Alma Mater Studiorum · Università di Bologna

Scuola di Scienze
Corso di Laurea in Informatica per il Management

Progettazione e Sviluppo di Logiche e Processi ETL per Supporto alle Vendite nel Settore Manifatturiero

Relatore:
Prof. Marco Di Felice

Correlatore:
Dott. Alex Incerti

Presentata da:
Elia Cannas

Anno Accademico 2023/2024

Indice

Abstract

L'elaborato analizza l'applicazione della Data Analytics nelle aziende manifatturiere, focalizzandosi sull'innovazione tecnologica a supporto delle vendite. Viene sviluppato un sistema ETL per il monitoraggio delle opportunità di vendita nel settore pneumatico, basato sull'analisi del parco auto circolante e delle performance commerciali. Inoltre, si costruiscono modelli previsionali per identificare articoli vendibili ai clienti attraverso mix di dati storici e di mercato. Infine, mediante l'implementazione di tecniche di machine learning si abilita la previsione delle quantità di acquisto, dimostrando come l'analisi dei dati possa migliorare l'efficacia delle strategie di vendita e fornire un vantaggio competitivo.

Listing

Elenco Figure

Elenco Tabelle

Capitolo 1

Introduzione

L'applicazione della Data Analytics in contesti aziendali ha consentito alle imprese di adottare un approccio "data driven", sfruttando la mole dei dati raccolti durante i processi interni per sviluppare soluzioni e strumenti a supporto delle decisioni strategiche.

La presente tesi si inserisce nel contesto dell'innovazione tecnologica a supporto delle vendite delle aziende manifatturiere, rivestendo un'importanza cruciale per quelle imprese che adottano una visione innovativa e rivoluzionaria al fine di incrementare la propria competitività e rispondere in modo efficiente e tempestivo alle dinamiche di mercato. Lo studio, sviluppato a partire da un tirocinio aziendale presso l'azienda di consulenza Iconsulting, mira a progettare e realizzare strumenti e algoritmi avanzati per supportare i processi di vendita di un'azienda manifatturiera nel settore pneumatico, utilizzando dati relativi ai prodotti aziendali e informazioni di marketing sui clienti.

Il primo obiettivo del progetto è stato lo sviluppo di un sistema ETL (Extract, Transform, Load) mirato a due aspetti fondamentali. In primo luogo, il monitoraggio dei

pneumatici potenzialmente vendibili, basato sull'analisi del parco auto circolante in specifici territori dove operano i clienti dell'azienda. Questo consente di identificare con precisione le opportunità di vendita in base alla domanda locale. In secondo luogo, il sistema ETL è stato utilizzato per monitorare una serie di statistiche di vendita relative ai clienti che acquistano prodotti dall'azienda madre, fornendo un quadro dettagliato delle performance commerciali.

La seconda parte della tesi si concentra sulla costruzione di modelli previsionali destinati a identificare i potenziali articoli vendibili ai clienti, elaborando due differenti "mix" di dati:

- **Customer Mix**: Questo approccio utilizza come fonte di partenza lo storico acquisti dei clienti che acquistano prodotti dall'azienda madre. L'analisi considera sia informazioni sui prodotti (ad esempio, sostituzione di prodotti con dei nuovi modelli, caratteristiche) sia sui clienti (storico degli acquisti). Questo permette di prevedere le esigenze future dei clienti basandosi sui loro comportamenti di acquisto passati e sulle tendenze di prodotto.

- **Market Mix**: In questo caso, l'analisi si focalizza sui dati di mercato di un determinato paese per formulare suggerimenti di acquisto per i clienti. Esaminando le tendenze di mercato e i comportamenti d'acquisto a livello mondiale, si possono ottenere previsioni che riflettono le dinamiche del mercato specifico.

Infine, la tesi presenta una fase sperimentale in cui sono state applicate tecniche di machine learning per sviluppare un modello predittivo delle quantità di acquisto di un determinato cliente. Analizzando dati mensili dal gennaio 2013 al giugno 2024, il modello ha permesso di prevedere le quantità future di acquisti basandosi sull'analisi dello storico.

In conclusione, questo lavoro di tesi rappresenta un significativo contributo alla trasformazione digitale nel settore delle vendite manifatturiere. Attraverso l'implementazione di strumenti ETL avanzati e modelli previsionali, è stato possibile dimostrare come l'analisi dei dati possa migliorare significativamente l'efficienza e l'efficacia delle strategie di vendita, offrendo alle aziende un vantaggio competitivo nel mercato globale.

L'elaborato, nei successivi cinque capitoli, discuterà i diversi aspetti legati alla realizzazione degli strumenti e degli algoritmi a supporto dell'azienda. Nel secondo capitolo, "Stato dell'arte", verranno discusse in maniera più concreta e approfondita le tecnologie utilizzate, analizzando studi e articoli inerenti alle tematiche trattate. Nel terzo capitolo, "Progettazione", verranno esaminati il problema d'origine e le scelte tecnologiche adottate per la sua risoluzione. Il quarto capitolo, "Implementazione", approfondirà l'implementazione, riportando logiche, codice e analisi eseguite per la realizzazione dell'elaborato. Nel quinto capitolo, "Validazione", verranno mostrati i risultati ottenuti per ciascuna delle parti descritte, con la dimostrazione di alcuni casi d'uso. Nell'ultimo capitolo, "Conclusioni e Sviluppi Futuri", si discuteranno i risultati riportati nel capitolo precedente e verranno proposti approcci futuri per migliorare l'accuratezza dei risultati ed estendere l'applicazione di logiche simili ad altri scopi.

Capitolo 2

Stato dell'Arte

Questo capitolo analizza le tecnologie e le metodologie esistenti pertinenti al lavoro di tesi, focalizzandosi su tre macro argomenti principali: la costruzione di sistemi ETL per il monitoraggio dei parchi auto e delle statistiche di vendita, la costruzione di logiche per la creazione di proposte di vendita basate su vari fattori e l'utilizzo di tecniche di machine learning, per la previsione dei trend di acquisto. Questi temi saranno trattati attraverso l'analisi di articoli scientifici e lavori di ricerca che forniscono una panoramica dello stato dell'arte e delle migliori pratiche nel settore.

2.1 Monitoraggio dei Parchi Auto e Statistiche di Vendita

Il monitoraggio e l'analisi dei parchi auto circolanti rappresentano un aspetto cruciale per la conduzione di studi approfonditi in vari ambiti, tra cui il marketing e la sostenibilità ambientale. Questa sezione si propone di esplorare l'importanza del flusso veicolare all'interno di un territorio, evidenziando come tali informazioni possano essere utilizzate in diversi contesti

Parallelamente, l'importanza dei sistemi ETL (Extract, Transform, Load) e dei data warehouse nella Business Intelligence (BI) è ormai consolidata. L'integrazione delle informazioni aziendali attraverso questi strumenti consente di ottenere un valore aggiunto dai dati raccolti durante i processi aziendali, migliorando significativamente la capacità decisionale delle imprese. I data warehouse sono fondamentali per l'aggregazione, l'integrazione e l'analisi dei dati provenienti da fonti eterogenee, permettendo alle aziende di ottenere una visione dettagliata e integrata delle proprie informazioni. La capacità di gestire informazioni complesse e produrre report analitici efficaci rende i data warehouse una componente essenziale per il successo delle operazioni aziendali basate sui dati.

2.1.1 Importanza del Monitoraggio e Analisi dei Parchi Auto Circolanti

Un parco auto circolante è un concetto che si riferisce all'insieme dei veicoli registrati e attivamente in circolazione all'interno di un determinato territorio, sia urbano che extraurbano. Questa definizione si estende a diverse categorie di veicoli, tra cui automobili, furgoni, camion e motocicli, ciascuna delle quali contribuisce in modo differente alla mobilità e alla logistica di una regione.

L'analisi del parco auto circolante riveste un'importanza cruciale per diverse ragioni:

1. **Indicatori Economici e Sociali**: Il parco auto circolante è un indicatore chiave della mobilità di una popolazione. Un aumento del numero di veicoli può riflettere una crescita economica e una maggiore disponibilità di reddito, mentre una diminuzione può suggerire problematiche economiche o cambiamenti nelle abitudini di trasporto.

2. **Pianificazione Urbanistica e Infrastruttur**: Le autorità locali e regionali utilizzano i dati relativi al parco auto circolante per pianificare lo sviluppo delle

infrastrutture di trasporto. Ciò include la costruzione di strade, parcheggi, e il potenziamento dei trasporti pubblici. Una comprensione accurata delle dinamiche del parco auto circolante consente di anticipare le necessità future di mobilità e di ridurre la congestione del traffico.

3. **Sostenibilità Ambientale e Infrastrutture**: Le autorità locali e regionali utilizzano i dati relativi al parco auto circolante per pianificare lo sviluppo delle infrastrutture di trasporto. Ciò include la costruzione di strade, parcheggi, e il potenziamento dei trasporti pubblici. Una comprensione accurata delle dinamiche del parco auto circolante consente di anticipare le necessità future di mobilità e di ridurre la congestione del traffico.

4. **Sostenibilità Ambientale**: La composizione del parco auto circolante ha ripercussioni dirette sull'ambiente. L'aumento di veicoli a combustione interna è associato a un incremento delle emissioni di gas serra e inquinanti atmosferici. Al contrario, una crescente adozione di veicoli elettrici o ibridi può contribuire a ridurre l'impatto ambientale del trasporto. Pertanto, monitorare la transizione verso veicoli più sostenibili è essenziale per le politiche ambientali.

5. **Politiche di Mobilità**: Le decisioni politiche riguardanti il trasporto pubblico, la gestione del traffico e la promozione di forme di mobilità alternativa (come biciclette e car-sharing) sono spesso guidate dai dati sul parco auto circolante. Le autorità possono implementare misure per incentivare l'uso di veicoli a basse emissioni o per promuovere la mobilità sostenibile in risposta ai cambiamenti nelle abitudini di trasporto.

6. **Implicazioni per la Sicurezza Stradale**: La dimensione e la composizione del parco auto circolante possono influenzare anche gli aspetti legati alla sicurezza

stradale. Un aumento del numero di veicoli può portare a un maggior numero di incidenti stradali, rendendo necessarie politiche di sicurezza e campagne di sensibilizzazione.

In conclusione, il parco auto circolante è un elemento fondamentale per comprendere e gestire le dinamiche di mobilità di una società. La sua analisi permette di trarre conclusioni importanti su aspetti economici, sociali e ambientali, fornendo un quadro utile per le politiche di pianificazione e gestione del territorio.

Prendendo in considerazione l'applicazione dello studio in due situazioni specifiche, si discute il primo esempio all'interno di un contesto di ricerca ecosostenibile condotto da Mrinal K. Ghose, R. Paul e S.K. Banerjee [8]. L'articolo analizza come la composizione del parco auto circolante in Cina influenzerà le future emissioni di carbonio. Utilizzando dati dettagliati sui veicoli in circolazione, gli autori sviluppano scenari di emissioni future basati su diversi tassi di crescita del parco veicoli, sull'introduzione di tecnologie più pulite e sulle politiche di gestione ambientale.

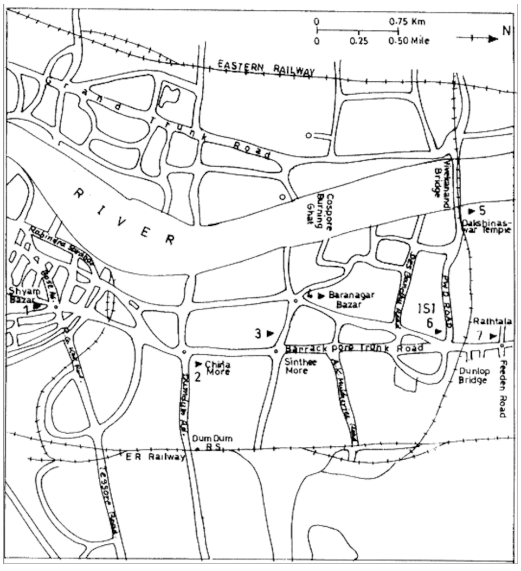

Figura 2.1: Analisi Parco Auto Circolante [8]

Il secondo esempio da riferimento all'articolo di Nowak e Kowalski (2022), intitolato "Comparative Analysis of Selected Car Parks" [3], presenta un'analisi comparativa di diversi tipi di parcheggi, focalizzandosi sull'impatto ambientale e sull'efficienza operativa delle soluzioni di parcheggio. Gli autori esaminano vari modelli di parcheggio, evidenziando le loro caratteristiche distintive e le prestazioni in contesti urbani specifici. Inoltre, lo studio integra l'analisi del parco auto circolante, discutendo come la crescente flotta di veicoli influisca sulla progettazione e gestione dei parcheggi. Attraverso un approccio analitico e l'uso di metriche quantitative, l'articolo fornisce indicazioni utili per la progettazione di parcheggi sostenibili, contribuendo a una migliore gestione dello spazio urbano e a una riduzione dell'impatto ecologico associato alla mobilità. Questa ricerca si rivela di grande rilevanza per le autorità locali e i pianificatori urbani, poiché offre spunti per migliorare le politiche di parcheggio e affrontare le sfide legate alla congestione del traffico nelle aree urbane.

2.1.2 Data Warehouse nella Business Intelligence

L'utilizzo dei sistemi ETL (Extract, Transform, Load) risulta essere di particolare importanza anche per la Business Intelligence, rappresentando una pietra miliare nella gestione e nell'analisi dei dati aziendali. Questi sistemi consentono di estrarre informazioni da diverse fonti di dati, trasformarle in un formato utile e caricarle in un data warehouse, dove possono essere facilmente accessibili per l'analisi. Tale processo di integrazione dei dati è cruciale, poiché le aziende spesso operano con informazioni sparse in molteplici sistemi e formati, rendendo difficile ottenere una visione unificata e coerente delle loro operazioni.

Punti chiave dei sistemi ETL:

1. **Estrazione (Extract)**: Processo di acquisizione dei dati da diverse fonti, tra cui database, file, API e sistemi legacy. Questa fase è fondamentale per raccogliere

informazioni disperse in vari formati e sistemi.

2. **Trasformazione (Transform)**: Fase in cui i dati estratti vengono elaborati e convertiti in un formato coerente e utile. Questo può includere la pulizia dei dati, la normalizzazione, l'arricchimento e l'applicazione di regole aziendali specifiche.

3. **Caricamento (Load)**: Fase finale del processo ETL, in cui i dati trasformati vengono caricati in un data warehouse o in un altro sistema di archiviazione per facilitare l'analisi e la reportistica.

4. **Integrazione dei Dati**: L'ETL consente di combinare dati provenienti da diverse fonti, fornendo una visione unificata delle operazioni aziendali e migliorando la qualità delle informazioni disponibili.

5. **Qualità dei Dati**: Attraverso il processo di trasformazione, i sistemi ETL garantiscono che i dati siano accurati, completi e coerenti, riducendo il rischio di errori e migliorando la fiducia nelle analisi effettuate.

6. **Supporto alla Business Intelligence**: I sistemi ETL sono essenziali per le iniziative di Business Intelligence, poiché forniscono le basi per reportistica, dashboard e analisi dei dati, permettendo decisioni aziendali informate.

7. **Efficienza Operativa**: L'automazione dei processi ETL riduce il tempo e gli sforzi necessari per gestire e analizzare i dati, consentendo alle aziende di rispondere più rapidamente alle esigenze del mercato.

8. **Adattabilità alle Normative**: I sistemi ETL aiutano le aziende a conformarsi a normative di protezione dei dati, garantendo la gestione e l'archiviazione sicura delle informazioni sensibili.

9. **Analisi Predittiva**: L'integrazione e la trasformazione dei dati consentono l'implementazione di modelli analitici avanzati, facilitando previsioni e strategie proattive.

10. **Scalabilità**: I sistemi ETL possono essere progettati per gestire un aumento del volume dei dati e per adattarsi alle crescenti esigenze di un'organizzazione in espansione.

Attraverso l'integrazione delle informazioni contenute nelle basi di dati con strumenti ETL e strumenti di reportistica, è possibile ottenere un valore aggiunto dai dati registrati dalle aziende durante i processi aziendali. Ad esempio, i dati relativi agli acquisti e alle vendite possono essere analizzati per identificare tendenze di mercato, comportamenti dei clienti e opportunità di miglioramento. Le aziende possono così non solo monitorare le loro performance storiche, ma anche effettuare previsioni e prendere decisioni strategiche più informate.

Inoltre, i sistemi ETL favoriscono la pulizia e la normalizzazione dei dati, garantendo che le informazioni siano accurate e consistenti. Questo è particolarmente importante in un contesto in cui le aziende devono conformarsi a normative sempre più stringenti riguardanti la gestione dei dati e la privacy. Un data warehouse ben progettato, alimentato da un processo ETL efficace, consente di generare report dettagliati e dashboard interattive, facilitando l'analisi e la visualizzazione delle informazioni chiave.

Come dimostrato dallo studio condotto da T. Jun, C. Kai, F. Yu e T. Gang [11], i data warehouse svolgono un ruolo cruciale nei sistemi di business intelligence (BI). Questi sistemi facilitano l'aggregazione, l'integrazione e l'analisi dei dati provenienti da fonti disparate, migliorando la capacità delle aziende di prendere decisioni basate sui dati. Lo studio sottolinea che i data warehouse sono una componente essenziale delle architetture di business intelligence moderne, offrendo strumenti per una visione dettagliata e integrata dei dati aziendali.

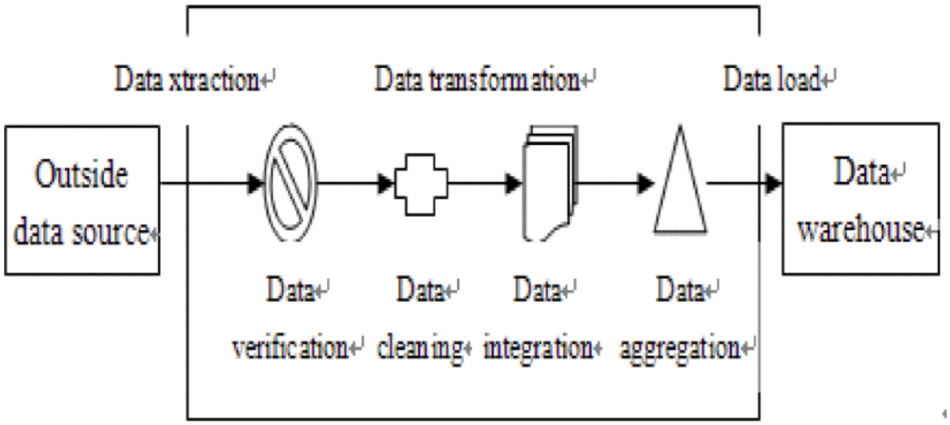

Figura 2.2: Creazione Data Warehouse [11]

L'articolo di Apanowicz (2009), intitolato "Data Warehousing and Business Intelligence: Benchmark Project for the Platform Selection" [2] esplora le sfide e le opportunità associate alla selezione di piattaforme per il data warehousing e la business intelligence (BI). Nell'ambito della teoria dei database e delle applicazioni, l'autore discute l'importanza di un approccio sistematico nella scelta delle piattaforme, enfatizzando come una selezione appropriata possa migliorare l'efficacia delle strategie di BI e ottimizzare i processi decisionali all'interno delle organizzazioni. Attraverso un progetto di benchmark, il documento offre una panoramica dettagliata delle diverse piattaforme disponibili, analizzando le loro caratteristiche, i punti di forza e le debolezze. Questo lavoro rappresenta una risorsa utile per i professionisti e i ricercatori che cercano di comprendere meglio il panorama delle tecnologie di data warehousing e BI, fornendo indicazioni pratiche per una scelta informata.

2.2 Sviluppo di Logiche per il Suggerimento di Proposte di Acquisto

La capacità di creare proposte di vendita efficaci, basate su vari fattori come lo storico degli ordini e le tendenze di mercato, è cruciale per le aziende che desiderano migliorare la loro competitività. In un contesto commerciale sempre più dinamico e competitivo, le aziende devono adottare approcci strategici e informati per differenziarsi dai concorrenti e attrarre clienti. Questo è particolarmente importante in un'epoca in cui le aspettative dei consumatori stanno evolvendo rapidamente, e dove la personalizzazione e la capacità di anticipare le esigenze dei clienti sono diventate fattori chiave per il successo.

L'analisi dei dati storici degli acquisti dei clienti è emersa come una pratica cruciale per le strategie di marketing nel contesto contemporaneo. Comprendere i modelli di acquisto passati non solo aiuta le aziende a prevedere le esigenze future dei clienti, ma

consente anche di personalizzare le offerte e migliorare la fedeltà del cliente. Le informazioni derivanti dagli storici degli acquisti possono fornire approfondimenti significativi sui comportamenti di consumo, facilitare la segmentazione dei clienti e ottimizzare le campagne promozionali.

Un panorama più ampio mostra che l'analisi dei dati storici non è solo una questione di identificazione di tendenze; si tratta di costruire un ecosistema informativo che aiuti le aziende a comprendere a fondo il proprio mercato. Le aziende possono utilizzare tecnologie avanzate come l'intelligenza artificiale e il machine learning per analizzare grandi volumi di dati e identificare schemi che non sarebbero visibili con metodi di analisi tradizionali. Questo approccio consente di prevedere con maggiore precisione il comportamento dei clienti e di adeguare le strategie di marketing di conseguenza.

La segmentazione dei clienti diventa quindi un processo più sofisticato, consentendo alle aziende di sviluppare offerte specifiche per diversi gruppi di clienti, a seconda delle loro preferenze e dei loro comportamenti di acquisto. Questa personalizzazione non solo migliora l'efficacia delle campagne di marketing, ma contribuisce anche a costruire relazioni più forti e significative con i clienti, aumentando la loro lealtà e soddisfazione.

Inoltre, l'ottimizzazione delle campagne promozionali basate sui dati storici degli acquisti consente alle aziende di allocare in modo più efficiente le risorse di marketing. Invece di investire in strategie generali che potrebbero non raggiungere il pubblico desiderato, le aziende possono indirizzare i loro sforzi verso canali e messaggi che hanno dimostrato di generare risultati positivi in passato. Ciò porta non solo a un aumento delle vendite, ma anche a una maggiore redditività, poiché le risorse vengono utilizzate in modo più strategico.

