UNIVERSITÀ DEGLI STUDI DI CATANIA

DIPARTIMENTO DI INGEGNERIA ELETTRICA ELETTRONICA E INFORMATICA

Dottorato di Ricerca in Ingegneria dei Sistemi, Energetica, Informatica e delle Telecomunicazioni

XXXVIII Ciclo

Mariaelena Berlotti

Machine Learning for Urban Intelligence: Building Predictive Smart Cities

Ph.D Thesis

Coordinatore:
Chiar.mo Prof. P. Arena

Tutor:
Chiar.mo Prof. Ing. S. Cavalieri

Academic Year 2024/2025

Contents

1. INTRODUCTION 1

1.1 Motivations and Goals 2
1.2 Structure of The Thesis 4

2. OVERVIEW OF MACHINE LEARNING IN SMART CITIES 5

2.1 Role of Data and AI in Smart Cities 6
2.2 Categories of ML Tasks in Urban Applications 9
2.2.1 Supervised Learning 9
2.2.2 Unsupervised Learning 10
2.2.2.1 Clustering for Urban Pattern Discovery 11
2.2.2.2 Dimensionality Reduction and Feature Learning 12
2.2.2.3 Anomaly Detection in Infrastructure Systems 13
2.2.2.4 Limitations and Hybrid Approaches 14
2.2.3 Deep Learning 15
2.3 Conclusions 17

3. AN INTEGRATIVE MACHINE LEARNING METHODOLOGY FOR SMART CITY APPLICATIONS 19

3.1 Data Acquisition and Integration 20
3.2 Data Preprocessing and Quality Enhancement 22
3.3 Exploratory Data Analysis 27
3.4 Feature Engineering 30
3.5 Feature Selection 32
3.6 Model Configuration 33
3.6.1 Leak Detection in WDSs 34
3.6.2 Clustering of water demands profiles 35
3.6.3 Traffic flow forecasting 35
3.6.4 Air quality prediction and cross-city transfer learning 37
3.7 Training and testing strategies 37
3.8 Performance Evaluation 38
3.9 Conclusions 39

4. LEAK DETECTION IN WATER DISTRIBUTION SYSTEMS 41

4.1 State of the Art 41
4.2 Case Study 43
4.3 Results 45
4.3.1 Leak Detection 45
4.3.2 Leak Prediction 45
4.4 Conclusions 47

5. MODELLING AND CLUSTERING WATER DEMAND PATTERNS IN WATER DISTRIBUTION SYSTEMS 48

5.1 State of the Art 48
5.2 Case Study 50
5.3 Results 51
5.4 Conclusions 58

6. MACHINE LEARNING FOR URBAN TRAFFIC FLOW ESTIMATION AND FORECASTING 60

6.1 State of the Art 61
6.2 Case Study 62
6.3 Results 66
6.4 Conclusions 70

7. TOWARDS TRANSFERABLE AIR QUALITY PREDICTION: A DOMAIN-ADVERSARIAL FRAMEWORK FOR CROSS-CITY GENERALIZATION 71

7.1 State of the Art 72
7.2 Case Study 73
7.3 Results 74
7.4 Conclusions 81

8. CONCLUSIONS 83

9. BIBLIOGRAPHY 86

List of Abbreviations

IoT	……..	Internet of Thing
NFC	……..	Near Field Communication
AI	……..	Artificial Intelligence
ML	……..	Machine Learning
ZTL	……..	Limited Traffic Zone
WDS	……..	Water Distribution System
RNNs	……..	Recurrent Neural Networks
LSTM	……..	Long Short-Term Memory
$PM_{2.5}$	……..	Particulate matter with 2.5 micrometers diameter
NO_2	……..	Nitrogen Oxides
LR	……..	Linear Regression
SVR	……..	Support Vector Regression
SVM	……..	Support Vector Machine
DBSCAN	……..	Density-Based Spatial Clustering of Applications with Noise
PCA	……..	Principal Component Analysis
t-SNE	……..	t-distributed Stochastic Neighbor Embedding
UMAP	……..	Uniform Manifold Approximation and Projection
OC-SVM	……..	One-Class Support Vector Machine
HVAC	……..	Heating, Ventilation, and Air Conditioning
DL	……..	Deep Learning
ANNs	……..	Artificial Neural Networks

UAVs	……..	Unmanned Aerial Vehicles
CCTV	……..	Closed-Circuit Television
CNNs	……..	Convolutional Neural Networks
GRUs	……..	Gated Recurrent Units
ST-GCNs	……..	Spatio-Temporal Graph Convolutional Networks
TL	……..	Transfer Learning
SHAP	……..	Shapley Additive exPlanations
LIME	……..	Local Interpretable Model-agnostic Explanations
EDA	……..	Exploratory Data Analysis
RMSE	……..	Root Mean Squared Error
MAE	……..	Mean Absolute Error
WNTR	……..	Water Network Tool for Resilience
SCADA	……..	Supervisory Control and Data Acquisition
NLP	……..	Natural Language Processing
DMA	……..	District Metered Area
IQR	……..	Interquartile Range
KNN	……..	K-Nearest Neighbors
FCD	……..	Floating Car Data
MSTL	……..	Multiple Seasonal Trend decomposition with LOESS
LOESS	……..	Locally Estimated Scatterplot Smoothing
ESD	……..	Extreme Studentized Deviate
LOF	……..	Local Outlier Factor
DTW	……..	Dynamic Time Warping

SBD	……..	Shape-Based Distance
GNNs	……..	Graph Neural Networks
DANN	……..	Domain-Adversarial Neural Network
AIC	……..	Akaike Information Criterion
BIC	……..	Bayesian Information Criterion
MAPE	……..	Mean Absolute Percentage Error
ADTK	……..	Anomaly Detection Toolkit
IWA	……..	International Water Association
TSK-Means	……..	Time Series K-Means
SMAPE	……..	Symmetric Mean Absolute Percentage Error

List of Figures

1.1 Smart City key areas.	……	3
2.1 Data-driven smart city workflow.	……	7
3.0 ML Methodology.	……	19
3.1 Data Preprocessing Steps.	……	22
3.2 Example Normal vs Leakage Pressure Condition.	……	27
3.3 Cumulative vs effective water user demand.	……	28
3.4 Example heatmap correlation between traffic and pollutants.	……	29
3.5 Comparison scaled original time series vs scaled seasonal components.	……	31
3.6 Original Milan WDS.	……	32
3.7 Skeletonized Milan WDS.	……	33
4.1 Simulated water demand curves.	……	43
4.2 Example 1H prediction Leak Node N00971.	……	46
4.3 Example 3H prediction Leak Node N00971.	……	46
5.1 Map of the WDS.	……	50
5.2 Average weekly pattern for each cluster.	……	52

5.3 Example anomalous consumption pattern residential user.	……	53
5.4 TSK-Means: Final combined average weekly pattern for each cluster.	……	55
5.5 TSK-Means: Residential vs non-residential average weekly pattern for each cluster.	……	55
5.6 K-Shape: Final combined average weekly pattern for each cluster.	……	57
5.7 K-Shape: Residential vs non-residential average weekly pattern for each cluster.	……	57
6.1 Architecture Urban Flow Estimation/Forecasting.	……	63
6.2 Urban are of Catania – in black operational traffic counters.	……	64
6.3 Two-level ML forecasting strategy.	……	65
6.4 Example one-week pattern sensors divided in three clusters.	……	67
6.5 Example test on excluded time series for the two weeks.	……	69
7.1 Cross-City Generalization approach.	……	73
7.2 Radar plot with public transport indicators.	……	74
7.3 PCA with public transport indicators.	……	75
7.4 Dendrogram with public transport indicators.	……	75
7.5 Test results for Berlin PM_{10}.	……	76

7.6 Test results for Berlin $PM_{2.5}$.	.....	76
7.7 Test results for Berlin NO_2.	.....	77
7.8 Test results for Helsinki $PM_{2.5}$.	.....	77
7.9 Radar plot without public transport indicators.	.....	78
7.10 PCA without public transport indicators.	.....	78
7.11 Dendrogram without public transport indicators.	.....	79
7.12 Test results for Paris NO_2.	.....	79
7.13 Test results for Paris PM_{10}.	.....	80
7.14 Test results for Paris $PM_{2.5}$.	.....	80
7.15 Mapping Paris air quality sensors.	.....	80
7.16 Test results for Helsinki PM_{10}.	.....	81

List of Tables

4.I Pressure dataset.	.....	44
4.II Leak history dataset.	.....	44
5.III Example final dataset.	.....	51

Chapter 1

Introduction

In recent years, the rapid proliferation of sensor networks, IoT devices, and connected infrastructures has transformed urban environments into complex data-rich ecosystems. This exponential growth in data generation has laid the foundation for what is now commonly referred to as the *smart city* — an interconnected, intelligent environment where physical infrastructure, digital technologies, and citizens interact continuously to improve quality of life, sustainability, and operational efficiency [1][2].

The concept of smart cities emerged as a response to the challenges posed by urbanization, resource constraints, and the need for sustainable development. Initially driven by advances in communication networks and embedded systems, smart cities have evolved into cyber-physical systems in which real-world phenomena are sensed, processed and acted upon in real time [3]. These cities integrate technologies such as IoT, edge computing, and 5G to support applications ranging from traffic optimization and environmental monitoring to energy management and public safety [4]. In particular, IoT has emerged as a key research area within Information and Communication Technologies, enabling the interconnection of smart devices and sensors through networks that support seamless data exchange via Wi-Fi, NFC, Bluetooth, and other protocols [5]. This connectivity generates large volumes of real-time data, especially in urban domains like transportation, where sensors monitor traffic flow and infrastructure conditions. The proliferation of connected devices within smart cities has led to unprecedented data availability, presenting new opportunities to enhance intelligent decision-making and optimize urban systems [6].

As cities become increasingly intelligent, the capacity to anticipate and proactively address real-world challenges becomes essential. In this context, AI and ML play a transformative role [7]. Unlike traditional reactive systems, AI-based prediction models enable cities to shift toward a proactive management paradigm, by leveraging historical and real-time data to forecast system behaviors. Owing to their flexibility and scalability, AI-based models can be deployed across a wide range of urban domains, including predictive traffic management, pollution level prediction, fault detection in water network and efficient resource allocation in water networks [8].

The shift from static to predictive urban management is crucial for addressing the growing complexity of urban systems. AI models trained on heterogeneous urban data streams — including sensor data, weather reports, human mobility traces, and social media inputs — enable not only real-time reaction, but also proactive planning. For example, smart transportation systems can dynamically adjust traffic signals based on predicted congestion [9],

while environmental monitoring platforms can forecast air quality issues and notify vulnerable populations in advance [10].

Despite these opportunities, implementing predictive intelligence in smart cities poses several challenges. Data collected from diverse sources often suffer from heterogeneity, sparsity, or privacy concerns [11]. Moreover, ensuring interoperability across various systems, platforms, and vendors remains a critical barrier [12]. In the same way that modern industries have adopted common approaches to better organize and connect their systems, smart cities also need shared methods and standards to bring together data from different sources, so that this information can be effectively used [13].

The aim of this research is to explore how machine learning-based predictive models can be effectively deployed across different domains of smart cities — including transportation, pollution monitoring, water management, and anomaly detection — to address everyday challenges faced by citizens.

By focusing on the intersection of urban data, prediction algorithms, and system interoperability, this thesis seeks to contribute to the development of smart cities that are not only connected and data-driven, but also resilient, anticipatory, and human-centered.

1.1 Motivations and Goals

The goal of this thesis is to investigate how predictive ML models can be effectively developed, adapted, and deployed across various domains of smart cities to address critical challenges in smart city environments, supporting more efficient, anticipatory, and data-driven urban management. By applying ML in specific domains, this research demonstrates how intelligent models can generate actionable insights and enable proactive operations across various urban services.

This work is motivated by the increasing availability of real-time data from interconnected urban infrastructures, including transportation systems, environmental monitoring platforms, and water networks. Despite the abundance of data, cities often lack the analytical tools necessary to translate this information into timely and effective decisions. Predictive ML provides a powerful means of bridging this gap, allowing urban systems to forecast conditions, detect anomalies, and better allocate resources.

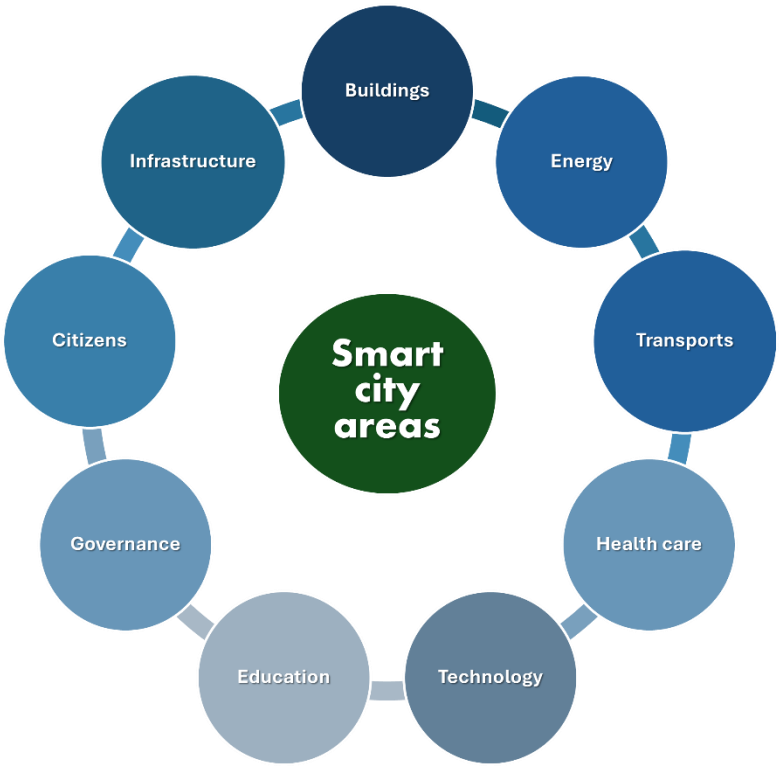


Figure 1.1 Smart City key areas.

Figure 1.1 illustrates the nine key domains that typically define the structure of a smart city: smart infrastructure, smart buildings, smart energy, smart transportation, smart health care, smart technology, smart education, smart governance and smart citizens [7].

While smart cities may prioritize different areas based on their specific needs and resources, domains such as **smart transportation** and **smart infrastructures** are often considered strategic, as they have a direct influence on key urban challenges like environmental sustainability, public health, and overall quality of life.

In alignment with this, the thesis focuses on four key use cases where predictive machine learning can offer concrete value:

1. **Traffic prediction,** supporting the smart transportation domain by forecasting congestion and informing urban mobility decisions.
2. **Air quality prediction,** contributing to both smart energy and smart health care through a unified transferable methodology for predicting pollution levels across similar but different urban environments and enabling targeted mitigation strategies— such as interventions for ZTLs or in high-risk zones.
3. **Anomaly detection in WDSs,** aligned with smart infrastructure, aimed at identifying faults and inefficiencies like water leakages.
4. **Clustering water consumption data,** serving both smart governance and smart citizens by classifying user types and detecting unusual demand profiles to support more adaptive water management policies.

These four use cases were selected not only for their social relevance, but also for the methodological diversity they represent. The research involves a combination of supervised,

unsupervised, and time-series modelling technologies, applied to heterogeneous data collected from real-world urban contexts. Each predictive model is developed independently and evaluated with respect to its effectiveness in enabling anticipatory decision making and operational improvements.

By grounding the study with multidimensional framework of smart city domains and applying ML to distinct yet interconnected challenges, this thesis aims to demonstrate the strategic role of predictive intelligence in shaping future urban environments. The outcomes contribute to a broader vision of smart cities that are not only connected and data-rich, but also sustainable, proactive and human-centered.

1.2 Structure of the Thesis

The structure of the thesis reflects a progressive exploration of ML methodologies applied to different facets of smart city management, moving from general foundations to specific case studies. Following the introductory chapter, which outlines the motivations and goals, Chapter 2 provides an overview of the role of AI in smart cities, presenting the main categories of ML tasks and discussing their relevance and limitations in urban applications.

Chapter 3 introduces the integrative methodological framework adopted throughout the work, covering data acquisition, preprocessing, exploratory analysis, feature engineering, and model configuration, before describing the training and evaluation strategies. This framework serves as the foundation upon which the subsequent application-driven chapters are built.

