel tempo, ma al contempo soggetta a una sensibilità maggiore al rischio di problemi in questi canali. Ad esempio, la recente stretta di Google riguardo al filtro anti spam[58] può costituire un pericolo valevole per il canale "CRM" dell'azienda, che fa leva sull'Email Marketing.

[58] https://blog.google/products/gmail/gmail-security-authentication-spam-protection/



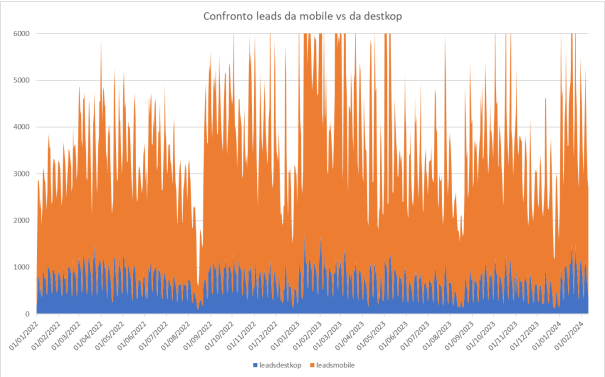


Figura 16 Confronto delle leads complete e parziali

Figura 17 Confronto leads da mobile e da destkop

In seguito a questi due grafici è possibile notare che la maggioranza di chi usufruisce del servizio di facile.it lo fa da dispositivo mobile, così come sempre la stessa maggioranza si ferma a fare il primo form, a visualizzare il prezzo dei mutui disponibili secondo le loro anagrafiche, ma senza completare anche il secondo form. In altre parole, la maggioranza delle leads effettua una richiesta 'parziale' rispetto a una 'completa'.

Per la previsione di leads nella settimana futura, l'analisi esplorativa dei dati è stata fatta per analizzare la stagionalità del numero di leads durante gli anni, i mesi, i giorni festivi e i giorni della settimana. Per costruire questa analisi è stato utilizzato il modello Prophet.

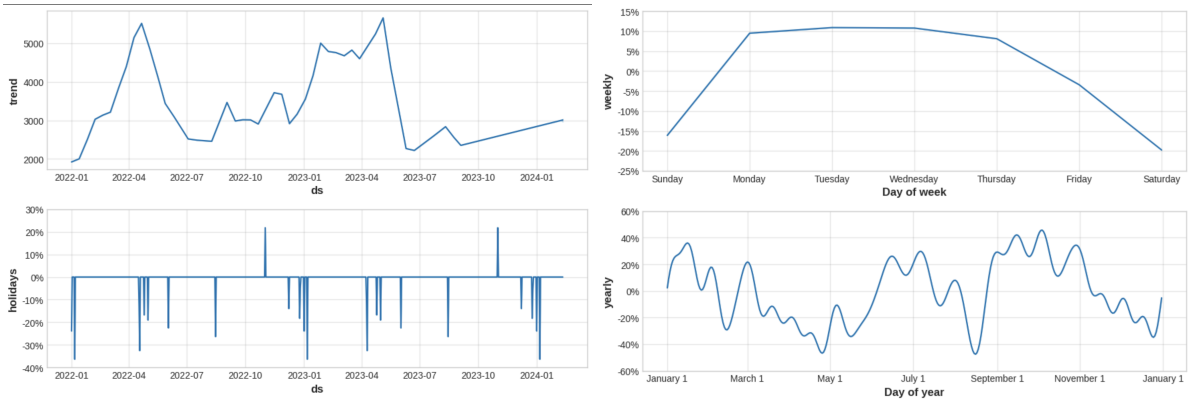


Figura 18 Grafici macrotrend su più anni e nei giorni di festività

Figura 19 Grafici trend settimanale e mensile

Dal primo e l'ultimo grafico è possibile visualizzare il macro trend stagionale dei vari mesi in un arco di 24 mesi. In particolare, nel primo grafico è possibile visualizzare il numero assoluto delle leads e si notano dei picchi nei mesi primaverili, mentre nell'ultimo grafico si vede lo scostamento percentuale rispetto alla media, visualizzando dei picchi negativi durante la stagione estiva inoltrata. Questo si riflette nella realtà nella volontà dei richiedenti mutuatari di fare più richiesta di mutuo in determinati periodi dell'anno.

Tipicamente, le festività influenzano negativamente le vendite.[59] Infatti, la maggior parte delle festività italiane influenza negativamente il numero delle leads, a simboleggiare che gli italiani tendono a dedicare le vacanze ad altro rispetto al mutuo. Si nota invece un picco positivo per quanto riguarda la festività "Tutti i santi" in entrambi gli anni nel campione, sopra il 20%, di richieste in più.

Durante la settimana, i giorni con maggior numero di leads sono martedì e mercoledì in prima posizione, seguiti dal lunedì e il giovedì in seconda posizione. Termina la classifica il weekend, che vede in modo piuttosto costante una diminuzione di richieste e "interesse" sulla richiesta di un mutuo.

L'analisi della correlazione è stata fatta mediante la creazione di una heatmap, o mappa di calore.

[59] Orlando, G., & Bufalo, M. (2023). Time series forecasting with the CIR# model: from hectic markets sentiments to regular seasonal tourism. Technological and Economic Development of Economy, 29(4), 1216-1238.

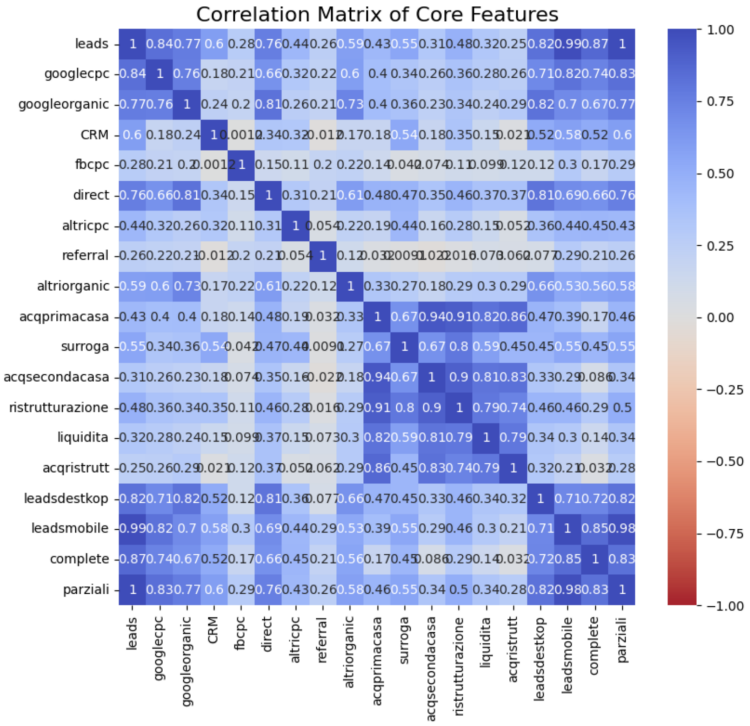


Figura 20 Matrice di correlazione Heatmap

Per analizzare i risultati è stata utilizzata la scala descritta nel capitolo 2 “Strumenti di analisi dati”.

Dalla heatmap si notano 3 macro cluster di correlazioni forti o moderatamente forti:

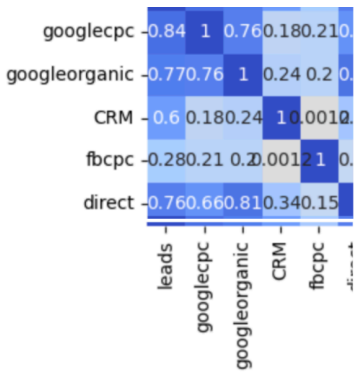


Figura 21 Correlazione tra fonti delle leads. Prima parte

La correlazione tra le fonti, dove si nota che la crescita delle leads è, in ordine di intensità, dipendente da Google Ads (googlecpc), Google Organic, Direct[60], CRM, e così via

[60] Una lead direct significa che è andato direttamente sul sito di facile.it invece che entrarci da un qualsiasi motore di ricerca

La correlazione tra il dispositivo utilizzato e la fonte dove si nota che:

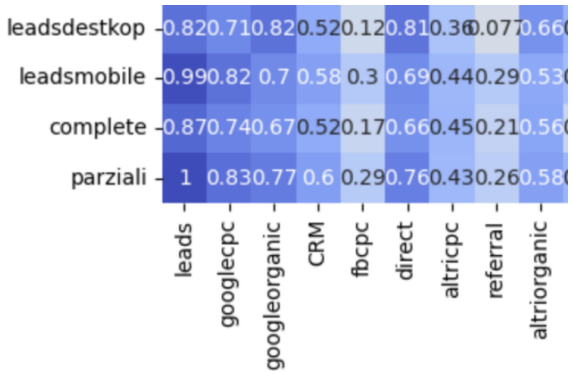


Figura 22 Correlazione tra fonti. Seconda parte

- Gli utenti su dispositivo mobile sono più numerosi da (in ordine decrescente): google ads, google organico e direct
- Gli utenti su dispositivo destkop sono più numerosi da (in ordine decrescente): google organico, direct e google ads
- Gli utenti da CRM sono relativamente maggiori su mobile che destkop, che presumibilmente significa che leggono, aprono e effettuano la CTA[61] più spesso su mobile che destkop

[61] CTA: acronimo di Call To Action, in questo contesto esprime il click che riporta il lettore della mail a entrare su facile.it ed effettuare la richiesta

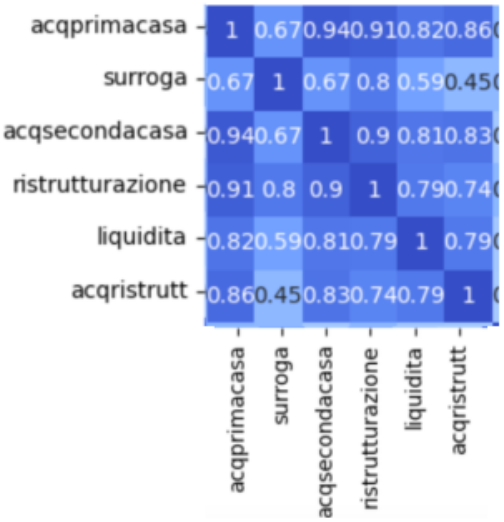


Figura 23 Correlazione tra finalità delle leads

La correlazione tra le varie finalità degli utenti richiedenti il mutuo:

- La crescita di leads di acquisto è correlata solo moderatamente alla crescita di leads di surroga. Questo è l'effetto dovuto al fatto che la maggior parte delle campagne surroga viene fatta attraverso l'Email Marketing, aumentando le leads surroga, ma non quelle di acquisto prima casa.