Un contributo fondamentale a questo campo è rappresentato dallo studio di Kumar e Shah (2004) [13], intitolato "Customer Purchase History and Its Effect on Customer

Loyalty", pubblicato nel *Journal of Marketing*. Questo lavoro esamina come i dati storici degli acquisti influenzino la fedeltà dei clienti e suggerisce che l'analisi di tali dati può portare a strategie di marketing altamente personalizzate e più efficaci. Tramite modelli statistici è possibile valutare la relazione tra i comportamenti di acquisto passati e la fedeltà futura, gli autori dimostrano che clienti con acquisti frequenti e di alto valore tendono a rimanere più fedeli. Le implicazioni dello studio offrono indicazioni pratiche per l'implementazione di programmi di fedeltà e promozioni mirate, rendendo questo articolo un riferimento cruciale per l'uso dei dati degli acquisti nella strategia di marketing.

L'adozione di approcci "data-driven" in contesti aziendali, come evidenziato da numerosi studi, offre significativi vantaggi nel campo del marketing. In particolare, l'articolo di Ettl, Harsha, Papush e Perakis, intitolato "A Data-Driven Approach to Personalized Bundle Pricing and Recommendation" [5], spiega come l'integrazione della disponibilità dei prodotti nelle raccomandazioni basate sui dati possa migliorare la precisione e la rilevanza delle raccomandazioni per i clienti. Questo lavoro fornisce una panoramica fondamentale su come le aziende possano sfruttare i dati per affinare le loro strategie di raccomandazione e ottimizzare le vendite.

Figura 2.3: Data-Driven

Nella creazione di proposte di acquisto, esistono diverse strategie per formulare un documento efficace. Il documento "Guide to Proposal Planning and Writing" [7] della Florida State University fornisce una visione dettagliata e metodica sulla redazione di proposte, in particolare per progetti di ricerca e finanziamenti. Analizza le fasi cruciali della pianificazione, dalla definizione degli obiettivi alla creazione di un budget appropriato. Inoltre, offre raccomandazioni pratiche per una presentazione chiara e persuasiva, evidenziando l'importanza di un approccio sistematico nella scrittura delle proposte.

2.3 Predizione di Volumi di Acquisto tramite Tecniche di Machine Learning

Le tecniche innovative di Machine Learning hanno permesso alle aziende di ottenere una visione alternativa e più approfondita rispetto alle analisi tradizionali. Utilizzando librerie come Prophet, è possibile prevedere con precisione l'evoluzione dei trend temporali nel corso del tempo, offrendo strumenti avanzati per l'analisi dei dati e migliorando la capacità decisionale in vari contesti aziendali.

2.3.1 Prophet

Prophet [1] è un tool di previsione sviluppato da Facebook, progettato per gestire serie temporali che presentano tendenze non lineari e stagionalità. È stato rilasciato come software open source e si distingue per la sua capacità di fornire previsioni accurate in presenza di dati storici complessi e con caratteristiche specifiche come trend non lineari, effetti stagionali annuali, settimanali e giornalieri, e la presenza di vacanze e giorni speciali.

[1]Prophet: `https://facebook.github.io/prophet/`

Prophet risulta particolarmente utile per le seguenti caratteristiche:

1. **Modellazione della Stagionalità**: Include componenti stagionali che possono essere adattati in base ai dati storici, rendendolo adatto per dati con tendenze annuali, mensili, settimanali e giornaliere.

2. **Gestione dei Dati Mancanti**: Ha la capacità di gestire automaticamente i punti dati mancanti e gli outlier, rendendo il modello robusto in caso di dataset incompleti.

3. **Flessibilità**: Permette l'inclusione di informazioni esterne come le festività, che possono influenzare le previsioni.

4. **Interfaccia Intuitiva**: È progettato per essere utilizzato sia da esperti di dati che da analisti aziendali, con un'interfaccia facile da usare e integrata con linguaggi di programmazione comuni come Python e R.

Prophet utilizza un approccio additivo per decomporre la serie temporale nei suoi componenti fondamentali: trend, stagionalità e festività. Questo approccio lo rende particolarmente efficace per applicazioni aziendali come la previsione delle vendite, la pianificazione della produzione e la gestione delle scorte.

2.3.2 ARIMA

ARIMA (AutoRegressive Integrated Moving Average)[2] è un modello statistico ampiamente riconosciuto e utilizzato nell'analisi delle serie temporali, che combina tre componenti principali: autoregressione (AR), differenziazione integrata (I) e media mobile (MA). Questo modello è progettato per affrontare dati temporali caratterizzati

[2]ARIMA: `https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average`

da tendenze e cicli, risultando particolarmente efficace quando le serie temporali sono stazionarie o possono essere trasformate in tale condizione attraverso processi di differenziazione.

ARIMA si distingue per le seguenti caratteristiche:

1. **Modellazione della Stazionarietà**: ARIMA richiede che la serie temporale sia stazionaria. In presenza di trend o stagionalità, è necessario rimuoverli tramite differenziazione o appropriati interventi di trasformazione.

2. **Complessità e Flessibilità**: Il modello offre la possibilità di modellare una vasta gamma di pattern temporali attraverso la combinazione di parametri AR, I e MA, consentendo un adattamento a diverse dinamiche di dati.

3. **Previsione a Breve Termine**: ARIMA si dimostra particolarmente efficace per le previsioni a breve termine, grazie alla sua capacità di catturare relazioni temporali intrinseche nei dati.

4. **Diagnosi e Verifica dei Modelli**: Il modello fornisce strumenti per la diagnosi della qualità della modellazione, consentendo un'analisi approfondita dei residui per verificare l'adeguatezza del modello.

Il modello ARIMA adotta un approccio parametrico per descrivere la serie temporale, perseguendo l'obiettivo di identificare le relazioni tra i dati storici e le previsioni future. Tale caratteristica lo rende utile in vari ambiti, inclusi l'economia, la finanza e la gestione della produzione, per attività quali la previsione della domanda e l'analisi delle vendite.

2.3.3 Contesti di Utilizzo di Prophet e ARIMA

Le capacità additive di questi modelli li portano ad essere particolarmente utili in diversi contesti.

Per citare alcuni dei lavori svolti con Prophet, un articolo intitolato "*Comparing Prophet and Deep Learning to ARIMA in Forecasting Wholesale Food Prices*" [14] ha confrontato Prophet con altri modelli di previsione come ARIMA e algoritmi di deep learning per la previsione dei prezzi all'ingrosso di alimenti. L'obiettivo era valutare l'accuratezza e l'efficacia dei diversi modelli nella previsione dei prezzi medi settimanali di prodotti. In questo studio, Prophet è stato utilizzato per modellare una serie temporale decomponibile in componenti di trend, stagionalità e festività. Questo ha permesso di catturare e prevedere i trend non lineari nei prezzi alimentari, migliorando la precisione delle previsioni settimanali rispetto ai metodi tradizionali come ARIMA. Prophet si è dimostrato particolarmente efficace nella gestione di dataset con dati mancanti, senza necessità di interpolazione, e ha fornito previsioni affidabili per prodotti con una lunga storia di vendite e vendite frequenti.

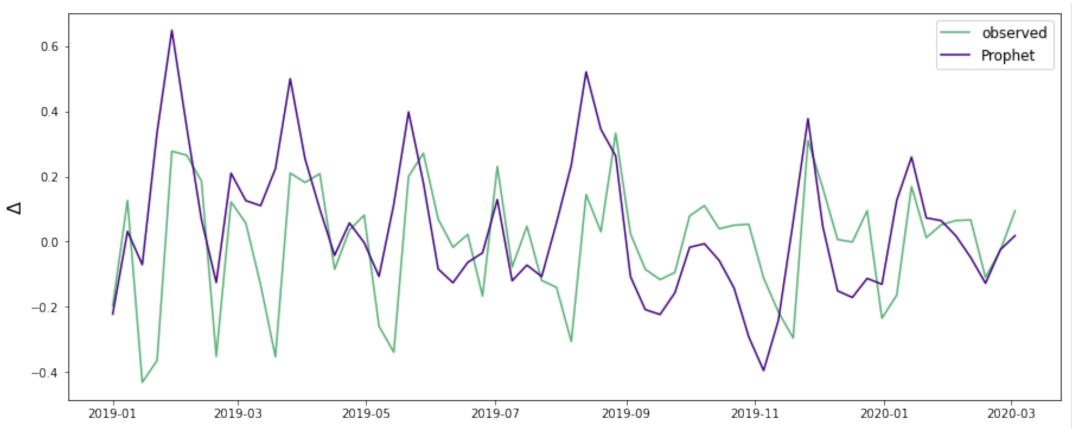

Figura 2.4: Confronto previsioni Prophet e ARIMA [15]

Un'ulteriore studio condotto tramite l'utilizzo di Prophet, intitolato "*Financial Time Series Forecasting Using Prophet*"[17] ha utilizzato Prophet per prevedere i tassi di crescita dei mercati finanziari. In particolare, lo studio ha analizzato le serie temporali dei tassi di mercato di vari indici azionari, dimostrando come Prophet possa essere impiegato per modellare con precisione le fluttuazioni e i trend stagionali dei mercati finanziari, migliorando così la capacità di previsione e gestione degli investimenti.

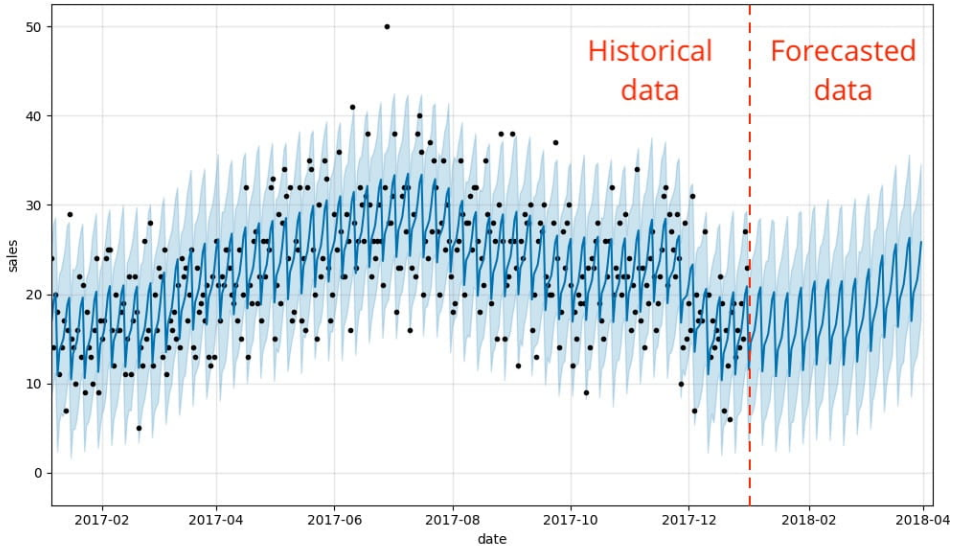

Figura 2.5: Analisi serie temporali con Prophet [18]

Per fornire un contesto ulteriore sull'utilizzo di Prophet, consideriamo lo studio condotto da Duarte, Diego, Walshaw, Chris e Ramesh, Nadarajah, intitolato "*A Comparison of Time-Series Predictions for Healthcare Emergency Department Indicators and the Impact of COVID-19*" [4]. Questo articolo esamina l'applicazione di modelli di previsione per la gestione dei Key Performance Indicators (KPIs) nelle strutture sanitarie, un settore particolarmente sollecitato e aggravato dalla pandemia di COVID-19. I KPIs, che sono frequentemente rappresentati da dati temporali, sono essenziali per la pianificazione delle unità di emergenza, il miglioramento della qualità delle cure e l'ottimizzazione delle risorse. L'articolo confronta tre modelli di previsione dei KPIs: l'Autoregressive

Integrated Moving Average (ARIMA), Prophet e il General Regression Neural Network (GRNN). Utilizzando un dataset proveniente da un ospedale del Regno Unito, che include indicatori orari come il numero di pazienti nel dipartimento, le presenze, i pazienti non allocati con Decision to Admit (DTA) e i pazienti pronti per la dimissione, lo studio evidenzia schemi regolari e tendenze stagionali, influenzati anche da fattori esterni come il clima o incidenti significativi, con il COVID-19 che rappresenta un esempio estremo di cambiamenti drammatici nel comportamento dei dati.

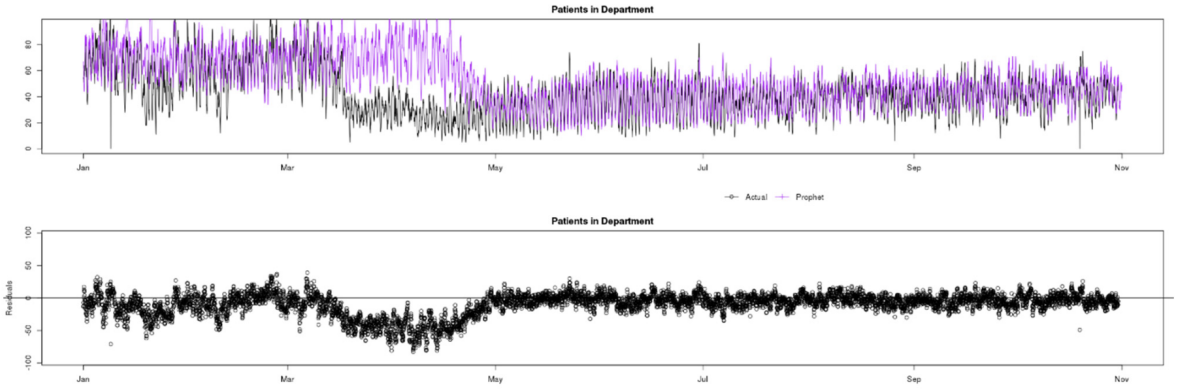

Figura 2.6: Confronto serie attuali e predizioni con Prophet [4]

Citando uno dei lavori svolti con il modello ARIMA, si prende in considerazione l'articolo intitolato "ARIMA Model for Accurate Time Series Stocks Forecasting" [12] di Khan e Alghulaiakh esplora l'applicazione del modello ARIMA (AutoRegressive Integrated Moving Average) nella previsione dei prezzi delle azioni. Gli autori sottolineano l'importanza di una previsione accurata dei prezzi delle azioni, cruciale per investitori e stakeholder che desiderano prendere decisioni informate.

Nel loro studio, i ricercatori delineano la metodologia adottata per implementare il modello ARIMA, inclusa la selezione dei parametri e le tecniche di validazione del modello. L'articolo mette in evidenza le performance del modello rispetto ad altri metodi di previsione, dimostrando la sua efficacia nel catturare le tendenze e i pattern sottostanti

nei dati dei prezzi delle azioni.

Gli autori forniscono un'analisi approfondita dei risultati ottenuti, enfatizzando le capacità predittive del modello e le sue potenziali implicazioni per i mercati finanziari. Questa ricerca contribuisce al dibattito in corso sull'affidabilità dei modelli statistici nel complesso ambito della previsione del mercato azionario.

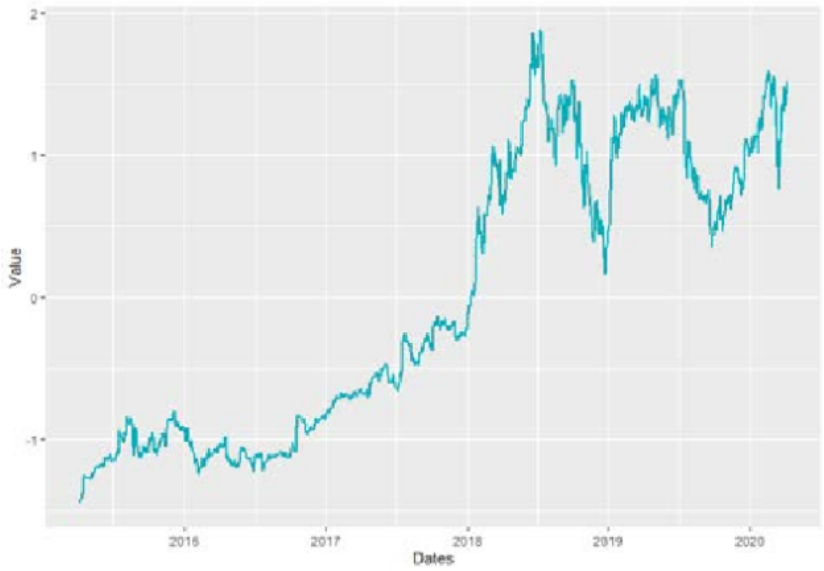

Figura 2.7: Previsione Temporale ARIMA [12]

In conclusione, Prophet e ARIMA si dimostrano due modelli di previsione altamente versatili e precisi, adatti a diversi contesti applicativi. La loro capacità di gestire i dati, li rende particolarmente efficaci nel fornire previsioni affidabili e dettagliate in scenari complessi e variabili.

2.4 Strategie di Vendita nel Settore Pneumatico

Nell'ambito di un elaborato finalizzato alla creazione di una strategia di vendita per un'azienda operante nel settore dei pneumatici, è stata condotta un'analisi approfondita di diversi studi e ricerche esistenti, volti a comprendere il panorama complessivo che

caratterizza la vendita di questi prodotti specifici. La letteratura disponibile evidenzia non solo le dinamiche di mercato, ma anche le preferenze dei consumatori e le tendenze emergenti che influenzano le decisioni d'acquisto.

In particolare, sono stati esaminati vari modelli di vendita e marketing, con un focus su come le aziende possono adattarsi alle sfide contemporanee e ottimizzare le loro strategie per soddisfare le esigenze del mercato. Gli studi rivelano che la personalizzazione delle offerte e un forte orientamento al cliente sono fattori chiave per il successo nella vendita di pneumatici. Inoltre, l'importanza di integrazioni tecnologiche, come l'e-commerce e l'analisi dei dati, è emersa come elemento cruciale per sviluppare un approccio competitivo.

L'articolo "Tire Sales and Marketing Strategies in the 21st Century" di Sharma, R.,Singh, M. (2008) [16] fornisce un'analisi approfondita delle dinamiche di vendita e delle strategie di marketing nel settore dei pneumatici. Esplora come le aziende di pneumatici possano adattarsi alle mutevoli esigenze del mercato, evidenziando l'importanza di comprendere le preferenze dei consumatori e le innovazioni tecnologiche emergenti. Il documento discute le sfide attuali e le opportunità che le aziende devono affrontare per mantenere la loro competitività, enfatizzando un approccio orientato al cliente e la necessità di implementare pratiche sostenibili nel settore.

Inoltre, l'articolo offre raccomandazioni pratiche su come le aziende possono migliorare le loro strategie di marketing, sottolineando la rilevanza della sostenibilità e dell'innovazione nel contesto attuale. Questo studio rappresenta una risorsa preziosa per i professionisti del settore, fornendo intuizioni utili per guidare le decisioni strategiche e promuovere una crescita sostenibile nel mercato dei pneumatici.

Osservando in maniera generica il panorama del settore pneumatico, si prende in considerazione il report di IBISWorld, intitolato "Tire Dealers in the US - Market Research Report" [9], fornisce un'analisi dettagliata del settore dei rivenditori di pneumatici negli

Figura 2.8: Strategie di Vendita nel Settore Pneumatico

Stati Uniti, evidenziando le tendenze di mercato e le dinamiche competitive. L'articolo esamina vari fattori che influenzano il settore, come le fluttuazioni nei costi delle materie prime, le innovazioni tecnologiche e le strategie di distribuzione. Inoltre, il report analizza l'importanza della customer experience e della fidelizzazione del cliente nel processo di vendita. Questa risorsa si rivela utile per i professionisti del settore, poiché offre dati e approfondimenti fondamentali per orientare le decisioni strategiche e ottimizzare le operazioni nel mercato dei pneumatici.

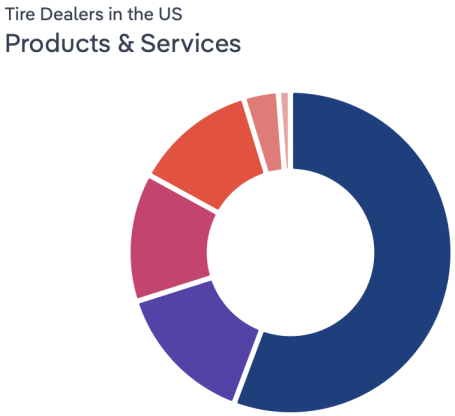

Figura 2.9: Rivenditori di Pneumatici negli US

2.5 Conclusioni

Il capitolo ha offerto una panoramica esaustiva delle tecnologie e metodologie disponibili per il supporto alle decisioni, evidenziando l'importanza degli approcci basati sui dati per guidare le scelte strategiche in vari settori. In un contesto in continua evoluzione, dove le aziende e le organizzazioni si trovano a fronteggiare sfide complesse e dinamiche, la capacità di prendere decisioni informate e tempestive è diventata un fattore cruciale per il successo. Gli approcci data-driven non solo migliorano la qualità delle decisioni, ma consentono anche di identificare opportunità nascoste e di ottimizzare le risorse disponibili.

Le applicazioni di tali approcci sono numerose e variegate, come illustrato negli esempi precedenti, che spaziano dall'analisi predittiva per il marketing alla pianificazione delle vendite e alla gestione delle operazioni. Questi esempi dimostrano come l'integrazione di dati storici, tendenze di mercato e modelli analitici possa tradursi in strategie più efficaci e mirate. Le aziende che adottano metodologie di supporto alle decisioni possono rispondere in modo più agile e proattivo alle esigenze del mercato, migliorando così la loro competitività e sostenibilità.

Inoltre, l'impiego di metodi innovativi, come l'uso di Prophet e ARIMA, arricchisce le analisi, fornendo una prospettiva dettagliata e all'avanguardia tanto in ambito aziendale quanto in altri contesti. Prophet e ARIMA, in particolare, si distinguono per la loro capacità di gestire serie temporali complesse e di adattarsi a variabili esterne, rendendoli uno strumento prezioso per le previsioni a lungo termine.

In conclusione, il capitolo sottolinea come le tecnologie e le metodologie per il supporto alle decisioni non siano semplicemente strumenti tecnici, ma rappresentino un cambiamento paradigmatico nel modo in cui le organizzazioni affrontano le sfide strategiche. La crescente disponibilità di dati, unita a metodi analitici avanzati, offre alle aziende l'opportunità di sviluppare un vantaggio competitivo sostenibile, di migliorare le

performance operative e di garantire un allineamento più stretto tra le strategie aziendali e le esigenze del mercato.