Chapters 4 through 7 are devoted to four case studies that exemplify the challenges and opportunities of applying ML in urban systems: leak detection in water distribution networks, clustering of water demand patterns, urban traffic flow forecasting, and transferable air quality prediction. Each case study follows a common structure—state of the art, methodological adaptation, empirical results, and discussion—allowing for a systematic comparison across domains.

Finally, the concluding chapter synthesizes the findings across the case studies, highlighting methodological contributions, reflecting critically on the limitations encountered, and outlining promising avenues for future research at the intersection of data science and sustainable urban development.

Chapter 2

Overview of Machine Learning in Smart Cities

The ongoing digital transformation of urban environments — characterized by the proliferation of sensors, the deployment of IoT infrastructures, and the accumulation of vast and diverse data streams— has made data-driven intelligence a foundational requirement for managing the complexity of modern cities. As established in Chapter 1, the transition from reactive, static systems to anticipatory, adaptive urban management is no longer a theoretical ideal but a practical necessity. In this context, ML, a subfield of AI, emerges as a powerful paradigm for enabling cities to move from passive data collection to proactive decision-making [14].

ML enables the development of models that can learn from historical and real-time data to identify trends, detect anomalies, and make informed predictions without being explicitly programmed for each task. This is particularly valuable in smart cities, where the scale, heterogeneity, and dynamic nature of data far exceed the capacity of traditional techniques [15]. Urban domains like the afore-mentioned transportation, environmental monitoring, and water usage increasingly depend on predictive analytics to support operational efficiency, citizen well-being, and sustainable development. Through ML it becomes possible not only to monitor system performance in real time, but also to forecast future states, anticipate risks, and guide adaptive responses across interconnected infrastructures [16].

Crucially, the versatility of ML allows it to be adapted to a wide range of data types and problem formulations, making it suitable for diverse urban applications. For example, time-series models can forecast traffic congestion or pollutant concentrations; clustering algorithms can reveal patterns in energy or water usage; and classification techniques can detect infrastructures anomalies or categorize urban behaviors. As outlined in the motivations of this thesis, the ability to apply predictive intelligence across underscores the cross-cutting value of ML in smart city ecosystems.

However, the successful application of ML in urban contexts also requires a systematic understanding of the different types of learning tasks it involves, the algorithms best suited for each use case, and the challenges that arise when scaling these models to city-wide infrastructure [17]. ML in smart cities must contend with issues such as noisy and incomplete data, real-time processing constraints, the need for model interpretability, and ethical considerations around data privacy and algorithmic fairness [18]. As such, developing effective predictive solutions requires not only technical proficiency but also contextual awareness of urban systems and their social, environmental, and infrastructural dynamics.

This chapter provides a comprehensive overview of the role of ML in smart cities, laying the conceptual and methodological groundwork for the use cases explored in subsequent chapters. Section 2.1 introduces the importance of predictive analytics in urban governance and the types of real-time data that support it. Section 2.2 categorizes the primary ML tasks—regression, classification, clustering, and forecasting—as they apply to different urban domains. Section 2.3 presents the most commonly used algorithms across these tasks, including both traditional and DL models. Finally, Section 2.4 discusses the technical and operational challenges of implementing ML in real-world smart city systems. Together, these sections offer a foundation for understanding how ML underpins the predictive intelligence required for next-generation cities.

2.1 Role of Data and AI in Smart Cities

The evolution of smart cities is fundamentally underpinned by the capacity to generate, process, and analyze vast and heterogeneous data streams collected from urban environments. As detailed in Chapter 1, the rapid integration of IoT devices, sensor networks, connected infrastructures, mobile technologies, and pervasive digital platforms has led to a new urban paradigm: cities as complex data-centric ecosystems. These systems constantly produce diverse data sources— *traffic flows, energy use, environmental monitoring, social media, economic transactions*, and *behavioral traces of citizens* — which form the backbone of informed urban planning and real-time management.

The importance of data in this context cannot be overstated. The concept of a "data-centric smart environment", as highlighted by [19], places data as the primary enabler for responsive, adaptive, and predictive city systems. With the global population expected to reach 70% in urban areas by 2050, cities face increasing demands on transportation, healthcare, water, energy, and waste management systems. These demands produce an unprecedented scale of information that must be collected, organized, and analyzed to support efficient decision-making. According to [20], the velocity, volume and variety of urban data have created new challenges that require advanced analytical frameworks far beyond traditional databases. Such frameworks integrate edge computing, cloud platforms, and distributed IoT infrastructures that allow millions of devices to operate collaboratively while streaming data in real time.

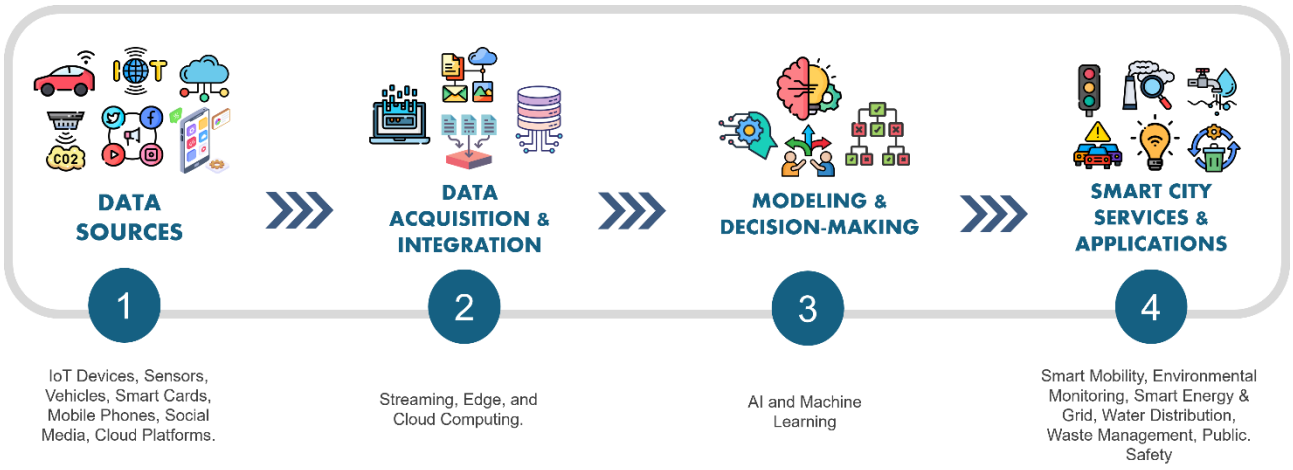


Figure 2.1 Data-driven smart city workflow.

Figure 2.1 depicts the foundational workflow of a data-driven smart city. The figure highlights four main stages: (1) *Data Sources*, where information is collected from IoT devices, vehicles, sensors, smart cards, mobile phones, social media, and cloud platforms; (2) *Data Acquisition and Integration*, in which raw data are gathered, cleaned, aggregated, and stored using streaming, edge, and cloud technologies; (3) *Modeling and Decision-Making*, where AI and ML models analyze the integrated datasets to detect patterns, forecast conditions, and support data-driven decisions; and finally, (4) *Smart City Services and Applications*, where the insights generated are applied to practical services such as smart mobility, environmental monitoring, energy management, water distribution, waste management, and public safety. This cyclical process enables a continuous transformation of data into actionable insights that directly influence urban operations.

As outlines so far, in this ecosystem, AI and specifically ML have become the central analytical engine of this data-rich ecosystem. ML models can automatously learn patterns, extract knowledge from noisy and high-dimensional data, and make accurate predictions that guide city planners and operators. This represents a fundamental shift from static, rule-based urban systems to adaptive, self-learning frameworks that can forecast future conditions and recommend optimal actions [17].

The applications of AI in smart cities are vast and deeply interconnected. In [21], the authors introduce the concept of "urban AI" to describe how intelligent algorithms are embedded in the everyday fabric of city life. AI-driven systems now influence mobility decisions, urban logistics, and environmental planning on a daily basis. Urban AI also underpins "digital twins" of cities [22], where real-time data feeds continuously update virtual replicas of the physical city to support forecasting, simulations, and scenario planning. These systems can predict the impact of changes to traffic flows, new construction, or public policy on air quality, energy consumption, and resource use, allowing more informed decisions at multiple scales.

IoT and ML are symbiotic in smart cities: IoT acts as the data collection infrastructure, while ML transforms this raw data into actionable insights. As described by [19], IoT devices installed across transport networks, utility grids, and environmental monitoring stations continuously gather data, which are then processed by ML models to guide automated decision-making. For

instance, in mobility management, ML models such as RNNs and LSTM networks predict traffic congestion and allow adaptive traffic signal control. In environmental domains, regression models, ensemble methods (e.g., *Random Forest, XGBoost*), and hybrid DL approaches predict concentrations of pollutants such as $PM_{2.5}$ and NO_2 hours or even days in advance, enabling early warning systems and mitigating health impacts. In water distribution systems, anomaly detection models—Isolation Forest, Autoencoders, and one-class SVMs—monitor the flow and pressure of water, automatically detecting leaks or bursts before significant losses occur [17].

Urban AI, however, does not operate in isolation. These technologies rely on the integration of cross-domain data sources to develop a holistic view of city dynamics— e.g., mobility data combined with air quality data can predict how traffic flows contribute to pollution peaks, while water usage data combined with climate models can forecast droughts conditions. As emphasized by [21] and [23], this convergence of data sources supports a systematic view of cities that recognizes the interdependence of different urban services.

Yet, despite these advances, the deployment of AI in smart cities is not without challenges. The first set of obstacles relates to the complexity of the data itself: heterogeneity, sparsity, incompleteness, and noise complicate the training of robust and generalizable models.

The second challenge involves the need for real-time responsiveness on a scale. As millions of devices stream data simultaneously, models must be optimized for efficiency, distributed computing, and low-latency inference to respond effectively to rapidly changing urban conditions [19][20].

Equally important are ethical and social challenges. In [18], the authors highlight that the increasing integration of AI into the operation of cities introduces significant concerns regarding privacy, transparency, interpretability, and fairness. Urban AI systems often process sensitive personal data, making privacy preserving methods—such as federated learning and encryption— essential. The interpretability of complex models, particularly deep neural networks, remains a crucial requirement in governance settings where trust, accountability, and explainability are non-negotiable [17]. In addition, there are issues of bias and fairness: AI systems trained on unbalanced data may perpetuate inequalities, exacerbating rather than alleviating urban disparities.

Furthermore, the socio-technical complexity of AI deployments necessitates significant investment in computing infrastructures and skilled human capital. Without such investments, there is a risk that technologically advanced cities will accelerate ahead while others are left behind, increasing inequality between regions and countries [21].

In summary, data and AI form the backbone of the smart city vision. IoT-generated data streams, when coupled with advanced AI and ML algorithms, transform static urban infrastructures into dynamic, predictive, and adaptive ecosystems. The ability to anticipate events and to respond proactively makes AI a fundamental driver of urban resilience and sustainability.
Building on the concepts, this section has established the central role of data-driven intelligence as the foundation of smart city development.

In the following section, we move to a more structured view of the main ML tasks and approaches that enable these predictive capabilities. This transition from concepts to methods underscores how AI translates urban data into actionable insights that guide real-time decision-making in modern cities.

2.2 Categories of ML Tasks in Urban Applications

ML models are widely applied across various urban domains in smart cities, offering scalable and adaptive tools for analyzing large-scale, real-time, and heterogeneous data. In smart city contexts, ML tasks typically fall into four core functional categories: *classification, regression, clustering* and *forecasting.* These tasks are implemented using three main types of learning paradigms: **supervised learning, unsupervised learning** and **deep learning,** each offering unique advantages for managing specific urban challenges.

2.2.1 Supervised Learning

Supervised learning is the most commonly employed ML approach in smart city systems because it formalizes the concept of learning a mapping between input data and corresponding outputs through a labeled dataset. Formally, given a dataset:

$$D = \{(x_i; y_i)\}^{N}_{i=1}$$

where $x_i \in \mathbb{R}^d$ are the feature vectors (e.g. *traffic volume, meteorological variables, sensor readings)* and y_i are the corresponding labels, a supervised learning algorithm seeks to approximate a function $f_\Theta : \mathbb{R}^d \rightarrow y$ parametrized by Θ that minimizes a predefined loss function over the training set. Depending on the nature of y, this task is categorized as a **regression,** if y is continuous, or **classification** problem, if y is discrete.

Regression tasks focus on estimating a continuous-valued output variable. The objective is to learn a mapping such that:

$$\hat{y} = f_\Theta(x)$$

with $y \subseteq \mathbb{R}$, where the parameters Θ are chosen to minimize a predefined loss function.

In the context of smart cities, regression is extensively used to predict continuous quantities such as pollutant concentrations, traffic flows, or energy consumption. For instance, ensemble-based models such as gradient boosting and random forests have been deployed to predict hourly $PM_{2.5}$ and NO_2 concentrations using traffic data, meteorological factors, and historical air quality observations [19]. Similarly, short- and medium-term electricity demand prediction at the neighborhood level has been addressed using regression techniques that incorporate socio-economic indicators and weather conditions [24]. In water resource management, multiple LR and SVR models have been used to predict daily water demand patterns based on seasonal trends and smart meter readings [25].

Classification tasks, instead, involve learning a decision boundary that partitions the input space into discrete classes. In this case, the label space is categorical, $y = \{1, \ldots, K\}$, and the function $f_\Theta(x)$ outputs either a discrete class prediction:

$$\hat{y} = \arg\max_k P(y = k \mid x; \theta)$$

or a vector of posterior probabilities $P(y = k \mid x; \theta)$.

Classification in smart cities underpins applications where discrete decisions are required. Examples include the detection and categorization of traffic incidents (e.g. *normal flow vs. congestion vs. accident)* using spatio-temporal traffic patterns from inductive loop sensors and video analytics [17]. Classification models are also used for fault detection in water networks, where decision trees and SVM classifiers distinguish between leaks, bursts, and sensor anomalies by analyzing time-series patterns [23].

Furthermore, classification is applied to categorize buildings or users into energy consumption profiles, allowing demand-side management strategies and incentives to be tailored accordingly.

While supervised learning techniques are attractive due to their strong predictive capabilities and, in some models (e.g., linear models or decision trees), their interpretability, their applicability in real-world smart city environments is often limited by the availability of labeled data. Obtaining labeled data at scale in real urban settings is particularly challenging for several reasons. First, the labeling process often requires human experts (e.g., annotating traffic incidents from video streams or identifying the type of failure in a water network), which is costly and time-consuming [18]. Second, in many domains—such as rare fault detection, extreme weather events, or abnormal traffic conditions— the relevant labeled examples are sparse by nature. These rare but critical events (low frequency, high impact) create highly imbalanced datasets, making supervised learning models prone to bias toward the majority class [25]. Finally, the dynamic nature of urban environments introduces the problem of concept drift, where the relationship between inputs and outputs changes over time, making historical labels less useful as systems evolve [18]. For these reasons, although supervised learning remains a central paradigm, its success depends on continuous data collection pipelines and strategies to handle scarcity and imbalance, such as data augmentation, semi-supervised learning, federated transfer learning, and cross domain adaptation methods [26]-[28].

2.2.2 Unsupervised Learning

In contrast to supervised learning, unsupervised learning operates in the absence of labeled datasets, where data consists solely of feature observations collected from urban sensors, IoT devices, or social data streams. The objective is to discover hidden patterns, structures, or statistical regularities in the data without predefined output variables. This paradigm is essential in urban analytics for several reasons:

1. **Scarcity of labeled data —** as discussed in Section 2.2.1, many urban systems do not have large-scale annotated datasets due to the cost, time, and expertise required for labeling [25][27].
2. **Exploratory knowledge discovery —** urban planners often need to understand underlying behavioral patterns before designing interventions.

3. **Adaptability to evolving systems —** since cities are dynamic, unsupervised models can be retrained on new, unlabeled data to capture changes without requiring constant manual annotation.

2.2.2.1 Clustering for Urban Pattern Discovery

Clustering is one of the most widely adopted unsupervised learning techniques in smart city analytics due to its ability to reveal latent structures and groupings in large-scale, heterogeneous datasets without requiring prior labeling. In essence, clustering seeks to partition a dataset into a set of distinct, non-overlapping groups—referred to as *clusters*—such that observations within the same group share high intra-cluster similarity, while those in different groups exhibit significant inter-cluster dissimilarity. The similarity or dissimilarity between data points is typically measured using distance metrics (e.g., *Euclidean, Manhattan,* or *cosine distance*) or density-based criteria, depending on the specific algorithm employed [29]. In the smart city context, clustering is particularly valuable for behavioral segmentation, spatiotemporal pattern recognition, and infrastructure usage profiling, enabling more targeted policy interventions and resource allocation strategies.