- Una crescita delle richieste di acquisto prima casa è correlata fortemente a una crescita di richieste di una seconda casa, ristrutturazione, liquidità e acquisto ristrutturazione, ma non. Il risultato così alto significa che gli utenti di Facile.it sono perlopiù generalisti e non rappresentano invece una nicchia di mercato che utilizza il portale solo al fine di comprare un mutuo per una finalità specifica (tolta la finalità di surroga)
- Una forte correlazione c'è tra la ristrutturazione e l'acquisto della seconda casa, e una chiave di lettura potrebbe essere che quando la domanda di mercato per i mutui per ristrutturazione aumenta, aumenta anche la domanda di mercato per i mutui per l'acquisto di una seconda casa. Questo può essere un dettaglio rilevante sia per le aziende nel mercato immobiliare, che per le aziende che si occupano di ristrutturazione.

Addestramento del modello Prophet

Si comincia con l'addestramento del modello Prophet. Nel seguente capitolo verranno spiegati passo per passo la funzione di ogni codice che ha portato alla soluzione finale. Le parti ripetitive dei codici, dovute all'operazione di suddivisione delle fonti dei leads, verranno troncati nell'elaborato in modo da non allungarne la lettura.

```
# Crea e addestra il primo modello Prophet per df_1
prophet_model1 = Prophet(n_changepoints=50,
seasonality_mode='multiplicative', changepoint_prior_scale=10)
prophet_model1.fit(df_1)

# Crea e addestra il secondo modello Prophet per df_2
prophet_model2 = Prophet(n_changepoints=50,
seasonality_mode='multiplicative', changepoint_prior_scale=10)
prophet_model2.fit(df_2)
```

Il comando Prophet(n_changepoints=50, seasonality_mode='multiplicative', changepoint _prior_scale=10) inizializza un modello di Prophet specificando tre parametri chiave: n_changepoints, seasonality_mode e changepoint_prior_scale. Ciascuno di questi parametri ha un ruolo cruciale nella configurazione del modello e influisce direttamente sulla sua capacità di apprendimento e predizione.

n_changepoints: Questo parametro determina il numero di potenziali cambiamenti nella tendenza che il modello può identificare lungo la serie temporale. Un valore di 50 indica che il modello considererà fino a 50 punti, distribuiti uniformemente lungo la serie temporale, dove potrebbero verificarsi cambiamenti significativi nella tendenza. La scelta di un valore più alto aumenta la flessibilità del modello nel rilevare variazioni di tendenza, a costo però di un maggiore rischio di overfitting, poiché il modello potrebbe adattarsi eccessivamente alle fluttuazioni casuali dei dati.

seasonality_mode: Il parametro seasonality_mode può assumere due valori: additive o multiplicative. Nel caso specifico, impostando il valore su multiplicative, si assume che l'effetto stagionale sulla serie temporale si moltiplichi man mano che il livello della tendenza cambia. Questo approccio è particolarmente adatto a serie temporali in cui l'ampiezza delle fluttuazioni stagionali cresce o decresce proporzionalmente al livello della serie temporale stessa. Al contrario, una modalità additive sarebbe stata appropriata per serie temporali in cui le fluttuazioni stagionali rimangono costanti nel tempo, indipendentemente dal livello della tendenza.

changepoint_prior_scale: Questo parametro regola la flessibilità del modello nel modificare la sua tendenza ai punti di cambiamento. Un valore di 10 indica una forte predisposizione del modello a cambiare la sua tendenza, facilitando l'adattamento a cambiamenti repentini nei dati. Tuttavia, un valore elevato può anche portare a un'eccessiva reattività del modello a variazioni minori, potenzialmente interpretando il rumore come cambiamenti significativi di tendenza. La scelta ottimale di questo parametro dipende dalla volatilità dei dati: per serie temporali più stabili, un valore inferiore potrebbe essere preferibile per evitare overfitting, mentre per dati altamente volatili, un valore più alto potrebbe essere necessario per catturare adeguatamente la dinamica dei cambiamenti.

Successivamente, il comando prophet_model1.fit(df_1) addestra il modello precedentemente configurato sui dati contenuti in df_1. Durante questa fase, il modello apprende le caratteristiche intrinseche della serie temporale - includendo tendenza, stagionalità e effetti dei giorni festivi - basandosi sui parametri specificati e sui dati forniti.[62]

[62] https://facebook.github.io/prophet/

```
# Configura e addestra il primo modello Prophet per df_p
prophet_model.add_seasonality('weekly', period = 7, fourier_order
= 5)
prophet_model.add_seasonality('yearly', period = 365,
fourier_order = 25)
prophet_model.add_country_holidays(country_name='Italy')
prophet_model.fit(df_p)

# Configura e addestra il primo modello Prophet per df_1
prophet_model1 = Prophet(n_changepoints=50,
seasonality_mode='multiplicative', changepoint_prior_scale=10)
prophet_model1.add_seasonality(name='weekly', period=7,
fourier_order=5)
prophet_model1.add_seasonality(name='yearly', period=365,
fourier_order=25)
prophet_model1.add_country_holidays(country_name='Italy')
prophet_model1.fit(df_1)
```

In questa fase si aggiungono le stagionalità al modello, in particolare quelle settimanali, annuali e legate alla vacanze “holidays”. Per quanto riguarda quest’ultimo, si utilizza una libreria apposita. La libreria holidays in Python è molto utile per gestire e identificare le festività nei vari paesi. Fornisce una semplice interfaccia per recuperare le date delle festività nazionali, permettendo agli utenti di controllare se una specifica data è una festività in un dato paese o anche in specifiche regioni o stati all'interno di alcuni paesi. Questa funzionalità è particolarmente preziosa in contesti come l'analisi delle serie temporali, dove le festività possono influenzare significativamente i modelli di dati, ad esempio nei consumi di vendita al dettaglio, nel traffico web, o in altri ambiti stagionali.

Forecasting con Prophet

La procedura per la generazione di previsioni future attraverso Prophet si articola in due fasi principali, delineate dai comandi make_future_dataframe(periods=forecast_window) e predict(future). Questi comandi rappresentano passaggi chiave nel processo di forecasting, permettendo rispettivamente la creazione di un dataframe per le date future su cui effettuare la previsione e l'applicazione del modello addestrato per generare le previsioni stesse.

```
# Genera previsioni future per il modello 1 (Google cpc)
future1 =
prophet_model1.make_future_dataframe(periods=forecast_window)
forecast1 = prophet_model1.predict(future1)
```

Il metodo make_future_dataframe(periods=forecast_window) crea un dataframe che estende la serie temporale originale del periodo indicato fra parentesi (forecast_window). Questo dataframe non contiene valori per la variabile target, ma elenca le date future per le quali si desidera prevedere il comportamento della serie temporale. Il parametro periods specifica il numero di periodi futuri da aggiungere al dataframe, mentre la frequenza di questi periodi (ad esempio, giorni o mesi) viene dedotta dalla frequenza della serie temporale originale.

La generazione di questo dataframe futuro prende in considerazione l'ultimo timestamp presente nel dataset originale, estendendolo per il numero di periodi specificato.

Successivamente, il metodo predict(future1) viene impiegato per effettuare le previsioni. Questo metodo applica il modello Prophet addestrato al dataframe contenente le date future, generando una serie di output che includono non solo le previsioni della variabile target ma anche componenti come tendenza, stagionalità e intervalli di incertezza. Il metodo predict assegnerà a ogni riga nel dataframe futuro un valore predetto, che viene denominato yhat. Se si inseriscono date storiche, il metodo fornirà un adattamento ai dati interni (in-sample fit). L'oggetto forecast qui è un nuovo dataframe che include una colonna yhat con la previsione, così come colonne per le componenti del modello e gli intervalli di incertezza.

La previsione si basa sull'applicazione del modello alle date future, sfruttando i parametri appresi durante la fase di addestramento (tendenza, stagionalità, effetti delle festività, ecc.). Il modello, quindi, calcola i valori previsti per ogni punto temporale futuro, fornendo anche stime dell'incertezza

associata a queste previsioni. Il risultato è un dataframe che, oltre alle date e alle previsioni, include colonne per componenti specifiche del modello, come trend, yearly, weekly, daily (a seconda della configurazione del modello), e intervalli di confidenza (yhat_lower, yhat_upper) che offrono una misura della variabilità attesa attorno alle previsioni.

Una volta fatti i forecast per tutte le fonti, si devono sommare insieme e immagazzinare in un dataframe.

```
# Seleziona le colonne di interesse dal primo set di previsioni
forecast_df1 = forecast1[['ds', 'yhat', 'yhat_lower', 'yhat_upper']]

# Assicura che tutti i DataFrame siano ordinati per 'ds'
forecast_df1 = forecast_df1.sort_values('ds')

# Verifica che tutti i DataFrame abbiano lo stesso 'ds'
if all(forecast_df1['ds'].equals(df['ds']) for df in [forecast_df2, forecast_df3, forecast_df4, forecast_df5, forecast_df6, forecast_df7, forecast_df8]):
    # Calcola la somma dei valori 'yhat' da tutti i DataFrame di previsione
    total_yhat = forecast_df1['yhat'] + forecast_df2['yhat'] + forecast_df3['yhat'] + \
                 forecast_df4['yhat'] + forecast_df5['yhat'] + forecast_df6['yhat'] + \
                 forecast_df7['yhat'] + forecast_df8['yhat']

    # Crea il DataFrame finale con la somma delle previsioni
    df_final = forecast_df1[['ds']].copy()
    df_final['total_yhat'] = total_yhat

    print(df_final.tail())  # Mostra le prime righe del DataFrame finale per verifica
else:
    print("Le date nei DataFrame di previsione non corrispondono.")
```

```
# Si assume che df_final abbia già una colonna 'ds' che corrisponde alle date in forecast_df1
# Si aggiunge i valori 'yhat' da forecast_df1 a df_final

# Se df_final non ha ancora una colonna 'yhat', la si aggiunge
if 'yhat' not in df_final.columns:
    df_final = df_final.merge(forecast_df[['ds', 'yhat']], on='ds', how='left')
else:
    # Se df_final ha già una colonna 'yhat' e vuoi aggiornarla con i valori da forecast_df1
    df_final.update(forecast_df[['ds', 'yhat']])
# Rinomina la colonna 'ds' in 'date'
df_final.rename(columns={'ds': 'date'}, inplace=True)

# Imposta la colonna 'date' come indice del DataFrame
df_final.set_index('date', inplace=True)

# Ora df_final ha 'date' come indice

# Ora df_final contiene una colonna 'yhat' che corrisponde ai valori predetti da forecast_df1
print(df_final.tail())
```

In questo frammento di codice si ha già generato il forecast di tutte le fonti, che sono state memorizzate in dataframe da df1 a df8. Non resta perciò che sommare ogni riga ed essere sicuri che le date corrispondano correttamente. In questo viene aiuto il ciclo for combinato con un if. Si conclude l'operazione andando ad aggiungere ad df_final non solo il total_yhat dato dalla somma dei df1-df8, ma anche il df_p che corrisponde alla versione del modello senza aver suddiviso le leads per fonte.