Capitolo 3

Progettazione

Il presente capitolo è dedicato alla descrizione dettagliata della fase di progettazione del presente lavoro di tesi. In esso verranno delineati i principali aspetti del problema iniziale che ha motivato lo studio, nonché le soluzioni proposte per affrontare i vari punti critici identificati. La sezione introduttiva del capitolo fornirà una panoramica del problema di base, evidenziandone le implicazioni teoriche e pratiche. Successivamente, si procederà con la spiegazione delle soluzioni concepite, le quali sono state accuratamente pianificate per rispondere in modo efficiente ed efficace alle sfide poste dal problema. Inoltre, verranno descritte le tecnologie adottate durante il processo di progettazione e sviluppo. Questo include sia gli strumenti software e hardware utilizzati, sia le piattaforme e i framework che hanno facilitato l'implementazione delle soluzioni proposte. Infine, verranno illustrati i metodi sperimentali impiegati per predire informazioni utili ai fini delle analisi svolte. Questa parte comprenderà una descrizione dei protocolli sperimentali, dei criteri di valutazione e delle modalità di analisi dei dati raccolti. L'obiettivo di questo capitolo è fornire una visione chiara e completa del percorso progettuale intrapreso, evidenziando la coerenza e la robustezza delle metodologie adottate nella risoluzione dei problemi iniziali.

3.1 Obiettivi

Nel paragrafo seguente verranno delineati gli obiettivi del presente elaborato redato in relazione al tirocinio azidele, suddivisi nelle tre parti fondamentali trattate al suo interno.

3.1.1 Obiettivi di Reportistica

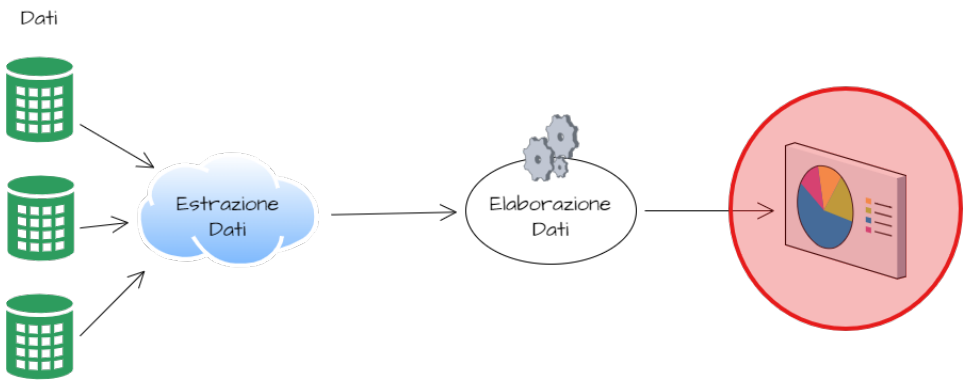

Figura 3.1: Obiettivo Reportistica

Il primo obiettivo dello studio consiste nello sviluppo di un sistema avanzato utilizzando strumenti di ETL e reportistica, finalizzato alla gestione di due compiti fondamentali:

- Progettazione di uno strumento di reportistica per la gestione e l'analisi del parco auto circolante all'interno di un determinato territorio.
- Creazione di uno strumento di reportistica per la gestione e l'analisi dei dati di vendita e degli obiettivi orientati verso i clienti di un'azienda manifatturiera.

Analizzando nello specifico il primo punto, l'obiettivo è sviluppare, attraverso strumenti di ETL, una soluzione che integri diverse sorgenti di dati per creare una piattaforma di reportistica avanzata. Questa piattaforma permetterà all'azienda manifatturiera di monitorare i propri clienti dal punto di vista degli acquisti e delle vendite. La piattaforma fornirà un'analisi dettagliata delle vendite di pneumatici effettuate, presentando i dati

suddivisi secondo diverse metriche temporali. Inoltre, sarà inclusa una funzionalità di geolocalizzazione per i punti vendita dei clienti, con una rappresentazione su mappa che evidenzia le aree del territorio coperte complessivamente.

Per quanto concerne il secondo compito, l'obiettivo specifico è sviluppare, utilizzando anch'esso strumenti di ETL, una soluzione che integri diverse sorgenti di dati. Questa soluzione consentirà di visualizzare attraverso la piattaforma sviluppata le statistiche relative ai clienti, offrendo una panoramica dettagliata delle vendite in tempo reale, del monitoraggio del target di vendita e di altre informazioni pertinenti al marketing. La piattaforma sarà inoltre dotata di una pagina dedicata che permetterà di confrontare gli acquisti effettuati dai clienti e le giacenze di magazzino con i dati dell'anno precedente, fornendo così un'analisi comparativa utile per valutare le performance e le tendenze nel tempo.

3.1.2 Obiettivi Logiche di Supporto

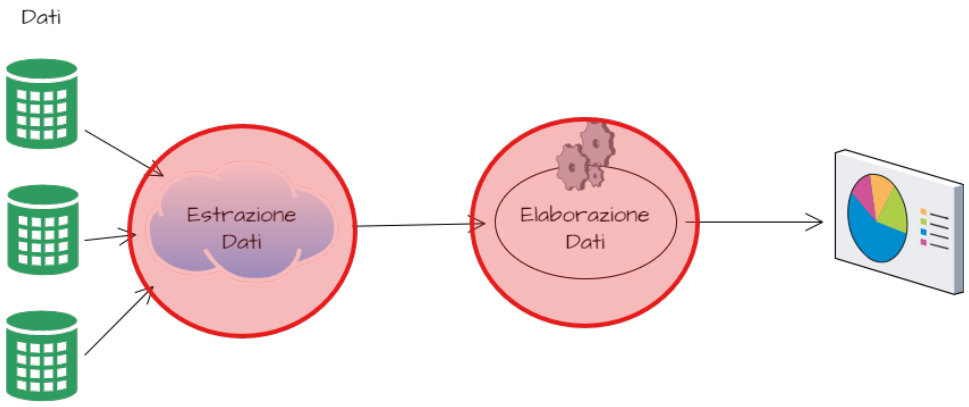

Figura 3.2: Obiettivo Logiche di Supporto

Gli obiettivi relativi alle logiche di supporto rappresentano il fulcro della realizzazione di questo elaborato. La progettualità prevede lo sviluppo di strategie per la proposta

di vendita ai clienti dell'azienda, finalizzate a suggerire quali pneumatici acquistare e in quale quantità, elaborando una sorta di preventivo di vendita. Questo strumento, generalmente utilizzato dal marketing dell'azienda cliente, ha l'obiettivo di diversificare le vendite e incrementare il fatturato dell'azienda manifatturiera, puntando sulla varietà dei prodotti offerti e sull'aumento degli articoli venduti. Le logiche alla base della creazione di tali proposte si basano su due algoritmi distinti. Ogni algoritmo è associato a un peso specifico, che consente di generare una proposta personalizzata per ciascun cliente. Questo approccio mirato permette di ottimizzare le raccomandazioni di vendita, migliorare la soddisfazione del cliente e potenziare le opportunità di vendita per l'azienda.

3.1.3 Obiettivi Predizione Volumi di Acquisto

Nell'ultima parte di questa sezione, si analizzano gli obiettivi relativi alla sezione sperimentale. Il focus dell'ultimo studio condotto è sull'impiego dei dati storici degli acquisti di un cliente, riferiti agli anni dal 2013 al 2024. L'obiettivo è utilizzare tecniche di Machine Learning per stimare i valori di acquisto del cliente nei mesi futuri. Questa stima ha lo scopo di migliorare le prestazioni degli algoritmi precedentemente discussi, ottimizzando così le previsioni e le proposte di vendita basate sui dati storici.

3.2 Requisiti

All'interno del seguente paragrafo veranno descritti i requisiti necessari ai fini del completamento degli obiettivi elencanti precedentemente.

3.2.1 Requisiti Reportistica

Requisiti Funzionali

Analizzando i requisiti funzionali relativi al primo obiettivo del lavoro di tesi, emergono i seguenti aspetti:

1. **Integrazione delle Sorgenti Dati**

   - **ETL:** La piattaforma deve essere in grado di estrarre dati da diverse sorgenti, trasformarli in un formato coerente e caricarli in un database centralizzato.
   - **Gestione Dati:** Deve supportare la connessione e l'integrazione con vari sistemi di origine dei dati, quali database, file CSV, API esterne, ecc.

2. **Generazione Reportistica**

   - **Analisi Parco Auto:** La piattaforma deve fornire report dettagliati sull'analisi del parco auto circolante, includendo informazioni sui rivenditori, sui clienti e sul livello di potenziale per ogni zona.
   - **Analisi Vendite Pneumatici:** Deve consentire la creazione di report relativi alle vendite di pneumatici per i clienti dell'azienda manifatturiera, utilizzando scale temporali per il confronto delle vendite.
   - **Obiettivi di Vendita:** La piattaforma deve generare report sull'andamento rispetto agli obiettivi di vendita stabiliti, rappresentandoli attraverso grafici.

3. **Visualizzazione Dati**

   - **Geolocalizzazione:** Deve fornire una mappa interattiva che rappresenti i punti vendita dei clienti e la loro distribuzione territoriale.

- **Dashboard Personalizzabile:** La piattaforma deve permettere la creazione di dashboard personalizzabili per visualizzare i dati e i report secondo le preferenze dell'utente.

4. **Filtri e Selezione Dati**

   - **Filtro Clienti:** Deve consentire all'utente di filtrare i clienti dell'azienda per nome e codice identificativo.
   - **Filtro Prodotti:** Deve consentire all'utente di filtrare i prodotti in base alle diverse tipologie.

Requisiti Non Funzionali

Tra gli aspetti che riguardano i requisiti non funzionali emergono i seguenti:

1. **Performance e Scalabilità**

   - **Tempi di Risposta:** Il sistema deve garantire tempi di risposta rapidi per la generazione dei report e l'analisi dei dati, anche quando vengono utilizzati grandi volumi di dati.
   - **Scalabilità:** La piattaforma deve essere scalabile per gestire un incremento del volume di dati senza compromettere le prestazioni.

2. **Usabilità**

   - **Interfaccia Utente:** Il sistema deve fornire un'interfaccia intuitiva e di facile utilizzo, per facilitare l'interpretazione dei report e dei dati.
   - **Documentazione e Supporto:** Deve essere fornita documentazione adeguata e supporto per gli utenti, al fine di facilitare l'uso del sistema.

3. **Manutenibilità**

- **Aggiornamenti e Manutenzione:** Il sistema deve essere facilmente manutenibile e aggiornabile, per incorporare nuove funzionalità e correggere eventuali bug.

4. **Affidabilità**

   - **Disponibilità:** Deve garantire un'alta disponibilità e continuità del servizio, includendo meccanismi di backup e ripristino in caso di malfunzionamenti.

5. **Compatibilità**

   - **Compatibilità con i Browser e Dispositivi:** La piattaforma deve essere compatibile con diversi browser e dispositivi.

3.2.2 Requisiti Logiche di Supporto

Requisiti Funzionali

1. **Elaborazione Proposte di Vendita**

   - **Utilizzo degli Algoritmi:** Implementare tre algoritmi distinti per la creazione delle proposte di vendita, ciascuno con un peso specifico assegnato. Questi algoritmi devono essere capaci di elaborare e integrare i dati del cliente per produrre raccomandazioni personalizzate.

2. **Personalizzazione delle Proposte**

   - **Personalizzazione:** Il sistema deve generare proposte di vendita su misura per ogni cliente, basandosi sui dati raccolti e sui risultati degli algoritmi.

3. **Integrazione con il Marketing**

- **Strumento di Supporto Marketing:** Il sistema deve essere integrato con gli strumenti di marketing dell'azienda cliente per facilitare l'uso delle proposte di vendita personalizzate nella strategia di marketing.

Requisiti Non Funzionali

1. **Performance e Scalabilità**

   - **Efficienza degli Algoritmi:** Gli algoritmi devono operare in modo efficiente anche con grandi volumi di dati, garantendo tempi di risposta rapidi per la generazione delle proposte.
   - **Scalabilità del Sistema:** Il sistema deve essere scalabile per gestire un numero crescente di clienti e proposte di vendita senza compromettere le prestazioni.

2. **Usabilità**

   - **Formazione e Supporto:** Deve essere fornita documentazione e supporto adeguati per gli utenti finali, inclusi i team di marketing, al fine di garantire un utilizzo efficace dello strumento.

3. **Affidabilità**

   - **Disponibilità:** Deve essere garantita un'elevata disponibilità del sistema, con meccanismi di backup e ripristino per prevenire la perdita di dati e assicurare la continuità del servizio.
   - **Correttezza dei Risultati:** Deve essere assicurata l'affidabilità dei risultati proposti, che devono essere logicamente coerenti con le aspettative del marketing dell'azienda cliente.

4. **Compatibilità**

- **Integrazione con Altri Sistemi:** Il sistema deve essere compatibile con altri strumenti e sistemi in uso presso l'azienda.

3.2.3 Requisiti Predizione

Requisiti Funzionali

1. **Raccolta e Preprocessing dei Dati**

   - **Acquisizione dei Dati:** Il sistema deve essere in grado di acquisire e gestire i dati storici degli acquisti del cliente, relativi al periodo dal 2013 al 2024.
   - **Preprocessing dei Dati:** Devono essere implementate tecniche di preprocessing per pulire e preparare i dati per l'analisi.

2. **Applicazione di Tecniche di Machine Learning**

   - **Modelli Predittivi:** Sviluppare e addestrare modelli di Machine Learning per stimare i valori di acquisto futuri basati sui dati storici.
   - **Ottimizzazione dei Modelli:** Implementare tecniche di ottimizzazione e tuning dei modelli per migliorare l'accuratezza delle previsioni.

3. **Integrazione dei Risultati**

   - **Integrazione con le Proposte di Vendita:** Integrare le stime dei valori di acquisto futuri nei modelli di raccomandazione e nelle proposte di vendita precedentemente sviluppati.
   - **Aggiornamento delle Raccomandazioni:** Utilizzare le previsioni per supportare e aggiornare le raccomandazioni di vendita basate sui dati storici.

4. **Validazione e Verifica**

- **Valutazione delle Previsioni:** Implementare metodi per valutare la precisione e l'affidabilità delle previsioni generate dai modelli di Machine Learning.
- **Feedback e Miglioramenti:** Utilizzare il feedback ottenuto dalle previsioni per affinare ulteriormente i modelli e migliorare la loro performance.

Requisiti Non Funzionali

1. **Performance e Scalabilità**

   - **Tempo di Elaborazione:** I modelli di Machine Learning devono operare in tempi ragionevoli, anche con grandi volumi di dati storici.
   - **Scalabilità:** Il sistema deve essere scalabile per gestire un incremento nella quantità di dati e nella complessità dei modelli senza compromettere le prestazioni.

2. **Accuratezza e Precisione**

   - **Precisione dei Modelli:** I modelli devono garantire un elevato livello di precisione nelle previsioni per ottimizzare le raccomandazioni di vendita.
   - **Metriche di Performance:** Utilizzare metriche standard per misurare l'accuratezza delle previsioni e assicurarsi che soddisfino i requisiti di performance.

3. **Usabilità e Manutenibilità**

   - **Documentazione e Supporto:** Fornire documentazione adeguata e supporto per l'uso e la manutenzione dei modelli di Machine Learning e del sistema di previsione.

4. **Affidabilità**

- **Stabilità del Sistema:** Garantire che il sistema di previsione sia stabile e affidabile, con meccanismi per gestire e correggere eventuali malfunzionamenti.

3.3 Soluzioni Proposte

Nel contesto della presente tesi, è fondamentale esplorare e discutere le soluzioni adottate per il raggiungimento degli obiettivi delineati all'inizio del lavoro. In particolare, questo paragrafo si propone di illustrare dettagliatamente le strategie e le metodologie impiegate per affrontare e risolvere le problematiche specifiche identificate, nonché per ottimizzare i processi e i risultati previsti.

3.3.1 Soluzioni Reportistica

In relazione al primo obiettivo dell'elaborato, è stata adottata la seguente soluzione:

- **Importazione e Selezione delle Sorgenti Dati tramite PySpark e AWS:** È stata implementata una strategia di importazione e selezione delle sorgenti dati utilizzando PySpark in combinazione con AWS. Questo approccio consente una gestione efficiente dei dati provenienti da diverse origini e una loro elaborazione scalabile.

- **Utilizzo di Qlik per la Creazione di un Sistema ETL:** È stato scelto di utilizzare Qlik per sviluppare un sistema ETL (Extract, Transform, Load). All'interno dell'applicativo Qlik, sono state importate le sorgenti dati tramite query specifiche, facilitando la trasformazione e l'integrazione dei dati necessari per il successivo utilizzo.

- **Creazione di una Dashboard in Qlik:** È stata realizzata una dashboard interattiva in Qlik, progettata per fornire un report facilmente utilizzabile dall'utente finale. Questo strumento offre una visualizzazione chiara e accessibile dei dati, supportando decisioni informate e semplificando l'analisi delle informazioni.

Queste soluzioni sono state selezionate per garantire un'efficace gestione dei dati e una presentazione intuitiva dei risultati, rispondendo in modo adeguato alle esigenze specifiche del primo obiettivo dell'elaborato.

3.3.2 Soluzioni Logiche di Supporto

Nel contesto del secondo obiettivo del lavoro di tesi, relativo alla creazione di logiche di supporto per le proposte di vendita di prodotti e quantità, è stata adottata la seguente metodologia:

- **Analisi delle Basi Dati tramite DBeaver:** È stata effettuata un'analisi approfondita delle basi dati utilizzando il tool DBeaver. Questo strumento ha permesso di esaminare e comprendere la struttura e il contenuto delle fonti dati, facilitando le operazioni di verifica e preparazione necessarie per le fasi successive.

- **Importazione delle Sorgenti tramite PySpark e AWS:** È stato scelto di importare le sorgenti dati utilizzando PySpark in combinazione con AWS. Questa scelta ha consentito di gestire e processare i dati in modo scalabile e flessibile, assicurando l'integrazione delle diverse fonti.

- **Unione delle Sorgenti per le Diverse Tipologie di Logiche:** Le sorgenti dati sono state successivamente unite per implementare le diverse logiche di supporto alle vendite. Questo processo ha previsto l'integrazione dei dati provenienti da differenti fonti, al fine di sviluppare una base informativa coerente e completa per le analisi.

- **Creazione di Tabelle di Supporto per la Definizione delle Logiche:** Sono state create tabelle di supporto per facilitare la costruzione delle logiche di vendita. Queste tabelle hanno fornito una struttura organizzata per l'elaborazione delle informazioni e la definizione delle regole di supporto alle proposte di vendita.

- **Salvataggio dei Risultati in File Parquet e CSV:** I risultati ottenuti sono stati salvati in file nei formati Parquet e CSV, garantendo così la loro leggibilità e accessibilità da parte del cliente. Questo approccio ha permesso di presentare i risultati in un formato facilmente interpretabile per la valutazione e l'analisi finale.

Questa metodologia è stata scelta per garantire una gestione efficace e una presentazione chiara delle logiche di supporto alle proposte di vendita, rispondendo in modo adeguato alle esigenze del secondo obiettivo del lavoro di tesi.

3.3.3 Soluzioni Predizioni

Per l'ultimo obiettivo dell'elaborato, riguardante la previsione delle quantità di acquisto da parte di un determinato cliente, è stata adottata la seguente metodologia:

- **Esplorazione del Dataset tramite Visualizzazione Dati:** È stato avviato un processo di esplorazione del dataset utilizzando tecniche di visualizzazione dei dati. Questa fase ha permesso di analizzare la distribuzione e le caratteristiche dei dati disponibili.

- **Preprocessing del Dataset:** È stato effettuato un preprocessing del dataset per identificare e gestire eventuali valori nulli o duplicati valutando l'eventuale necessità di applicare altre tecniche per la gestione dei dati.

- **Applicazione del Modello di Machine Learning Prophet:** È stato applicato il modello di machine learning Prophet e ARIMA per effettuare previsioni delle

quantità di acquisto del cliente nei successivi 12 mesi a partire da giugno 2024. Questi modelli sono stati utilizzati per generare stime basate sui dati storici degli acquisti.

- **Rappresentazione Grafica dei Risultati:** I risultati delle previsioni sono stati rappresentati graficamente per visualizzare le stime delle quantità di acquisto nel tempo.

Queste attività hanno contribuito a raggiungere l'obiettivo di prevedere le quantità di acquisto del cliente.

Capitolo 4

Implementazione

All'interno del quarto capitolo dell'elaborato, si procederà con un'analisi tecnica dettagliata dello sviluppo delle soluzioni proposte nel capitolo precedente. Saranno presentati frammenti di codice e tabelle di supporto che sono stati creati per facilitare l'esecuzione dello studio.

4.1 Strumenti e Tecnologie

All'interno della seguente sezione verranno analizzati strumenti e tecnologie utilizzati ai fini della realizzazione dell'elaborato, descrivendo in maniera generale cosa siano e per quale mansione siano stati impiegati.

4.1.1 DBeaver

DBeaver [1] è un'applicazione di database management open source, progettata per la gestione e amministrazione di una vasta gamma di sistemi di database.

[1] DBeaver: `https://dbeaver.io/`

Figura 4.1: Logo Dbeaver

Cos'è DBeaver?

In base a quanto descritto all'interno di un articolo riportato nel sito "*Il Software*" [10], DBeaver è un client universale per database che offre un'interfaccia grafica unificata per la gestione di diversi tipi di database. È compatibile con una varietà di sistemi di database, tra cui SQL e NoSQL, e supporta numerosi database relazionali come MySQL, PostgreSQL, Oracle, SQL Server e SQLite, nonché database non relazionali come MongoDB e Cassandra.

Come funziona DBeaver?

DBeaver funziona come una piattaforma di gestione dei database che fornisce strumenti per:

1. **Connessione ai Database:** Permette di stabilire connessioni a diversi database utilizzando driver JDBC. Gli utenti possono configurare e gestire connessioni multiple attraverso un'interfaccia centralizzata.

2. **Navigazione e Visualizzazione:** Offre una vista strutturale dei database, permettendo agli utenti di esplorare e visualizzare tabelle, viste, indici e altre strutture di dati in modo intuitivo.

3. **Esecuzione di Query:** Include un editor SQL avanzato con supporto per l'autocompletamento del codice, evidenziazione della sintassi e esecuzione di query SQL.

Gli utenti possono eseguire query direttamente sui database e visualizzare i risultati in tempo reale.

4. **Gestione dei Dati:** Consente di inserire, aggiornare e cancellare dati direttamente dalle tabelle del database attraverso un'interfaccia grafica, facilitando operazioni di manutenzione e aggiornamento.