K-means clustering is one of the most widely applied algorithms due to its computational efficiency and simplicity. In energy analytics, it has been used extensively for electricity consumption profiling, where households or commercial buildings are categorized into distinct usage patterns based on their load curves [30]. Such segmentation allows utility providers to design demand-response programs that tailor energy conservation incentives to different user groups, thereby improving both grid stability and energy efficiency. In smart water grids, [31] demonstrated the use of K-means for grouping consumers according to their daily and seasonal water consumption profiles. This enabled water utilities to optimize distribution schedules, detect abnormal demand surges, and design conservation campaigns targeted to high-usage clusters. Similarly, in waste management, K-means has been applied to group neighborhoods according to waste generation patterns, supporting dynamic waste collection routing and resource planning [32].

While K-means assumes spherical clusters and equal cluster sizes, many urban datasets do not conform to these assumptions. DBSCAN overcomes this limitation by identifying clusters based on density connectivity, allowing the detection of arbitrarily shaped clusters and isolating noise points as outliers [33]. In urban mobility analysis, DBSCAN has been successfully used to detect high-density commuting routes from GPS trajectory data while identifying outlier trips that may correspond to special events, traffic anomalies, or emergency evacuations [34]. It has also been applied in bicycle-sharing systems to segment stations according to usage patterns, helping operators anticipate peak demand periods and redistribute bikes efficiently [35].

Another important approach is hierarchical clustering, which builds a tree-like structure (*dendrogram*) representing nested groupings of data at different levels of granularity. This is particularly suitable for urban zone segmentation, where clustering can be conducted iteratively to group city districts according to multi-dimensional attributes such as traffic density, air quality indices, noise pollution, socio-economic variables, and land-use data [17]. This multi-level segmentation enables city planners to analyze urban phenomena at varying

spatial resolutions, from fine-grained neighborhood clusters to broader metropolitan patterns, thereby supporting decisions on zoning, infrastructure investment, and environmental monitoring.

Beyond these classical techniques, hybrid clustering frameworks combining K-means, DBSCAN, or hierarchical methods with dimensionality reduction techniques (e.g., *PCA, t-SNE, UMAP*) are increasingly used in smart city research to manage high-dimensional, noisy datasets from IoT infrastructures [28]. For instance, in transportation analytics, GPS trajectory data can be first reduced in dimensionality to capture essential mobility features, and then clustered to identify commuting communities or traffic flow bottlenecks. Such integration improves computational efficiency, enhances cluster separation, and facilitates visual interpretation of results.

2.2.2.2 Dimensionality Reduction and Feature Learning

Urban IoT ecosystems generate high-dimensional and heterogeneous datasets due to the variety of deployed sensing technologies, the diversity of measured variables, and the granularity of temporal and spatial resolutions. Examples include multi-sensor environmental monitoring stations recording pollutant concentrations, temperature, humidity, and meteorological parameters; GPS mobility traces enriched with road network features and speed profiles; or energy management systems tracking multi-phase voltage, current, and power quality indicators. Processing such high-dimensional data directly often leads to computational inefficiency, overfitting in predictive models, and difficulties in visualization and interpretation [37].

Dimensionality reduction addresses these challenges by transforming the original dataset into a lower-dimensional representation that preserves its most informative characteristics, thereby simplifying model training, reducing noise, and enabling more interpretable analyses. These techniques can be broadly categorized into linear and non-linear approaches.

Principal Component Analysis (PCA) is the most widely used linear dimensionality reduction technique. By projecting data onto a set of orthogonal components that maximize variance, PCA effectively identifies the most significant patterns in the data while removing redundant and collinear variables. In air quality monitoring, PCA has been applied to identify dominant pollution sources and to reduce redundancy among measurements of multiple pollutants (e.g., $PM_{2.5}$, PM_{10}, NO_2, O_3), improving the computational efficiency of subsequent forecasting models [28]. Similarly, in traffic analytics, PCA has been used to compress high-dimensional GPS trajectory features into compact representations for clustering and route similarity analysis, reducing computational costs while maintaining discriminatory power [32].

Non-linear techniques such as t-SNE and UMAP have gained significant traction in smart city applications, particularly for multi-modal data visualization and exploratory analysis. In concerns:

- **t-SNE** [37] is designed to preserve local neighborhood structures in the data when mapping high-dimensional points to a low-dimensional space (typically 2D or 3D). In urban contexts, t-SNE has been used to visualize clusters of citizen mobility behaviors, revealing how commuting patterns differ across neighborhoods and times of day [38].

- **UMAP** [39] extends these capabilities by relying on manifold learning and topological data analysis principles to achieve both speed and scalability, making it more suitable for large-scale IoT datasets. UMAP has been employed in urban data fusion pipelines to integrate mobility, pollution, and weather data streams into unified visual representations, enabling city planners to identify cross-domain correlations without direct supervision [40].

In addition to visualization, dimensionality reduction also serves as a feature learning stage, providing compact yet informative representations that can be fed into supervised or unsupervised predictive models. For example, environmental monitoring systems have used PCA-derived features as inputs to ML regression models for pollutant forecasting, while mobility analytics pipelines often use UMAP embeddings for clustering trajectories in large-scale transportation networks [17].

Hybrid approaches combining autoencoders (DL-based non-linear dimensionality reduction) with classical techniques like PCA are increasingly common in smart infrastructure anomaly detection. These models first learn low-dimensional latent spaces capturing normal system behaviors, and then detect anomalies as deviations in this compressed representation [41].

Ultimately, dimensionality reduction in smart cities is not merely a preprocessing step—it is a core enabler of computational feasibility, interpretability, and cross-domain integration in the analysis of complex urban datasets. As urban data sources continue to expand in volume, variety, and velocity, the role of scalable, interpretable, and domain-adapted dimensionality reduction methods will become increasingly critical for delivering actionable intelligence.

2.2.2.3 Anomaly Detection in Infrastructure Systems

Anomaly detection in urban infrastructure systems is a critical component of predictive maintenance and operational resilience in smart cities.

Anomalies—also referred to as *outliers* or *novelties*—are data points or patterns that deviate significantly from expected system behavior. In an urban context, such anomalies frequently correspond to critical failures or unusual events, including water leaks, power grid faults, structural damages, environmental hazards, or traffic incidents. Detecting these deviations early enables proactive interventions that can mitigate economic losses, minimize service disruptions, and enhance public safety [42].

A major challenge in urban anomaly detection is the rarity of failure events in well-maintained systems. This scarcity of labeled fault data makes supervised learning approaches less feasible, as they require large, balanced training datasets to generalize effectively. Moreover, anomalies often exhibit non-stationary characteristics, meaning their signatures can change over time due to seasonal trends, infrastructure upgrades, or evolving usage patterns. Consequently, unsupervised and semi-supervised approaches are often preferred, as they do not rely on large amounts of labeled data and instead model the normal operational state to identify deviations [43].

Autoencoders, a type of neural network designed for data compression and reconstruction, have become a prominent tool for anomaly detection in smart infrastructure. By learning a low-

dimensional latent representation of normal operating data, autoencoders can reconstruct expected patterns with high accuracy. When new inputs produce large reconstruction errors, these deviations are flagged as potential anomalies [41]. For example, in electric power systems, autoencoders have been used to detect unusual voltage waveform patterns or harmonic distortions, enabling maintenance teams to address faults before they escalate into large-scale blackouts [23]. In traffic monitoring, autoencoders have been applied to identify sensor malfunctions by detecting unrealistic speed or density readings in traffic flow datasets [44].

Another widely used approach is the OC-SVM, which learns a decision boundary that encapsulates the distribution of normal operational data in feature space. Any points falling outside this learned boundary are considered anomalies. This method is especially valuable in applications such as industrial IoT monitoring, where only normal operation data are typically available during system commissioning. OC-SVMs have been applied to urban HVAC systems to detect deviations in temperature and energy consumption profiles indicative of faults or inefficiencies [45].

Isolation Forests [46] take a different approach by exploiting the principle that anomalies are easier to isolate in a decision tree structure due to their sparse and distinct feature values. This makes the method particularly effective for high-dimensional datasets common in smart cities. In water distribution networks, [31] successfully applied Isolation Forests to detect hidden leakages by modeling expected flow–pressure relationships, identifying irregularities that would not be captured by threshold-based monitoring. In electricity usage monitoring, Isolation Forests have been deployed to detect unusual consumption spikes that may indicate equipment malfunctions, energy theft, or sensor faults [17].

Recent research trends in anomaly detection for smart cities have focused on hybrid frameworks that combine DL feature extraction (e.g., via convolutional autoencoders or recurrent neural networks) with classical anomaly detection algorithms such as OC-SVM or Isolation Forests. This integration leverages the representational power of deep networks to capture complex temporal and spatial dependencies in urban infrastructure data, while maintaining the robustness and interpretability of established outlier detection methods [47].

As smart cities expand their sensing capabilities and data volumes grow, scalable anomaly detection systems capable of operating in real time and handling concept drift will become essential for resilient and adaptive urban infrastructure management.

2.2.2.4 Limitations and Hybrid Approaches

Although unsupervised learning methods are indispensable in smart city analytics—particularly in exploratory data analysis, anomaly detection, and dimensionality reduction—they are not without significant challenges.

Interpretability remains one of the foremost limitations. Clusters, latent dimensions, or feature embeddings generated by unsupervised algorithms often lack direct semantic meaning, making it difficult for city planners or engineers to translate them into actionable interventions without extensive domain expertise [23]. For instance, a clustering algorithm might separate electricity usage patterns into distinct groups, but understanding whether these correspond to residential,

industrial, or commercial consumers often requires supplementary labeled data or expert annotation. This limitation is particularly problematic in multi-modal urban datasets, where patterns may be driven by hidden correlations between domains such as mobility, energy, and weather.

Hyperparameter sensitivity is another critical drawback. Many unsupervised algorithms depend on carefully tuned parameters—such as the number of clusters (k) in K-means or the density threshold ε in DBSCAN—to produce meaningful outputs. Small variations in these parameters can lead to drastically different clustering results, thereby affecting reproducibility and consistency in urban policy decisions [48]. This sensitivity is further amplified in heterogeneous urban data, where feature scaling, noise, and missing values can bias the learning process.

Scalability is an equally pressing concern. With the exponential growth of urban data from IoT devices, satellite imagery, and real-time sensor feeds, traditional unsupervised algorithms may become computationally prohibitive without distributed computing frameworks, incremental learning strategies, or streaming implementations. For example, processing petabyte-scale mobility data or fine-grained environmental monitoring data in real time requires specialized architectures that can handle high throughput and low latency [28].

To address these limitations, hybrid approaches have gained increasing traction in recent literature. These pipelines often employ unsupervised methods—such as autoencoders, clustering algorithms, or manifold learning—for representation learning or data segmentation, followed by supervised fine-tuning on a small set of labeled samples. This strategy allows models to benefit from large volumes of unlabeled urban data while leveraging limited ground truth to enhance predictive accuracy and semantic clarity.

A related development is the rise of self-supervised learning in smart city contexts. In this paradigm, models generate pseudo-labels or pretext tasks directly from raw data streams, such as predicting missing sensor values, forecasting the next timestep in mobility trajectories, or reconstructing masked regions of satellite imagery. These pseudo-labels then serve as supervisory signals for downstream predictive tasks, effectively bridging the gap between unsupervised and supervised paradigms [40]. Self-supervised approaches have been successfully applied in traffic forecasting, pollution modeling, and urban scene understanding, where labeled datasets are scarce or expensive to obtain.

Overall, the integration of hybrid and self-supervised learning strategies represents a promising path forward for urban AI systems, combining the pattern discovery strengths of unsupervised methods with the precision and interpretability of supervised learning, while addressing the data scarcity, scalability, and interpretability challenges that currently constrain purely unsupervised approaches.

2.2.3 Deep Learning

DL, a specialized subfield of ML grounded in multi-layered ANNs, has emerged as a transformative analytical paradigm in the context of smart cities. Its strength lies in its capacity to automatically extract multi-level hierarchical feature representations from large-scale,

heterogeneous, and often unstructured datasets, enabling the modeling of complex non-linear relationships beyond the reach of traditional statistical and shallow learning approaches [49][50]. Unlike conventional algorithms that rely heavily on domain-specific feature engineering, DL models learn task-relevant abstractions directly from raw input data—whether these are high-dimensional numerical sensor readings, georeferenced spatial grids, temporal sequences, or multimodal urban data streams.

The versatility of DL is particularly suited to smart city ecosystems, where data are inherently multi-modal and spatio-temporally correlated. Sources include real-time traffic sensors and GPS trajectories [51], environmental monitoring stations measuring pollutants and meteorological variables [52], high-resolution geospatial imagery from satellites and UAVs [53], CCTV and surveillance video streams [23], and unstructured citizen-generated content from social media platforms [20]. The spatio-temporal nature of these datasets—where patterns often emerge from complex interactions between geographic proximity and temporal dynamics—makes the ability of DL to model both local and long-range dependencies especially advantageous. For example, in urban mobility analysis, traffic congestion at a single intersection may be influenced by conditions several kilometers away and hours earlier; similarly, pollutant dispersion patterns depend not only on local emissions but also on meteorological conditions over extended time horizons.

Architectures such as CNNs excel at capturing spatial hierarchies and have been applied to represent urban sensor networks as structured grids or graphs for pollution mapping [54], land-use classification [53] and infrastructure condition monitoring [17]. Meanwhile, recurrent RNNs, particularly long short-term memory networks LSTMs and GRUs, are capable of modeling temporal dependencies in time-series data such as traffic flows [55], electricity demand [56] and water consumption [57]. In more complex settings, STL-GCNs combine CNN-based spatial learning with RNN- or attention-based temporal modeling to capture interactions over irregular urban networks, such as road maps or utility grids [58].

One of the defining strengths of DL in the smart city domain is its ability to handle heterogeneous data fusion—a core requirement for integrated urban analytics. Multimodal DL frameworks enable the joint learning of representations from diverse sources, such as combining CCTV footage with IoT sensor readings for anomaly detection in public safety systems, or integrating satellite imagery with social media reports for disaster response coordination [59]. The use of attention mechanisms and transformer architectures further enhances this capability by allowing models to focus selectively on the most relevant features across modalities and time frames [60].

Additionally, TL has become increasingly important in DL-based smart city applications, especially where labeled data are scarce—a common challenge in urban AI deployment [61]. TL allows models pre-trained in data-rich contexts (e.g., traffic forecasting in megacities) to be fine-tuned for smaller cities with limited data availability, drastically reducing training time and computational costs while maintaining predictive performance. Recent studies have demonstrated that air quality prediction models trained in one region can be successfully

adapted to another with minimal retraining [62], and similar approaches have been used to transfer congestion prediction models across cities with different road topologies [63].

However, the deployment of DL in operational smart city systems is not without challenges. The computational demands of deep models—particularly for real-time, city-scale inference—often necessitate distributed computing frameworks, GPU acceleration, or deployment on cloud platforms, which may introduce latency and data privacy concerns [18]. Furthermore, the opacity of deep models, often described as the "black-box problem," poses significant barriers to adoption in governance contexts, where explainability, accountability, and fairness are essential. Techniques from the explainable AI domain, including SHAP [64] and LIME [65], as well as inherently interpretable DL architectures, are increasingly being explored to address these concerns. Efforts are also underway to optimize DL for edge computing environments—such as traffic cameras and environmental sensors—through model compression, quantization, and lightweight architectures like MobileNet and TinyML, enabling on-device intelligence without reliance on constant cloud connectivity.

In summary, DL has established itself as a cornerstone of predictive intelligence in smart cities, uniquely capable of capturing complex spatio-temporal relationships, integrating heterogeneous data sources, and adapting knowledge across domains. Its ability to deliver high-accuracy predictions and actionable insights positions it as a critical enabler of anticipatory, adaptive, and resilient urban governance.

2.3 Conclusions

In this chapter, the authors have reviewed the major categories of ML techniques—*supervised learning, unsupervised learning*, and *deep learning*—and discussed their relevance in the context of smart cities. We explored how supervised methods such as regression and classification excel in scenarios where labeled datasets are available and predictive accuracy is paramount, how unsupervised methods such as clustering and anomaly detection provide valuable insights in data-scarce or exploratory contexts, and how DL architectures enable the extraction of complex spatio-temporal patterns from high-dimensional, multimodal data streams. We also examined cross-cutting challenges including data heterogeneity, sparsity, scalability, interpretability, and the rarity of labeled data in real-world urban environments, highlighting emerging approaches such as transfer learning, self-supervised learning, and explainable AI as potential solutions.