A questo punto si può già visualizzare un primo risultato di forecasting dato da Prophet, sia nel caso di aver suddiviso le leads per fonte, che senza. Nel caso rappresentato, correva il giorno 11 febbraio

e si voleva fare il forecast della settimana dal 11 al 17 febbraio. (il dataset aveva come dato più recente il 10 febbraio)

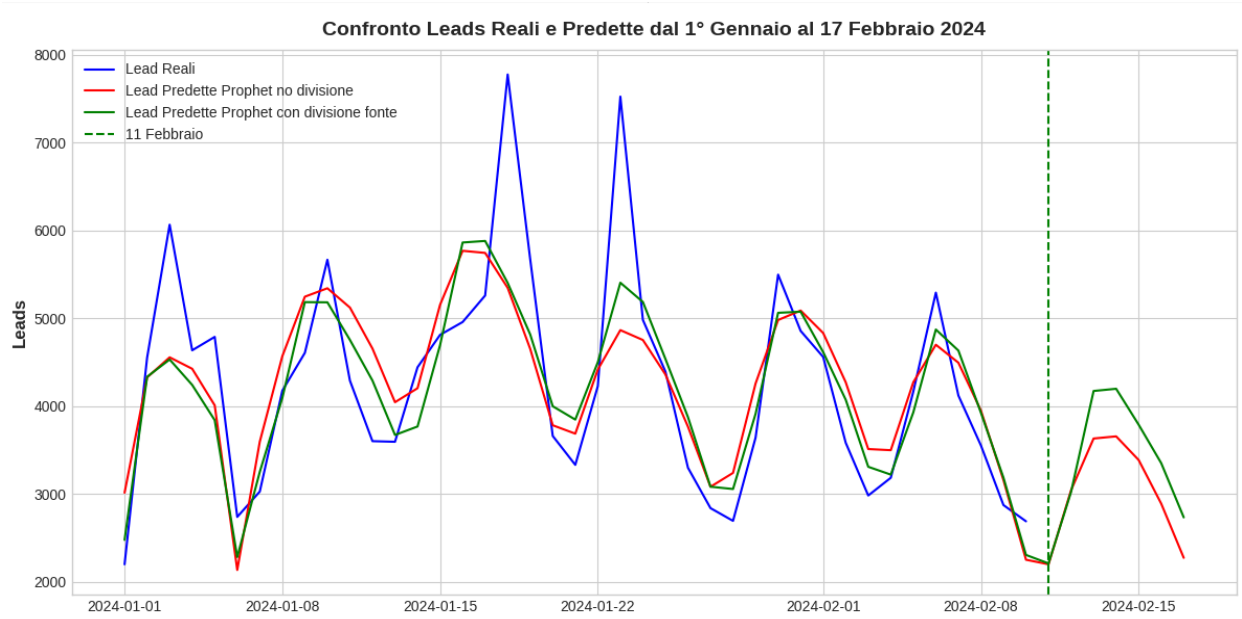


Figura 24 Risultati del forecast con Prophet con e senza divisione per fonte delle leads

Per visualizzare il grafico si è utilizzato il seguente codice, abilitato dalla libreria matplotlib:

```
# Filtra per l'orizzonte temporale specificato
start_date = '2024-01-01'
end_date = '2024-02-17'

df_filtered = df.loc[start_date:end_date]
df_final_filtered = df_final.loc[start_date:end_date]

# Crea il grafico
plt.figure(figsize=(12, 6))
```

```
# Usa lo stesso asse y per entrambe le serie
ax = plt.gca()  # Get current axis
ax.plot(df_filtered.index, df_filtered['leads'], color='blue',
label='Lead Reali')
ax.plot(df_final_filtered.index, df_final_filtered['yhat'],
color='red', label='Lead Predette Prophet no divisione')
ax.plot(df_final_filtered.index, df_final_filtered['total_yhat'],
color='green', label='Lead Predette Prophet con divisione fonte')
ax.set_ylabel('Leads')
ax.tick_params(axis='y')

# Aggiungi una linea verticale al 04 Febbraio 2024
plt.axvline(pd.to_datetime('2024-02-11'), color='green',
linestyle='--', label='11 Febbraio')

# Titolo e legenda
plt.title('Confronto Leads Reali e Predette dal 1° Gennaio al 17
Febbraio 2024')
ax.legend(loc='upper left')

plt.show()
```

Per le conclusioni e osservazioni si rimanda al capitolo 6, in cui ci sono i risultati anche del secondo modello (SARIMAX).

Addestramento del modello SARIMAX tramite AUTO ARIMA

Il modello SARIMAX è il secondo modello utilizzato per fare la previsione delle leads e confrontare i risultati di questo modello con il precedente (Prophet).

Si è incominciato andando a fare un'analisi adfuller del dataset, verificandone l'eventuale stazionarietà. Il risultato ha dato un p-value di circa 0.05, che è sulla soglia della non stazionarietà, perciò si è deciso di utilizzare il metodo .diff() per verificare che la serie possa essere stazionaria.

```
result=adfuller(df['data'].diff())
print(f'ADF Statistics:{result[0]}')
print(f'p-value:{result[1]}')
```

Il risultato ha dato “ADF Statistics:-3.36249130248029 e p-value:0.012305965554996013”, confermando di fatto la stazionarietà della serie.

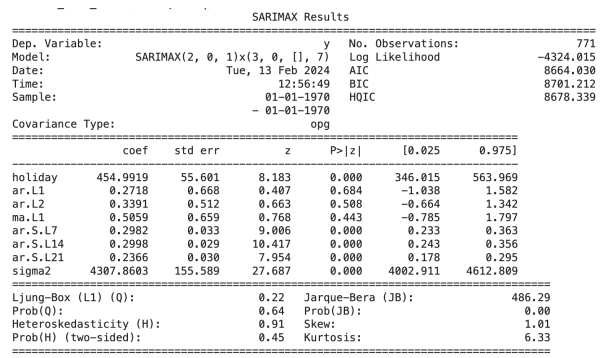
Adesso si riscontra il procedimento che ha richiesto più tempo nella run fatta per il progetto, con un tempo di 18 minuti totali. Il metodo è pm.auto_arima, derivante dalla omonima libreria, e che permette di calcolare gli iper parametri del modello SARIMAX in maniera trial and error, ovvero simulando il comportamento del modello con diversi set di iperparametri fino a dare come risultato il migliore.

```
# Inizializzazione della lista per memorizzare i modelli
risultanti
results = []

# Addestramento del modello su df_p
print("Addestramento su df_p")
model_p = pm.auto_arima(df_p['y'], start_p=0, d=None, start_q=0,
max_p=3, max_q=3,
                        seasonal=True, m=7, D=None, test='adf',
start_P=0, start_Q=0, max_P=3, max_Q=3,
                        information_criterion='aic', trace=True,
error_action='ignore',
                        trend=None, exog=df_p['holiday'],
with_intercept=True, stepwise=True)
results.append(model_p)

# Ripeti il processo per ciascuno degli altri DataFrame (df_1 a
df_8)
for i, df in enumerate([df_1, df_2, df_3, df_4, df_5, df_6, df_7,
df_8], start=1):
    print(f"Addestramento su df_{i}")
    model = pm.auto_arima(df['y'], start_p=0, d=None, start_q=0,
max_p=3, max_q=3,
                          seasonal=True, m=7, D=None, test='adf',
start_P=0, start_Q=0, max_P=3, max_Q=3,
                          information_criterion='aic',
trace=True, error_action='ignore',
                          trend=None, exog=df['holiday'],
with_intercept=True, stepwise=True)
    results.append(model)
```

Come risultato, il codice ha restituito tutti gli iperparametri, che è possibile notare all’interno del successivo codice. Come funzionamento, il metodo pm.auto_arima va pari passo al funzionamento

del metodo di ottimizzazione “Branch and Bound”. Il risultato del pmdarima è stato perciò inserito nel codice per inizializzare i modelli SARIMAX. (di cui se ne scrivono solamente due per evitare ridondanze)

```
# Modello per df_1
model_1 = SARIMAX(df_1['y'], order=(3,0,0),
seasonal_order=(3,0,1,7), exog=df_1['holiday'])
results_1 = model_1.fit()
print(results_1.summary())

# Modello per df_2
model_2 = SARIMAX(df_2['y'], order=(2,0,1),
seasonal_order=(3,0,0,7), exog=df_2['holiday'])
results_2 = model_2.fit()
print(results_2.summary())
```



SARIMAX Results

Dep. Variable:		y	No. Observations:			771
Model:	SARIMAX(2, 0, 1)x(3, 0, [], 7)		Log Likelihood			-4324.015
Date:	Tue, 13 Feb 2024		AIC			8664.030
Time:	12:56:49		BIC			8701.212
Sample:	01-01-1970		HQIC			8678.339
	- 01-01-1970					
Covariance Type:	opg					

	coef	std err	z	P>|z|	[0.025	0.975]
holiday	454.9919	55.601	8.183	0.000	346.015	563.969
ar.L1	0.2718	0.668	0.407	0.684	-1.038	1.582
ar.L2	0.3391	0.512	0.663	0.508	-0.664	1.342
ma.L1	0.5059	0.659	0.768	0.443	-0.785	1.797
ar.S.L7	0.2982	0.033	9.006	0.000	0.233	0.363
ar.S.L14	0.2998	0.029	10.417	0.000	0.243	0.356
ar.S.L21	0.2366	0.030	7.954	0.000	0.178	0.295
sigma2	4307.8603	155.589	27.687	0.000	4002.911	4612.809

Ljung-Box (L1) (Q):	0.22	Jarque-Bera (JB):	486.29
Prob(Q):	0.64	Prob(JB):	0.00
Heteroskedasticity (H):	0.91	Skew:	1.01
Prob(H) (two-sided):	0.45	Kurtosis:	6.33

Figura 25 Risultati del modello SARIMAX

Si notano già alcuni fattori dai risultati del modello, come ad esempio il calcolo dei coefficienti dei termini autoregressivi (es. ar.L1, ar.L2, e così via). Questi termini fanno parte del polinomio autoregressivo della funzione matematica di SARIMAX.

Similarmente, il termini ma.L1 si riferisce alla media mobile calcolata dal modello e utilizzata all’interno della funzione matematica. Sigma2 infine è la varianza del termine di errore.