5. **Strumenti di Amministrazione:** Fornisce strumenti per la gestione e l'amministrazione del database, inclusi la creazione e la modifica di schemi, la gestione degli utenti e la configurazione delle impostazioni di sicurezza.

6. **Supporto per Estensioni e Plugin:** Permette l'aggiunta di estensioni e plugin per estendere le funzionalità, come il supporto per nuovi tipi di database o strumenti di visualizzazione avanzati.

Per cosa è stato utilizzato DBeaver?

Nel contesto del progetto di tesi, DBeaver è stato utilizzato per leggere e gestire diverse basi di dati provenienti da fonti differenti. L'impiego di questo strumento si è rivelato essenziale per ottenere una visione d'insieme dei dati contenuti nelle varie fonti. Grazie a DBeaver, è stato possibile selezionare solo le informazioni rilevanti e necessarie per lo sviluppo dell'elaborato, facilitando così la gestione e l'analisi dei dati pertinenti al progetto.

4.1.2 ETL

Le ETL, acronimo di Extract, Transform, Load, rappresentano un processo fondamentale nel campo della gestione dei dati e dell'integrazione dei sistemi informativi. Esse

costituiscono un insieme di tecniche e strumenti utilizzati per il trasferimento, la trasformazione e il caricamento dei dati da diverse fonti verso un sistema di destinazione, tipicamente un data warehouse o un altro tipo di deposito dati centralizzato.

Cos'è ETL?

Il processo ETL si articola in tre fasi principali:

- **Extract** (Estrazione): Questa fase comporta la raccolta dei dati da diverse fonti, che possono includere database relazionali, file flat, sistemi ERP, applicazioni cloud e altre origini di dati. L'obiettivo è acquisire i dati necessari in modo completo e accurato.

- **Transform** (Trasformazione): Una volta estratti, i dati vengono trasformati per soddisfare i requisiti del sistema di destinazione. Questa fase può includere operazioni quali la pulizia dei dati (rimozione di duplicati, correzione di errori), la normalizzazione (standardizzazione dei formati dei dati), l'aggregazione (sommare o combinare dati), e la conversione (modifica dei formati o delle strutture dei dati). La trasformazione garantisce che i dati siano coerenti, integri e utili per l'analisi e la reportistica.

- **Load** (Caricamento): Nella fase di caricamento, i dati trasformati vengono inseriti nel sistema di destinazione, come un data warehouse, un data mart o un altro repository di dati. Questo processo assicura che i dati siano disponibili per l'analisi, la generazione di report e altre attività decisionali.

Per cosa è stato utilizzato ETL?

Il sistema ETL si è rivelato essenziale per integrare e consolidare le diverse fonti di dati precedentemente analizzate e selezionate. Questo processo ha consentito la combinazione

delle informazioni necessarie alla creazione di un report destinato all'utente finale. Attraverso l'utilizzo di query, le informazioni elaborate dal sistema ETL sono state estratte e presentate nel report, permettendo così la loro visualizzazione e utilizzo per le successive analisi.

4.1.3 Qlik

Qlik [2] è una piattaforma di business intelligence e analisi dei dati che fornisce strumenti per la visualizzazione, l'analisi e la gestione delle informazioni aziendali. Fondata nel 1993, Qlik offre soluzioni per trasformare i dati grezzi in informazioni significative e fruibili, supportando la presa di decisioni basata sui dati.

Figura 4.2: Logo Qlik

Cos'è Qlik?

Per un'analisi approfondita e dettagliata su Qlik e il suo funzionamento, è stata utilizzata come riferimento la sezione di documentazione del sito ufficiale. Questa risorsa discute in modo specifico il funzionamento e le caratteristiche di Qlik [15]. Qlik è principalmente nota per due prodotti principali:

[2]Qlik: `https://www.qlik.com/`

1. **Qlik Sense:** Una piattaforma di analisi dei dati e visualizzazione che consente agli utenti di creare dashboard interattivi e report. Qlik Sense offre un'interfaccia intuitiva e una vasta gamma di opzioni di visualizzazione per esplorare i dati, identificare tendenze e ottenere approfondimenti. Supporta l'analisi self-service, permettendo agli utenti di interagire con i dati senza necessità di competenze tecniche avanzate.

2. **Qlik View:** Un'applicazione di business intelligence progettata per fornire analisi ad alta velocità e capacità di reporting. QlikView permette agli utenti di costruire report complessi e dashboard interattivi, utilizzando un approccio basato su modelli di dati associativi che facilita la scoperta di relazioni tra i dati.

A cosa serve Qlik?

Qlik serve a numerosi scopi nell'ambito della gestione e analisi dei dati:

1. **Visualizzazione dei Dati:** Permette la creazione di visualizzazioni interattive e dashboard, facilitando la comprensione dei dati attraverso rappresentazioni grafiche come grafici, mappe e tabelle.

2. **Analisi Interattiva:** Consente agli utenti di esplorare i dati in modo dinamico e interattivo, fornendo strumenti per filtrare, segmentare e approfondire le informazioni. Questo approccio supporta l'analisi ad hoc e la scoperta di approfondimenti.

3. **Integrazione dei Dati:**Facilita l'integrazione di dati provenienti da fonti diverse, combinando informazioni in un'unica vista coerente. Questo è particolarmente utile per ottenere una visione complessiva delle performance aziendali e per effettuare analisi cross-funzionali.

4. **Reporting e Monitoraggio:** Supporta la creazione di report dettagliati e dashboard per il monitoraggio delle metriche chiave. Gli utenti possono personalizzare

i report per soddisfare le esigenze specifiche di business e ottenere aggiornamenti in tempo reale.

5. **Supporto alla Decisione:** Fornisce strumenti per analisi avanzate e previsioni, aiutando le organizzazioni a prendere decisioni informate basate su dati concreti e approfondimenti approfonditi.

6. **Accesso e Condivisione dei Dati:** Offre funzionalità per la condivisione e la collaborazione sui dati, consentendo agli utenti di distribuire report e dashboard all'interno dell'organizzazione e collaborare sui risultati.

Per cosa è stato Utilizato Qlik?

Per completare il primo obiettivo dell'elaborato, è stato impiegato Qlik principalmente per la costruzione del processo ETL, che ha integrato le sorgenti di dati sia per l'analisi del parco circolante, sia per le analisi di marketing relative ai clienti dell'azienda manifatturiera. In seconda istanza, l'utilizzo di Qlik è stato fondamentale per la realizzazione della dashboard. Grazie a Qlik, è stato possibile sviluppare un sistema semplice e concreto per l'azienda, capace di leggere i file presenti nelle sorgenti e valorizzare i dati nei database. Questo ha permesso di utilizzare le informazioni sia per prevedere il potenziale di vendita dei pneumatici, sia per analizzare le statistiche di vendita temporali.

4.1.4 Open Street Map

OpenStreetMap (OSM)[3] è un progetto collaborativo volto a creare mappe digitali libere, aggiornate e modificabili da chiunque. La piattaforma si basa sul contributo di utenti volontari che raccolgono dati geografici da diverse fonti, inclusi GPS, immagini satellitari,

[3]OpenStreetMap: `https://www.openstreetmap.org/`

cartografie cartacee e osservazioni sul campo. Questi dati vengono poi combinati per formare una mappa globale, accessibile a tutti e utilizzabile per molteplici scopi.

A cosa serve Open Street Map?

OpenStreetMap serve principalmente a fornire una rappresentazione dettagliata e aperta del territorio, senza le restrizioni di licenza tipiche delle mappe commerciali. Viene utilizzato in vari ambiti, tra cui pianificazione urbanistica, gestione di emergenze, attività outdoor, ricerca accademica e sviluppo di applicazioni e servizi che necessitano di dati geografici. Essendo un progetto aperto, permette a sviluppatori e organizzazioni di integrare mappe nei loro progetti senza costi aggiuntivi, favorendo l'innovazione e la diffusione di informazioni geografiche accurate e aggiornate.

Per cosa è stato Utilizzato Open Street Map?

L'utilizzo di OpenStreetMap è stato fondamentale durante la creazione della dashboard di reportistica per l'analisi del potenziale di vendita in relazione al parco auto circolante in un determinato territorio. Grazie a questa libreria, è stato possibile rappresentare graficamente la mappa, evidenziando i diversi territori mediante l'uso di differenti tonalità di colore. Queste tonalità rappresentavano il potenziale vendibile all'interno di ciascun'area, permettendo così una visualizzazione chiara e intuitiva dei dati di vendita potenziale correlati alla distribuzione dei veicoli.

4.1.5 AWS (Amazon Web Service)

Amazon Web Services (AWS)[4] è una piattaforma di servizi cloud offerta da Amazon, progettata per fornire una vasta gamma di soluzioni informatiche scalabili e flessibili

[4]AWS: `https://aws.amazon.com/`

tramite internet. AWS è uno dei principali fornitori di servizi cloud a livello globale e offre una suite completa di servizi per l'infrastruttura IT, l'archiviazione dei dati, l'analisi dei dati, la gestione delle applicazioni e molto altro.

Figura 4.3: Logo AWS

Cosa è AWS?

Per poter descrivere nel completo tutte le funzionalità che offre la piattaforma è possibile consultare il sito ufficiale di Amazon Web Services [1]. In generle AWS è una piattaforma cloud che include un'ampia gamma di servizi e risorse distribuiti attraverso un'infrastruttura globale di data center. I servizi di AWS sono progettati per essere altamente scalabili, sicuri e accessibili, e possono essere utilizzati per gestire applicazioni e carichi di lavoro di qualsiasi dimensione. AWS opera su un modello di pagamento basato sull'uso, consentendo alle organizzazioni di pagare solo per le risorse effettivamente consumate.

A cosa serve AWS?

AWS serve a molteplici scopi nel contesto della gestione e dell'elaborazione delle risorse informatiche, inclusi:

1. **Infrastruttura IT e Virtualizzazione:** Offre servizi di computing come Amazon EC2 (Elastic Compute Cloud), che consente di eseguire server virtuali scalabili, e Amazon RDS (Relational Database Service) per la gestione di database relazionali, permettendo alle aziende di ridurre i costi e migliorare l'efficienza operativa.

2. **Archiviazione dei Dati:** Fornisce soluzioni di archiviazione scalabili e sicure come Amazon S3 (Simple Storage Service) e Amazon EBS (Elastic Block Store), che permettono di archiviare e gestire grandi volumi di dati in modo affidabile.

3. **Analisi e Big Data:** Include servizi avanzati per l'analisi dei dati e l'elaborazione di big data, come Amazon Redshift per l'analisi dei dati su larga scala e Amazon EMR (Elastic MapReduce) per l'elaborazione di dati distribuiti, supportando le attività di data mining e reportistica.

4. **Networking e Connettività:** Offre soluzioni per la gestione delle reti e della connettività, come Amazon VPC (Virtual Private Cloud) per creare reti virtuali isolate e Amazon Route 53 per il servizio di DNS (Domain Name System).

5. **Sicurezza e Conformità:** Fornisce strumenti e servizi per garantire la sicurezza e la conformità dei dati, inclusi Amazon IAM (Identity and Access Management) per la gestione delle identità e dei permessi, e Amazon KMS (Key Management Service) per la crittografia dei dati.

6. **Sviluppo e Gestione delle Applicazioni:** Supporta lo sviluppo e la gestione delle applicazioni attraverso servizi come AWS Lambda per l'esecuzione di codice in risposta a eventi e AWS Elastic Beanstalk per il deployment e la gestione delle applicazioni web.

7. **Intelligenza Artificiale e Machine Learning:** Include servizi di intelligenza artificiale e machine learning come Amazon SageMaker per la creazione, l'addestramento e il deployment di modelli di machine learning.

Per cosa è stato Utilizzato AWS?

L'utilizzo della piattaforma cloud AWS è stato impiegato principalmente per lo sviluppo del codice, attraverso l'uso di diversi framework disponibili all'interno della piattaforma.

Utilizzando Amazon EMR (Elastic MapReduce), è stato avviato un cluster dedicato al progetto specifico, configurando i parametri necessari per eseguire il lavoro in modo efficiente. Attraverso la sezione Workspace di AWS, è stato creato un ambiente dedicato alla scrittura del codice, collegato al cluster preconfigurato per sfruttare le risorse computazionali in cloud. L'ambiente di sviluppo scelto è stato un notebook Jupyter, utilizzato per gestire i vari file di codice. I risultati ottenuti durante lo sviluppo del progetto sono stati salvati in due formati distinti (Parquet e CSV), a seconda delle esigenze, all'interno di Amazon S3. Questo ha permesso una successiva fase di validazione e la presentazione dei risultati all'azienda manifatturiera.

4.1.6 Jupyter

Jupyter è un'applicazione open source progettata per la creazione e la condivisione di documenti che contengono codice eseguibile, visualizzazioni e testi descrittivi. È ampiamente utilizzato nel campo della scienza dei dati, dell'analisi statistica e della ricerca accademica per facilitare la documentazione e la presentazione dei risultati.

Figura 4.4: Logo Jupyter

Cosa è Jupyter?

Jupyter è un progetto che ha originato dal progetto IPython, e il suo nome è un acronimo che rappresenta i linguaggi di programmazione Julia, Python e R, sebbene ora supporti

molti altri linguaggi. Il componente principale di Jupyter è il Jupyter Notebook, che è un ambiente interattivo per l'esecuzione di codice e la creazione di documenti interattivi.

A cosa serve Jupyter?

Jupyter funziona attraverso i seguenti componenti principali:

1. **Jupyter Notebook:** Un'applicazione web che consente agli utenti di creare e interagire con documenti interattivi chiamati notebook. Questi notebook possono contenere celle di codice eseguibili, celle di testo in formato Markdown, e visualizzazioni grafiche. Le celle di codice possono essere eseguite direttamente all'interno del notebook, mostrando i risultati immediatamente sotto il codice.

2. **Kernel:** Il kernel è il motore che esegue il codice contenuto nelle celle del notebook. Jupyter supporta diversi kernel per diversi linguaggi di programmazione. Quando un notebook viene eseguito, il codice viene inviato al kernel appropriato, che lo esegue e restituisce l'output al notebook.

3. **Interfaccia Utente:** L'interfaccia utente di Jupyter Notebook è accessibile tramite un browser web e fornisce strumenti per l'interazione con i notebook. Gli utenti possono eseguire codice, visualizzare output, aggiungere e modificare celle di testo e codice, e salvare i loro lavori.

4. **File Format:** I notebook Jupyter sono salvati in un formato di file JSON con estensione .ipynb. Questo formato supporta la memorizzazione di codice, output, e testo descrittivo, rendendo i notebook facilmente condivisibili e riproducibili.

5. **Estensioni e Plugin:** Jupyter supporta un'ampia gamma di estensioni e plugin che possono ampliare le sue funzionalità. Queste estensioni possono aggiungere nuove opzioni di visualizzazione, strumenti di analisi e integrazioni con altre applicazioni.

Per cosa è stato Utilizzato Jupyter?

L'utilizzo di Jupyter è stato fondamentale per il completamento dell'intero progetto di tesi, poiché tutto il lavoro è stato sviluppato attraverso questo applicativo. Jupyter ha permesso la costruzione e la gestione del codice alla base di ciascun obiettivo dell'elaborato, facilitando l'implementazione e la documentazione delle soluzioni proposte.

4.1.7 PySpark

PySpark [5] è una libreria open source per l'elaborazione dei dati su larga scala, sviluppata da Apache Spark e progettata per essere utilizzata con il linguaggio di programmazione Python. PySpark consente di lavorare con grandi volumi di dati in modo distribuito e parallelo, sfruttando l'architettura di calcolo in memoria di Apache Spark.

Figura 4.5: Logo PySpark

Cosa è Pyspark?

PySpark è l'interfaccia Python di Apache Spark, una piattaforma di elaborazione dei dati che gestisce operazioni complesse come il calcolo distribuito, l'analisi dei big data e la machine learning. Spark è noto per la sua capacità di elaborare grandi quantità di dati in modo veloce e scalabile grazie alla sua architettura in memoria e alla distribuzione dei dati tra più nodi di un cluster.

[5]PySpark: `https://spark.apache.org/docs/latest/api/python/`

A cosa serve Pyspark?

PySpark opera attraverso i seguenti componenti e processi principali:

1. **Spark Context:** È il punto di ingresso principale per tutte le funzionalità di Spark. PySpark richiede la creazione di un'istanza di SparkContext, che gestisce la connessione al cluster di Spark e coordina le operazioni di calcolo distribuito.

2. **Resilient Distributed Datasets (RDDs):** Gli RDDs sono la struttura di dati fondamentale di Spark. Rappresentano una collezione distribuita di oggetti immutabili che possono essere elaborati in parallelo. Gli RDDs supportano operazioni come map, filter e reduce, e offrono tolleranza ai guasti attraverso la loro natura resiliente.

3. **DataFrames:** I DataFrames sono una struttura di dati più avanzata rispetto agli RDDs e sono ispirati ai DataFrames di R e Pandas. Forniscono un'astrazione di livello più alto per l'elaborazione dei dati e sono ottimizzati per le operazioni di query e trasformazione. I DataFrames possono essere utilizzati per eseguire operazioni SQL-like e sono compatibili con una vasta gamma di formati di dati.

4. **Spark SQL:** Questa componente consente di eseguire query SQL su DataFrames e RDDs. Spark SQL offre un'interfaccia SQL e un motore di esecuzione ottimizzato per interrogare i dati distribuiti.

5. **Spark Streaming:** Consente l'elaborazione di flussi di dati in tempo reale. Spark Streaming può gestire e processare dati in streaming, consentendo analisi e risposte immediate a flussi di dati continui.

```
from pyspark import SparkContext
sc = SparkContext(appName="MyApp")
```

4.1 Esempio di codice

Per cosa è stato Utilizzato Pyspark?

PySpark è stato impiegato come principale libreria di programmazione per lo sviluppo del codice dell'elaborato. Grazie all'uso di PySpark, è stata stabilita una connessione con i server di AWS, consentendo l'interazione con Amazon S3 per la lettura e la scrittura di file. L'intero processo di esplorazione dei dati e di sviluppo degli algoritmi è stato realizzato mediante l'utilizzo di PySpark.

4.1.8 Dataset

Cosa è un Dataset?

Un dataset è una raccolta organizzata di dati strutturati, spesso raccolti e conservati in formato tabellare. Ogni dataset è composto da un insieme di record (o righe), ciascuno dei quali rappresenta un'istanza o un'osservazione, e da un insieme di attributi (o colonne), che definiscono le caratteristiche o le variabili misurate per ciascun record.

Quali sono le Caratteristiche di un Dataset?

1. **Struttura Tabellare:** I dati sono generalmente organizzati in righe e colonne. Ogni riga rappresenta un'unità di analisi (ad esempio, un individuo, un oggetto o un evento), mentre ogni colonna rappresenta una variabile o un attributo misurato per quell'unità.

2. **Campi e Record:** I campi (o colonne) definiscono le categorie di dati raccolti, come nomi, date, valori numerici, ecc. I record (o righe) contengono i dati specifici per ogni entità o osservazione.

3. **Formato:** I dataset possono essere memorizzati in vari formati, inclusi file CSV, Excel, database relazionali, e formati di dati più complessi come JSON o XML.

4. **Metadati:** Spesso, un dataset include metadati che forniscono informazioni aggiuntive sulla struttura, il contenuto e la qualità dei dati. I metadati possono includere descrizioni dei campi, unità di misura, e indicazioni su eventuali trasformazioni dei dati.

Per cosa sono stati Utilizzati i Dataset?

I dataset utilizzati per lo sviluppo dell'elaborato sono stati selezionati mediante l'applicazione DBeaver. Questi dataset rappresentano il cuore pulsante dell'intero progetto, in quanto contengono tutte le informazioni di base essenziali per l'esecuzione del lavoro. Durante la fase iniziale del progetto, i dataset sono stati estratti tramite un notebook dedicato. Le informazioni raccolte includono dettagli sui clienti dell'azienda manifatturiera, specifiche dei pneumatici, dati sulle vendite e informazioni riguardanti il parco auto circolante. Tali dati hanno costituito la base su cui sono state sviluppate le logiche e gli strumenti descritti nell'elaborato. Attraverso un'analisi approfondita di questi dataset, sono state implementate le metodologie necessarie per il conseguimento degli obiettivi del progetto, assicurando così che le soluzioni proposte fossero basate su informazioni accurate e pertinenti.

4.1.9 Prophet

Prophet è una libreria open source sviluppata da Facebook per la modellizzazione e previsione delle serie temporali, utilizzabile facilmente tramite la guida descritta al interno del sito *"Prophet"* ufficile [6] che illustra tutti gli step da seguire ai fini di utilizzare questa librerira. È progettata per gestire dati con forti componenti stagionali e trend, rendendola particolarmente adatta per previsioni di serie temporali in scenari reali.

Cosa è Prophet?

Prophet è un modello di previsione basato su un approccio additivo, che combina componenti di trend, stagionalità e festività per produrre previsioni. È progettato per essere robusto e facilmente interpretabile, e può gestire dati con frequenze diverse e mancanze di dati, rendendolo adatto a un'ampia gamma di applicazioni.

Componenti Principali

1. **Trend:** Prophet utilizza un modello di trend per catturare le tendenze a lungo termine nei dati. Questo componente può essere rappresentato da una crescita lineare o logistica, a seconda delle caratteristiche del dataset.

2. **Stagionalità:** Prophet include componenti stagionali per rappresentare variazioni periodiche nei dati, come quelle giornaliere, settimanali o annuali. Le componenti stagionali possono essere modellate attraverso funzioni sinusoidali.

3. **Festività:** Prophet permette l'inclusione di effetti dovuti a festività o eventi speciali che possono influenzare i dati. Gli utenti possono specificare date e impatti delle festività per migliorare la precisione delle previsioni.

4. **Componenti di Errore:** Prophet assume che i dati contengano una componente di errore, modellata come un rumore gaussiano, che viene gestita nel processo di previsione.

Come funziona Prophet?

Il funzionamento di Prophet può essere suddiviso nei seguenti passaggi principali:

1. **Preprocessing dei Dati:** I dati delle serie temporali vengono preparati e strutturati in due colonne principali: una per la data e una per il valore osservato. Prophet richiede che i dati siano in formato pandas DataFrame.

2. **Definizione del Modello:** Gli utenti configurano il modello specificando i parametri di trend, stagionalità e festività. Prophet fornisce anche parametri per gestire la robustezza del modello e la gestione degli outlier.