The remainder of this thesis operationalizes the theoretical and methodological concepts outlined in this chapter, applying them to distinct yet interconnected domains within the smart city ecosystem. In the following chapters of the thesis, the author will demonstrate that the choice of ML paradigm, whether supervised, unsupervised, or DL, was not arbitrary but carefully tailored to the specific characteristics of the available data and the operational needs of the target urban domain.

In the field of urban water distribution, [66] presents an anomaly detection framework that integrates statistical pre-processing with ML-based predictive modeling to detect and anticipate leakages from smart meter pressure and flow data. Complementary to this, [67] and [68] employ clustering algorithms such as K-means and hierarchical clustering to segment

water consumers by daily and seasonal demand patterns, thereby enabling adaptive resource allocation and informed infrastructure planning.

For environmental monitoring, [69] introduces a transfer learning strategy based on domain-adversarial neural networks, allowing for the prediction of multiple pollutants in cities with scarce air quality data by adapting models trained in data-rich urban contexts. This approach reduces computational costs while maintaining predictive accuracy, offering a scalable solution for cross-city environmental forecasting.

In the mobility domain, a set of works [70)-[73] explore supervised regression, DL, and data fusion techniques to combine traffic sensor data with third-party mobility datasets (e.g., TomTom) for road condition forecasting in Catania, under both sensor-rich and sensor-less scenarios. These contributions illustrate how different ML methods can be adapted to data availability constraints, achieving accurate forecasts in real operational contexts.

Each of the subsequent chapters will then focus on one of these smart city domains, presenting the corresponding case study in detail and discussing the findings in relation to the existing literature. In particular, the next chapter will describe the common data analysis pipeline adopted across all these studies, including data acquisition, preprocessing, exploratory analysis, feature engineering, model selection, training, validation, and evaluation. These methodological steps form the backbone of the empirical work presented in the following domain-specific chapters, ensuring that each application is understood not as an isolated experiment but as part of a coherent, replicable, and well-justified analytical framework.

Chapter 3

An Integrative Machine Learning Methodology for Smart City Applications

The diversity of challenges faced by modern cities—ranging from traffic congestion and air pollution to water distribution management—requires a unified yet adaptable approach to data analysis and predictive modeling. As discussed in Chapter 2, ML techniques form the backbone of intelligent urban systems, enabling the transformation of raw, heterogeneous data into actionable insights. However, the success of these techniques does not rely solely on the choice of algorithms; it depends equally on a rigorous, well-structured methodological pipeline that ensures data quality, model robustness, and domain adaptability.

This chapter presents the common methodological framework that underpins all the studies conducted by the author in the context of this thesis. Regardless of whether the target application is anomaly detection in water networks, clustering of water demand profiles, transfer learning for cross-city air quality prediction, or supervised forecasting of traffic flows, each investigation follows a consistent analytical workflow. This ensures methodological coherence while allowing for domain-specific adjustments in response to the nature of the data and the problem at hand.

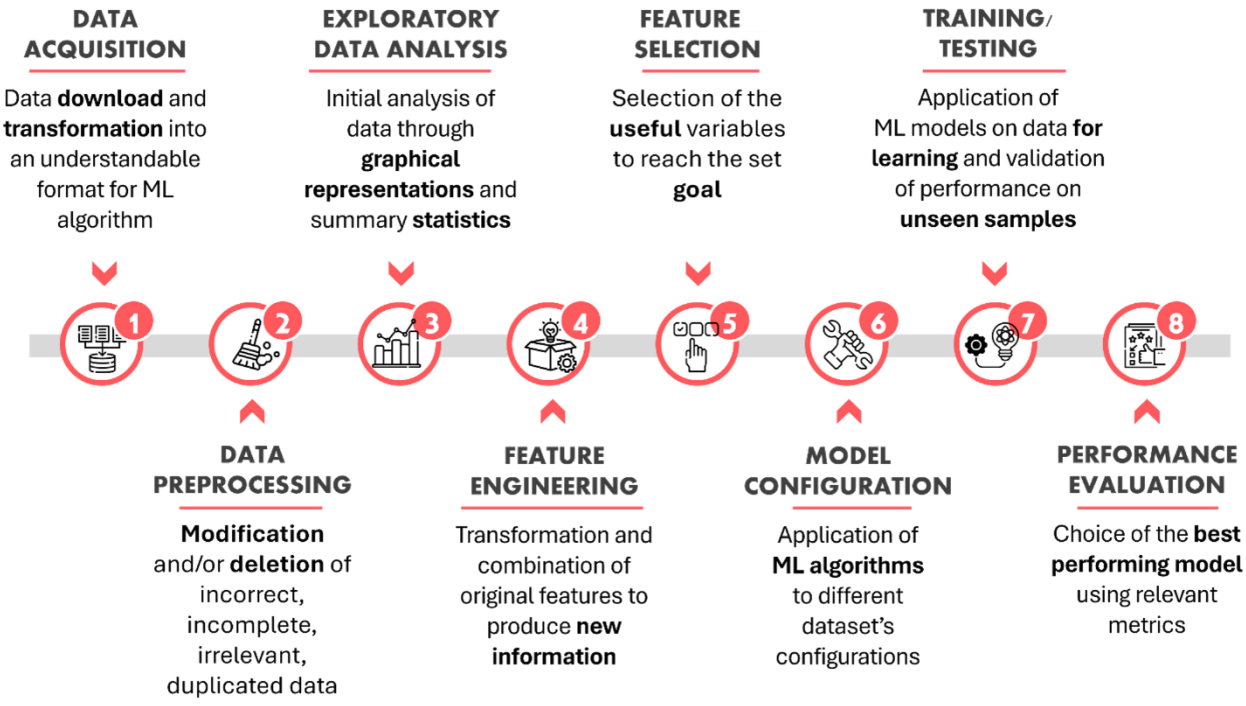


Figure 3.0 ML Methodology.

The logical progression followed by the proposed methodological framework, illustrated in Figure 3.0, outlines a structured sequence of data analysis steps. The process begins with Data Acquisition, which involves collecting and integrating heterogeneous datasets from IoT devices, sensor networks, mobility platforms, and open data repositories.

This step is, then, followed by Data Preprocessing, where raw data are cleaned to address missing values, noise, inconsistencies, and outliers, ensuring that downstream analyses are built upon reliable inputs.

Once the data are preprocessed, EDA is conducted to reveal underlying distributions, detect anomalies, and identify key patterns or correlations relevant to the urban domain under study. These insights inform the subsequent Feature Engineering phase, where new variables are derived, and Feature Selection, where the most relevant predictors are retained to reduce dimensionality, improve interpretability, and enhance computational efficiency.

In the Model Configuration stage, appropriate ML algorithms are selected and tailored to the datasets characteristics, often experimenting with different hyperparameter configurations to optimize performance. The data are then split into Training and Testing subsets, enabling the model to learn from historical patterns before being evaluated on unseen samples to ensure generalizability.

Finally, Performance Evaluation is carried out using metrics such as RMSE, MAE, accuracy, precision, recall, and F1-score, depending on the nature of the task (regression, classification, clustering, etc.). This step not only validates the predictive capabilities of the model but also guides iterative improvements. Together, these stages form an integrated workflow that can be consistently applied across diverse smart city applications, ensuring methodological coherence throughout the thesis.

3.1 Data Acquisition and Integration

In any data-driven smart city application, data acquisition and integration constitute the cornerstone of the entire analytical and predictive pipeline. The accuracy, robustness, and generalizability of ML models are fundamentally constrained by the quality, coverage, and interoperability of the input data. Given the inherently multi-domain and multi-modal nature of urban systems, the ability to collect, harmonize, and align heterogeneous data sources is not merely a preparatory step but a critical methodological component that shapes all subsequent stages of the framework.

Smart cities produce data at an unprecedented velocity, variety, and volume—the so-called *3Vs of big data* [74] —originating from physical infrastructure, environmental monitoring systems, mobility services, social platforms, and administrative records. This thesis adopts a multi-source acquisition strategy, encompassing the following categories:

- **IoT-Based Urban Sensing and Smart Metering:** fixed and mobile sensing infrastructures generate high-frequency, real-time measurements across multiple domains. In several studies, real-world IoT sensors were the primary source of data, providing high-frequency, domain-specific measurements. For urban mobility forecasting, as presented in [70]-[73],

data were collected from 21 MobilTraf300 microwave traffic counters deployed in the city of Catania, Italy. These devices operate at 24 GHz, detect vehicles using radar technology, and transmit aggregated measurements every 5 minutes. The dataset includes counts, travel direction, lane usage, and timestamps. For analysis, 12 operational counters covering different road types (single-lane, two-lane same direction, and two-lane opposite directions) were selected, providing a complete year (2022) of high-resolution traffic flow time-series data.

For consumer profiling in water distribution systems (WDSs), the studies [67]-[68] used hourly consumption readings from smart meters installed in a central Italian WDS. Data was collected between September 2023 and May 2024 across six District Metered Areas (DMAs), with user counts ranging from 31 to 403. Each smart meter transmitted cumulative water usage (liters/second), enabling the segmentation of consumption behaviors through clustering methods.

- **Simulation-Based Synthetic Data Generation:** when real-world data availability was limited, simulation tools were employed to create realistic synthetic datasets. This was the case in [66], where labeled leakage events were needed to train anomaly detection models. Due to the scarcity of recorded real losses—caused by incomplete maintenance logs and absence of digital asset monitoring systems—the WNTR [75] was used to simulate hydraulic operations of the Milan WDS. The system was simulated using daily and weekly demand variation coefficients extracted from SCADA data, aggregated to 30-minute intervals and perturbed with noise to mimic realistic demand fluctuations. Leakages of varying magnitudes were randomly introduced at network nodes, producing pressure time-series suitable for both supervised and unsupervised fault detection experiments.
- **Third-Party, Open Data, and Multi-City Integration:** in addition to IoT and in-situ sensor networks, smart city research increasingly benefits from third-party mobility analytics platforms and crowd-sourced geographic data. Services such as TomTom, Google Maps, and HERE Technologies provide aggregated traffic speed, flow, and congestion metrics at high temporal resolution, enabling large-scale mobility pattern analysis. Crowd-sourced initiatives like OpenStreetMap contribute detailed and continuously updated geospatial infrastructure maps, while social media streams from platforms such as Twitter and Facebook—although unstructured—can be mined through NLP techniques to detect real-time events, disruptions, and citizen-reported incidents [20].

  In the context of air quality prediction and TL, such heterogeneous third-party and open-access data sources were integrated in the study [69]. In details, this work compiled and harmonized multi-city datasets from Paris, Madrid, Berlin, and Helsinki, combining hourly traffic counts, meteorological variables, road network attributes, and pollutant concentrations (NO_2, $PM_{2.5}$, PM_{10}). To ensure spatial consistency, traffic sensors in each city were geographically linked to their nearest air quality monitoring stations, resulting in unified, multi-variable time-series datasets suitable for comparative and cross-domain modeling. The deployment density of these sensors varied considerably—from the extensive network in Paris, comprising 242 traffic counters and 6 pollution stations, to the sparser configuration in Helsinki, with 57 traffic counters and a single pollution station—highlighting the need for robust domain adaptation strategies. This spatial and semantic

data fusion enabled consistent cross-city ML experiments, supporting the application of transfer learning and domain-adversarial techniques to generalize predictive models across urban contexts with differing sensor infrastructures.

- **Cross-Domain Data Fusion and Synchronization:** across all domains, the integration of heterogeneous datasets required precise temporal and spatial alignment. Traffic and environmental data were matched by timestamp, and geospatial mapping techniques (e.g., nearest neighbor search using OSMnx in Python [76]) were used to link physical locations across data layers. This synchronization ensured that multimodal inputs—such as road conditions, meteorology, and pollution levels—were co-registered for model training. In the WDS studies, temporal alignment ensured that consumption patterns and operational events were matched to the correct DMA or network node, while in traffic forecasting tasks, the data from multiple sensors were aggregated or normalized to allow comparability between roads of different configurations.

Across all domains, the overarching objective remained the same: to construct spatially and temporally coherent, semantically consistent, and analytically relevant datasets capable of supporting robust and transferable ML models for predictive smart city applications.

3.2 Data Preprocessing and Quality Enhancement

Data preprocessing and quality enhancement constitute a critical stage in the ML pipeline, acting as the bridge between raw data acquisition and effective model training. Regardless of the application domain—whether water distribution systems, traffic forecasting, or air quality prediction—the goal is to transform heterogeneous, and often imperfect, data sources into consistent, reliable, and analytically meaningful datasets.

The general strategy adopted in this thesis can be summarized in the following sequential steps depicted in Figure 3.1, which were adapted and implemented differently depending on the characteristics of each dataset and task.

Figure 3.1 Data Preprocessing Steps.

– **Step 1: Data Consolidation and Harmonization**

The first stage involved data consolidation and harmonization, where heterogeneous data formats (CSV, JSON, API streams), often originating from multiple acquisition systems, were merged into a unified analytical schema. This step required ensuring consistent timestamp formats, spatial references, and variable naming conventions across sources.

In the anomaly detection study for water distribution networks, this process entailed merging WNTR simulation outputs, which included both pressure readings and leak history tables, while retaining only variables that reflected leakage events [66].

For the clustering analyses of water demand, multiple smart meter datasets from different DMAs were aggregated into a unified time series table per user [67][68].

In traffic forecasting applications, wide-format tables were created in which each column represented the vehicle count for a specific road-direction pair, obtained by pivoting multi-road traffic counter outputs. A further integration step in traffic forecasting involved merging TomTom FCD with sensor-based traffic counts. The FCD provided aggregated speed, travel time, and congestion metrics across the Catania road network. These were spatially and temporally aligned with the corresponding sensor data to enable cross-validation and scaling for roads lacking physical counters.

In the air quality prediction study, spatial data fusion was performed to associate each traffic sensor with its nearest pollution monitoring station, creating integrated multi-variable time series.

— **Step 2: Duplicate and missing data treatment**

Following consolidation, the process moved to duplicate and missing data treatment, a critical step in ensuring dataset completeness, reliability, and statistical validity.

Duplicate records—entries that appear more than once due to repeated sensor transmissions, API retries or merging of overlapping data sources—can artificially inflate sample sizes and bias model training if left unaddressed. Missing data, on the other hand, may arise from hardware malfunctions, network outages, data corruption, or irregular reporting intervals, and, if not properly handled, can distort model parameters or reduce predictive accuracy. Common strategies for managing these issues include de-duplication through exact-match removal or fuzzy matching for near-identical entries, and missing value treatment via deletion of variables or records with excessive gaps, or imputation techniques that estimate missing values based on temporal, spatial, or statistical patterns in the available data.

In the studies presented in this thesis, exact duplicates were removed, and variables with excessive missingness—typically above 30%—were excluded from further analysis.

In water demand clustering, users with more than 30% missing hourly readings were removed, while remaining gaps were imputed using KNN spatial interpolation [77], which leverages the similarity between neighboring observations to reconstruct missing values [67][68].

In traffic forecasting, missing values caused by sensor malfunctions or special events (e.g., weather alerts) were filled using time-based averaging for the same road, day of week, and hour within the same month. For TomTom FCD, occasional gaps in speed or congestion metrics were interpolated using moving averages calculated within the same temporal context to preserve short-term traffic dynamics [70]-[73].

In air quality prediction, sensors with excessive missingness were discarded, while gaps in the remaining datasets were imputed using the mean for the corresponding hour and weekday in the same month, maintaining temporal consistency in pollutant concentration patterns [69].

— **Step 3: Outlier Detection and Correction**

Once duplicates and missing data were addressed, the next step involved outlier detection and correction, which is essential to prevent extreme or erroneous values from disproportionately influencing model training. Outliers—data points that deviate markedly from the general distribution—can result from sensor malfunctions, data entry errors, unusual operating conditions, or genuine but rare events. If unaccounted for, they may bias statistical estimates, distort scaling, or mislead machine learning models, particularly those sensitive to range and variance. Common strategies for identifying outliers include visual inspection via boxplots or scatterplots, statistical thresholds such as the IQR method, and model-based anomaly detection. Once detected, outliers may be removed or replaced with statistically informed estimates, depending on whether they reflect erroneous measurements or meaningful but rare events relevant to the analysis.