Per quanto riguarda invece il contenuto dei parametri:

- coef: è il valore stimato del coefficiente del modello. Indica l’effetto della variabile corrispondente sulla variabile dipendente.

- std err: è l'errore standard del coefficiente. Indica la precisione della stima del coefficiente. Più è basso, più la stima è precisa.
- z: è il valore z del coefficiente. Si ottiene dividendo il coefficiente per il suo errore standard. Indica il numero di deviazioni standard che il coefficiente si discosta da zero. Più è alto in valore assoluto, più il coefficiente è significativo.
- P>|z|: è il p-value del coefficiente. Indica la probabilità di ottenere un valore z uguale o più estremo di quello osservato, se il coefficiente fosse nullo. Più è basso, più il coefficiente è significativo.
- [0.025 0.975]: è l'intervallo di confidenza al 95% del coefficiente. Indica l'intervallo di valori in cui il coefficiente è compreso con il 95% di probabilità. Più è stretto, più la stima è precisa.

Una volta creato il modello, si può procedere con il realizzare le previsioni ed il forecast dai risultati ottenuti.

Forecasting con SARIMAX

```
# Previsioni da results_1
prediction_1 = results_1.get_prediction(start=-prediction_window,
exog=df_1['holiday'][-prediction_window:])
mean_prediction_1 = prediction_1.predicted_mean

# Intervalli di confidenza e limiti per results_1
confi_int_1 = prediction_1.conf_int()
lower_limits_1 = confi_int_1.iloc[:, 0]
upper_limits_1 = confi_int_1.iloc[:, 1]
lower_today_1 = np.full([1, prediction_window], np.nan).flatten()
upper_today_1 = np.full([1, prediction_window], np.nan).flatten()
lower_today_1[-1] = confi_int_1.iloc[-1, 0]
upper_today_1[-1] = confi_int_1.iloc[-1, 1]

# Per results_1
sarimax_prediction_1 = pd.DataFrame({'yhat': mean_prediction_1,
'y_lower': lower_today_1, 'y_upper': upper_today_1})
forecast_1 = results_1.get_forecast(steps=forecast_window,
exog=df_1['holiday'].iloc[-forecast_window:])
mean_forecast_1 = forecast_1.predicted_mean
confi_int_f_1 = forecast_1.conf_int()
```

```
lower_limits_f_1 = confi_int_f_1.iloc[:, 0]
upper_limits_f_1 = confi_int_f_1.iloc[:, 1]
sarimax_forecast_1 = pd.DataFrame({'yhat': mean_forecast_1,
'y_lower': lower_limits_f_1, 'y_upper': upper_limits_f_1})
# Unisce le previsioni e le previsioni future per results_1
sarimax_results_1 =
sarimax_prediction_1.append(sarimax_forecast_1)
```

Inizialmente, si procede con la generazione delle previsioni (prediction_1) per un determinato intervallo di tempo nel passato (prediction_window), sfruttando i dati esogeni (exog), in questo caso rappresentati da una variabile indicatrice di periodi di vacanza (df_1['holiday']). Tale scelta metodologica permette di integrare nel modello l'effetto che giorni specifici possono avere sulla variabile di interesse. La media delle previsioni (mean_prediction_1) viene quindi calcolata per ottenere una stima puntuale del valore atteso della serie temporale.

Successivamente, il codice si focalizza sulla determinazione degli intervalli di confidenza (confi_int_1) per le previsioni generate, separando i limiti inferiori (lower_limits_1) e superiori (upper_limits_1). Questo passaggio è cruciale per valutare la variabilità attesa delle previsioni e per fornire un'indicazione della certezza associata ai valori predetti. Per la giornata corrente, vengono preparati due array (lower_today_1, upper_today_1) pieni di valori NaN, tranne per l'ultima posizione che viene popolata con i limiti dell'intervallo di confidenza dell'ultima previsione, evidenziando così un'attenzione particolare per l'immediato futuro.

Dopo aver elaborato le previsioni per il passato recente, il codice si dedica alla generazione di previsioni future (forecast_1) per un numero specificato di passi futuri (forecast_window), utilizzando ancora dati esogeni per migliorare l'accuratezza delle previsioni. Si calcolano poi la media di tali previsioni future (mean_forecast_1) e i relativi intervalli di confidenza (confi_int_f_1), procedendo con la stessa metodologia descritta precedentemente.

Infine, le previsioni relative al periodo passato e quelle future vengono aggregate in un unico dataframe (sarimax_results_1), fornendo una visione complessiva delle performance del modello sia in termini di previsioni effettuate che di stime future. Questa aggregazione facilita l'analisi comparativa e l'interpretazione dei risultati, permettendo di valutare l'efficacia del modello SARIMAX nell'adattarsi e prevedere la serie temporale in esame, tenendo conto sia della componente stagionale sia dell'influenza di variabili esogene come le vacanze.

Per il funzionamento del modello si distinguono prediction e forecast in base alla window. In altre parole, la prediction calcola i valori attesi per l'orizzonte temporale passato e presente, che nel caso del progetto sono i 28 giorni che precedono la giornata di domani, mentre il forecast calcola i valori attesi nella settimana futura.

```
df_sarimax_1 = pd.DataFrame({
    'date': date_range,
    'yhat': sarimax_results_1['yhat'].values
})
```

Nell'ultimo passo prima di procedere con il merge del totale delle simulazioni, si immagazzinano i risultati di ciascun modello all'interno di un dataframe apposta dedicato. Ora che si hanno tutti dataframe contenenti le leads previste e "forecast" si può procedere con il generare un dataframe totale del modello SARIMAX, contenente il caso di nessuna divisione per fonte delle leads e nel caso di divisione per fonte.

```
# Inizializzazione di total_yhat con zeri, della stessa lunghezza
del date_range
total_yhat = pd.Series([0] * len(date_range), index=date_range)

# Somma i valori di yhat da df_sarimax_1 a df_sarimax_8
for df in [df_sarimax_1, df_sarimax_2, df_sarimax_3,
df_sarimax_4, df_sarimax_5, df_sarimax_6, df_sarimax_7,
df_sarimax_8]:
    total_yhat += df['yhat'].values

# Creazione di df_sarimax_total con la colonna total_yhat
df_sarimax_total = pd.DataFrame({
    'date': date_range,
    'total_yhat': total_yhat
})
df_sarimax_total = pd.DataFrame({
    'date': date_range,
    'total_yhat': total_yhat,
    'yhat_p': df_sarimax_p['yhat'].values  # Aggiunge la colonna
yhat dal modello p
})
df_sarimax_total
```

	date	total_yhat	yhat_p
2024-01-14	2024-01-14	3258.564442	3534.476067
2024-01-15	2024-01-15	4558.912208	4710.622531
...	...	...	...
2024-02-16	2024-02-16	3611.003327	3297.255997
2024-02-17	2024-02-17	2972.696993	2660.142819

35 rows × 3 columns

Il risultato finale di questo merge di dataframe è una tabella con "date", "total_yhat" che rappresenta le leads con divisione di fonte e "yhat_p" che rappresenta le leads senza divisione.

Va notato come date è ripetuto due volte: una volta nell'indice ed una volta come colonna a sé stante. Questa ridondanza si potrebbe rimuovere con un'operazione di snowflaking per liberare la memoria.[63]

Utilizzando un codice simile a quello del plot precedente, si visualizzano tutte e 2 i modelli, con le 2 varianti ciascuna, messe a confronto con le leads reali (con la data più recente al 10 febbraio).

[63] Golfarelli, M., & Rizzi, S. (2006). Data Warehouse. Teoria e pratica della progettazione.

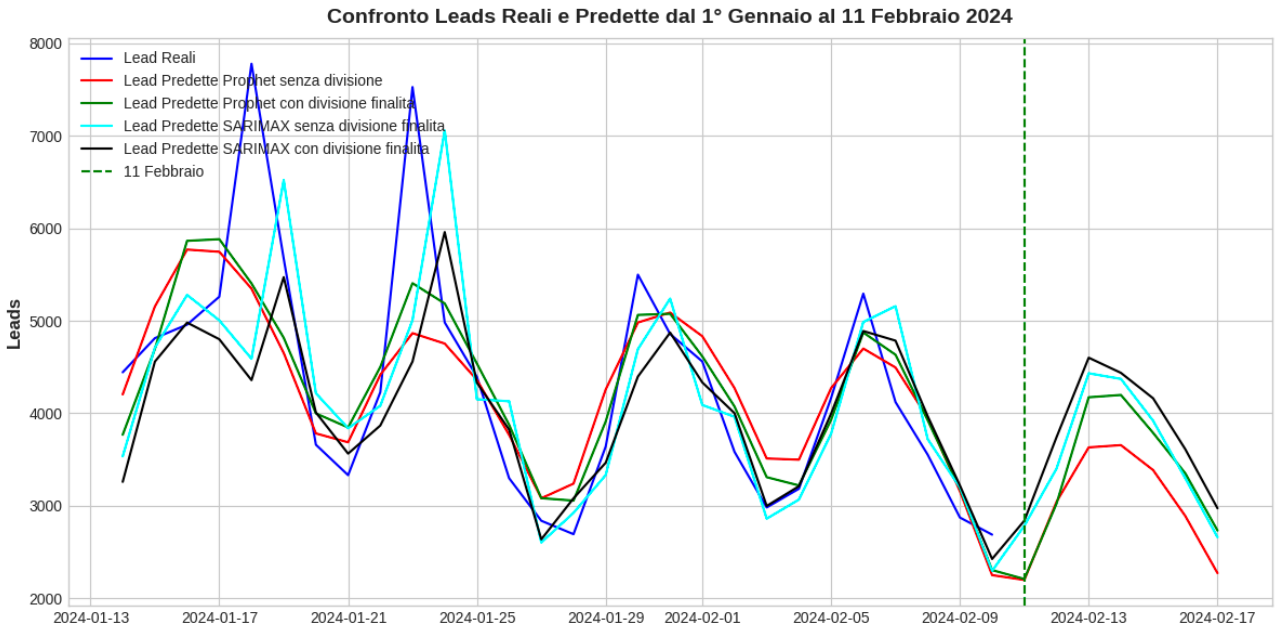


Figura 26 Risultati del forecast dei modelli Prophet e SARIMAX, con e senza divisione. Per facile.it