3. **Addestramento del Modello:** Prophet stima i parametri del modello basandosi sui dati storici. Utilizza tecniche di ottimizzazione per adattare i componenti di trend e stagionalità ai dati osservati.

4. **Generazione delle Previsioni:** Dopo l'addestramento, il modello può essere utilizzato per fare previsioni future. Prophet fornisce intervalli di confidenza per le previsioni, offrendo stime accompagnate da un margine di incertezza.

5. **Visualizzazione dei Risultati:** Prophet include strumenti per visualizzare i risultati delle previsioni, consentendo di esaminare i trend, le stagionalità e gli effetti delle festività nel contesto delle previsioni.

Per cosa è stato Utilizzato Prophet?

Per la realizzazione dell'ultimo obiettivo dell'elaborato, è stata impiegata la libreria Prophet. Utilizzando questo strumento, è stato possibile, a partire da un dataset contenente le vendite effettuate verso un determinato cliente dal 2013 al 2024, suddiviso in intervalli mensili, stimare le vendite future per quel cliente nei successivi 12 mesi. Questo approccio ha permesso di analizzare e comprendere come il trend di vendita potrebbe evolversi nel futuro in base ai dati storici delle vendite.

4.1.10 ARIMA

ARIMA, acronimo di AutoRegressive Integrated Moving Average, è un modello statistico ampiamente utilizzato per la previsione delle serie temporali. Questo approccio combina elementi di regressione autoregressiva, differenziazione e media mobile, rendendolo

efficace per la modellazione di dati con caratteristiche complesse e non stazionarie.

Cosa è ARIMA?

ARIMA è un modello di previsione che si basa sull'analisi delle serie temporali storiche per fare previsioni sui dati futuri. Esso si compone di **tre parametri** fondamentali: p (autoregressione), d (differenziazione) e q (media mobile), che insieme definiscono il comportamento del modello. La flessibilità di ARIMA permette di adattarlo a diverse tipologie di dati, sia stazionari che non stazionari.

Componenti Principali:

1. **Autoregressione (AR):** Questa componente utilizza i valori passati della serie temporale per prevedere i valori futuri. L'idea principale è che i dati storici possano influenzare i dati futuri.

2. **Differenziazione (I):** La differenziazione è utilizzata per rendere la serie temporale stazionaria. Questo passaggio è fondamentale, poiché molte tecniche di previsione richiedono che i dati non presentino trend o stagionalità.

3. **Media Mobile (MA):** Questa componente considera la media mobile dei residui passati, consentendo al modello di correggere le sue previsioni in base all'errore passato.

Come funziona ARIMA?

Il funzionamento di ARIMA può essere descritto nei seguenti passaggi:

1. **Preprocessing dei Dati:** I dati delle serie temporali devono essere preparati, assicurandosi che siano in formato adeguato per l'analisi, e che soddisfino i requisiti di stazionarietà.

2. **Identificazione del Modello:** La scelta dei parametri p, d e q viene effettuata utilizzando funzioni di autocorrelazione (ACF) e autocorrelazione parziale (PACF), strumenti utili per determinare il lag ottimale.

3. **Stima dei Parametri:** Attraverso tecniche statistiche, i parametri del modello vengono stimati basandosi sui dati storici.

4. **Generazione delle Previsioni:** Una volta che il modello è stato addestrato, può essere utilizzato per generare previsioni future, complete di intervalli di confidenza per rappresentare l'incertezza delle previsioni.

5. **Validazione del Modello:** È essenziale valutare le prestazioni del modello confrontando le previsioni con i dati reali, al fine di determinare la sua accuratezza e affidabilità.

Per cosa è stato Utilizzato ARIMA?

Nell'ambito di questo studio, il modello ARIMA è stato impiegato per analizzare un dataset contenente informazioni sulle vendite effettuate da un cliente specifico dal 2013 al 2024, suddiviso in intervalli mensili. Attraverso l'applicazione di ARIMA, è stato possibile stimare le vendite future per i successivi 12 mesi, fornendo così un'analisi approfondita dell'evoluzione del trend di vendita, basata su dati storici. Questo approccio ha rivelato utilità pratica in contesti aziendali, consentendo decisioni strategiche informate.

4.2 Sviluppo ETL e Report

La sezione seguente illustrerà le fasi di sviluppo relative al primo obiettivo dell'elaborato, ovvero la creazione di un processo ETL per la costruzione di una dashboard di reportistica

di supporto. Saranno discusse, a livello implementativo, le operazioni effettuate, a partire dall'importazione delle sorgenti dei dati fino al raggiungimento della dashboard finale.

4.2.1 Sviluppo importazione Sorgenti

La selezione delle sorgenti è stata effettuata utilizzando il software DBeaver. Considerando che le fonti erano distribuite in diversi ambienti, è stata necessaria una preliminare individuazione delle stesse tramite l'utilizzo del suddetto software. Successivamente, sono stati sviluppati diversi notebook preliminari per il trattamento dei dati.

I notebook sono stati implementati in PySpark, con l'obiettivo di stabilire una connessione al server di riferimento contenente le sorgenti. Mediante l'esecuzione di query SQL, passate in input ai notebook, è stato possibile estrarre i dati richiesti dalle sorgenti identificate.

Di seguito, viene riportato un esempio di applicazione in un caso d'uso specifico.

```
from pyspark.context import SparkContext, SparkConf
from pyspark.sql import HiveContext, SQLContext, SparkSession
from pyspark.sql import functions as F
from pyspark.sql.window import Window
from pyspark.sql.types import IntegerType
from pyspark.sql.functions import concat
from pyspark.sql.types import *
import datetime
import os
from io import StringIO
import sys
```

4.2 Esempio Importazione Librerire

Utilizzando le librerie sopra indicate, è possibile elaborare dati, inclusa l'instaurazione di un collegamento con l'API di Spark e la gestione della connessione ai cluster.

4.2.2 Sviluppo importazione Tabelle in Qlik

Per la realizzazione dell'applicativo di reportistica è stato impiegato Qlik, come precedentemente descritto. La fase di creazione dell'ETL ha richiesto lo sviluppo di script per la generazione delle tabelle delle dimensioni e dei fatti. Gli script in Qlik sono stati sviluppati utilizzando una notazione denominata Backus-Naur Form (BNF).

Per chiarire il processo di creazione delle tabelle delle dimensioni e dei fatti, si riportano di seguito due esempi di script in BNF, uno relativo alla creazione di una tabella delle dimensioni e l'altro alla creazione di una tabella dei fatti.

```
// Caricamento informazioni dalla tabella client
LOAD
client_code,
client_name,
client_code&' - '&client_name AS client_code_name,
client_level,
AutoNumber(client_code) AS client_key;

LOAD point_code, point_name, latitude, longitude;
...
// Cariamento informazioni area client
[CLIENT_AREA]:
SELECT
client_code,
id_country,
area_code
FROM $(GEO_DB).lookup_client_area;
```

4.3 Creazione tabella dimensioni Qlik

```
// Caricamento informazioni parco auto circolanti
LOAD
auto_parc,
fullyear,
yearToDay,
AutoNumber(id_country&area_code&brand&.(attributi)..&stagione) AS
    ↪ key_area;

// Caricamento informazioni potenziale area
[AREA_POTENTIAL]:
...
(Query creazione area potezionale vendite)

// Caricamento informazioni potenziale area pneumatici premium
[AREA_PREMIUM_POTENTIAL]:
...
(Query creazione area potezionale vendite pneumatici premium)
```

4.4 Creazione tabella fatti Qlik

I codici riportati sopra offrono una panoramica riassuntiva del processo seguito per la creazione delle tabelle dei fatti e delle dimensioni che compongono l'ETL.

4.2.3 Sviluppo Dashboard

La creazione della Dashboard finale sia per il monitoraggio del potenziale calcolato in base al parco auto circolante, sia per la gestione delle statistiche di vendita ha costituito un processo di analisi grafica con l'applicazione di concetti derivanti dall UX Designer

(User Experience Designer) per la costruzione di un report chiaro e intuitivo da far utilizzare all'utente senza problemi.

La costruzione degli elenti grafici all'interno dell'applicativo è stata fatta tramite l'operazioen di "drag and drop" degli elementi selezionati che costituiscono la dashboard finale.

Per ogni elemento utilizzato sono state utilizzate delle query associate ad ognuno di essi che hanno permesso di estrarre i valori di interesse all'interno delle tabelle precedemente utilizzate per la costruzione dell'ETL. Ottenendo così l'ìnterfaccia finale.

Si riporta di seguito un'esempio di query eseguita per l'estrazione di informazioni utili dall'ETL.

```
// Caricamento dei dati dalla tabella 'Sales'
Sales:
LOAD
    OrderID,
    CustomerID,
    OrderDate,
    SalesAmount
SQL SELECT
    OrderID,
    CustomerID,
    OrderDate,
    SalesAmount
FROM SalesTable;

// Caricamento dei dati dalla tabella 'Client'
Client:
LOAD
    Client_code,
    Client_name,
    Client_level,
```

```
SQL  SELECT
     Client_code,
     Client_name,
     Client_level
FROM Client_table;
```

4.5 Query dashboard

Per fornire una rappresentazione più accurata del risultato ottenuto attraverso il processo di costruzione grafica della dashboard, viene riportata un'anteprima di una pagina della dashboard finale. Di seguito è illustrato un esempio di caso d'uso relativo all'applicazione per il monitoraggio del parco auto circolante.

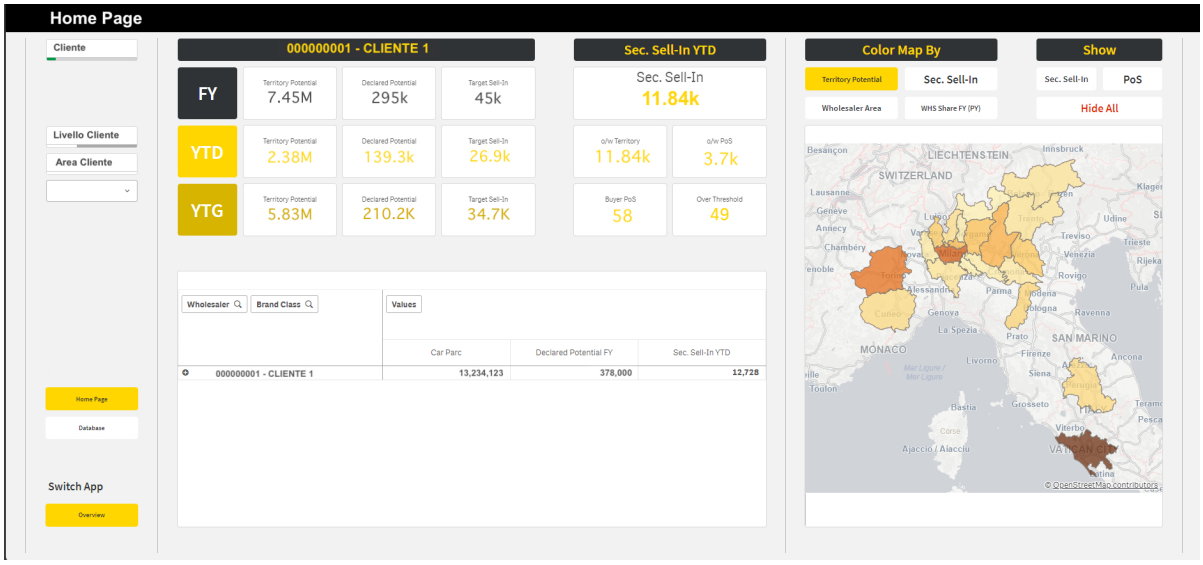

Figura 4.6: Dashboard Parco Auto Circolante

4.2.4 Conclusione

Il processo descritto, relativo alla creazione di una dashboard di reportistica, è stato realizzato con due obiettivi principali: il calcolo del potenziale vendibile di pneumatici basato sul parco auto circolante all'interno di un territorio specifico per un cliente e il monitoraggio delle statistiche di vendita dal punto di vista del marketing per i clienti di un'azienda manifatturiera. Nel capitolo successivo, verranno presentati i risultati ottenuti e le operazioni eseguibili per raggiungere entrambi gli scopi.

4.3 Sviluppo Logiche di Supporto

Nel paragrafo seguente saranno analizzate le logiche adottate per il supporto alle vendite di pneumatici. Verranno esaminati vari aspetti, a partire dalle sorgenti utilizzate fino alla creazione delle tabelle di supporto necessarie per l'applicazione delle logiche.

4.3.1 Sviluppo importazione Sorgenti

I dati che sono stati raccolti per la creazione delle logiche sono stati estrapolati da diverse fonti. Le fonti utilizzate sono state estratte tramite dei notebook creati con Jupyter in Pyspark.

In seguito si riporta un esempio di connesione a server per l'importazione di un dataset tramite query SQL su S3, salvandolo in parquet.

```
// Connessione a DB
url = "jdbc:oracle:thin:@sunset.it.x.com:1599:STING"
user = "DB"
password = os.environ['STING_DB_PASSWORD']

//Query
query = f"""
```

```
(SELECT
PROD_ID, MERCATO, CLIENTE,
VAREA_ORIG, VINCOLO, DT_VALIDITA,
PROD_ID_SOST, YAGG_DATE, YAGG_USER, "TIMESTAMP"
FROM netgamma.dataset
WHERE MERCATO in ('ITA', 'US')
) t
"""
// Assegnamento dataset importato
db = hc.read.format('jdbc')\
    .option("url", url)\
    .option("dbtable", query)\
    .option("user", user)\
    .option("password", password)\
    .option("queryTimeout", 0)\
    .option("driver", "oracle.jdbc.driver.OracleDriver").load()

// Salvataggio su S3
db = db.withColumn("last_update", F.current_timestamp())
db.write.mode("overwrite").parquet(f"s3://path.db")
```

4.6 Importazione dataset in S3

Il codice riportato consente di stabilire una connessione al SOURCE DB, la fonte contenente il database, utilizzando le credenziali di accesso. Successivamente, il database importato viene memorizzato nella variabile ‘db‘ e successivamente trasferito nello storage S3 di AWS per essere utilizzato nelle logiche applicative.

Per quanto riguarda la lettura di un dataset da S3, il processo risulta essere più semplice e veloce.

```
// Lettura dataset da S3
```

```
db = hc.read.parquet(f"s3://path.db)
```

4.7 Lettura dataset da S3

4.3.2 Sviluppo Logiche

Dopo l'importazione di tutte le sorgenti necessarie per la creazione delle logiche, è stato creato un nuovo notebook denominato *mix.ipynb*, contenente tutti gli sviluppi relativi alle logiche dei clienti e del mercato.

Per fornire una visione complessiva dello studio condotto per la realizzazione delle logiche, nella figura seguente è riportato uno schema che illustra la costruzione dei due mix.

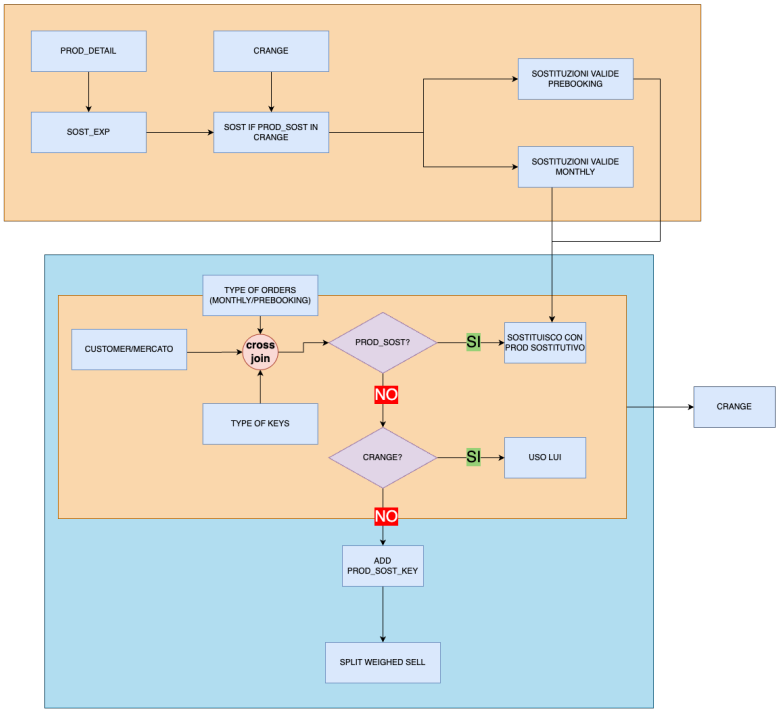

Figura 4.7: Schema Logiche di Supporto

Analizzando in dettaglio lo schema riportato, si prende come punto di partenza il rettangolo rosa, in particolare il rettangolo etichettato **'CLIENTE/MERCATO'**. Il processo di entrambe le logiche inizia dalle fonti iniziali, alle quali vengono aggiunte, tramite un'operazione di *join*, ulteriori informazioni sui pneumatici relative rispettivamente ai clienti e al mercato.

A partire dalle due fonti iniziali, le logiche vengono sviluppate tenendo conto di diversi criteri specifici per i prodotti. Un aspetto fondamentale delle analisi condotte per questo scopo è la ricerca dei prodotti sostitutivi, che devono essere suggeriti quando un cliente effettua un ordine.

Gli ordini sono suddivisi secondo due metriche temporali: **prebooking** (pre-ordine in previsione dell'arrivo di una nuova stagione) e **monthly** (ordini generici effettuati entro la fine del mese successivo a quello corrente).

Utilizzando queste due metriche temporali e identificando i corretti prodotti sostitutivi, quando necessario, le logiche per i due mix (clienti e mercato) genereranno in output un file CSV contenente il prodotto finale da suggerire al cliente (nel caso del mix clienti), insieme alla quantità da acquistare.

4.3.3 Prodotti Sostitutivi

Un aspetto cruciale nella realizzazione delle logiche è l'individuazione dei prodotti sostitutivi. Il processo di ricerca dei prodotti sostitutivi è identico per entrambi i mix.

Problema

La creazione di una tabella di supporto per i prodotti sostitutivi si rende necessaria per garantire che al cliente o al mercato venga proposto il prodotto attualmente "valido", evitando così di considerare prodotti che potrebbero uscire dal mercato prima dell'effettivo ordine. Questo processo ha inoltre l'obiettivo di diversificare le vendite, offrendo

una varietà di articoli differenti rispetto a quelli acquistati periodicamente dai clienti, con l'intento di intensificare le vendite e promuovere la distribuzione di una gamma più ampia di prodotti.

Di seguito, è riportato il codice utilizzato per costruire la tabella delle sostituzioni storiche per prodotto.

```
// Lettura dataset info prodotti
df1 = df_info.select('ip5', 'prod_id_sost','dt_validita',
    ↪ 'vincolo').distinct().filter("vincolo = 2 or vincolo =
    ↪ 1").orderBy('dt_validita')
df1 = df1.withColumnRenamed('ip5', 'prod_id')

// Join estrazione data di ingresso e di uscita del prodotto
    ↪ sostitutivo
df2 = df1.alias("df1").join(
    df1.alias("df2"),
    col("df1.prod_id_sost") == col("df2.prod_id"),
    "left"
).select(
    col("df1.prod_id"),
    col("df1.prod_id_sost"),
    col("df2.vincolo"),
    col("df1.dt_validita").alias("df1_dt_validita"),
    col("df2.dt_validita").alias("df2_dt_validita"),
    col("df2.prod_id_sost").alias("sost_sost")
)

// Set delle date corrette per il prodotto sostitutivo
df_history_sost = df2.withColumn(
    "dal",
```

```
    when(col("vincolo") == 2, col("df1_dt_validita"))
    .when((col("vincolo") == 1), col("df2_dt_validita"))
    .otherwise(col("df2_dt_validita"))
).withColumn(
    "al",
    when((col("vincolo") == 2), col('df2_dt_validita'))
    .when(col("vincolo") == 1, lit(None))
    .otherwise(None)
).select("prod_id", "prod_id_sost", "dal", "al", "vincolo",
   ↪ "sost_sost")

// Inizializzazione variabili di supporto
new_rows = df_history_sost
check = 0

// Ciclo ricorsivo per l'aggiunta dei prodotti sostitutivi nella nuova
   ↪ tabella
while  ((new_rows.cache() \
         .filter("vincolo = 2 and sost_sost is not null").count() > 0)):
    df3 = new_rows.filter("vincolo = 2 and sost_sost is not null")

    // Creazione nuovo df per info del sostituto del prodotto
       ↪ sostitutivo
    df4 = df3.alias("d3").join(
        df1.alias("df4"),
        col("d3.sost_sost") == col("df4.prod_id"),
        "left").select(
        col("d3.prod_id").alias("prod_origin"),
        col("d3.prod_id_sost").alias("prod_new"),
        col("d3.dal"),
        col("d3.al"),
```

```
    col("d3.sost_sost"),
    col("df4.*")
).filter("prod_new != prod_id_sost or prod_id_sost is null")

// Assegnazione ultimo sostituto al prodotto di origine
df5 = df4.withColumn(
    "dal",
    when(col("vincolo") == 2, col("al"))
    .when((col("vincolo") == 1), col("dt_validita"))
    .otherwise(lit(None))
).withColumn(
    "al",
    when(col("vincolo") == 1, lit(None))
    .when(col("vincolo") == 2, col("dt_validita"))
    .otherwise(None)
).withColumn(
    "prod_new", lit(col("prod_id"))
).withColumn(
    "sost_sost", (col("prod_id_sost"))
).select(
    col("prod_origin").alias("prod_id"),
    col("prod_new").alias("prod_id_sost"),
    col("dal"),
    col("al"),
    col("vincolo"),
    col("sost_sost")
)

df_history_sost = df_history_sost.unionByName(df5)
new_rows = df5
```

```
    // Check in caso di loop
    check = check+1
    assert check < 10, "Probabile Loop"

df_history_sost = df_history_sost.select(
"prod_id", "prod_id_sost", "dal", "al", "vincolo") \
.filter("vincolo is not null").orderBy('dal')
```