In this thesis, outlier management was tailored to the domain and data type. In traffic forecasting, outliers in vehicle count time series—often due to temporary sensor faults or one-off incidents—were detected using boxplots or IQR thresholds and replaced with values obtained via time-based averaging for the same road, day of week, and hour in the same month [70]-[73]. In detail, for TomTom Floating Car Data, transient spikes in speed or congestion metrics caused by GPS errors or irregular updates were smoothed using short-window moving averages, ensuring continuity while retaining genuine traffic fluctuations.

In air quality prediction, the IQR method was applied to pollutant and traffic variables to identify anomalous spikes or drops, which were then imputed using the same temporal mean strategy as for missing data [69].

For water demand clustering, outliers in effective hourly consumption were detected through visual exploration of time series plots and statistical cut-offs, where abnormally high or negative values indicated faulty smart meter readings or data transmission errors. These outliers were replaced using KNN spatial interpolation [77] ensuring realistic consumption profiles while preserving the overall behavioral patterns [67]-[68].

Finally, in leak detection for water distribution systems, no outlier correction was applied to pressure data. This was a deliberate choice, as unusually high- or low-pressure readings may correspond to the very events the model aims to detect—namely, the onset or progression of leakages. Removing such points could have inadvertently erased valuable predictive signals from the dataset [66].

— **Step 4: Data Normalization and Scaling**

Data normalization and scaling are fundamental preprocessing operations that ensure all features contribute proportionally to model training. In raw datasets, variables often have different units and magnitudes—for instance, traffic counts may range from tens to thousands, while pollutant concentrations or pressure readings can have much smaller numeric ranges. Without normalization, algorithms sensitive to feature magnitude—such as distance-based models (e.g., K-means, KNN) or gradient-based optimizers—can become biased toward

variables with larger absolute values, distorting decision boundaries and degrading predictive performance**.**

Two common scaling approaches are:

- Z-score standardization which centers each feature to have a mean of zero and a standard deviation of one, making it suitable for normally distributed variables.
- Min–Max scaling which rescales features to a fixed range commonly [0,1], while preserving the relative distances between values, useful when the model benefits from bounded input ranges.

The choice between these methods depends on the data distribution, the sensitivity of the algorithm to feature magnitude, and the interpretability requirements of the task.

In this thesis, scaling strategies were selected according to domain characteristics and model requirements. Indeed, for anomaly detection task in WDS, pressure readings, which exhibited Gaussian-like distributions after simulation, were standardized using the *z-score* approach via the StandardScaler function in Scikit-learn [78]. This ensured uniform variance across features, improving the stability of algorithms sensitive to feature dispersion [66].

To profile the users through time series water demand using clustering, households water consumption profiles were normalized using *Min–Max scaling* to a [0,1] range. This approach was critical to cluster households based on usage patterns rather than total volume, preventing the algorithm from grouping consumers by size alone [67][68].

For traffic flows forecasting, vehicle count time series from both physical sensors and TomTom FCD were scaled using *Min–Max normalization*. This harmonized data from roads with vastly different traffic volumes, enabling models to learn temporal patterns without bias toward heavily trafficked roads [70]-[73].

Finally, in the air quality prediction, all traffic, meteorological, and pollutant concentration variables were normalized to [0,1] scale via *Min–Max scaling*, facilitating convergence in multi-variable DL architectures and ensuring comparability across cities with differing measurement ranges [69].

Through these tailored normalization and scaling strategies, the datasets used in each study were brought to a common numeric basis, enabling fair comparison between variables, stable training dynamics, and improved cross-domain generalization.

— **Step 5: Dataset Enrichment**

This step refers to the process of augmenting raw and cleaned data with additional contextual, engineered, or derived features that can enhance the predictive performance and interpretability of a model [78]. While earlier preprocessing stages focus on removing inconsistencies and ensuring quality, enrichment focuses on adding value. In many smart city applications, the phenomena being modeled—traffic flows, water demand patterns, air quality fluctuations—are not solely explained by the core measurements (e.g., vehicle counts, water

volumes, pollutant levels). Instead, they are influenced by a range of temporal, spatial, infrastructural, and environmental factors.

By integrating these additional descriptors, dataset enrichment allows machine learning algorithms to capture latent relationships, seasonal patterns, or cross-domain dependencies that would otherwise remain hidden. Enrichment can be achieved through:

- **Temporal features**: day of week, time of day, seasonality, public holidays.
- **Spatial features**: location coordinates, network topology, infrastructure attributes.
- **Environmental features**: weather variables, pollution levels, event indicators.
- **Derived/engineered features**: aggregated metrics, moving averages, rates of change, domain-specific transformations.

The key challenge is to ensure that these features are *informative* without introducing redundancy or noise. In this thesis, dataset enrichment was tailored to the domain and data source of each study, with a focus on features that provide operationally meaningful signals for predictive modeling [32].

As concerns the application in this thesis, for traffic flow forecasting, enrichment was central to improving forecasting accuracy by incorporating features beyond raw vehicle counts. Temporal indicators such as *day of week* and *hour of day* were added to capture periodic traffic patterns. Spatial and infrastructural descriptors included the *number of lanes*, *lane width class*, and *presence of on-street parking*, all of which influence traffic flow capacity and congestion behavior. For datasets integrating TomTom FCD, congestion metrics and average speeds were merged with sensor counts to provide both demand (flow) and supply (capacity) perspectives [70]-[73].

For air quality prediction, predictive performance was enhanced by including meteorological variables (e.g., temperature, humidity, wind speed, and direction), which have well-established impacts on pollutant dispersion and concentration levels. In addition, spatial descriptors were obtained from road network data using OSMnx [76], including *road density*, *proximity to major roads*, and *intersection density*. These features enriched the link between traffic activity and pollutant concentrations by accounting for both emission sources and dispersion conditions [69].

In water demand clustering, while the clustering focused primarily on consumption profiles, temporal features such as season, day type (weekday/weekend), and hour of day were added to capture behavioral variability. These variables support the identification of consumption patterns that are temporally dependent, such as peak hours or seasonal irrigation use, improving the interpretability of resulting clusters [67][68].

Finally for leak detection in WDS, although the primary data source consisted of simulated hydraulic pressure values, temporal attributes (e.g., time of day, simulation stage) were included to differentiate between routine fluctuations and leakage-induced anomalies. These enrichments assisted the model in distinguishing between genuine fault events and regular operational variability [66].

Across all domains addressed in this thesis, dataset enrichment is aligned with best practices in feature engineering for smart city analytics. The adopted strategies emphasized: (1) relevance, ensuring that each added feature had a plausible relationship with the target variable; (2) domain knowledge integration, where infrastructural and meteorological descriptors were included based on established physical and operational principles; and (3) scalability, enabling enriched datasets to support TL and cross-city applications. The enrichment phase, therefore, not only improved model accuracy but also facilitated deeper interpretability—critical for decision-making in operational urban systems.

3.3 Exploratory Data Analysis

EDA is a pivotal step in the ML pipeline, bridging preprocessing and feature engineering by enabling researchers to *interrogate data structure, distribution, and relationships* before modeling [80][81]. The goal of EDA is twofold: (i) to validate assumptions about data quality and relevance, and (ii) to generate insights into latent patterns, anomalies, and dependencies that can inform both feature selection and model design. Typical EDA procedures include descriptive statistics (mean, variance, skewness), distributional analysis via histograms or kernel density plots, time-series decomposition, correlation matrices, and visualization of spatial or network-based attributes. In smart city applications, EDA is particularly important because urban data streams are *multi-source, heterogeneous, and often spatio-temporal*, requiring careful inspection to uncover patterns such as diurnal cycles in mobility, seasonal shifts in consumption, or episodic spikes in pollutants.

In the anomaly detection study, EDA focused on the inspection of pressure time series generated by WNTR simulations. Distributions of pressure values were analyzed both under normal conditions and in the presence of simulated leaks.

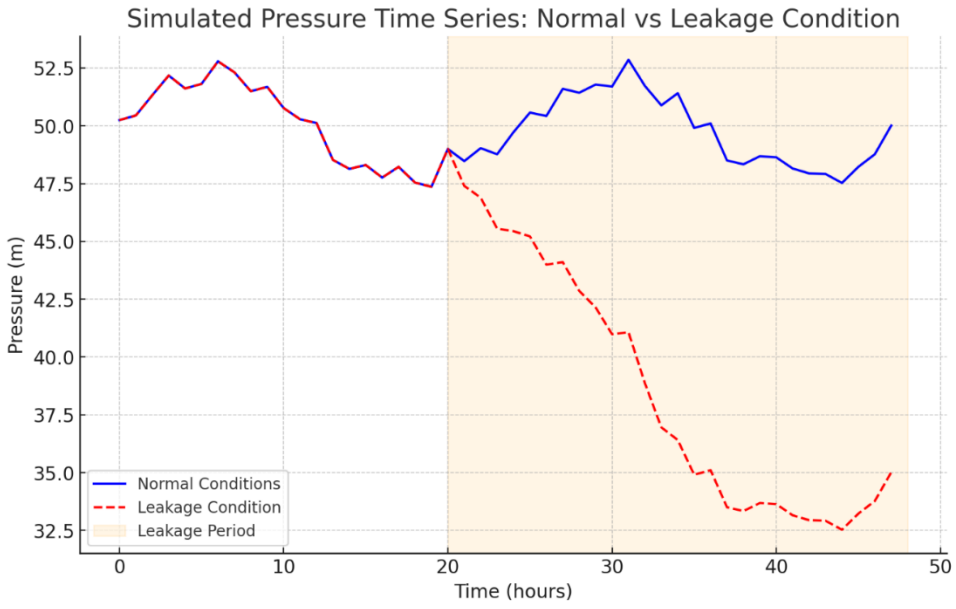


Figure 3.2 Example Normal vs Leakage Pressure Condition.

In Figure 3.2, an example of visual comparison of pressure drops across nodes helped to identify which variables carried discriminatory power for leakage events. Time series plots and

difference curves between “normal” and “leak” states were employed to verify that simulated anomalies aligned with expected hydraulic behavior [66].

For both clustering papers, EDA was central in understanding user consumption behavior.

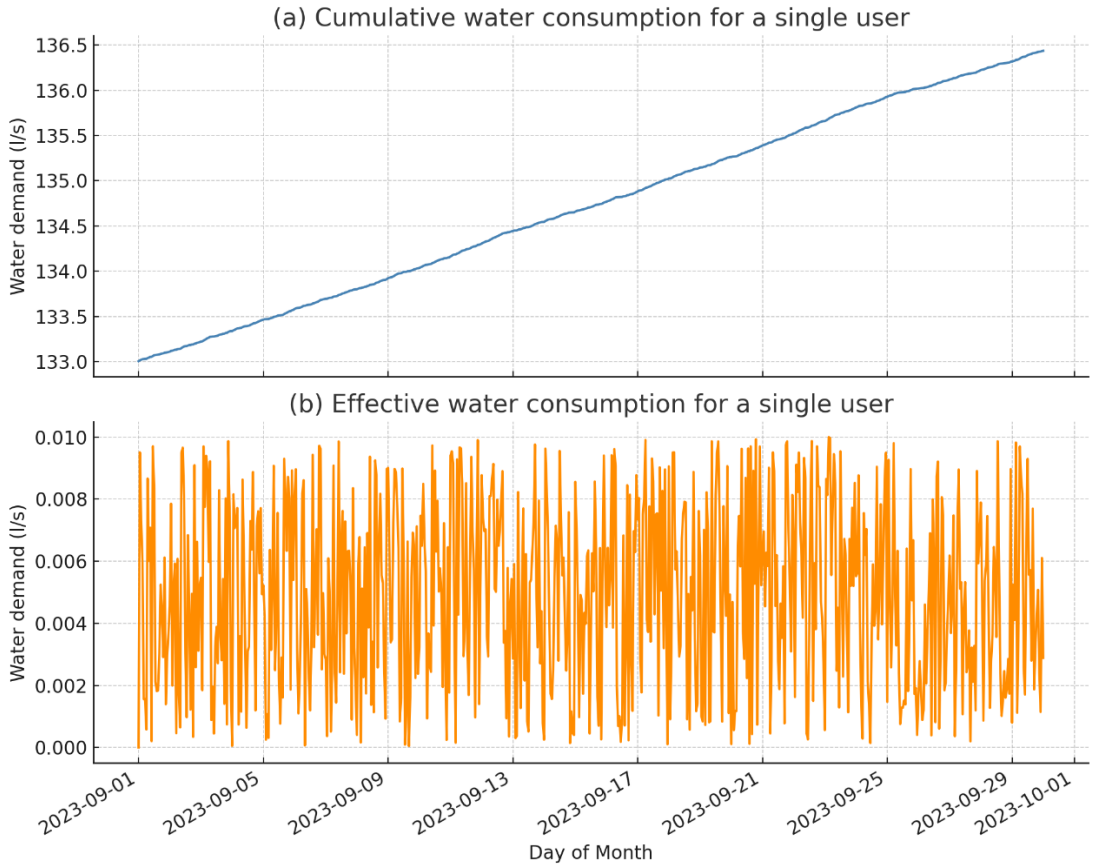


Figure 3.3 Cumulative vs effective water user demand.

As shown in Figure 3.3, initial steps involved visualizing *cumulative vs. effective consumption curves* for representative households to confirm that the differencing transformation properly revealed hourly usage patterns. Indeed, in (a) the cumulative consumption curve steadily increases, which indicates that water consumption is continuous over time. Each new data point is added to the total amount of water consumed up to that point. The slight fluctuations in the slope suggest variations in the rate of water consumption but, overall, the analyzed user consistently consumes water over time. With this type of information, it is not possible to study behavioral patterns in water consumption, such as daily, weekly, or yearly fluctuations. Unlike the cumulative, the effective water consumption depicted in graph (b) offers a more detailed view of variations in water use throughout the day. The spikes in the graph indicate periods of high-water usage, while the lower points or near-zero values suggest little to no water usage during certain hours. Further histograms and boxplots were used to assess variability across DMAs, while correlation analysis identified similarities among user groups. Finally, an important aspect of the exploratory phase was the analysis of user type distribution. Thanks to geographic information provided for each user, an application was developed to identify activity types based on longitude, latitude and address data. Each user was then classified into one of two categories—residential or non-residential, according to the geographic information of the

user. Data exploration revealed a significant sample imbalance, with only 60 non-residential users, representing activities such as restaurants, banks, food manufacturing, and other commercial services. This situation reflects real-world conditions, where residential users typically dominate water consumption in cities [82]. This imbalance, although expected given the structure of most DMAs, was a critical insight into the subsequent clustering process, as it highlighted the potential dominance of residential consumption patterns and the need to carefully interpret results for minority user groups [67][68].

In the traffic forecasting studies, EDA examined diurnal and weekly fluctuations in vehicle counts through aggregated line plots and heatmaps, which revealed strong periodicities (e.g., morning/evening rush hours). Correlation analysis between different sensors demonstrated spatial dependencies across the network, while scatterplots and density plots were used to inspect the relationships between TomTom Floating Car Data (speeds, congestion indices) and sensor counts. Outlier detection performed during preprocessing was complemented here by exploratory visualizations, which confirmed whether spikes were attributable to abnormal events or genuine traffic surges [70][73].

Finally, for air quality prediction across multiple European cities, EDA served to investigate the joint dynamics of pollutants, traffic, and meteorological variables.

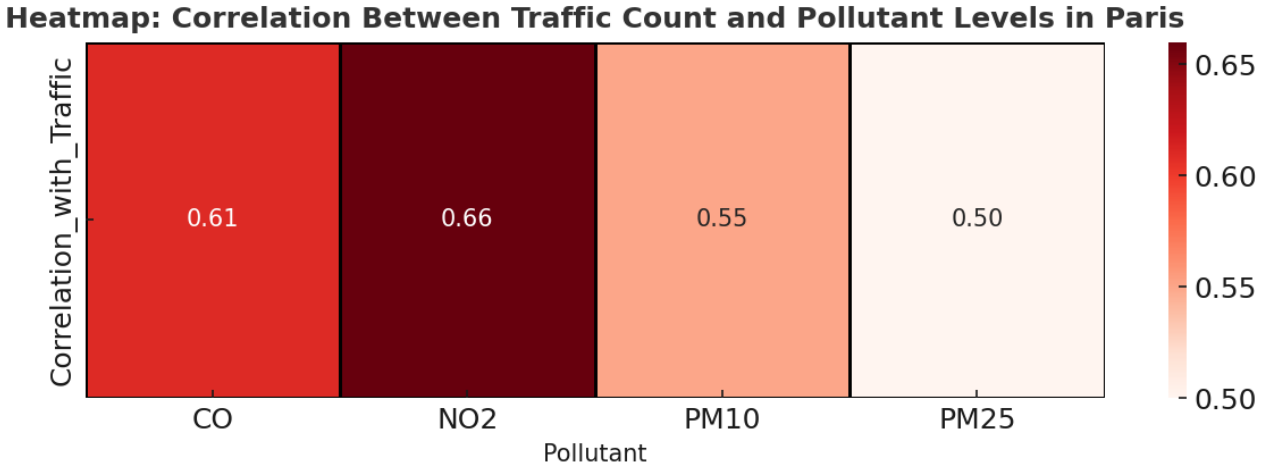


Figure 3.4 Example heatmap correlation between traffic and pollutants.