È possibile ora fare un calcolo preliminare di RMSE, MAPE e MAE.

```
from sklearn.metrics import mean_squared_error,
mean_absolute_error, mean_absolute_percentage_error
import numpy as np
import pandas as pd

# Crea i nuovi DataFrame filtrati
df_prophet_filtered2 = df_final_filtered[:-7]
df_sarimax_filtered2 = df_sarimax_total_filtered[:-7]

metrics = {
    'Model': [],
    'RMSE': [],
    'MAPE': [],
    'MAE': []
}

# Aggiorna il dizionario dei modelli
models = {
    'Prophet senza divisione': df_prophet_filtered2['yhat'],
    'Prophet con divisione': df_prophet_filtered2['total_yhat'],
    'SARIMAX senza divisione': df_sarimax_filtered2['yhat_p'],
    'SARIMAX con divisione': df_sarimax_filtered2['total_yhat']
}

for model_name, y_pred in models.items():
```

```
    rmse = np.sqrt(mean_squared_error(df_filtered['leads'],
y_pred))
    mape = mean_absolute_percentage_error(df_filtered['leads'],
y_pred)
    mae = mean_absolute_error(df_filtered['leads'], y_pred)

    metrics['Model'].append(model_name)
    metrics['RMSE'].append(rmse)
    metrics['MAPE'].append(mape)
    metrics['MAE'].append(mae)

# Creazione del DataFrame
df_metrics = pd.DataFrame(metrics)

# Mostra il DataFrame
print(df_metrics)
```

Tramite la libreria sklearn.metrics è possibile calcolare gli errori confrontando i vari modelli con le leadas reali, che costituiscono la ground truth. E' possibile fare questo solamente con i dati reali che esistono già, perciò si escludono i dati attesi nella settimana futura. I risultati ottenuti, approssimati alla seconda cifra decimale, sono stati:

Tabella 9 Confronto dei valori di errore tra i modelli solo per facile.it

Modello e divisione (solo facile.it)	RMSE (Errore quadratico medio)	MAPE (Errore medio assoluto percentuale)	MAE (Errore medio assoluto)
Prophet senza divisione	812,02	11,92%	554,96
Prophet con divisione di fonte	734,90	11,00%	514,22
SARIMAX senza divisione	979,69	13,46%	648,70
SARIMAX con divisione di fonte	971,27	11,29%	565,95

Secondo questi dati e l'orizzonte temporale preso in considerazione, il modello più preciso sembrerebbe essere Prophet con divisione di fonte, che in tutti i risultati ha ottenuto un errore

inferiore rispetto agli altri modelli. Questo risultato considera tuttavia solo le leads provenienti dal sito principale, ovvero Facile.it, ignorando ancora l'altro sito da cui derivano le leads ovvero mutui.it.

Aggiunta delle leads da mutui.it e finalizzazione dataset di forecast

Per l'esperimento non è stato possibile dividere le leads mutui.it, perciò si sono fatti unicamente due modelli con una sola variante ciascuna. I passaggi per arrivare al plot di confronto sono stati simili ai precedenti, e perciò omessi in questo elaborato. Si inviano perciò unicamente i risultati finali delle simulazioni. Macro trend stagionale

Confrontando i grafici ottenuti dai modelli Prophet e SARIMAX con le leads (isolate) da mutui.it, insieme alle leads reali, si nota una fluttuazione più marcata nel sito "satellite" dell'azienda (mutui.it), rispetto a quello principale. (facile.it)

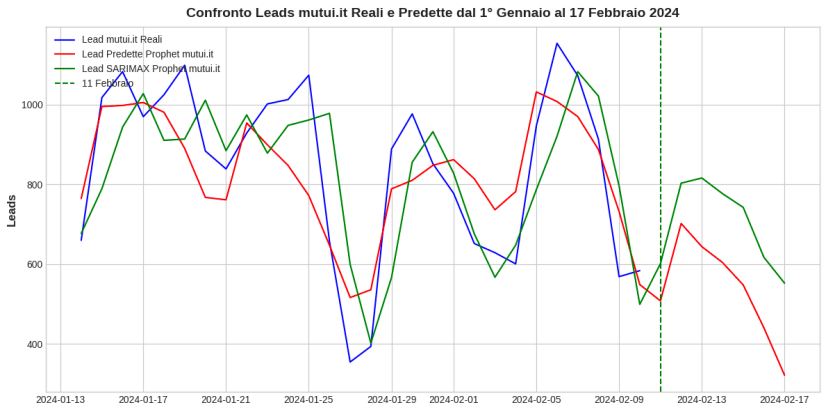


Figura 27 Risultati forecast delle leads da mutui.it, con Prophet e SARIMAX

Prima di arrivare al plot finale, si uniscono le leads derivante da entrambi siti, unendoli per gli stessi modelli, ovvero Prophet con Prophet e SARIMAX con SARIMAX.

```
# Calcolo delle somme delle colonne desiderate
prophet_no_division_sum = df_final_filtered['yhat'] +
df_m_filtered['y']
prophet_con_divisione_sum = df_final_filtered['total_yhat'] +
df_m_filtered['y']
sarimax_no_division_sum = df_sarimax_total_filtered['yhat_p'] +
df_sarimax_m['yhat']
sarimax_con_divisione_sum =
df_sarimax_total_filtered['total_yhat'] + df_sarimax_m['yhat']
```

```
# Utilizzo delle date da df_final_filtered per il nuovo DataFrame
df_unificato = pd.DataFrame({
    'Prophet senza divisione': prophet_no_division_sum.values,
    'Prophet con divisione': prophet_con_divisione_sum.values,
    'SARIMAX senza divisione': sarimax_no_division_sum.values,
    'SARIMAX con divisione': sarimax_con_divisione_sum.values
}, index=df_final_filtered.index)  # Imposta l'indice per mantenere l'allineamento delle date

# Se necessario, si può resettare l'indice per avere 'date' come una colonna regolare
df_unificato.reset_index(inplace=True)
```

Grafico finale del dataset di forecast

date	Prophet senza divisione	Prophet con divisione	SARIMAX senza divisione	SARIMAX con divisione
2024-01-14	4968.913197	4532.620303	4211.737291	3935.825666
2024-01-15	6150.234744	5690.755869	5500.004817	5348.294493
...	...	...	...	...
2024-02-16	3330.110028	3791.497036	3914.781344	4228.528674
2024-02-17	2593.735046	3054.466448	3212.667140	3525.221315

35 rows × 4 columns

Si ottiene il dataset finale con i due modelli e le due suddivisioni, che contengono la totalità delle leads che arrivano all'azienda dai due siti da cui attingono le richieste.

Figura 28 Dataset generato dai dati del forecast

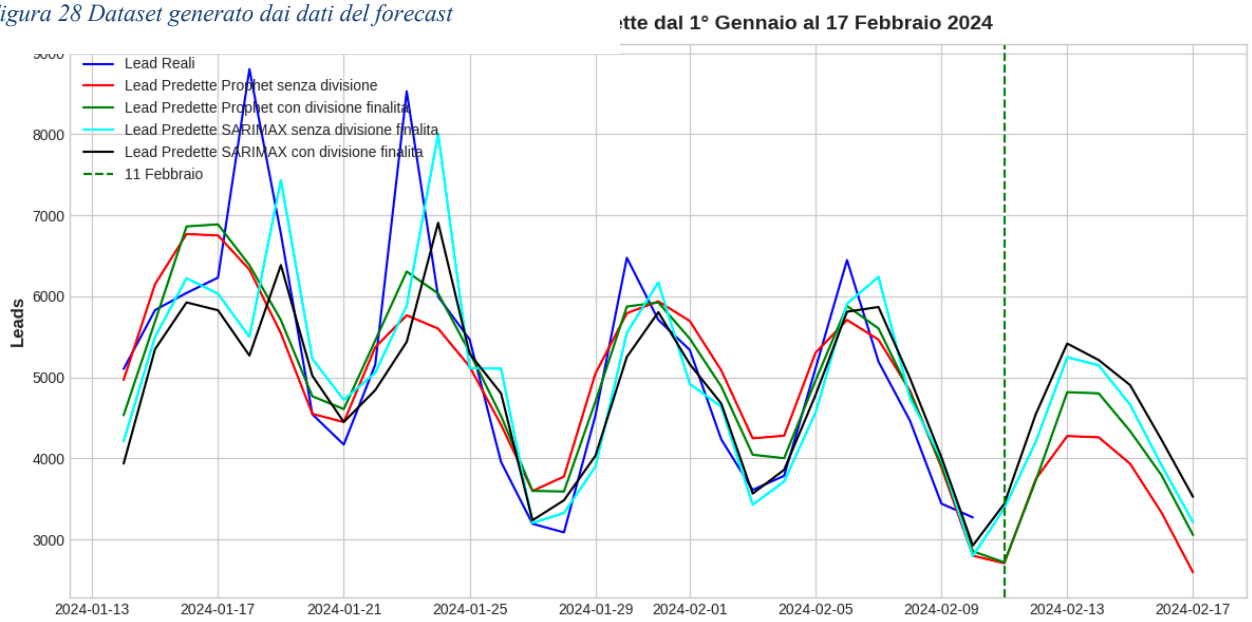


Figura 29 Confronto dei risultati finali, con facile.it e mutui.it, con i modelli Prophet e SARIMAX e con e senza divisione per fonte. Confronto con le lead reali (in blu)

Si possono distinguere 5 settimane osservando la composizione delle sinusoidi. In una giornata ciascuno nelle prime due settimane si sono toccati dei picchi sopra gli 8000 leads reali, mentre i

valori attesi di ogni modello non superava i 7500 leads. La situazione si è stabilizzata nelle due settimane a seguire, in cui i modelli hanno predetto in maniera meno erronea le singole giornate, con valori attesi di poco differenti rispetto al valore reale.

L'ultima settimana rappresenta una settimana futura in cui è stato eseguito il forecasting, in cui osserviamo che il numero di leads predetto tocca il picco più alto nel SARIMAX con divisione di fonte, mentre la simulazione con il picco più basso è rappresentato dal modello Prophet senza divisione. Si può infine procedere con l'analisi degli errori:

Tabella 10 Confronto dei valori di errore tra i modelli dopo l'aggiunta di mutui.it

Modello e divisione (facile.it + mutui.it)	RMSE (Errore quadratico medio)	MAPE (Errore medio assoluto percentuale)	MAE (Errore medio assoluto)
Prophet senza divisione	864,59 (+)	11,32% (-)	614,56 (+)
Prophet con divisione di fonte	770.43 (+)	10,06% (-)	547,3 (+)
SARIMAX senza divisione	1031,29 (+)	12,56% (-)	708,75 (+)
SARIMAX con divisione di fonte	1031,28 (+)	11,28% (-)	652,17 (+)

Sia l'errore quadratico medio che quello assoluto sono aumentati in tutti i modelli, mentre l'errore medio assoluto percentuale è inferiore rispetto al test con solo le leads del sito di facile.it. I volumi di leads sono una delle cause scatenanti di questo fenomeno, ovvero che l'errore nel numero di leads è aumentato dato che si tratta di volumi più ampi (nei giorni di picco: 7500 leads/giorno vs 8500 leads/giorno), tuttavia l'errore percentuale è calato in ogni modello.