4.8 Creazione tabella prodotti sostitutivi

4.3.4 Suddivisione Temporale

Utilizzando come fonte di partenza la tabella degli storici sostotituzioni sono state prese in considerazione le due metriche temporali (prebooking e monthly) per creare due tabelle distinte a seconda della metrica utilizzata per l'ordine.

```
// Calcolo monthly
monthly_date = F.last_day(F.add_months(current_date(), 1))

// Controllo che la data monthly sia valida per il prodotto
df_history_sost_monthly = df_history_sost.filter(
    (monthly_date >= col('dal')) &
    (monthly_date <= col('al'))
)

// Conteggio numero di righe con condizione valida
windowSpec = Window.partitionBy("prod_id").orderBy(col("al").desc())

df_history_sost_monthly =
    ↪ df_history_sost_monthly.withColumn("row_number",
    ↪ row_number().over(windowSpec))
```

```
// Selezione prima riga valida
df_history_sost_monthly = df_history_sost_monthly.filter("row_number =
    ↪ 1")

// Creazione nuovo dataframe per ordini monthly
df_history_sost_monthly =
    ↪ df_history_sost_monthly.withColumnRenamed('dal', 'dal_monthly')
df_history_sost_monthly =
    ↪ df_history_sost_monthly.withColumnRenamed('al', 'al_monthly')
df_history_sost_monthly =
    ↪ df_history_sost_monthly.withColumnRenamed('prod_id_sost',
    ↪ 'sost_monthly')
```

4.9 Creazione tabella sostituzioni monthly

Processo analogo eseguito per la creazione della tabella degli ordini sostitutivi prebooking.

```
// Calcolo date fisse per stagionalit  (15 Aprile e 15 Settembre)
    current_year = datetime.now().year
    april_15_pre = datetime(datetime.now().year, 4, 15)
    sept_15_pre = datetime(datetime.now().year, 9, 15)
    april_15_post = datetime(datetime.now().year, 4, 15)

    // Selezione data in base al giorno di esecuzione codice
    if(datetime.now() <= april_15_pre):
        prebooking_data = april_15_pre
    elif((datetime.now() <= sept_15_pre) & (datetime.now() >
       ↪ april_15_pre)):
        prebooking_data = sept_15_pre
    elif(datetime.now() <= april_15_post):
        prebooking_data = april_15_post
```

```
// Filtro prodotti validi
df_history_sost_pre = df_history_sost.filter(
    (prebooking_data >= col('dal')) &
    (prebooking_data <= col('al'))
)

// Conteggio numero righe prodotti validi
windowSpec =
   ↪ Window.partitionBy("prod_id").orderBy(col("al").desc())

df_history_sost_pre = df_history_sost_pre.withColumn("row_number",
   ↪ row_number().over(windowSpec))

// Filtro per il primo prodotto valido
df_history_sost_pre = df_history_sost_pre.filter("row_number = 1")

// Creazione dataframe prodotto sostitutivi prebooking
df_history_sost_pre = df_history_sost_pre.withColumnRenamed('dal',
   ↪ 'dal_prebooking')
df_history_sost_pre = df_history_sost_pre.withColumnRenamed('al',
   ↪ 'al_prebooking')
df_history_sost_pre =
   ↪ df_history_sost_pre.withColumnRenamed('prod_id_sost',
   ↪ 'sost_prebooking')
```

4.10 Creazione tabella sostituzioni prebooking

4.3.5 Calcolo delle Quantità di Vendita

L'ultima sezione, ma non per importanza, nel contesto della creazione delle logiche di supporto alle vendite riguarda l'algoritmo di suddivisione delle quantità di acquisto per i clienti relativamente a un determinato prodotto. Le quantità vengono calcolate in base agli storici di acquisto, sia per i singoli clienti sia per il mercato corrispondente.

```
// Considerazione prodotti con chiave non nulla
step_5 = step_4_5.filter("prod_key IS NOT NULL")

// Calcolo del valore di vendita totale data la chiave
windowSpec = Window.partitionBy('whs_code', 'order_type', 'ip5')
step_5 = step_5.withColumn('tot_value_key',
    ↪ F.sum('value_key').over(windowSpec))

// Calcolo del peso
step_5 = step_5.withColumn('weight', F.col('value_key') /
    ↪ F.col('tot_value_key'))

// Calcolo quantit
step_5 = step_5.withColumn('final_value', F.col('weight') *
    ↪ F.col('sellin_fy'))
step_5 = step_5.withColumn('key_final_prod', lit(col('key')))
```

4.11 Calcolo quantità per mercato o cliente

La logica sviluppata prende in considerazione le righe che hanno un valore di chiave del prodotto non nulla e dunque con un valore di vendita, successivamente calcola il valore totale per tutte le vendite effettuate per quel cliente o mercato per poi calcolare il peso dato dal valore di vendita effettuato per quel prodotto al cliente o mercato. Dati questi valori si calcola infine la quantità finale da proporre al cliente o mercato.

4.3.6 Conclusioni

In conclusione, per questo paragrafo dedicato allo sviluppo delle logiche di supporto, il risultato finale può essere rappresentato da una tabella sia per il mix clienti che per il mix mercato. Ciascuna di queste tabelle contiene i prodotti finali da proporre, con le relative quantità calcolate in base a fattori come la stagionalità (tipo di ordine) e lo storico degli ordini effettuati.

4.4 Sviluppo Modelli di Machine Learning

4.4.1 Dataset

Il dataset impiegato per l'analisi e la previsione dei volumi di vendita futuri è stato estratto da fonti specifiche, riferendosi ai dati relativi a un singolo cliente di un'azienda manifatturiera. Questo dataset include i valori delle vendite effettuate a tale cliente nel periodo compreso tra il 1 gennaio 2013 e il 1 giugno 2024.

N° Vendite	**Giorno**	**Mese**	**Anno**
12.456	01	05	2014
37.990	01	11	2017
4.568	01	02	2021
...	...	...	...

Tabella 4.1: Esempio Struttura Dataset

4.4.2 Data Exploration

L'esplorazione dei dati e la visualizzazione rappresentano fasi cruciali in un progetto analitico, poiché consentono di comprendere in modo approfondito le caratteristiche e le relazioni all'interno dei dati stessi. Attraverso queste attività, è possibile identificare pattern, anomalie e tendenze, facilitando così l'interpretazione dei risultati.

Inoltre, la visualizzazione dei dati permette di comunicare in modo chiaro e efficace le scoperte e le intuizioni emerse durante l'analisi. Questi processi non solo supportano la validazione delle ipotesi iniziali, ma contribuiscono anche a informare decisioni strategiche basate su evidenze, aumentando la probabilità di successo del progetto.

Dall'analisi del dataset iniziale sono emersi diversi risultati, che verranno presentati e discussi nelle immagini seguenti.

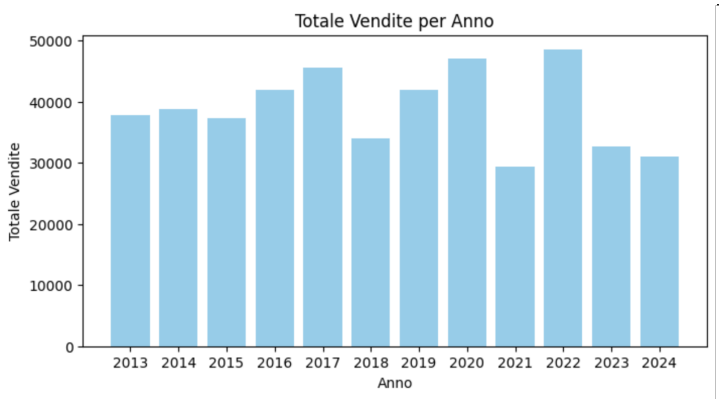

Figura 4.8: Istogramma Vendite Totale per Anno

Nella Figura 4.8 sono rappresentate le vendite totali di pneumatici effettuate verso il cliente per ciascun anno analizzato. Dall'immagine si evince che non si sono registrati anni con vendite nulle, evidenziando una costante interazione con il cliente.

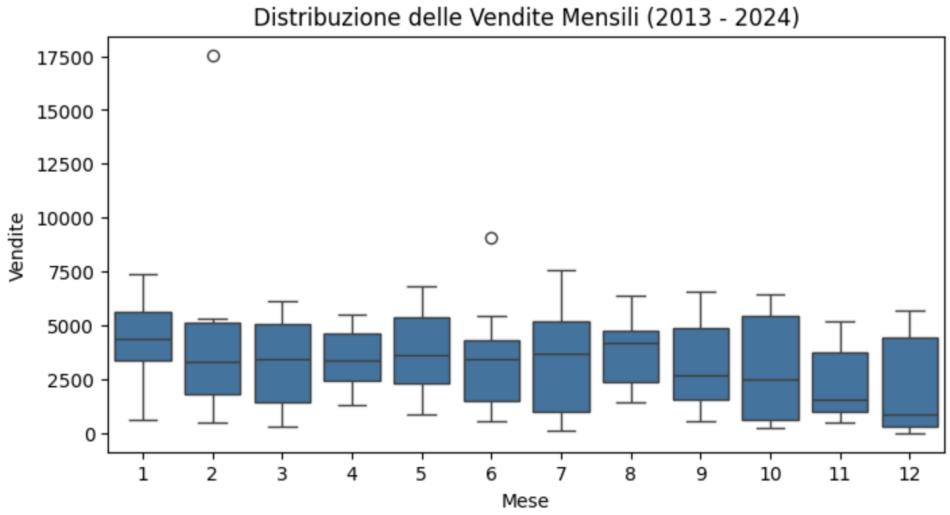

Figura 4.9: Box Plot Vendite Mensiili

Il Box Plot mostrato nella Figura 4.9 consente di visualizzare la distribuzione delle vendite mensili, risultando utile per identificare eventuali variazioni stagionali e per osservare la presenza di outlier nei dati.

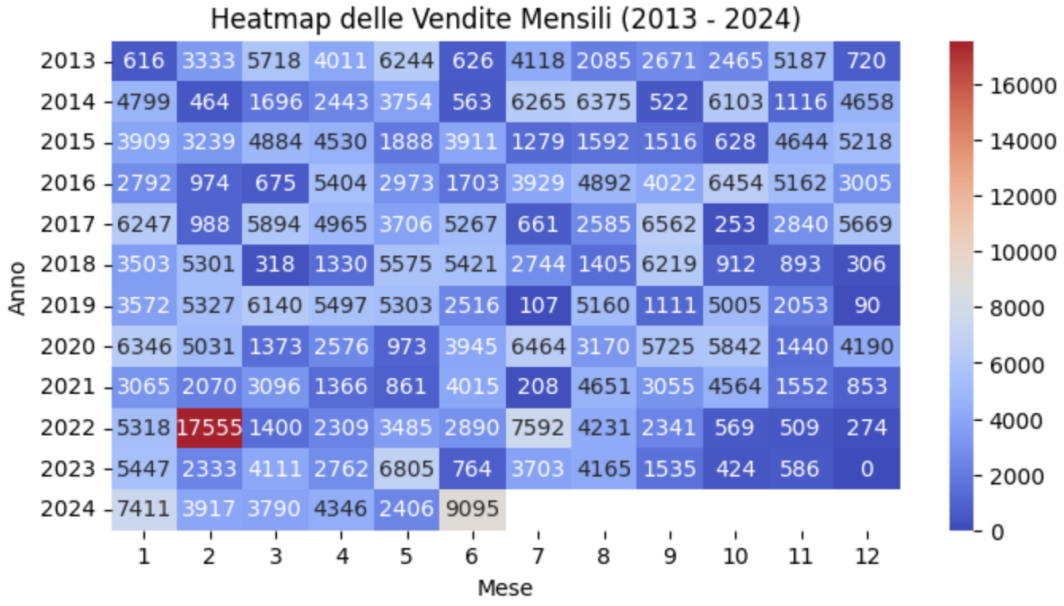

Figura 4.10: Heatmap Vendite Mensiili nel Tempo

La Figura 4.10 presenta, in forma di matrice, le vendite registrate mese per mese e anno per anno, offrendo una rappresentazione più chiara dei dati di vendita. Attraverso l'uso di una scala cromatica variabile, la figura evidenzia i mesi e gli anni in cui le vendite sono state superiori o inferiori alla media, contrassegnata con una tonalità grigia neutra.

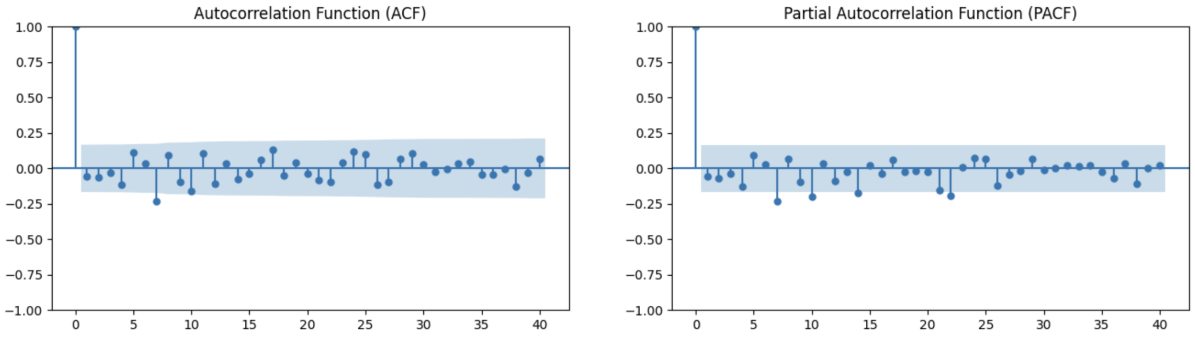

Figura 4.11: ACF e PACF

La Figura 4.11 raffigura l'Autocorrelation Plot (ACF e PACF), strumento utile per verificare la correlazione tra i valori della serie temporale, al fine di determinare il grado di dipendenza tra i dati a vari lag. L'obiettivo è individuare eventuali autocorrelazioni che potrebbero indicare ciclicità nei dati. Come evidenziato dalla figura, non si riscontrano ciclicità, poiché le vendite nel tempo risultano indipendenti tra loro.

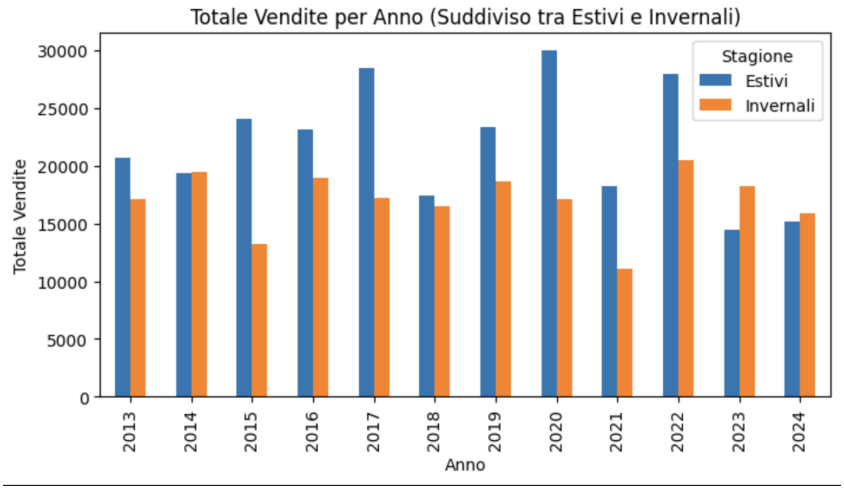

Figura 4.12: Istogramma Vendite Stagionali

Nell'ultima figura, la 4.12, è stato impiegato un istogramma per confrontare le vendite utilizzando la stagionalità come criterio di analisi. In particolare, le vendite di pneumatici invernali sono evidenziate in arancione, mentre quelle di pneumatici estivi in blu, permettendo di identificare, per ciascun anno, quale tipologia di pneumatico (estivo o invernale) sia stata venduta maggiormente rispetto al totale delle vendite.

4.4.3 Prophet

Il primo modello impiegato per la previsione dei volumi di vendita è stato Prophet, un modello precedentemente discusso, che consente di effettuare previsioni per un numero selezionabile di mesi, utilizzati come unità di misura temporale, basandosi sui dati forniti in input.

La creazione del modello e la configurazione dei parametri sono riportate nel codice seguente:

```
# Preprocess dati Prophet
prophet_data = data.rename(columns={'Date': 'ds', 'Sales': 'y'})
```

```
# Inizializzare modello
model = Prophet()
model.add_seasonality(name='monthly', period=30, fourier_order=3)
model.fit(prophet_data)

# Creazione dataframe futuro per la previsione
future = model.make_future_dataframe(periods=1, freq='M') # Predice il
    ↪ mese successivo
forecast = model.predict(future)
```

4.12 Creazione Modello Prophet

Il codice presentato inizia con la rinomina delle colonne della tabella *Date* e *Sales* in *ds* e *y*, un passaggio necessario per garantire il corretto funzionamento del modello Prophet. Successivamente, il modello Prophet viene inizializzato e assegnato alla variabile 'model', al quale viene aggiunta una suddivisione mensile con intervalli di 30 giorni. Si è scelto di utilizzare un valore di 3 per i termini di Fourier, al fine di catturare la complessità dei dati stagionali. Infine, il modello viene addestrato e il risultato salvato, impostando un periodo di previsione pari a 1, corrispondente al numero di mesi da predire.

Successivamente, sono state integrate le informazioni precedentemente rilevate riguardanti la stagionalità, analizzando i valori previsti per le stagioni invernale ed estiva.

```
forecast['Season'] = forecast['ds'].dt.month.apply(classify_season)
forecast_seasonal = forecast.groupby(['ds',
    ↪ 'Season'])['yhat'].sum().reset_index()
```

4.13 Previsione Predizioni Stagionali

4.4.4 ARIMA

Il compito di previsione è stato successivamente eseguito utilizzando il secondo modello considerato, ARIMA. Tramite questo modello è stato possibile prevedere le vendite per un numero selezionabile di mesi futuri.

```
data_arima = data.rename(columns={'Date': 'ds', 'Sales': 'y'})
data_arima = data_arima.set_index('ds')['y']
# Definizione modello ARIMA
arima_model = ARIMA(data_arima, order=(10, 0, 2))
arima_fit = arima_model.fit()

# Previsione mese successivo
forecast_arima = arima_fit.forecast(steps=1)
```

4.14 Previsione Vendite con ARIMA

Anche per questo modello è stata effettuata una rinomina delle colonne per garantire il corretto funzionamento. Successivamente, sono stati impostati tre parametri di input corrispondenti a p, d e q:

- **p = 10**: rappresenta l'ordine autoregressivo (AR), ossia il numero di lag passati della serie temporale utilizzati per prevedere il valore corrente.
- **d = 0**: indica il grado di differenziazione necessario per rendere la serie stazionaria. In questo caso, il valore è 0, quindi non viene applicata alcuna differenziazione.
- **q = 2**: rappresenta l'ordine della media mobile (MA), cioè il numero di lag degli errori passati considerati nel modello.

Dopo aver impostato questi parametri, il modello viene addestrato utilizzando il metodo *fit()* e si imposta un numero di passi pari a 1 per determinare quanti mesi futuri prevedere.

Infine, per studiare la stagionalità anche per questo modello è stato seguito lo stesso procedimento applicato al precedente modello (Prophet) riportato all'interno del codice 4.13.

Capitolo 5

Validazione

Il presente capitolo ha l'obiettivo di presentare i risultati ottenuti dall'applicazione delle metodologie e degli strumenti sviluppati nei capitoli precedenti.
La validazione delle soluzioni implementate si basa su un confronto sistematico tra i risultati ottenuti e gli obiettivi prefissati durante la fase di progettazione, consentendo di evidenziare punti di forza e possibili aree di miglioramento. Saranno inoltre presentati alcuni casi studio esemplificativi, che dimostrano l'efficacia delle soluzioni sviluppate nel rispondere alle esigenze specifiche del settore pneumatico, in termini di previsione delle vendite e ottimizzazione delle opportunità di mercato.
Questo capitolo intende fornire una valutazione critica delle soluzioni adottate, mettendo in luce l'impatto positivo che un approccio data-driven può avere sulla competitività aziendale, e dimostrando come l'analisi avanzata dei dati possa contribuire a una gestione più efficiente delle strategie di vendita.

5.1 Validazione Sistema ETL e Dashboard

In relazione al primo obiettivo di questo elaborato, che consiste nella creazione di un sistema ETL e di un applicativo di reportistica, si procederà alla discussione dei risultati ottenuti in merito a due specifiche finalità. La prima finalità riguarda il calcolo del potenziale di vendita basato sull'analisi del parco auto circolante; la seconda, il monitoraggio delle statistiche di vendita. Nel corso della discussione, verranno presentati i risultati raggiunti per entrambi gli scopi, illustrando lo schema ETL sviluppato in seguito all'integrazione delle sorgenti dati necessarie al conseguimento degli obiettivi prefissati. Successivamente, si esaminerà la dashboard finale, mettendo in evidenza un caso d'uso concreto dell'applicativo di reportistica, al fine di dimostrare l'efficacia delle soluzioni implementate e l'impatto positivo che esse hanno avuto nell'ottimizzazione dei processi di vendita dell'azienda.

5.1.1 ETL

A partire dalla struttura interna, vengono presentate entrambe le figure che illustrano la struttura finale dell'ETL ottenuta.