Figure 3.4 shows an example of correlation heatmap for the city of Paris, used to reveal expected relationships between pollutants concentrations and traffic intensity [69].

In sum, EDA across the different studies combined *visual, statistical, and domain-specific techniques* to validate data quality, uncover meaningful spatio-temporal patterns, and guide methodological choices. Far from being a mere descriptive step, EDA in this thesis acted as a diagnostic tool ensuring that the downstream machine learning models were built upon robust empirical understanding of the underlying data-generating processes.

3.4 Feature Engineering

Feature engineering represents a pivotal stage in the data preparation pipeline. Its primary goal is to derive new, informative representations of the data, refine existing features, and select only those attributes most relevant to the predictive task. In general, this step involves generating domain-specific features, extracting statistical or temporal descriptors, encoding categorical variables, and reducing dimensionality where necessary. Well-designed features can significantly enhance model interpretability and predictive power, particularly when dealing with heterogeneous urban data sources such as water networks, smart meters, traffic counters, floating car data, and air quality monitors.

In this thesis, feature engineering was tailored to the requirements of each application domain and research question. As concerns the anomaly detection in WDS, no dimensionality reduction or engineered statistics were introduced, since filtering out extreme values could have erased true leakage events [66].

For both clustering studies, engineered features were designed to capture behavioral patterns in water consumption rather than absolute volumes. First, as introduced in paragraph *3.3 Exploratory Data Analysis*, cumulative smart meter readings were transformed into effective hourly consumption via differencing, enabling the identification of daily and weekly usage patterns. Additionally, to better characterize consumption dynamics, MSTL algorithm has been used. MSTL iteratively decomposes the time series into three main components: *seasonality*, which represents the repeating patterns in the data, *trend*, which captures the long-term changes of the series, indicating whether the data is generally increasing, decreasing or remaining stable over timer, and *residuals*, which are the remaining part of the data, typically considered as noise or irregular fluctuations [83]. Compared with other decomposition alternatives, MSTL is highly efficient and well-suited to handling large datasets due to its computational scalability [84]. To estimate the different components, MSTL relies on LOESS, a non-parametric regression method that fits local polynomial regressions to the data, smoothing the series at each time point within a defined window. The outcome of the procedure is a smooth curve that represents the seasonality in the data, filtered out from noise to reveal clear patterns. Two key hyperparameters have been set in the MSTL algorithm: *periods*, which regulate the number of time points in each seasonal cycle, and *windows*, which refer to the size of the smoothing window used in the LOESS. While the window size has been left at its default value, the authors set period=168, corresponding to weekly water consumption patterns for hourly data. Setting the period to 168 is a strategic choice when working with water consumption data, in order to capture variations between residential and non-residential users. For example, residential users typically consume more water on weekends, while commercial users show the opposite trend, with higher consumption on weekdays. This strategic choice enables MSTL to accurately reflect these cyclical behaviors and improve the quality of clustering results.

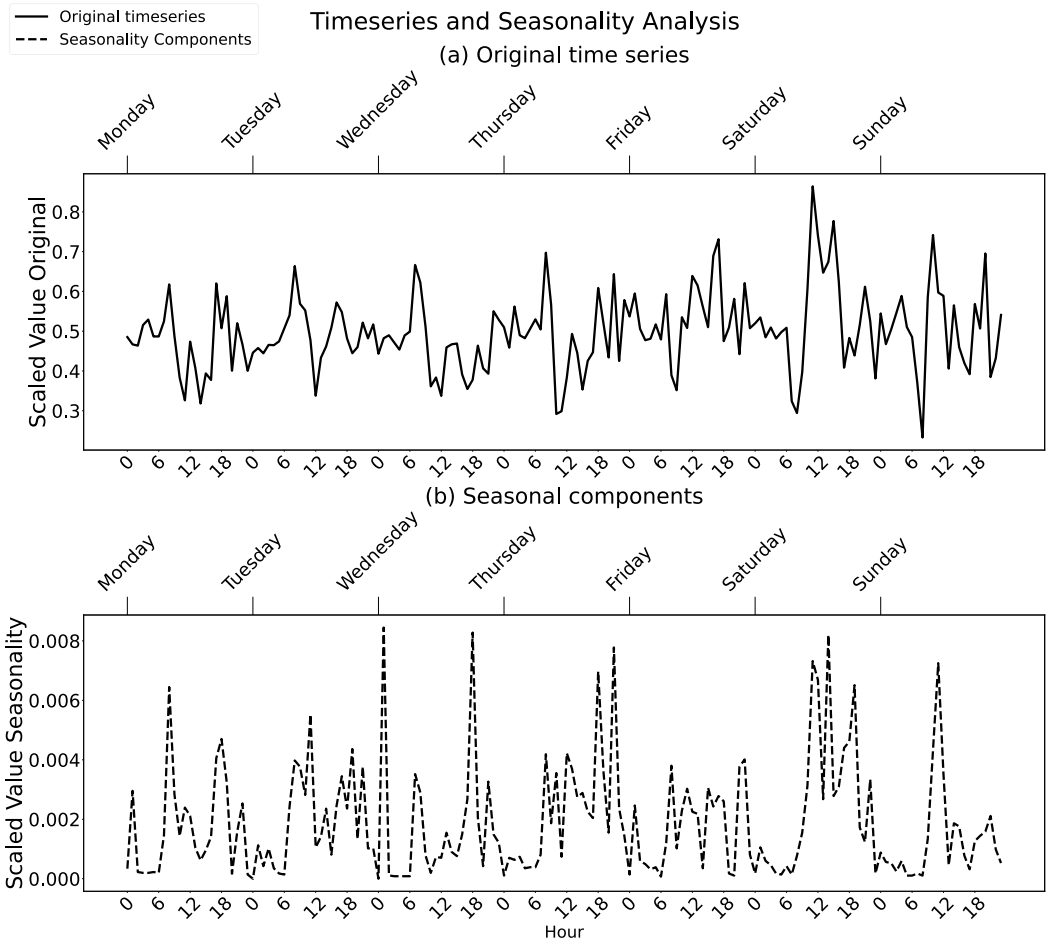


Figure 3.5 Comparison scaled original time series vs scaled seasonal components.

Figure 3.5 shows a comparison between the scaled original time series data (a) and the scaled seasonal components (b) extracted after applying the MSTL algorithm. In detail, in (a) an example of the hourly pattern of water consumption in a week of January for a single user is shown. The bottom graph (b) focuses just on the seasonal component extracted from the original time series. As shown in (b), the seasonal graph shows how water usage typically increases or decreases at specific times within the week. The analysis of seasonal components makes it easier to identify distinct patterns among different users. Lastly, seasonal data was aggregated using the mean as aggregation function to obtain a typical weekly water consumption pattern for each month. Specifically, for each day of the week, the average hourly consumption was calculated by aggregating the values from all occurrences of that particular day within the month. This aggregation was applied to all nine; as a result, a typical weekly consumption pattern for the month for each user is obtained. By focusing on a representative week, general daily and weekly consumption patterns were captured without being skewed by unusual dates within the month. This approach is well-suited for water consumption analysis, which tends to follow daily, weekly, and yearly cycles rather than monthly ones [67][68].

Feature engineering in traffic studies emphasized both temporal and spatial descriptors. Raw 5-minute vehicle counts were aggregated to hourly totals, aligned with forecasting horizons (short- and long-term). Lanes in the same direction were summed, while those in opposite directions were retained as separate features to preserve directional flow dynamics [70]–[73].

Furthermore, additional contextual features were engineered, including day of week, time of day, number of lanes, lane width class, and parking presence, enriching the predictive space with structural and temporal attributes. When TomTom FCD was integrated, additional features such as average travel speed, congestion level, and travel time were extracted, harmonized with sensor-based counts, and used to scale traffic flow predictions for under-sensed road segments.

Finally, for air quality prediction, feature engineering was essential to capture the complex interplay between traffic, environment, and pollution. Spatial linking associated traffic sensors with their nearest air quality monitoring stations, enabling the construction of integrated multi-variable time series. Features included hourly traffic volume, pollutant concentrations (NO_2, $PM_{2.5}$, PM_{10}, CO), and meteorological descriptors such as temperature, humidity, wind speed, and atmospheric pressure [69]. In addition, road network indicators (e.g., centrality, density) were extracted using OSMnx, providing structural context to traffic-pollution interactions. Together, these features enabled advanced models such as domain-adversarial neural networks to generalize across multiple cities with heterogeneous data distributions.

3.5 Feature Selection

Feature selection is a critical step in the ML pipeline, aimed at identifying the most informative subset of variables while discarding redundant or irrelevant ones. This process not only improves computational efficiency but also enhances model interpretability and predictive performance by reducing noise and mitigating the risk of overfitting [85]. In practice, feature selection can be carried out using filter methods (e.g., correlation thresholds, mutual information), wrapper methods (e.g., recursive feature elimination), or embedded approaches (e.g., regularization techniques). The choice of strategy is usually tailored to the problem domain, data type, and the modeling objectives.

For anomaly detection, since pressure readings in a water network are highly correlated across nodes, feature selection aimed at reducing dimensionality without losing the hydraulic information necessary to detect leakages.

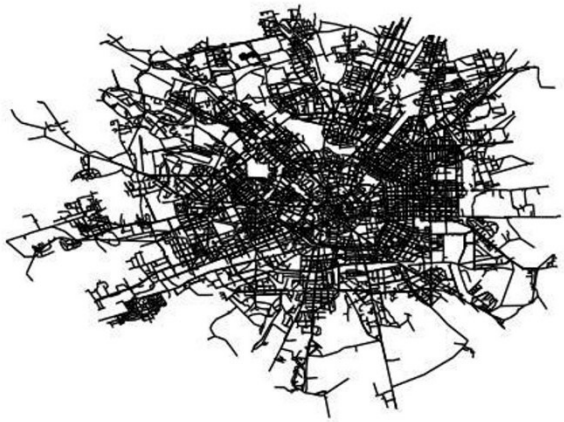


Figure 3.6 Original Milan WDS.

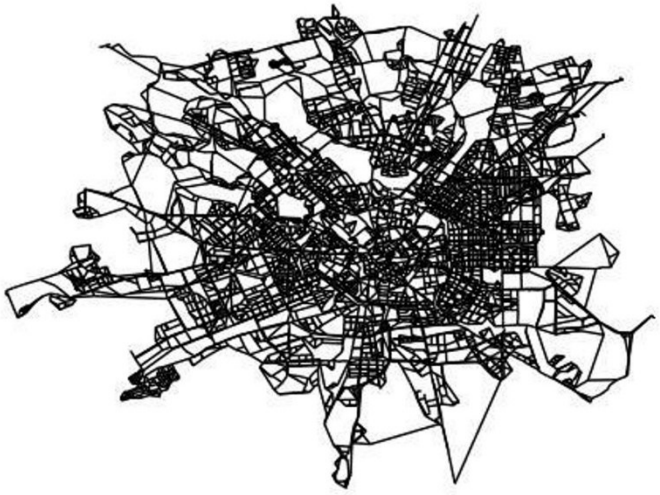


Figure 3.7 Skeletonized Milan WDS.

Figure 3.6 and Figure 3.7 show a planimetry of the Milan network and of its reduced version. A skeletonization process was applied to the original Milan WDS, removing low-impact nodes and pipes to retain only those with significant hydraulic influence. This reduced complexity while preserving the essential dynamics of the system for leakage modeling.

For clustering water demand studies, feature selection went beyond retaining raw consumption variables. After transforming cumulative smart meter readings into effective hourly demand, PCA was applied to reduce the dimensionality of the user consumption profiles. The first few principal components captured the dominant variance associated with daily and weekly consumption patterns, while discarding noise and minor fluctuations. This allowed clustering to focus on major behavioral trends rather than individual outliers or high-frequency variability. In practice, PCA complemented normalization and LOESS smoothing by emphasizing structural differences between residential and non-residential users, whose imbalance was previously highlighted during exploratory analysis [67][68].

As regards the traffic forecasting, the preprocessing steps (described before aggregation by lane/direction, pivoting tables), produced a structured dataset, and all resulting variables were kept for model training [70]-[73].

For air quality prediction, multicollinearity checks and correlation analysis were applied to remove redundant predictors among traffic, meteorological, and pollutant variables. This ensured that the final feature set contained only independent, non-redundant variables [69].

3.6 Model Configuration

The modeling stage represents the core of the ML pipeline, where cleaned and feature-engineered data are used to train algorithms capable of capturing underlying patterns and making reliable predictions. In general, this step involves selecting suitable model families based on the problem type (classification, regression, clustering, or forecasting), training them on historical data, tuning hyperparameters to optimize performance, and validating generalization capabilities through cross-validation or hold-out strategies. The choice of

modeling approach is strongly influenced by domain characteristics, data availability, and task objectives: anomaly detection requires methods sensitive to deviations from normal behavior, clustering demands unsupervised learning to uncover latent structures, while forecasting and regression tasks often benefit from temporal models capable of capturing seasonality and trends.

In this thesis, modeling approaches were tailored to each study as explained below.

3.6.1 Leak Detection in WDSs

In the leak detection study, the modeling phase was designed as an unsupervised anomaly detection task, where the central goal was to identify pressure signal deviations symptomatic of leaks. Since leaks are rare events compared to normal operations, the dataset was highly unbalanced, which required the use of specialized anomaly detection techniques rather than conventional supervised classifiers. To this end, a diverse set of algorithms was applied, spanning statistical, regression-based, distance-based, clustering-based, and ML methods [66].

Statistical approaches included the Generalized ESD Test and the IQR method. The ESD test identifies extreme outliers in a distribution by iteratively removing the most anomalous points, while the IQR method uses quartile thresholds to detect abnormal pressure fluctuations outside the expected range. These approaches are lightweight and interpretable, but they are limited in detecting complex temporal patterns [86][87].

Shift and persistence detection algorithms such as LevelShiftAD and PersistAD were also tested. LevelShiftAD is designed to capture sudden changes in the mean level of a time series, which may correspond to leak-induced disruptions, whereas PersistAD identifies deviations that persist beyond a specified window, helping to filter out transient anomalies [88].

Regression-based approaches were represented by AutoRegressionAD, which models expected pressure values based on historical lags of the same signal. Deviations from the autoregressive prediction were flagged as anomalies. This method is particularly suitable for hydraulic data where pressure is strongly autocorrelated in time [89].

Distance- and density-based methods included the LOF, which estimates the local density of each observation and flags points that significantly deviate from their neighbors as anomalies. LOF is effective in capturing local irregularities in high-dimensional data but can be sensitive to parameter choices [90].

Ensemble-based methods such as the Isolation Forest were also applied. This algorithm isolates anomalies by recursively partitioning the data; anomalous points are easier to isolate since they are few and different. This makes it effective in unbalanced scenarios such as leak detection [91].

Finally, clustering-based techniques like K-Means and Affinity Propagation were evaluated. The idea here is to group pressure time series data into clusters of similar behavior, with points lying far from cluster centers being marked as potential leaks. K-Means assumes spherical cluster structures, whereas Affinity Propagation automatically determines the number of clusters based on data similarities [92].

3.6.2 Clustering of water demands profiles

Clustering water demand data was designed as an unsupervised pattern discovery task, where the objective was to identify groups of users with similar consumption behaviors without relying on predefined labels. This approach is particularly relevant in water distribution systems, where consumption profiles vary widely between households, commercial activities, and industrial users. After data preprocessing and normalization, clustering was applied to reveal hidden structures in the demand profiles, supporting more effective demand management and infrastructure planning. A recurring challenge in clustering is the determination of the optimal number of clusters and the assurance of stability, ensuring that the clusters remain robust to variations in input data and initialization. Both studies addressed these issues through a combination of resampling, hyperparameter tuning, and advanced time-series clustering algorithms.