Si conclude che l'aggiunta delle leads mutui.it non ha inficiato in maniera relativamente negativa la precisione del modello.

```
Il numero di leads di questa settimana secondo il modello Prophet
senza divisione sono stati: 24471
Il numero di leads di questa settimana secondo il modello Prophet
con divisione sono stati: 26835
Il numero di leads di questa settimana secondo il modello SARIMAX
senza divisione sono stati: 27314
Il numero di leads di questa settimana secondo il modello SARIMAX
con divisione sono stati: 27382
Il numero di leads di questa settimana reali sono stati: 26139
```

Essendo l'esperimento dedicato al dimensionamento del numero di operatori necessarie nel call center, si procede con l'ultima fase che determina la capacità richiesta nella settimana futura.

Dimensionamento del numero necessario di operatori del call center

Nel capitolo 4 si è affrontato il tema delle leads complete e parziali, e la differenza chiave nel che hanno.

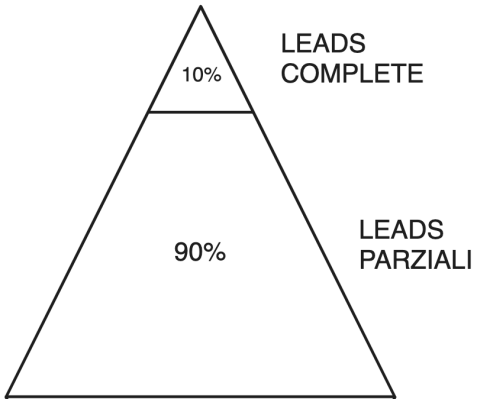


Figura 30 Divisione approssimata delle leads complete e parziali

Le leads parziali corrispondono in media al 90% delle leads totali, mentre il restante 10% sono le leads complete.

Essendo diversi i team che gestiscono le due leads, si dividono due team di call center: "Team complete" e "Tean parziali". (inizialmente)

La differenza principale in questi due team risiede nel volume settimanale di leads gestibili dal singolo operatore.

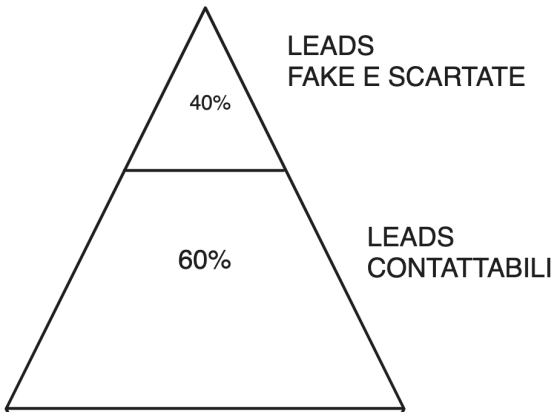


Figura 31 Divisione approssimata delle leads false e scartate da quelle contattabili e valide

Settimanalmente, un operatore del “Team complete” può gestire in media 100 leads, mentre un operatore del “Team parziali” può gestire in media 550 leads.

Non tutte le leads sono contattabili, ma in media solo un 60% viene contattato dal call center, mentre il restante 40%, in media, sono scartate dal sistema che individua le richieste fake e quelle prive di valore.

Si considera infine una settimana lavorativa di 6 giorni, ovvero dal lunedì al sabato.

Ci sono due metodi per il dimensionamento degli operatori: per settimana o per ora. Si procede con il primo metodo, più semplice, per poi passare al secondo metodo, più completo.

Primo metodo: per settimana

Tabella 11 Numero di leads gestibili per operatore del team. Dati medi sull’anno 2023

Team	**n° leads gestibili alla settimana**
Team Complete	100 leads/operatore
Team Parziali	550 leads/operatore

Avuta la capacità disponibile, è necessario enumerare la capacità necessaria, ovvero le leads complessive ottenute dal sito durante la settimana. Tramite il seguente codice, è possibile calcolarlo per ognuna delle 4 simulazioni.

```
# Somma gli ultimi 7 valori per ogni modello
sums = {}
for model_name, y_pred in models.items():
    sums[model_name] = y_pred[-7:].sum()

# Stampa i risultati
for model_name, sum in sums.items():
    print(f"il numero di leads della prossima settimana secondo
il modello {model_name} è di: {math.ceil(sum)}")
```

Si è deciso di tenere anche la domenica nel riferimento delle leads, nonostante vengano chiamati nella settimana successiva, in quanto entrambi i team devono chiamare al lunedì le leads della domenica precedente. I risultati, con approssimazione per eccesso all'intero successivo, sono i seguenti:

```
Settimana 28 feb - 5 mar
il numero di leads della prossima settimana secondo il modello
Prophet senza divisione è di: 28449
il numero di leads della prossima settimana secondo il modello
Prophet con divisione è di: 26518
il numero di leads della prossima settimana secondo il modello
SARIMAX senza divisione è di: 26443
il numero di leads della prossima settimana secondo il modello
SARIMAX con divisione è di: 26112
```

Per arrivare alla capacità necessaria, si dividono queste leads con suddivisione 10/90 per identificare quali sono le leads parziali e quali quelle complete.

$$n°\ Operatori = \frac{Leads\ totali\ \cdot\ \%\ complete\ \cdot (1 - \%fake\ e\ scartate) + Leads\ Parziali \cdot 0{,}17}{capacità\ leads\ per\ operatore\ complete} + \frac{Leads\ totali\ \cdot\ \%parziali\ \cdot (1 - \%fake\ e\ scartate)}{capacità\ leads\ per\ operatore\ parziali}$$

Equazione 9 Numero di operatori

Sostituendo i valori alla formula:

$$n°\ Operatori = \frac{Leads\ totali\ \cdot\ 0{,}1 \cdot (0{,}6)\ +\ Leads\ totali\ \cdot\ 0{,}9\ \cdot\ 0{,}6\ \cdot\ 0{,}17}{100} + \frac{Leads\ totali\ \cdot\ 0{,}9\ \cdot 0{,}6}{550}$$

Equazione 10 Numero di operatori

Tabella 12 Tabella di calcolo del numero di operatori con il metodo per settimana, per modello e divisione

Team	Modello e divisione	n° leads settimana futura	n° leads del cluster (approx.)	N° operatori richiesti secondo il modello
Team Complete	Prophet, nessuna divisione	28449	4319	44
	Prophet, divisione su fonte	26518	4026	41

	SARIMAX, nessuna divisione	26443	4015	41
	SARIMAX, divisione su fonte	26112	3964	40
Team Parziali	Prophet, nessuna divisione	28449	15363	29
	Prophet, divisione su fonte	26518	14320	27
	SARIMAX, nessuna divisione	26443	14280	27
	SARIMAX, divisione su fonte	26112	14101	27
TOTALE OPERATORI: 67 ÷ 73				

Secondo tutti i modelli, saranno necessari nella settimana dal 11 al 17 febbraio, tra 40 e 44 operatori nel "team complete" e tra 27 e 29 operatori nel "team parziali", per un totale di range tra 67 e 73 operatori.

Questo dimensionamento è meno preciso del secondo metodo in quanto si assume che ogni giorno ci sia un carico di lavoro uguale. Per capirne la precisione bisogna confrontare il risultato con il secondo metodo.

Secondo metodo: per ore lavorate

In una settimana, un operatore che non fa straordinari e non prende ferie, lavora sei giorni settimanali, 36 ore alla settimana e, in media, 6 ore al giorno.

Prendendo come riferimento la settimana dal 4 al 10 febbraio 2024, si è osservato che gli operatori del Team "Complete" lavoravano una media di 82 leads, mentre gli operatori del Team "Parziali" una media di 527,5 leads, in linea con le stime fatte nel primo metodo.

Considerando una media lavorata per ciascun operatore di 33,7 ore, togliendo perciò pause e operatori che hanno richiesto dei permessi, si ottengono i seguenti numeri:

Tabella 13 Leads lavorate all'ora dai team. Dati presi dalla settimana dal 4 al 10 febbraio 2024

Team	Leads lavorate/settimana*operatore	Leads/ora lavorate
Team Complete	82	2,43
Team Parziali	540	15,6

Questo dato leads/ora permette di trovare il numero di ore di lavoro necessarie ogni giorno e, di conseguenza, il numero di operatori necessari in entrambi i team.

Prima di fare il calcolo, considerando le leads prese dal forecast e il dato di leads orarie, bisogna considerare anche che alcune delle leads del team parziali vengono gestite prima da loro, e poi dal team complete. Questo accade perché il team parziali converte le leads "parziali" in leads "complete", che poi vengono assegnate al team complete. La % di questa conversione varia tra il 12 e il 20% del totale, si può considerare una mediana di 17%.

Considerando una simulazione al 27 di febbraio 2024, che riporta i seguenti dati:

Tabella 14 Calcolo del numero di operatori per ciascuna giornata

Data	28-feb-24	29-feb-24	1-mar-24	2-mar-24	3-mar-24	4-mar-24	5-mar-24
Leads forecast (AVG)	5128	4744	4106	3366	3392	4359	5172
Leads Parziali	2769	2562	2217	1818	1832	2354	2793
Leads Complete	778	720	623	511	515	662	785
Ore Necessarie Team	178	165	143	117	118	151	180

Parziali							
Ore Necessarie Team Complete	321	297	257	211	212	273	324
Op. Team Parziale full time necessarie	30	28	24	20	20	26	30
Op. Team Complete full time necessarie	54	50	43	36	36	46	54
Totale Operatori Necessari	84	78	67	56	56	72	84

Il risultato del primo metodo (67 - 73) è un valore compreso tra il minimo (56) e il massimo (84) del secondo metodo, come prevedibile dalla natura stagionale delle leads durante la settimana. Idealmente il risultato del secondo metodo è più preciso, ma il risultato del primo metodo può essere una stima sufficiente per ipotizzare il carico di lavoro nella settimana successiva. Va specificato che il 3 marzo gli operatori necessari sono 56, ma è un giorno festivo.

Creazione di una dashboard per aumentare la comprensione dei risultati al management

L'area aziendale che si occupa di Data Science e Analytics ha il compito di mostrare i risultati dei modelli di machine learning nella maniera più comprensibile possibile all'area che si occupa delle decisioni strategiche, che nel caso aziendale è l'area Business.

Uno dei metodi per rendere comprensibili e utilizzabili i dati ricevuti dal forecast del modello di machine learning, è quello di creare una dashboard e trasformare i dati del forecast in metriche, KPIs ed euro.

Le metriche principali calcolate per la creazione della dashboard sono basate sul numero di operatore da assumere ipoteticamente per soddisfare la domanda di mercato, il numero di ore di straordinario

ipotetico e i costi, il confronto tra le leads forecast e budget, la % di copertura del fabbisogno e il confronto tra le ore operatori disponibili e quelle necessarie.