Parco Auto Circolante

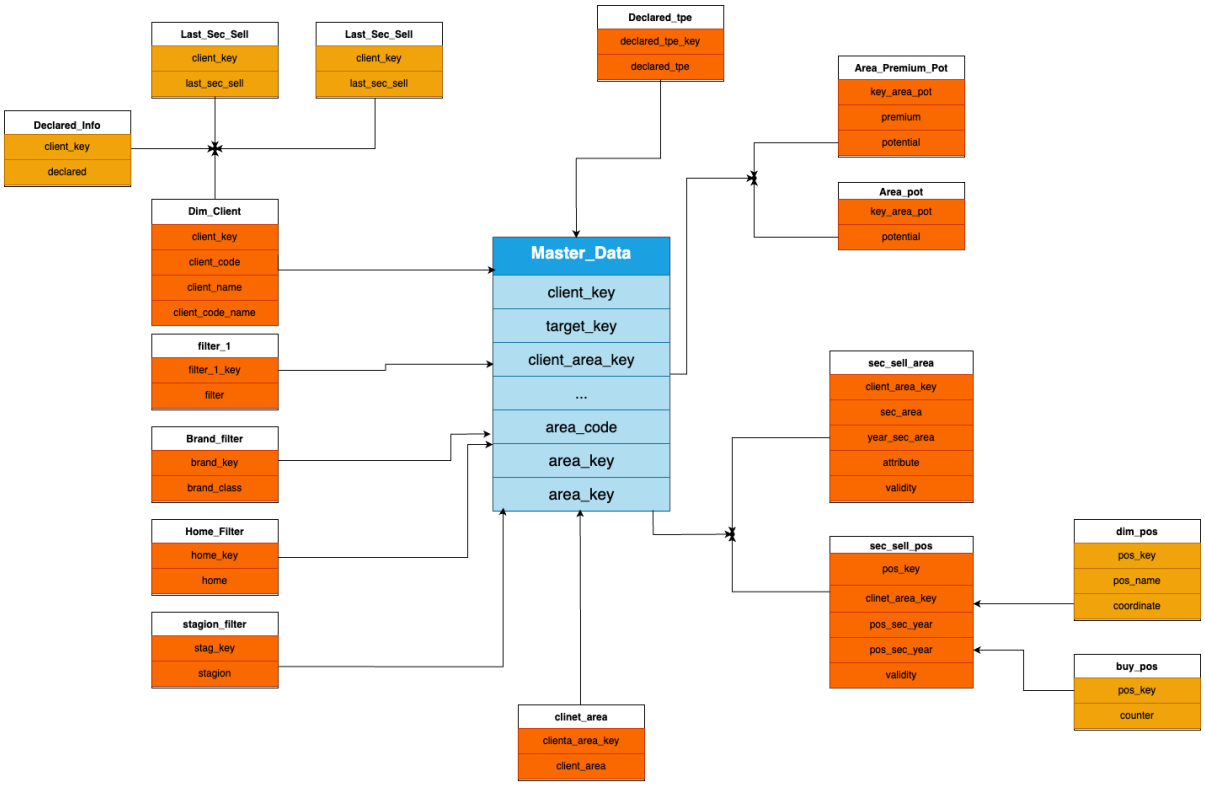

Figura 5.1: ETL Analisi Parco Auto Circolante

Lo schema ETL presentato evidenzia una struttura complessa in cui diverse tabelle sono collegate tra loro per supportare il processo di analisi e reportistica in un contesto aziendale. All'interno di questo schema, è possibile distinguere tra tabelle di dimensioni e tabelle dei fatti, che svolgono ruoli fondamentali nell'organizzazione e nella manipolazione dei dati.

Le tabelle di **dimensioni** forniscono contesti descrittivi per i dati numerici presenti

nelle tabelle dei fatti, tra queste si possono indicare alcune come: *dim-client* e *dim-pos*, permettono di ottenere informazioni dettagliati sotto vari aspetti.

Le tabelle dei **fatti** al contrario contengono dati quantitativi che sono oggetto di analisi, alcune di esse sono: *sec-sell-pos* e *area-pot* per citarne alcune, permettono di osservare i valori numerici che riguardano fattori di registrazione, vendita, auto circolanti.

Le relazioni tra le tabelle sono indicative di un processo di analisi articolato che coinvolge il monitoraggio delle vendite in diverse dimensioni, come le aree geografiche, i magazzini e i punti vendita, oltre alla valutazione del potenziale di mercato basato su differenti attributi. La tabella centrale *Master-Data* funge da punto di convergenza per molte dimensioni, facilitando l'integrazione dei dati necessari per l'analisi avanzata.

Analisi Statistiche di Vendita

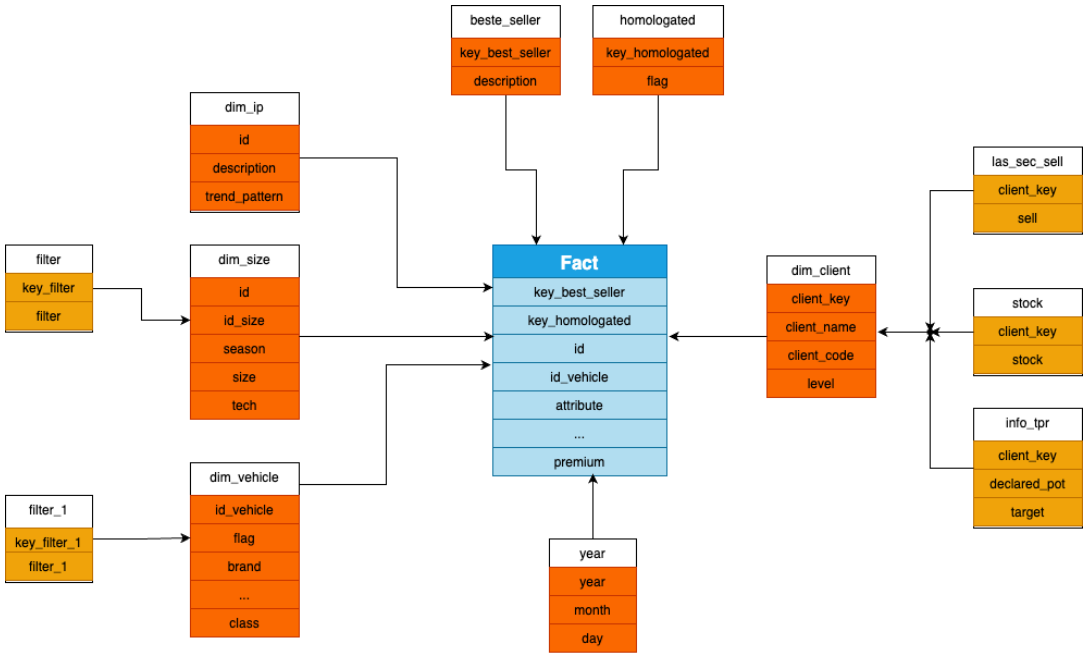

Figura 5.2: ETL Statistiche di Vendita

Lo schema ETL presentato in questa figura evidenzia una struttura dati mirata all'analisi dettagliata delle performance di vendita. Anche in questo caso, è possibile

distinguere tra tabelle di dimensioni e tabelle dei fatti, ognuna delle quali svolge un ruolo cruciale nella costruzione del sistema di analisi.
Per questo caso possiamo individuare tra le tabelle delle dimensioni alcune come: *dim-client* e *dim-size*, le quali permettono di ottenere informazioni realativi a diversi fattori come magazzini e veicoli.
Come per la struttura precedente le diverse tabelle dei fatti presenti mostrano i dati quantitativi cruciali per l'analisi.
All'interno dello schema ETL si riportano connessioni articolate tra la tabella dei fatti e le tabelle di dimensioni, facilitando un'analisi approfondita delle vendite, con la possibilità di segmentare i dati in base a variabili come le dimensioni del prodotto, i veicoli, i grossisti e le omologazioni. La presenza di filtri specifici e di una tabella dello scenario corrente suggerisce un'analisi dinamica e flessibile, capace di adattarsi a diverse esigenze di reporting e di ottimizzazione delle vendite.

5.1.2 Dashboard

Lo sviluppo dei due schemi ETL ha consentito, tramite l'utilizzo dell'applicativo Qlik, l'implementazione di un'interfaccia grafica strutturata. Sono state definite diverse pagine per ciascuno dei due ambiti applicativi.

Per quanto concerne il primo obiettivo, è stata progettata un'interfaccia che presenta, in una pagina, il potenziale derivato dal parco auto circolante, mentre una seconda pagina visualizza il dataset dei prodotti disponibili relativi a un cliente selezionato.

In riferimento al secondo obiettivo, sono state create tre pagine. La prima è dedicata all'analisi delle statistiche di vendita; la seconda offre una rappresentazione grafica, tramite diagrammi comparativi rispetto all'anno precedente; infine, la terza pagina riporta i prodotti acquistati da un determinato cliente.

Parco Auto Circolante

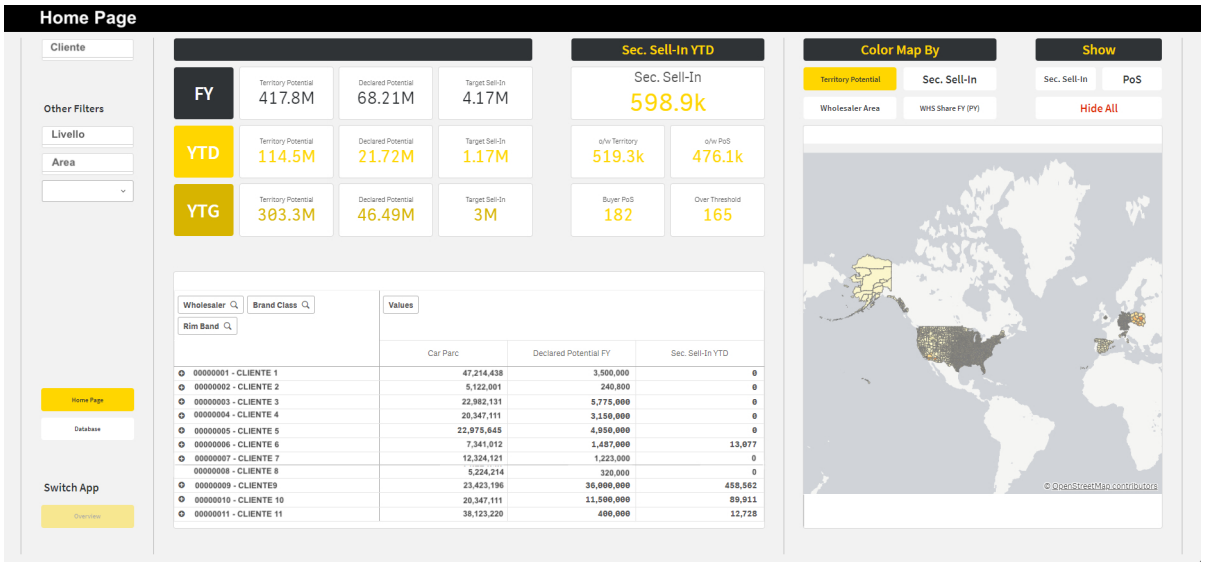

Figura 5.3: Home Page - Parco Auto

La pagina iniziale della dashboard, illustrata nella Figura 5.3, si presenta con una struttura ben definita. Sul lato sinistro della schermata, è possibile selezionare diversi filtri per eseguire ricerche specifiche. La parte centrale mostra una serie di valori relativi al potenziale territoriale, calcolato in base al numero di auto circolanti, al potenziale determinato dal reparto marketing, e al target di pneumatici che si intende vendere rispetto al totale. Questi valori sono suddivisi in tre metriche distinte. La prima, denominata "FY", indica i valori relativi all'anno precedente a quello corrente (in questo caso, si riferiscono all'anno 2023). La seconda metrica, "YTD", rappresenta i potenziali e il target dell'anno precedente fino al giorno corrente (ad esempio, se la data odierna è il 10/06/2024, il valore si riferisce al periodo dal 01/01/2023 al 10/06/2023). L'ultima metrica, "YTG", indica i potenziali di vendita previsti dal giorno corrente in cui i valori sono stati aggiornati fino al termine dell'anno solare. Nella parte inferiore della pagina, è riportata un'anteprima del dataset contenente i vari clienti e i rispettivi valori, calcolati

in base alla loro "espansione". Si segnala che, per motivi di privacy, i nomi e i valori reali dei clienti sono stati oscurati. Infine, sulla destra, è visibile una mappa, realizzata tramite la libreria 'OpenStreetMap', che raffigura il globo, evidenziando i territori in cui sono presenti i clienti dell'azienda manifatturiera.

Di seguito si riporta un caso d'uso mostrante il funzionamento dell'applicativo per un cliente selezionato.

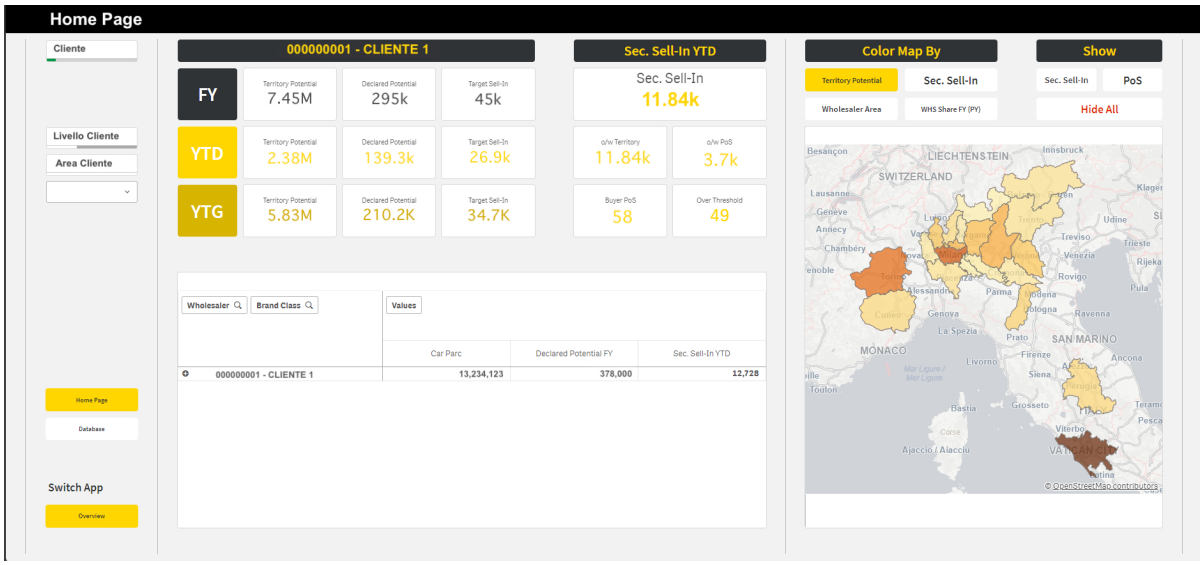

Figura 5.4: Home Page - Parco Auto - Caso d'uso

Come illustrato nella Figura 5.4, un caso d'uso generico dell'applicativo prevede la selezione di un cliente specifico, consentendo così di visualizzare il potenziale di vendita in base ai territori in cui sono presenti i suoi punti vendita. All'interno della mappa, è possibile distinguere i territori evidenziati da diversi colori: le tonalità più scure indicano un potenziale maggiore.

Selezionando una regione del territorio è possibile visualizzare dettagli relativi ad essa.

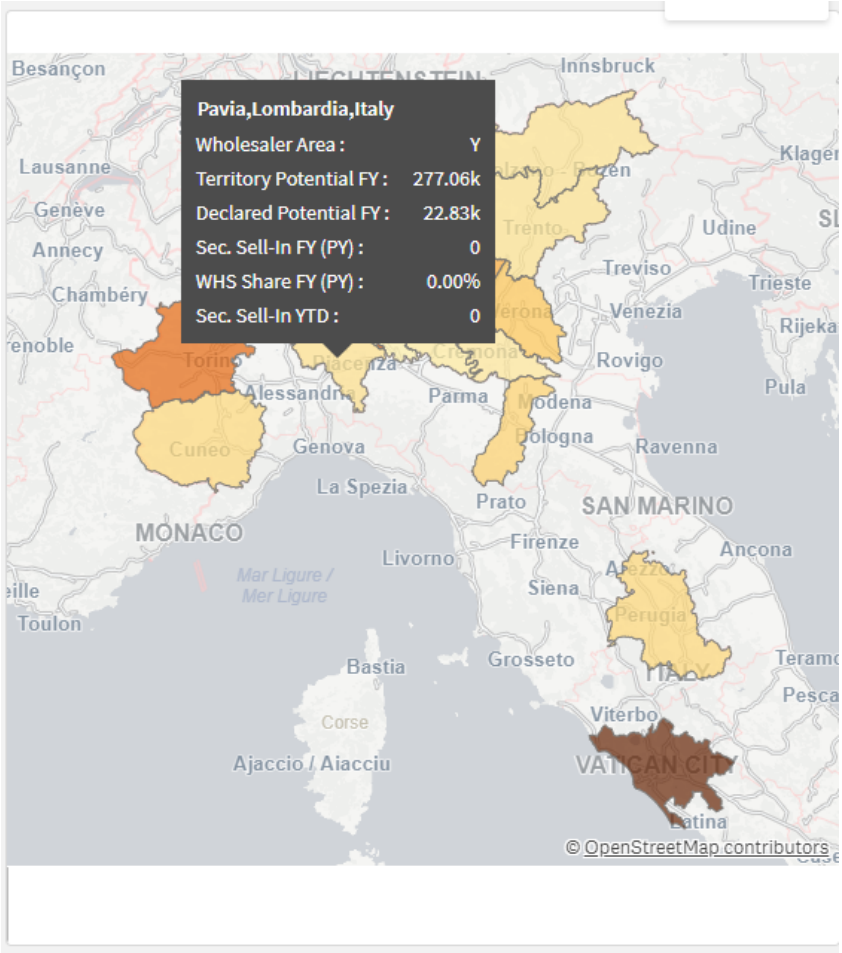

Figura 5.5: Info Potenziale Territorio

All'interno della Figura 5.5 viene mostrato, per il territorio selezionato, il potenziale indicato per le diverse metriche descritte in precedenza.

Analisi Statistiche di Vendita

La seconda dashboard creata per l'analisi delle statistiche di vendita presenta una pagina iniziale che visualizza differenti valori relativi ad un cliente.

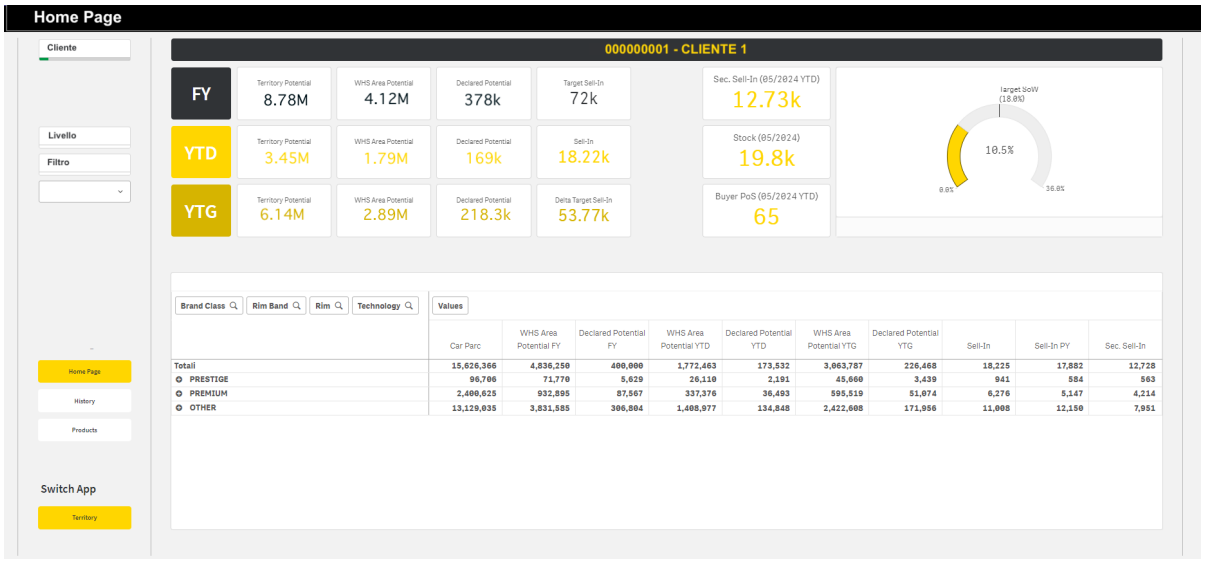

Figura 5.6: Home Page - Statistiche Vendita

Come illustrato nella Figura 5.6, la dashboard presenta un caso d'uso che riporta le statistiche di vendita di un cliente. I valori mostrati nella parte centrale sono gli stessi descritti nella Figura 5.3. Tuttavia, la dashboard fornisce ulteriori informazioni riguardanti il raggiungimento del target, ovvero l'obiettivo di vendita dei pneumatici entro la fine dell'anno solare.

Nella tabella situata nella parte inferiore della pagina, sono riportati i dati relativi alle tre tipologie di pneumatici, con valori presentati in modo più dettagliato, al fine di consentire analisi supplementari da parte del reparto marketing.

Figura 5.7: Storico Vendite

La Figura 5.7 mostra la seconda pagina della dashboard dedicata all'analisi delle statistiche di vendita. Questa pagina è stata progettata per offrire un confronto più rappresentativo dei dati di vendita distribuiti tra le diverse metriche temporali. Inoltre, nei tre riquadri situati nella parte inferiore della pagina, sono visualizzati i dati relativi alle statistiche delle diverse tipologie di pneumatico, in modo tale da monitorarne le vendite.

5.2 Validazione Logiche di Supporto

Sulla base delle analisi condotte e descritte nei capitoli precedenti, nel seguente paragrafo verranno presentati e discussi i risultati relativi allo sviluppo delle logiche di supporto per la proposta di vendita di prodotti e quantità.

Per ciascun mix precedentemente descritto, è stato generato un output in formato tabellare, contenente nelle colonne le informazioni necessarie per il suggerimento dei prodotti e delle relative quantità di acquisto.

Prima di esaminare i risultati finali dei due mix, saranno analizzati i risultati ottenuti durante lo sviluppo dei diversi algoritmi volti a supportare la creazione delle logiche.

(**N.B.** A causa delle politiche di privacy dell'azienda manifatturiera per la quale è stato svolto il lavoro, i risultati riportati all'interno delle tabelle saranno presentati con modifiche alla struttura e ai valori. Questa scelta ha l'obiettivo di mantenere intatto il concetto dei risultati ottenuti, pur oscurando i dati reali.)

Tabella Storico Prodotti Sostitutivi

Come evidenziato nel capitolo precedente, la costruzione di una tabella in grado di tracciare la sostituzione dei prodotti con nuove versioni nel corso del tempo riveste un'importanza cruciale per lo svolgimento dello studio e la formulazione delle logiche necessarie.

La tabella risultante dal codice riportato nel listing *4.7 Creazione tabelle prodotti sostitutivi* è la seguente:

Prodotto Originale	**Prodotto Sostitutivo**	**Dal**	**Al**
p2	pF	10/12/2018	null
p1	p2	01/05/2016	10/12/2018
null	p1	01/01/2010	01/05/2016

Tabella 5.1: Storico sostituzioni prodotti

Come riportato nella Tabella 5.1, quando un prodotto originale viene introdotto sul mercato (ad esempio, il prodotto p1), esso viene inserito nella colonna *'Prodotto Sostitutivo'*, specificando le date di ingresso e uscita dal mercato (dal 01/01/2010 al 01/05/2016). Nel momento in cui un prodotto viene sostituito, la sostituzione è indicata nella stessa

colonna, insieme alle date relative al periodo di sostituzione. Qualora il prodotto sostitutivo non disponga di una data di uscita dal mercato, esso viene considerato l'ultimo modello disponibile.