In [67], TSkmeans was employed as clustering algorithm, due to its ability to handle temporal distortions in smart meter data. Unlike standard K-Means, TSkmeans leverages DTW as a distance metric, enabling alignment of sequences with shifts or varying speeds in consumption events. This allowed for the identification of more realistic behavioral patterns, such as differences in morning vs. evening peak usage across households. Clustering stability was assessed through bootstrap resampling, or Bootstrapping, a resampling technique used in statistics and ML, which consists in drawing multiple samples from the original data with replacement, meaning that the same data point can appear more than once in the resampled dataset [93]. This technique enables clustering validation on "fake" datasets that differ from the original, with some user patterns potentially missing. Starting with an unknown population, X of n elements, m different bootstrap samples (S_1, S_2, …, S_m) are created, each containing the water consumption values over a series of timesteps for the users, who are equally likely to be selected. The number of bootstrap samples, m, is a key hyperparameter that needs to be tuned for optimal model performance.

The second study [68] extended this framework by adopting a more rigorous hyperparameter optimization strategy using the Optuna library [94]. A total of 100 trials were performed, systematically exploring parameters such as the number of clusters, maximum iterations, tolerance thresholds, and the number of bootstrap samples. The computationally intensive process (≈5 hours on a CPU-only setup) highlighted the trade-off between accuracy and efficiency in clustering tasks. Alongside TSkmeans, this study also tested the K-Shape algorithm which relies on SBD and cross-correlation measures to align and cluster time series based on their overall shape rather than magnitude [95]. This dual approach enabled the detection of both fine-grained temporal alignments (via DTW) and global shape similarities (via SBD).

3.6.3 Traffic flow forecasting

The modeling phase in the traffic forecasting studies was designed around the overarching objective of predicting traffic flows across the road network of Catania, including both sensor-equipped and sensor-less roads. The core challenge arises from the fact that TomTom FCD provides only sample-based mobility information (e.g., average speeds, congestion levels, travel times), while inductive loop and magnetic sensors deliver direct traffic counts. Since FCD and

sensor data represent complementary but non-linearly related views of the same phenomenon, the first step in the modeling framework was to establish and learn the statistical and functional relationship between them.

This calibration step was crucial: by using supervised machine learning algorithms as in [73], FCD data were mapped to sensor counts, enabling the scaling of sample-based data into traffic volumes consistent with ground-truth measurements. Once this relationship was modeled, the framework could be extended to forecast traffic flows both on sensor-equipped roads (where models could be directly trained and validated) and on sensor-less roads (where predictions relied on scaled FCD inputs).

Within this general framework, different methodological choices were explored across the four studies, progressively evolving from classical supervised regressors to DL and hybrid spatio-temporal models:

- In [70] a supervised learning approach was adopted to predict short-term traffic flows from calibrated FCD and sensor data. Algorithms including Random Forests, Gradient Boosting, and SVR were trained to capture nonlinear relationships, with hyperparameter optimization ensuring real-time applicability.
- The work in [71] extended the modeling to recurrent neural networks, focusing on LSTM and GRUs. These models were specifically designed to capture sequential dependencies and temporal correlations in traffic data, improving the ability to forecast congestion patterns at fine temporal resolutions.
- In [72] the authors introduced a spatio-temporal perspective by combining GNNs with recurrent models. Here, the road network was represented as a graph, where nodes corresponded to road segments and edges encoded spatial connectivity. This hybrid modeling strategy allowed the simultaneous exploitation of spatial interdependencies and temporal dynamics, offering a more holistic representation of traffic flow evolution.
- Finally, [73] focused on sensor-less roads, leveraging transfer learning to generalize predictive models trained on sensor-rich roads. Both tree-based ensembles (Random Forests, XGBoost) and neural networks were applied, with feature mappings enabling knowledge transfer across heterogeneous road segments. This demonstrated the feasibility of extending predictive capabilities to under-instrumented areas of the urban network.

Taken together, these studies illustrate a coherent methodological trajectory: starting with the calibration of FCD to sensor data, moving to robust supervised regressors, and gradually evolving toward advanced DL approaches. This progression reflects both the increasing complexity of the modeling strategies and the expanding scope of the predictive task — from sensor-specific forecasting to generalized predictions across the entire urban road network.

3.6.4 Air quality prediction and cross-city transfer learning

The modeling phase of the air quality prediction study was designed with the overarching goal of developing a domain-invariant framework capable of generalizing pollutant predictions across structurally similar cities. Unlike single-city prediction tasks, this study tackled the cross-city generalization problem, where differences in data distributions (traffic, infrastructure, emissions) pose a significant challenge for model transferability. The pipeline thus combined three key stages: (i) similarity analysis across cities, (ii) domain-invariant feature extraction, and (iii) regression modeling of pollutant concentrations.

The first stage focused on assessing inter-city similarity to determine which urban contexts were suitable for training transferable models. To evaluate similarities, a combination of radar plots, PCA, and hierarchical clustering was employed, enabling both visual and statistical comparisons. Two separate analyses were carried out: one using the full set of indicators, and another excluding public transport variables to account for cities lacking metro or tram systems. This ensured that structurally comparable cities could be paired for training and TL.

The second stage introduced DANN [96], employed as a feature extractor to generate domain-invariant representations. The DANN architecture included a feature extractor, label predictor, and domain classifier, trained in an adversarial setting. The feature extractor sought to capture patterns relevant to pollution prediction, while simultaneously learning to deceive the domain classifier, which attempted to identify the city of origin. This adversarial process encouraged the learning of generalizable, city-independent features that enabled knowledge transfer across urban domains with different local distributions.

Once the feature-extracted dataset was constructed, the third stage involved predictive modeling of pollutant concentrations. The task was framed as a multi-target regression problem, simultaneously predicting NO_2, PM_{10}, and $PM_{2.5}$ levels across monitoring stations. Several tree-based ML algorithms (e.g., Random Forest, Gradient Boosting, Decision Trees) were implemented using scikit-learn. Importantly, the models were trained and validated in a cross-city setup, where two structurally similar cities were used for training while two others were held out for testing. This design explicitly tested the transferability of the learned representations.

Finally, a generalization test was performed by applying the best-performing feature extractor regression model combination to previously unseen cities. This evaluation demonstrated the robustness of the proposed framework: models were expected to generalize well in structurally similar cities, but not in dissimilar ones. Such an outcome reinforced the central claim of the work — that air quality prediction requires domain-invariant, transferable models tailored to groups of comparable cities, rather than a one-size-fits-all global model.

3.7 Training and testing strategies

A critical stage in any ML workflow is the definition of appropriate training and testing strategies, which ensure that developed models generalize beyond the data used for model fitting. In supervised learning, datasets are typically split into training sets, used to fit model

parameters, and testing sets, employed to evaluate predictive ability on unseen data. Depending on the problem context, different approaches may be adopted, ranging from random hold-out splits and k-fold cross-validation to domain-specific partitioning strategies such as temporal splits in time series forecasting. These strategies mitigate risks of overfitting while allowing robust assessment of model generalizability.

In the leak detection study for water distribution systems, models were trained on normal operating conditions data, while synthetic leaks generated by WNTR simulations served as test cases to evaluate detection ability under abnormal scenarios [66].

For water demand clustering, since the objective was unsupervised pattern discovery, bootstrap resampling was used to repeatedly generate perturbed datasets from the original consumption profiles. This approach enabled testing the stability of cluster assignments and supported hyperparameter optimization via the Optuna library, ensuring reproducibility and robustness across different clustering configurations [67][68].

In traffic forecasting studies, the sequential nature of traffic data required a temporal hold-out strategy: training was performed on earlier intervals of traffic counts and TomTom FCD, while more recent intervals were reserved for testing, preventing information leakage across time horizons [70]-[73].

Finally, in the air quality prediction work, a cross-city TL strategy was adopted. Here, structurally similar cities were selected as the training domain, with the last week of each month in 2023 set aside as the temporal test set. Held-out cities that were not included in the training served as an additional validation layer, testing the generalizability of the domain-adversarial framework to unseen environments [69].

Together, these diverse training and testing approaches highlight the importance of tailoring validation strategies to the characteristics of the data and task: anomaly detection required separation of normal and abnormal conditions, clustering required stability testing under resampling, traffic forecasting demanded temporal splits, and cross-city pollution prediction necessitated domain transfer evaluation.

3.8 Performance Evaluation

Performance evaluation is a fundamental component of the ML pipeline, as it provides objective evidence of how well the developed models achieve their intended goals across different application domains. The selection of evaluation metrics depends on the nature of the task—classification, regression, clustering, or anomaly detection—as well as the properties of the data, such as balance between classes, temporal dependencies, and the presence of noise. In general, performance is assessed using a mix of accuracy-based, error-based, and cluster validation metrics, ensuring both predictive accuracy and generalizability.

In the leak detection study for water distribution systems, the task was framed as an anomaly detection problem, where the primary challenge was the strong imbalance between normal operating conditions and rare leakage events. In such cases, relying exclusively on overall accuracy can be misleading, since models may achieve high scores simply by favoring the

majority class. For this reason, evaluation relied on Precision, Recall, and the F1-score in addition to Accuracy. Precision measured the proportion of detected leaks that were truly leak events, Recall quantified the ability to capture rare leak conditions, and the F1-score provided a harmonic mean balancing the two [97]. This evaluation strategy ensured that models were not only accurate under normal conditions but also sensitive to the early detection of anomalies [66].

For the clustering of water demand, the objective was unsupervised discovery of behavioral groups among users. Evaluation relied on a combination of the Silhouette coefficient, which measures intra-cluster cohesion and inter-cluster separation [98], and information criteria such as the AIC and BIC, which penalize model complexity to avoid overfitting [99]. A combined score balancing these measures was used to select the optimal number of clusters [67][68].

In the traffic forecasting studies, the task was treated as a supervised regression problem with strong temporal dependencies. Given the time-series structure, chronological train–test splits were adopted to avoid data leakage and preserve temporal causality. Performance was assessed using error-based metrics, primarily the RMSE, which emphasizes large deviations, the MAE, which provides a more interpretable measure of average error, and the MAPE, which normalizes errors relative to observed values, making it useful for comparing roads with different traffic volumes [100]. These metrics collectively provided insight into both the absolute and relative accuracy of predictions across sensor-equipped and sensor-less roads [70]-[73].

Finally, in the air quality prediction study, performance evaluation was designed to address the cross-city generalization problem. Because the models had to predict pollutant levels (NO_2, PM_{10}, and $PM_{2.5}$) in unseen cities, traditional linear-model metrics such as R^2 and RMSE were intentionally excluded, as they assume linear relationships and may fail to capture the complex, non-linear traffic–pollution interactions [101]. Instead, the evaluation combined error-based measures such as MAE with rank-based metrics that assess monotonicity between observed and predicted pollutant levels, better reflecting the non-linear but directional dependencies between traffic activity and air pollution concentrations [102]. To further validate generalizability, the evaluation adopted a hold-out strategy, in which models were trained on structurally similar cities and tested on unseen ones, ensuring that predictive performance reflected the ability to transfer knowledge across urban environments rather than mere memorization [69].

Overall, performance evaluation across the different studies in this thesis was tailored to the specific characteristics of each task: rare-event detection in leak analysis, unsupervised stability in clustering, error minimization in traffic forecasting, and cross-domain generalization in air quality prediction. By adopting context-sensitive evaluation strategies, the analyses ensured both methodological rigor and practical relevance for urban infrastructure applications.

3.9 Conclusions

This chapter has outlined an integrative ML methodology for smart city applications, spanning the entire pipeline from data acquisition and preprocessing to exploratory analysis, feature engineering, feature selection, model development, and performance evaluation. While the

framework was presented in a general form, its flexibility was demonstrated across diverse domains, including water distribution networks, urban mobility and traffic forecasting, and air quality prediction. In each case, domain-specific adaptations were required, but the overarching methodology emphasized a rigorous combination of data integration, quality enhancement, robust feature design, and context-sensitive evaluation strategies.

The next chapter will delve in detail into one of the core application domains of this thesis: leak detection in water distribution systems. This focus reflects both the novelty and relevance of the state of the art in this field and the methodological contributions developed by the authors. Chapter 4 will present the case study investigated, describe the proposed methodology as applied in practice, and discuss the results obtained, highlighting the potential of data-driven anomaly detection for improving the reliability and sustainability of urban water infrastructure.

Chapter 4

Leak Detection in Water Distribution Systems

WDSs represent one of the most critical infrastructures for urban sustainability, ensuring reliable access to clean water for residential, commercial, and industrial use. However, these systems are highly vulnerable to leakages, which can lead to significant economic losses, service disruptions, and environmental impacts. Detecting such events in a timely and accurate manner is a complex task, as leaks are often subtle, masked by normal fluctuations in pressure or demand, and difficult to distinguish without advanced analytical tools. In recent years, MLg–based anomaly detection has emerged as a powerful approach to this challenge, enabling the automatic identification of abnormal hydraulic behaviors that may signal the presence of leaks.

This chapter focuses specifically on the application of the integrated ML methodology presented in Chapter 3 to the problem of leakage detection in WDSs. Building on high-fidelity simulations generated by WNTR, pressure and demand signals were collected under both normal and faulty operating conditions. These data served as the foundation for designing, preprocessing, and analyzing time series tailored to anomaly detection tasks. The study systematically compared multiple state-of-the-art unsupervised detection algorithms, reflecting diverse statistical, distance-based, and clustering paradigms, with the goal of assessing their suitability for highly unbalanced leak detection scenarios.

The remainder of this chapter is structured to provide a detailed account of this case study. First, the state of the art in leakage detection methods is reviewed, highlighting the transition from traditional hydraulic modeling approaches to modern ML-based anomaly detection frameworks. Then, the case study setup is presented, describing the simulated water distribution network, the data acquisition process, and the preprocessing steps. Finally, the results obtained with the proposed methodology are discussed, evaluating the strengths and limitations of different anomaly detection algorithms when applied to the leak detection problem. Together, this chapter illustrates how the general methodology introduced earlier can be adapted to a specific, high-impact urban infrastructure challenge, laying the foundation for the novel contributions of this thesis.

4.1 State of the Art

The detection and prediction of leakages in WDSs have been widely studied, with existing methods broadly classified into hardware-based and software-based approaches. Hardware methods can be divided into passive and active systems: passive techniques rely on visual

inspection or direct sensing devices, while active approaches exploit acoustic, vibration, flow, or pressure signals. Early studies demonstrated the use of acoustic measurements for leak identification in plastic pipes [103], while more recent works have integrated artificial intelligence, such as DL for hydroacoustic spectrogram analysis [104] and neural networks trained on simulated acoustic leak signals [105]. Vibration-based approaches have also been explored, including wavelet-based analysis of pipeline vibrations [106] and ML models applied to vibration signals captured by wireless accelerometers in real networks [107].

Active systems encompass transient-based, hydraulic model-based, and data-driven approaches. Transient-based methods exploit alterations in flow and pressure signals caused by structural changes in pipelines [108], but their reliance on high-frequency acquisition makes them costly and impractical for real-time monitoring of large WDSs [109]. Hydraulic model-based approaches use mathematical formulations to simulate network behavior, but they require extensive calibration data and assume static conditions, which is unrealistic given that pipe aging, and roughness evolve over time [110].

In recent years, data-driven approaches have emerged as the most flexible and scalable solution. These methods include supervised, semi-supervised, and unsupervised learning. Supervised learning, where classifiers are trained on labeled normal and abnormal data, can achieve good performance in simple networks but is rarely practical due to the scarcity of labeled leak data [111]. Semi-supervised learning, requiring only normal data, has been adopted in related applications such as water quality monitoring [112]. By contrast, unsupervised methods have gained prominence for leakage detection, as they do not rely on labeled datasets and are therefore more feasible in operational contexts [111]. For example, recurrent neural networks have been applied to the LeakDB dataset to identify anomalies in flow and pressure data, while the NAIADES project [113] adopted spatio-temporal anomaly detection on pressure and flow data in the Braila district.

Although detection has been widely studied, prediction of leakages remains a comparatively underexplored research direction. Proposed methods include radial basis function neural networks to predict leaks by modeling influencing factors [114], Bayesian network learning for dynamically updating failure probabilities [115], and digital twin models for predictive maintenance in hydraulic systems [116]. Nevertheless, most works still focus on the detection of abrupt leaks, with limited attention to incipient events.