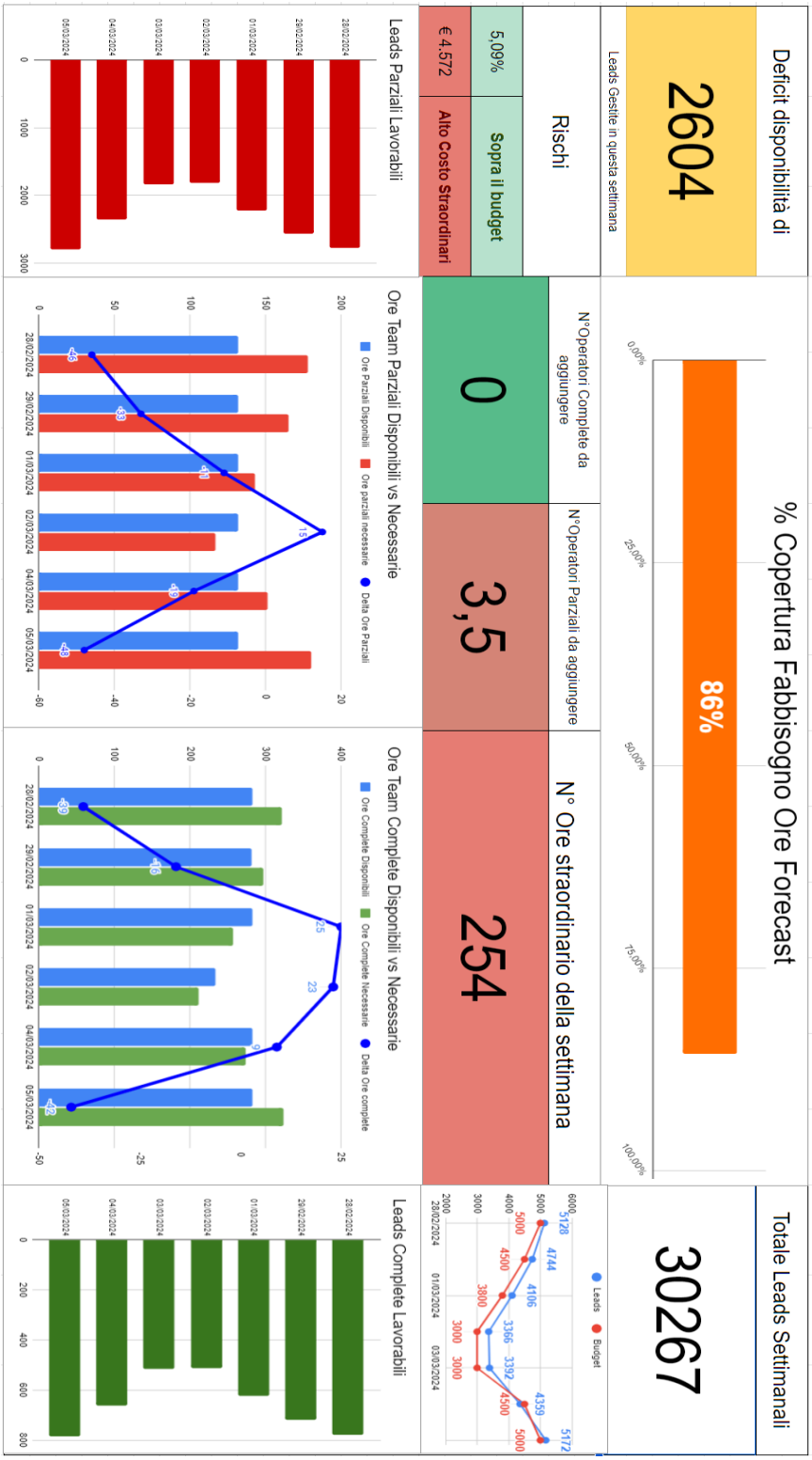


Figura 32 Dashboard per la comprensione dei risultati al management

CONCLUSIONI FINALI

Nel contesto di una crescente digitalizzazione dei servizi finanziari, il presente lavoro di tesi ha affrontato il problema della previsione accurata del volume di leads in una realtà aziendale attiva nel settore online. La questione centrale ha riguardato l'identificazione del modello predittivo più efficace per stimare il numero di leads generati dai siti facile.it e mutui.it, al fine di ottimizzare la gestione delle risorse nel call center aziendale.

L'approccio adottato per risolvere questa problematica si è articolato attraverso l'impiego dei modelli Prophet e SARIMAX, valutati sia individualmente sia in combinazione con una suddivisione delle leads per fonte. Tale metodologia ha permesso di analizzare i dati storici relativi alle leads, confrontando la precisione delle previsioni fornite dai due modelli e valutando l'impatto dell'integrazione delle leads provenienti da diverse fonti.

I risultati ottenuti hanno evidenziato che il modello SARIMAX, in particolare quando applicato con una divisione per fonte delle leads, ha superato in termini di accuratezza il modello Prophet e lo stesso SARIMAX senza suddivisione per fonte. In termini numerici, il modello SARIMAX con divisione ha mostrato una performance superiore rispetto agli altri modelli testati, migliorando la previsione del volume di leads con un significativo 99,83% di precisione rispetto al 93,16% ottenuto con Prophet, senza divisione. Questo esito sottolinea l'importanza di considerare le peculiarità delle diverse fonti di leads nella costruzione di modelli predittivi nel settore dei servizi finanziari online.

Inoltre, l'analisi ha confermato che l'integrazione delle leads da mutui.it non ha deteriorato la precisione dei modelli, dimostrando la robustezza degli approcci adottati anche in presenza di set di dati ampliati. Dal punto di vista operativo, l'utilizzo dei modelli di previsione ha facilitato una stima accurata del numero di operatori necessari nel call center, contribuendo a una pianificazione più efficace delle risorse umane.

Per quanto riguarda i possibili sviluppi futuri, si suggerisce l'esplorazione di tecniche di machine learning avanzate, come le reti neurali applicate al modello di previsione, che potrebbero offrire ulteriori miglioramenti nella previsione dei volumi di leads. Inoltre, potrebbe essere utile estendere l'analisi includendo variabili aggiuntive, come dati macroeconomici o indicatori di mercato, per verificare il loro impatto sulle previsioni e affinare ulteriormente i modelli.

Un altro ambito di sviluppo riguarda l'ottimizzazione della dashboard di monitoraggio per il management, implementando funzionalità predittive in tempo reale e integrando strumenti di intelligenza artificiale per l'elaborazione di scenari prospettici. Ciò consentirebbe una gestione ancora più dinamica e basata sui dati delle risorse aziendali, con ripercussioni positive sulla soddisfazione del cliente e sull'efficienza operativa.

In conclusione, questo studio ha dimostrato l'efficacia dei modelli Prophet e SARIMAX nella previsione dei volumi di leads nel settore dei servizi finanziari online, offrendo spunti preziosi per il miglioramento della gestione delle risorse nel call center. L'adozione di approcci predittivi avanzati e l'integrazione di nuove variabili e tecnologie rappresentano le frontiere future verso le quali orientare la ricerca e lo sviluppo aziendale nel contesto digitale.

BIBLIOGRAFIA

- Aghabozorgi, S., Shirkhorshidi, A. S., & Wah, T. Y. (2019). Time-series clustering - A decade review. Information Systems, 80, 16-38. https://doi.org/10.1016/j.is.2018.09.003
- Armando, E., & Craparotta, G. (2019). A Meta-Model for Fashion Retail Category Sales Forecasting. In Business Models and ICT Technologies for the Fashion Supply Chain: Proceedings of IT4Fashion 2017 and IT4Fashion 2018 7 (pp. 79-93). Springer International Publishing.
- Baesens, B., Bapna, R., Marsden, J. R., Vanthienen, J., & Zhao, J. L. (2015). Transformational issues of big data and analytics in networked business. MIS Quarterly, 39(2), 461-478.
- Bandara, K., Hyndman, R. J., & Bergmeir, C. (2021). MSTL: A seasonal-trend decomposition algorithm for time series with multiple seasonal patterns. arXiv preprint arXiv:2107.13462.
- Basatnia, N., Hossein, S. A., Rodrigo-Comino, J., Khaledian, Y., Brevik, E. C., Aitkenhead-Peterson, J., & Natesan, U. (2018). Assessment of temporal and spatial water quality in international Gomishan Lagoon, Iran, using multivariate analysis. Environmental Monitoring and Assessment, 190, 1-17.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: forecasting and control (5th ed.). Wiley.
- Buxton, E., Kriz, K., Cremeens, M., & Jay, K. (2019, December). An auto regressive deep learning model for sales tax forecasting from multiple short time series. In 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA) (pp. 1359-1364). IEEE.
- Castiglione, J., Astroza, R., Azam, S. E., & Linzell, D. (2020). Auto-regressive model based input and parameter estimation for nonlinear finite element models. Mechanical Systems and Signal Processing, 143, 106779.
- Chu, B., & Qureshi, S. (2023). Comparing out-of-sample performance of machine learning methods to forecast US GDP growth. Computational Economics, 62(4), 1567-1609.
- Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2013). Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences (3rd ed.). Mahwah, NJ: Lawrence Erlbaum Associates.

- Darjani, N., & Omranpour, H. (2022). Comprehensive Learning Polynomial Auto-Regressive Model based on Optimization with Application of Time Series Forecasting. International Journal of Industrial Electronics Control and Optimization, 5(1), 43-50.
- Dong, Y., Chung, M., Zhou, C., & Venkataraman, S. (2019). Banking on "Mobile Money": The Implications of Mobile Money Services on the Value Chain. Manufacturing & Service Operations Management, 21(2), 357-374.
- Duan, J., & Kashima, H. (2021). Learning to rank for multi-step ahead time-series forecasting. IEEE Access, 9, 49372-49386.
- Elamin, N., & Fukushige, M. (2018). Modeling and forecasting hourly electricity demand by SARIMAX with interactions. Energy, 165, 257-268.
- Esposito, L., Mastromatteo, G., Molocchi, A., Brambilla, P. C., Carvalho, M. L., Girardi, P., Marmiroli, B., & Mela, G. (2022). Green mortgages, EU taxonomy and environment risk weighted assets: A key link for the transition. Sustainability, 14(3), 1633. https://doi.org/10.3390/su14031633
- Fuster, A., Plosser, M., Schnabl, P., & Vickery, J. (2019). The role of technology in mortgage lending. The Review of Financial Studies, 32(5), 1854-1899. https://doi.org/10.1093/rfs/hhz018
- Godin, S. (2007). Permission Marketing: Turning Strangers into Friends and Friends into Customers. New York, NY: Simon & Schuster.
- Golfarelli, M., & Rizzi, S. (2006). Data Warehouse. Teoria e pratica della progettazione. McGraw-Hill Education
- Golitsis, P., Bellos, S. K., Fassas, A. P., & Demiralay, S. (2021). The Spillover Effect of Euribor on Southeastern European Economies: A Global VAR Approach. Journal of East-West Business, 27(1), 57-91.
- Gummesson, E. (2008). Total Relationship Marketing (3rd ed.). Oxford, UK: Butterworth-Heinemann.
- Gupta, S., & Sharma, D. (2022). Prediction of COVID-19 spread in world using pandemic dataset with application of auto ARIMA and SIR models. International Journal of Critical Infrastructures, 18(2), 148-158.
- Gurnani, M., Korke, Y., Shah, P., Udmale, S., Sambhe, V., & Bhirud, S. (2017). Forecasting of sales by using fusion of machine learning techniques. 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), 1, 1-10.