Tabella Calcolo Quantità

Per determinare le quantità di acquisto dei prodotti da parte dei clienti, sono stati analizzati gli storici degli acquisti. Sulla base di questi dati, sono stati calcolati dei pesi da assegnare a ciascun prodotto in relazione a uno specifico cliente. Di seguito è riportato un esempio dei risultati ottenuti, presentato in forma tabellare.

Prodotto	**Quantità**	**Peso**	**Totale**	**Cliente**
p1	3	0.08	35	cliente1
p1	3	0.08	35	cliente1
p1	4	0.11	35	cliente1
p2	15	0.42	35	cliente1
p2	10	0.28	35	cliente1

Tabella 5.2: Esempio calcolo pesi prodotti cliente1

A partire dalla Tabella 5.2, in cui sono stati calcolati i pesi dei prodotti rispetto al totale degli acquisti effettuati dal cliente 1, sono state determinate le quantità finali per ciascun prodotto. I risultati di tali calcoli sono riportati nella tabella seguente.

Prodotto Suggerito	**Quantità**	**Cliente**
p1	10	cliente1
p2	25	cliente1
p1	5	cliente2
p4	17	cliente3
p2	11	cliente 4

Tabella 5.3: Quantità finali prodotti suggeriti

La Tabella 5.3 rappresenta il risultato finale delle quantità suggerite per ciascun prodotto in relazione a un determinato cliente. Per un utilizzo più efficace, è possibile applicare un filtro per selezionare un cliente specifico, come illustrato nella tabella sottostante.

Prodotto Suggerito	**Quantità**	**Cliente**
p1	10	cliente1
p2	25	cliente1
p3	11	cliente1
p4	24	cliente1
p5	19	cliente1

Tabella 5.4: Quantità finali prodotti suggeriti Cliente 1

Procedendo in questo modo, è possibile elaborare una proposta di vendita sfruttando le informazioni contenute all'interno della tabella. Le quantità suggerite per ciascun prodotto, filtrate per cliente, possono essere utilizzate per formulare un'offerta personalizzata che risponda alle esigenze specifiche del cliente, ottimizzando così il processo di vendita e migliorando l'efficacia delle strategie commerciali.

Mix Mercato

Un discorso analogo a quello relativo al Mix Clienti può essere applicato per l'analisi del mercato, con l'unica differenza che, in questo caso, al posto della colonna *'Cliente'* è presente la colonna *'Mercato'*, che riporta il nome della nazione in cui è stata condotta l'analisi.

Le tabelle di supporto, come la Tabella 5.1, 5.2 e 5.3, vengono replicate anche durante la creazione di questo Mix, con la differenza che l'analisi è focalizzata sul mercato piuttosto che sui clienti. Questo approccio consente di adattare le strategie in base alle specificità del mercato di riferimento, mantenendo la struttura e la metodologia utilizzate per l'analisi del Mix Clienti.

Prodotto Suggerito	Quantità	Mercato
p1	10	Italia
p2	25	Italia
p3	11	Cina
p1	24	USA
p2	19	Germania

Tabella 5.5: Quantità e prodotti finali Mix Mercato

Conclusione

In conclusione, grazie alle logiche e alle tabelle ottenute precedentemente, è possibile stimare un suggerimento di acquisto per i prodotti e le relative quantità, basato sui dati storici delle vendite ai clienti e ai mercati. Assegnando pesi ai due mix—quello relativo ai clienti e quello relativo ai mercati—è possibile stabilire delle priorità sui valori da considerare con maggiore attenzione nella realizzazione del preventivo di vendita finale da proporre al cliente.

5.3 Validazione Predizioni di Vendita

I modelli di machine learning adottati per la previsione dei volumi d'acquisto hanno prodotto risultati interessanti. Per fornire un'interpretazione più solida delle capacità del machine learning, sono state effettuate diverse valutazioni utilizzando le seguenti metriche:

- **Mean Absolute Error (MAE)**
  - **Definizione**: Il MAE calcola la media degli errori assoluti tra le previsioni del modello e i valori reali.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{5.1}$$

  - **Signigicato nel contesto dell'elaborato**: Il MAE indica quanto, in media, le previsioni delle vendite si discostano dai valori reali. Un MAE più basso indica che il modello è in grado di fare previsioni più accurate.

- **Mean Squared Error (MSE)**

  - **Definizione**: Il MSE calcola la media degli errori al quadrato tra le previsioni e i valori reali.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{5.2}$$

  - **Signigicato nel contesto dell'elaborato**: Il MSE fornisce un'indicazione della variabilità delle previsioni rispetto ai valori reali. Penalizza maggiormente gli errori più grandi, il che significa che se ci sono previsioni significativamente errate, il MSE sarà più elevato.

- **Mean Absolute Percentage Error (MAPE)**

  - **Definizione**: Il MAPE misura l'accuratezza delle previsioni come una percentuale degli errori assoluti.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{5.3}$$

  - **Signigicato nel contesto dell'elaborato**: Il MAPE indica quanto le previsioni delle vendite si discostano dai valori reali in termini percentuali. Un MAPE basso significa che le previsioni sono generalmente molto vicine ai valori reali.

Attraverso l'utilizzo delle metriche precedentemente discusse, sono stati raccolti i valori effettuando diversi esperimenti su vari archi temporali di previsione. Sono stati calcolati il MAE, il MSE e il MAPE delle predizioni di entrambi i modelli per un periodo di 1 mese, 3 mesi e 6 mesi futuri.

Successivamente, sono stati confrontati i risultati ottenuti dai due modelli al fine di determinare quale fosse il migliore.

Prophet

In seguito all'utilizzo del modello Prophet, le predizioni delle vendite ottenute sono state rappresentate attraverso diverse visualizzazioni grafiche, al fine di fornire una visione più chiara dell'andamento delle vendite future in confronto a quelle "reali" del passato.

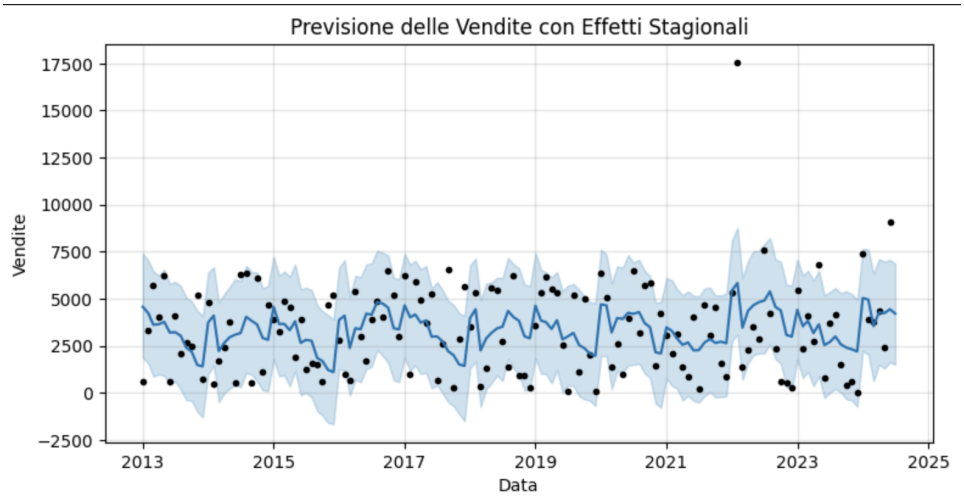

Figura 5.8: Predizione Vendite Prophet (1 mese)

Un procedimento analogo è stato effettuato per quanto concerne le previsioni, integrando l'informazione relativa alle stagionalità, il che ha prodotto i seguenti risultati:

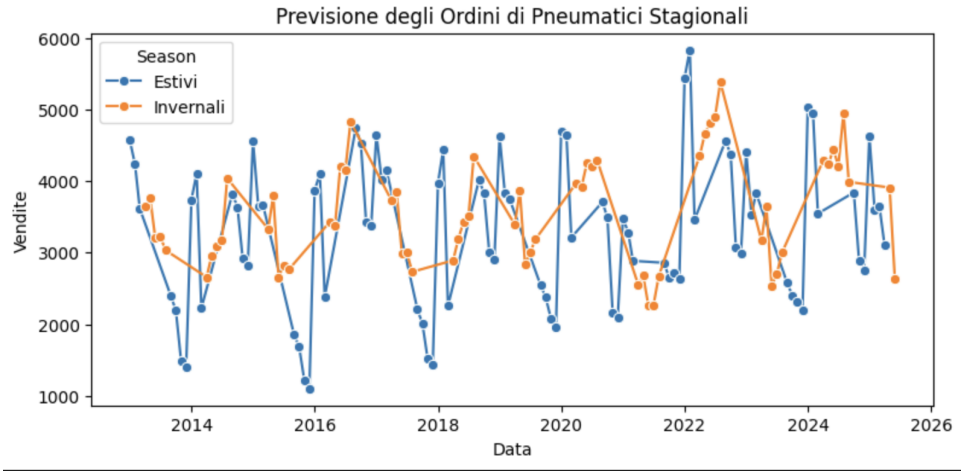

Figura 5.9: Predizione Vendite Prophet con Stagionalità (1 mese)

Per ottenere valori più comprensibili, sono state calcolate le metriche di valutazione del modello, producendo i risultati riportati nella seguente tabella:

Mesi Predizione	**MSE**	**MAE**	**MAPE%**
1	760.384	872	10.90
3	1.585.081	1259	15.04
6	2.521.744	1.588	19.71

Tabella 5.6: Metriche Valutazione Prophet

Dai risultati emersi, si può concludere come facilmente deducibile, che il modello risulta più affidabile quando si effettua la previsione su un numero limitato di mesi. In particolare, con un MAE di 872 unità, il modello indica un errore medio di circa 872 unità su una media di vendite mensili di circa 9.000 unità, corrispondente a un errore percentuale del 10.90%. Al contrario, aumentando il numero di mesi di previsione, l'errore tende ad aumentare, raggiungendo un MAPE del 19.71% per una previsione su sei mesi futuri.

ARIMA

Il lavoro svolto con il secondo modello è simile a quello realizzato con il primo. Sono state effettuate previsioni su tre orizzonti temporali differenti (1 mese, 3 mesi e 6 mesi), producendo una rappresentazione grafica dell'andamento delle vendite future. Inoltre, sono state calcolate le metriche necessarie per valutare le prestazioni del modello in ciascun caso.

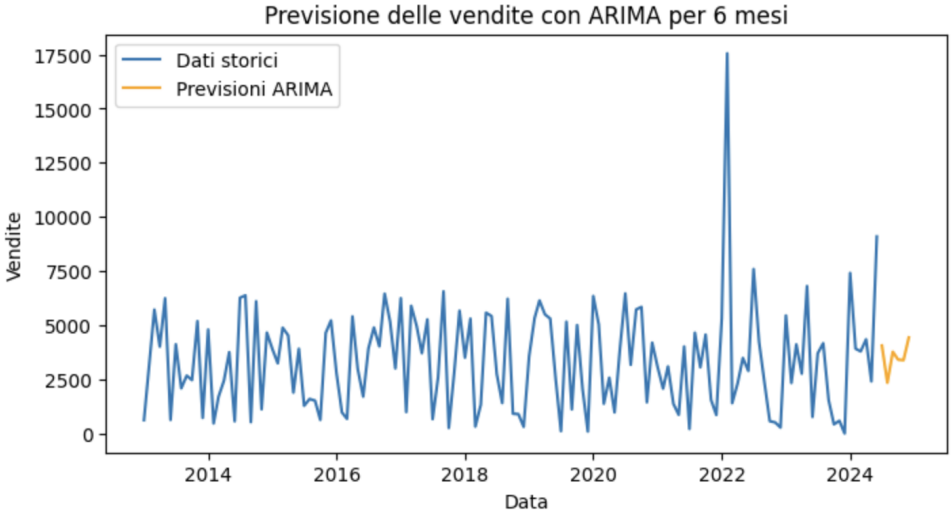

Figura 5.10: Predizione Vendite ARIMA (6 mesi)

Come per il precedente modello si è ripetuta la predizione utilizzando le informazioni relative alle stagionalità, ottenendo in questo modo la seguente figura:

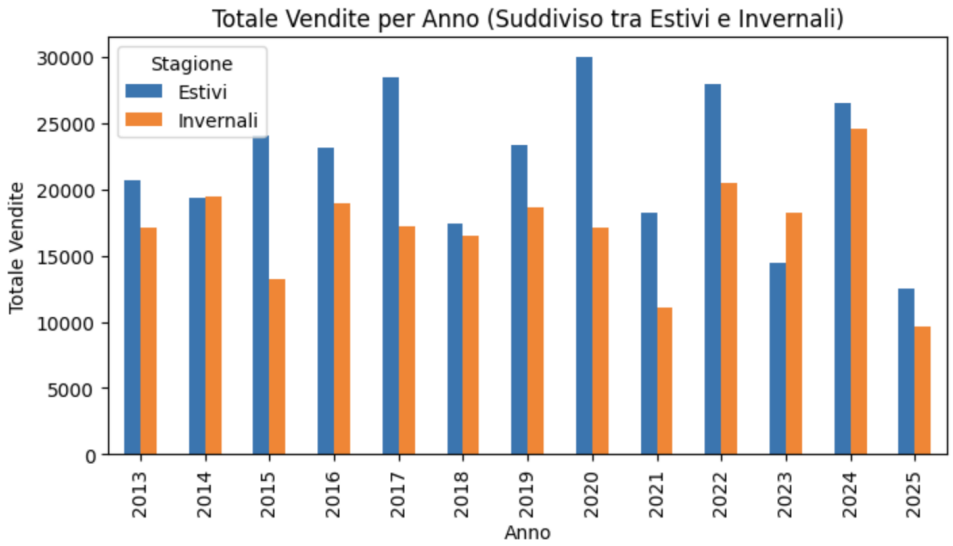

Figura 5.11: Predizione Vendite ARIMA con Stagionalità (6 mesi)

E infine sono state calcolate le metriche di valutazione relative ai risultati ottenuti utilizzando il modello ARIMA, per avere un confronto diretto rispetto a quanto ottenuto utilizzando il modello di Prophet.

Mesi Predizione	**MSE**	**MAE**	**MAPE%**
1	992.132	996	12.03
3	1.788.119	1337	17.57
6	2.891.321	1700	24.11

Tabella 5.7: Metriche Valutazione ARIMA

Confronto risultati Prophet e ARIMA

Dai risultati ottenuti mediante l'applicazione dei due modelli di Machine Learning, emerge una performance superiore del modello Prophet, che ha mostrato una percentuale di errore inferiore in tutte e tre le previsioni con diverse scale temporali. Sebbene la precisione delle previsioni tenda a diminuire con l'aumento del periodo di predizione, il modello Prophet può comunque essere considerato la scelta più appropriata per la previsione delle

vendite future, grazie alla sua capacità di fornire risultati più affidabili rispetto agli altri modelli analizzati.

Capitolo 6

Conclusioni e Sviluppi Futuri

In conclusione, il lavoro di tesi ha raggiunto pienamente gli obiettivi prefissati, dimostrando come l'applicazione di soluzioni avanzate di Data Analytics possa migliorare significativamente i processi decisionali di un'azienda manifatturiera nel settore dei pneumatici. Gli strumenti sviluppati hanno fornito un contributo essenziale nell'analisi delle opportunità di mercato e nella previsione delle vendite, utilizzando un approccio basato sui dati storici, di mercato, e sulle caratteristiche specifiche dei prodotti e dei clienti.

Il primo obiettivo, relativo allo sviluppo del sistema ETL è stato portato a compimento in modo efficace. Questo sistema ha consentito di raccogliere, trasformare e organizzare dati chiave per l'azienda, con due finalità principali: il monitoraggio del parco auto circolante nei territori di interesse e la raccolta di statistiche dettagliate sulle vendite dei clienti dell'azienda madre. Il risultato è stato un miglioramento significativo nella capacità dell'azienda di individuare con precisione le opportunità di vendita e ottimizzare le strategie commerciali.

Il secondo obiettivo, la costruzione di modelli previsionali volti a identificare i potenziali articoli vendibili, è stato raggiunto attraverso l'implementazione dei due distinti "mix" di dati: il **Customer Mix** e il **Market Mix**. Ciascun approccio ha prodotto

risultati importanti. Il **Customer Mix** ha permesso di prevedere le necessità future dei clienti sulla base dei loro comportamenti d'acquisto passati, mentre il **Market Mix** ha fornito preziose informazioni sulle tendenze di mercato a livello nazionale, favorendo una visione strategica più ampia. .

Il terzo obiettivo, riguardante lo sviluppo di modelli predittivi per stimare le quantità di acquisto future, è stato soddisfatto con l'applicazione di tecniche di machine learning ai dati di vendita mensili, coprendo il periodo da gennaio 2013 a giugno 2024. I modelli hanno dimostrato una notevole capacità predittiva, contribuendo all'ottimizzazione delle previsioni di vendita e supportando l'azienda nel rispondere prontamente ai cambiamenti della domanda.

Dei potenziali **sviluppi futuri** per questo elaborato riguardano soprattutto l'ampliamento e il miglioramento della parte sperimentale, che ha già prodotto risultati soddisfacenti in termini di previsione delle vendite, ma che potrebbe essere ulteriormente perfezionata. Un primo miglioramento potrebbe consistere nell'integrazione di nuovi set di dati, in particolare quelli relativi alle caratteristiche specifiche dei singoli prodotti. Questo ampliamento permetterebbe non solo di affinare le previsioni relative alle vendite per singoli clienti dell'azienda manifatturiera, ma anche di analizzare più in profondità l'andamento futuro del mercato in relazione alla produzione e alla commercializzazione di determinati prodotti.

Attraverso l'inclusione di dati come specifiche tecniche dei prodotti, cicli di vita, e preferenze dei consumatori, sarebbe possibile costruire modelli in grado di prevedere non solo le vendite, ma anche le tendenze di mercato emergenti. Ciò consentirebbe di anticipare i cambiamenti nelle preferenze dei consumatori e di identificare in anticipo i prodotti che potrebbero diventare popolari nel prossimo futuro, offrendo all'azienda una maggiore capacità di adattarsi tempestivamente alle richieste del mercato.

Un ulteriore sviluppo nella sperimentazione potrebbe consistere nell'adozione di tecniche di forecasting avanzate basate su modelli di deep learning, come le Long Short-Term Memory (LSTM). Questi modelli sono particolarmente efficaci nella gestione di dati temporali e sequenziali, permettendo di catturare pattern complessi e relazioni non lineari nei dati storici. L'implementazione di LSTM, grazie alla loro capacità di memorizzare informazioni a lungo termine, consentirebbe di ottenere previsioni più accurate non solo sulle vendite future, ma anche sulle tendenze emergenti del mercato. In questo contesto, l'inclusione di dati riguardanti le caratteristiche dei prodotti e il ciclo di vita, abbinata alla capacità degli LSTM di gestire serie temporali multi-variate, potrebbe fornire una visione più completa delle dinamiche di mercato, migliorando l'abilità dell'azienda manifatturiera di pianificare strategie a lungo termine e adattarsi ai cambiamenti nelle preferenze dei consumatori.

In conclusione, i potenziali sviluppi futuri dell'elaborato mirano a potenziare la parte sperimentale attraverso l'integrazione di nuovi set di dati relativi alle specifiche dei prodotti, ai cicli di vita e alle preferenze dei consumatori. Questi miglioramenti, combinati con l'adozione di tecniche di forecasting avanzate come i modelli LSTM, offriranno all'azienda manifatturiera la possibilità di affinare le previsioni di vendita e di anticipare le tendenze di mercato emergenti, migliorando così la capacità di adattarsi proattivamente alle dinamiche del mercato e alle esigenze dei clienti.

Bibliografia

[1] Amazon Web Services, Inc. What is aws?, 2024.

[2] C. Apanowicz. Data warehousing and business intelligence: Benchmark project for the platform selection. In D. Ślęzak, Th. Kim, Y. Zhang, J. Ma, and K. Chung, editors, *Database Theory and Application*, volume 64 of *Communications in Computer and Information Science*, pages 123–135. Springer, 2009.

[3] Jolanta Baran, Daria Tandos, and Iwona Żabińska. Comparative analysis of selected car parks. *Multidisciplinary Aspects of Production Engineering*, 4(1):365–375, 2021.

[4] Diego Duarte, Chris Walshaw, and Nadarajah Ramesh. A comparison of time-series predictions for healthcare emergency department indicators and the impact of covid-19. *Applied Sciences*, 11(8), 2021.

[5] Markus Ettl, Pavithra Harsha, Anna Papush, and Georgia Perakis. A data-driven approach to personalized bundle pricing and recommendation. *Manufacturing & Service Operations Management*, 22(5):461–480, 2019.

[6] Facebook. Prophet: Forecasting at scale, 2024.

[7] Florida State University. Guide to proposal planning and writing, 2022.

[8] Mrinal K. Ghose, R. Paul, and S.K. Banerjee. Assessment of the impacts of vehicular emissions on urban air quality and its management in indian context: the case of kolkata (calcutta). *Environmental Science & Policy*, 7(4):345–351, 2004.

[9] IBISWorld. Tire dealers in the us - market research report, 2024.

[10] IlSoftware.it. Dbeaver: cos'è e come funziona il client database universale, 2024.

[11] Tang Jun, Cui Kai, Feng Yu, and Tong Gang. The research application of etl tool in business intelligence project. *2009 International Forum on Information Technology and Applications*, 2:620–623, 2009.

[12] Shuja Khan and Muhammad Alghulaiakh. Arima model for accurate time series stocks forecasting. *Semantic Scholar*, 2021.

[13] V. Kumar and D. Shah. Customer purchase history and its effect on customer loyalty. *Journal of Marketing*, 68(4):15–35, 2004.

[14] Lorenzo Menculini, Loris Francesco Termite, Emanuele Bonamente, Alberto Garinei, Marcello Marconi, and Lorenzo Biondi. Comparing prophet and deep learning to arima in forecasting wholesale food prices. *arXiv preprint arXiv:2107.12770*, 2021.

[15] Qlik. What is qlikview? - qlik help, 2024.

[16] R. Sharma and M. Singh. Tire sales and marketing strategies in the 21st century. In *Proceedings of the 2008 International Conference on Business and Information (BAI 2008)*. CiteseerX, 2008.

[17] Umi Kalsom Yusof, Haziqah Shamsudin, Mohd Nor Akmal Khalid, and Abir Hussain. Financial time series forecasting using prophet. *SpringerLink*, 2021.