- Huang, Y. T., Bai, Y. L., Yu, Q. H., Ding, L., & Ma, Y. J. (2022). Application of a hybrid model based on the Prophet model, ICEEMDAN and multi-model optimization error correction in metal price prediction. Resources Policy, 79, 102969.
- Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice (2nd ed.). OTexts. https://otexts.com/fpp2/
- Hyndman, R. J., Wang, E., & Laptev, N. (2020). Large-scale unusual time series detection. IEEE Transactions on Knowledge and Data Engineering, 32(3), 569-579. https://doi.org/10.1109/TKDE.2018.2876855
- Johnson, A. E., Pollard, T. J., Shen, L., Li-wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. Scientific Data, 3, 160035.
- Johnson, M. (2018). Digital Marketing Strategies. Oxford, UK: Oxford University Press.
- Kanas, A., Hassan Al-Tamimi, H. A., Albaity, M., & Mallek, R. S. (2019). Bank competition, stability, and intervention quality. Journal of Financial Stability, 43, 100-710.
- Kotler, P., & Armstrong, G. (2010). Principles of Marketing (14th ed.). Upper Saddle River, NJ: Prentice Hall.
- Kotler, P., & Keller, K. L. (2016). Marketing Management (15th ed.). Upper Saddle River, NJ: Prentice Hall.
- Kshetri, N. (2014). Big data's impact on privacy, security and consumer welfare. Telecommunications Policy, 38(11), 1134-1145.
- Kumar, V., & Reinartz, W. (2016). Data-Driven Marketing: The 15 Metrics Everyone in Marketing Should Know. Hoboken, NJ: John Wiley & Sons.
- Le, T. T., Pham, B. T., Ly, H. B., Shirzadi, A., & Le, L. M. (2020). Development of 48-hour precipitation forecasting model using nonlinear autoregressive neural network. In CIGOS 2019, Innovation for Sustainable Infrastructure: Proceedings of the 5th International Conference on Geotechnics, Civil Engineering Works and Structures (pp. 1191-1196). Springer Singapore.
- Lee, Y. K., Mammen, E., Nielsen, J. P., & Park, B. P. (2018). In-sample forecasting: A brief review and new algorithms. ALEA-Latin American Journal of Probability and Mathematical Statistics, 15, 875-895.
- Lemon, K. N., & Verhoef, P. C. (2016). Understanding Customer Experience Throughout the Customer Journey. Journal of Marketing, 80(6), 69-96.

- Levine, R., Lin, C., Wang, Z., & Xie, W. (2017). Bank Liquidity, Credit Supply and the Environment.
- Li, Dongfang, Baotian Hu, Qingcai Chen, Xiao Wang, Quanchang Qi, Liubin Wang, and Haishan Liu. "Attentive capsule network for click-through rate and conversion rate prediction in online advertising." Knowledge-based systems 211 (2021): 106522.
- Li, M., & Liu, X. (2020). Maximum likelihood least squares based iterative estimation for a class of bilinear systems using the data filtering technique. International Journal of Control, Automation and Systems, 18(6), 1581-1592.
- Li, Y., Wang, N., Shi, J., Liu, J., & Hou, B. (2020). A survey on deep learning for time series data. Information Fusion, 63, 147-161. https://doi.org/10.1016/j.inffus.2020.05.012
- Litvinova, S. A., Ivanova, O. B., Chernobay, O. S., & Zarubina, V. R. (2023). Green mortgage as the key to energy-efficient and resource-saving real estate. In A. K. Bahl, S. Batra, & S. K. Dhameja (Eds.), Advances in entrepreneurial economics and sustainable development: Climate-smart innovation (pp. 269-284). Springer.
- Liu, J. (2018). Bank Stability, Sovereign Debt and Derivatives. Eastern Economic Journal, 44(1), 174-176.
- Loungani, P., & Razin, A. (2001). How beneficial is foreign direct investment for developing countries? Finance & Development, 38(2), 6-9.
- Maes, M. A., Breitung, K., & Dann, M. R. (2021, May). At issue: the Gaussian autocorrelation function. In International Probabilistic Workshop (pp. 191-203). Cham: Springer International Publishing.
- Maslim, M., & Arinanda, K. (2020). Motorcycle parts sales forecasting using auto-Regressive Integrated moving average model. International Journal of Computer Theory and Engineering,, 12(1), 28-31.
- McHugh, C., Coleman, S., Kerr, D., & McGlynn, D. (2019, December). Forecasting day-ahead electricity prices with a SARIMAX model. In 2019 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1523-1529). IEEE.
- McKinney, W. (2012). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. Sebastopol, CA: O'Reilly Media.
- Mi, B., & Han, L. (2018). Banking market concentration and syndicated loan prices. Review of Quantitative Finance and Accounting, 54(1), 1-28.

- Mullan, J., Bradley, L., Loane, S., Estelami, H., & Laukkanen, T. (2017). Bank adoption of mobile banking: stakeholder perspective. International Journal of Bank Marketing, 35(7), 1154-1174. https://doi.org/10.1108/IJBM-09-2015-0145
- Nontapa, C., Kesamoon, C., Kaewhawong, N., & Intrapaiboon, P. (2020). A new time series forecasting using decomposition method with SARIMAX model. In Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part V 27 (pp. 743-751). Springer International Publishing.
- Orlando, G., & Bufalo, M. (2023). Time series forecasting with the CIR# model: from hectic markets sentiments to regular seasonal tourism. Technological and Economic Development of Economy, 29(4), 1216-1238.
- Pavlyshenko, B. M. (2019). Machine-Learning Models for Sales Time Series Forecasting. Data, 4(1), 15.
- Puccinelli, N. M., Goodstein, R. C., Grewal, D., Price, R., Raghubir, P., & Stewart, D. (2010). Customer experience management in retailing: Understanding the buying process. Journal of Retailing, 85(1), 15-30.
- Rahimi, Z., Shafri, H. Z. M., & Norman, M. (2018). A GNSS-based weather forecasting approach using nonlinear auto regressive approach with exogenous input (NARX). Journal of Atmospheric and Solar-Terrestrial Physics, 178, 74-84.
- Robles-Garcia, C. (2019). Competition and incentives in mortgage markets: The role of brokers. Unpublished working paper.
- Rubin, D. B. (2004). Multiple Imputation for Nonresponse in Surveys. Hoboken, NJ: John Wiley & Sons.
- Sahoo, K., Samal, A. K., Pramanik, J., & Pani, S. K. (2019). Exploratory data analysis using Python. International Journal of Innovative Technology and Exploring Engineering, 8(12), 4727-4735.
- Smith, J. A., & Taylor, E. (2020). Consumer Behavior in the Digital Age. New York, NY: Penguin Random House.
- Stockenstrand, A.-K., & Nilsson, F. (Eds.). (2017). Bank Regulation. Routledge.
- Tan, P.-N., Steinbach, M., & Kumar, V. (2019). Introduction to Data Mining (2nd ed.). Boston, MA: Pearson.
- Taylor, S. J., & Letham, B. (2017). Forecasting at scale. PeerJ Preprints.

- Tufte, E. R. (2001). The Visual Display of Quantitative Information (2nd ed.). Cheshire, CT: Graphics Press.
- V. Zamagni (1993) The Economic History of Italy 1860-1990. Clarendon Press
- Vartholomaios, A., Karlos, S., Kouloumpris, E., & Tsoumakas, G. (2021, September). Short-term renewable energy forecasting in greece using prophet decomposition and tree-based ensembles. In International Conference on Database and Expert Systems Applications (pp. 227-238). Cham: Springer International Publishing.
- Vives, X. (2019). Competition and stability in modern banking: A post-crisis perspective. International Journal of Industrial Organization, 64, 55-69.
- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques (4th ed.). Burlington, MA: Morgan Kaufmann.
- Witzany, J. (2020). Interest Rate Derivatives. In Derivatives: Theory and Practice of Trading, Valuation, and Risk Management (pp. 43-75). Cham: Springer International Publishing.
- Yakubu, U. A., & Saputra, M. P. A. (2022). Time series model analysis using autocorrelation function (acf) and partial autocorrelation function (pacf) for e-wallet transactions during a pandemic. International Journal of Global Operations Research, 3(3), 80-85.

SITOGRAFIA

https://facebook.github.io/prophet/

https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/

https://alkaline-ml.com/pmdarima/modules/generated/pmdarima.arima.AutoARIMA.html

https://www.statsmodels.org/dev/examples/notebooks/generated/statespace_sarimax_stata.html

RINGRAZIAMENTI

L’università è stata un’esperienza che ha cambiato la mia vita.

Ringrazio in primis la mia famiglia, in particolare la mamma e il papà, i nonni e gli zii, per avermi permesso di fare quest'esperienza, ed incoraggiato prima, durante e nella conclusione.

Ringrazio la mia compagna Ludovica per spronarmi ogni giorno ad uscire dalla mia zona di comfort.

Ringrazio l’associazione JEBO per avermi permesso di creare, guidare, ispirare ed essere ispirato.

Ringrazio l’associazione AIESEC per avermi mostrato concretamente il significato delle parole “internazionalità”, “inclusione” e “valori”.

Ringrazio il Chia.mo Prof. Ing. Borghesi per la disponibilità e il supporto dimostrato nello svolgimento di questo elaborato.

Ringrazio i Chiar.mi professori e il dipartimento del corso di laurea di Ingegneria Gestionale per avermi messo alla prova accademicamente e personalmente negli ultimi cinque anni e mezzo.

Ringrazio i gentili professori ed ex-professori dell’indirizzo elettronico dell’Istituto d’Istruzione Superiore Einaudi-Scarpa di Montebelluna per la qualità dell’istruzione offerta a monte di questa esperienza.

Sono grato di aver vissuto questa esperienza, e lo devo a chi mi ha supportato lungo questo percorso.