Building on this background, the methodology developed in this thesis leverages unsupervised time-series anomaly detection algorithms for both detection and prediction of leakages. In contrast to many existing approaches, the proposed framework relies solely on pressure data, allowing algorithms to first identify anomalies associated with leaks and then extend this learned knowledge to prediction. By requiring only a single variable, the approach simplifies deployment while retaining the ability to capture both abrupt and incipient leaks, thereby offering a novel contribution to the literature, which has primarily emphasized detection rather than prediction.

4.2 Case Study

The case study developed in this thesis focuses on a two-phase methodology for leakage detection and prediction in WDSs, with the process organized into data generation, pre-processing, detection, and prediction. Since real-world leakage data are often scarce—due to the absence of detailed digital records of maintenance activities and the lack of long-term monitoring infrastructures— as anticipated in Chapter 3, synthetic data were generated using the WNTR Python package [75].

The simulations were based on the Milan WDS, a large-scale network that was first reduced through a skeletonization procedure to maintain the hydraulic behavior of the system while removing components with negligible impact. The resulting network included 12,354 nodes, 17,548 pipes, 26 pumping stations, and 95 booster pumps, which provided a realistic yet computationally tractable environment for experimentation. To model realistic variability in demand, coefficients of variation were extracted from SCADA records, aggregated at 30-minute intervals, and perturbed by adding random noise within ±1.5 units and scaled by a small percentage factor (0.05), a step designed to reproduce fluctuations in daily demand patterns without distorting their general shape. These demand curves were then fed into the WNTR simulator to generate the hydraulic conditions of the network over multiple days.

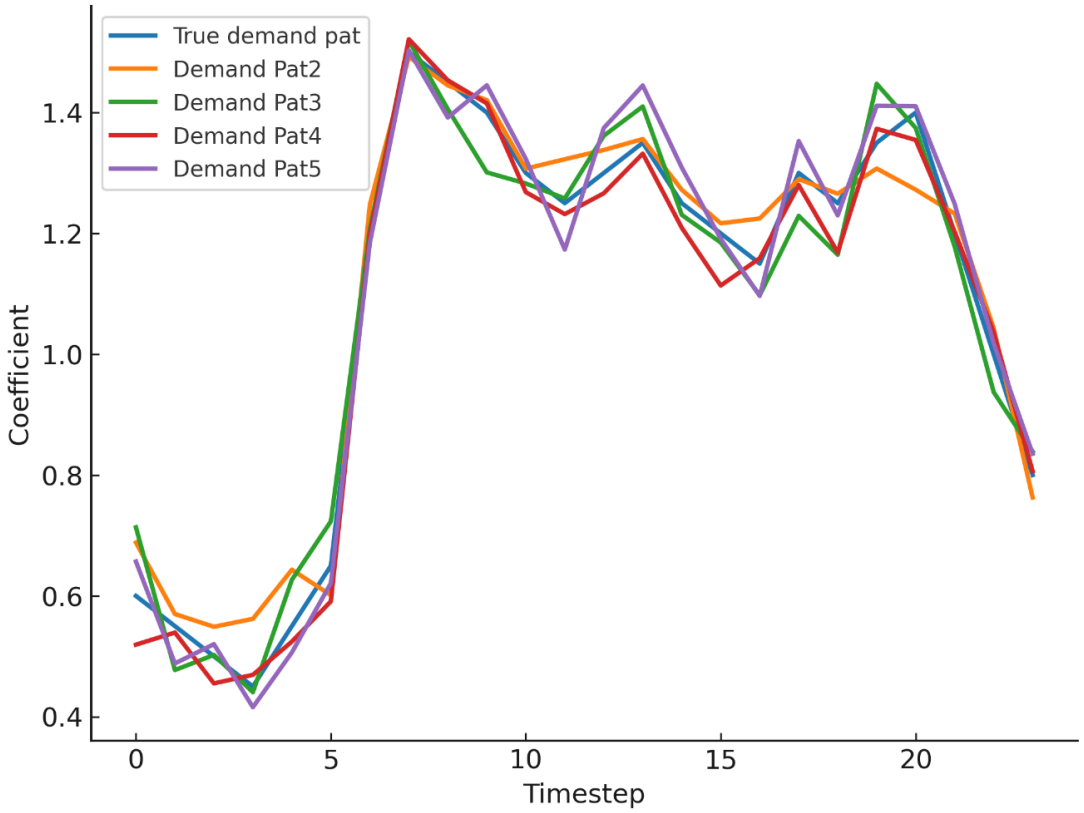


Figure 4.1 Simulated water demand curves.

Figure 4.1 shows an example of the obtained water curves demand. In the figure, the blue-line curve represents the original water demand curve, obtained starting from the real-world

coefficients. The other curves in the plot (Demand-Pat2, Demand Pat3, Demand Pat4 and Demand Pat5) are the simulated water demand curves.

Leakages were simulated at randomly selected nodes, where WNTR introduced artificial leaks by splitting the associated pipe and adding a new leak node with a randomly assigned orifice diameter, allowing the magnitude of the leakage to vary. This produced pressure datasets that captured both normal operating conditions and altered states caused by leakage. In the data pre-processing phase, duplicate columns introduced by the simulator (corresponding to normal nodes and their leak-node counterparts) were identified, and only the leak-representative pressure data were retained. In addition, timestamps originally recorded in seconds were converted into a standard datetime format to align with the time granularity of water demand inputs.

Timestamp	Abbiategrasso	Anfossi	...
2009-11-18 00:00:00	66.4090	64.30.0	...
2009-11-18 00:00:00	66.6450	64.3740	...
...	...	...	...
2009-11-22 23:00:00	67.6521	64.6836	...

Table 4.I Pressure dataset.

End Node	Timestamp	Diameter
N03185	2009-11-18 00:00:00	0.3717
N22998	2009-11-18 00:00:00	0.1328
...	...	...
N10174	2009-11-22 23:00:00	0.1174

Table 4.II Leak history dataset.

Table 1 and Table 2 show the final datasets used in the analysis. As shown in the two figures, the outcomes of these steps were two structured datasets composed the first by half-hourly pressure readings across all nodes of the skeletonized Milan WDS, and the second by a detailed leak history, which together formed the foundation for the subsequent unsupervised anomaly detection and predictive modeling experiments carried out in the next phases of the study.

4.3 Results

4.3.1 Leak Detection

The experimental evaluation of leakage detection began with the systematic application of a wide range of anomaly detection algorithms implemented in the ADTK, a Python library designed for unsupervised and rule-based anomaly detection in time series [117]. The algorithms tested covered diverse methodological paradigms, including statistical approaches, regression-based methods, clustering techniques, and isolation-based models, all applied to pressure time series generated through WNTR simulations of the Milan water distribution system [75]. A key challenge in this task was the strong imbalance of the dataset, with normal operating conditions vastly outnumbering leak events. This imbalance was reflected in the results: while accuracy scores remained very high across all models, precision, recall, and F1-scores were often extremely low, making accuracy alone an inadequate measure of performance [118][119]. Among the algorithms tested, the InterQuartileRangeAD (IQR-AD) consistently outperformed the others in terms of precision score. However, this algorithm suffers from a Quadratic Time Complexity $O(n^2)$, requiring a greater computational effort due to its quadratic time complexity. To alleviate the imbalance issue, leakage events were simulated with extended durations ranging between 8 and 72 hours, thereby enlarging the proportion of leakage data in the dataset. This adjustment significantly improved the precision scores of all models, while confirming the superior performance of the IQR-AD algorithm. On this basis, the IQR-AD was selected as the most suitable tool to be extended to the prediction task.

4.3.2 Leak Prediction

For leakage prediction, an additional feature engineering step was introduced through rolling window aggregation, transforming pressure time series into a representation that captures temporal dependencies in system behavior [120]. This allowed the model to anticipate anomalies by analyzing historical pressure patterns over defined time horizons. Two different predictive windows were considered: a shorter 1-hour horizon and a longer 3-hour horizon. Results demonstrated that the IQR-AD algorithm was capable of raising early warnings before the actual occurrence of leaks at several network nodes, including N00971 and N02197.

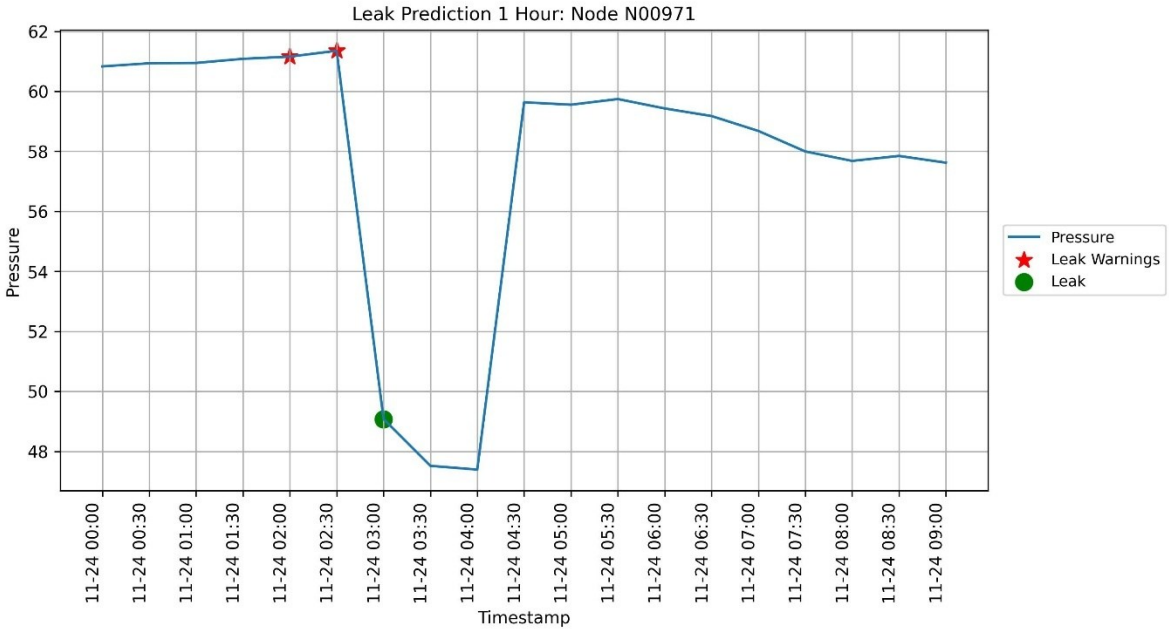


Figure 4.2 Example 1H prediction Leak Node N00971.

Figure 4.2 shows an example of the 1-hour prediction scenario for the node N00971, where the algorithm flagged anomalies between 30 and 60 minutes before the true leak onset.

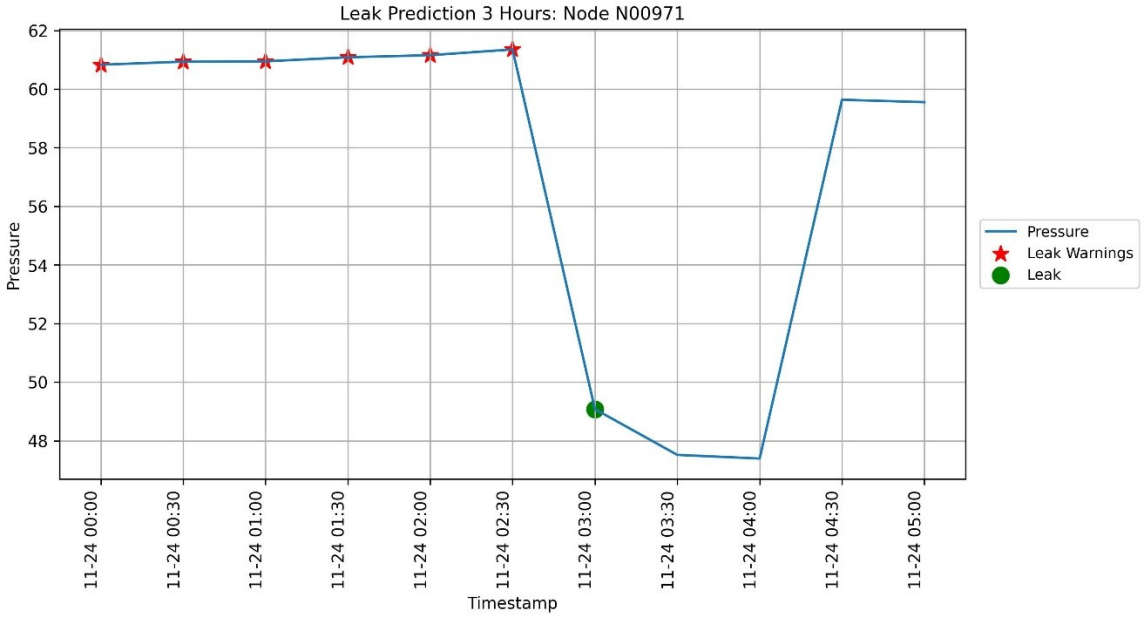


Figure 4.3 Example 3H prediction Leak Node N00971.

Figure 4.3 represents, instead, the 3-hour scenario, where the algorithm successfully signaled leak events with a lead time of up to three hours. These findings highlight the effectiveness of the proposed two-phase methodology, where detection and prediction are seamlessly integrated. The capacity to not only identify but also anticipate leakages supports the development of predictive maintenance strategies, enabling utilities to intervene proactively and mitigate economic, service, and environmental losses [116]. Overall, this work demonstrates how unsupervised anomaly detection, combined with careful data balancing and

temporal feature engineering, can provide a robust framework for addressing one of the most critical challenges in modern water distribution systems.

4.4 Conclusions

The investigation into leak detection and prediction within WDSs has demonstrated the effectiveness of unsupervised anomaly detection techniques for addressing one of the most persistent challenges in urban infrastructure management. By leveraging high-fidelity simulations and applying a wide set of state-of-the-art algorithms, the study highlighted the trade-offs between accuracy, precision, and computational complexity when operating in highly unbalanced datasets. The IQR-AD method emerged as the most promising approach, showing strong performance in detecting leaks and providing early warnings in prediction tasks through feature-engineered rolling window aggregation. This dual capability of both identifying and anticipating leaks emphasizes the potential of data-driven anomaly detection frameworks to support proactive maintenance strategies, reduce water losses, and improve system resilience. While results confirm the feasibility of the approach, they also underscore the importance of continued refinement in handling unbalanced conditions and exploring scalable solutions for real-time monitoring. Overall, this work contributes to advancing the digitalization of water utilities by moving toward predictive and preventive management practices, aligning with the broader vision of Water 4.0.

Building on this contribution, the next chapter shifts focus from anomaly detection to the characterization of consumer behavior, another application in smart cities. Specifically, it explores how clustering techniques can be applied to water demand patterns in order to profile users, uncover usage dynamics, and provide valuable insights for demand management and service optimization.

Chapter 5

Modelling and Clustering Water Demand Patterns in Water Distribution Systems

The increasing global population, rapid urbanization, and the accelerating impacts of climate change are placing unprecedented pressure on water resources worldwide, making the concept of a "water crisis" an urgent and recurring theme in both scientific and policy discussions [121]. Droughts caused by reduced rainfall and rising temperatures have dramatically diminished reservoir storage capacities, while the aging and mismanagement of water infrastructures further exacerbate inefficiencies in WDSs. The IWA estimates that 346 billion liters of water are lost daily due to leaks, while the World Bank reports annual global losses of 8.6 trillion gallons caused by failures and breakages in water networks [122]. Against this backdrop, improving the efficiency and sustainability of WDSs requires a deeper understanding of water consumption behaviors, which can inform predictive demand models, guide resource allocation, and support proactive maintenance strategies [123].

The advent of smart metering technologies has been transformative in this context, enabling the collection of high-resolution consumption data across both spatial and temporal scales [124]. These granular datasets create new opportunities for the development of advanced data-driven applications, including user profiling, anomaly detection, and demand forecasting. Profiling consumers by analyzing their water demand patterns is particularly valuable, as it allows utilities to distinguish between residential and non-residential users, identify typical consumption habits, and detect irregularities that may indicate leaks, unauthorized use, or other anomalies.

To address this challenge, this chapter introduces a clustering-based methodology for the analysis of hourly water demand time series. By grouping users with similar, though not identical, consumption patterns, the approach generates representative profiles capable of supporting demand-side management and enhancing system resilience.

5.1 State of the Art

Research on water demand profiling has historically concentrated on the residential sector, as households represent the largest share of end-users in many contexts [125]. Traditionally, the available data came from conventional water meters, aggregated monthly or yearly for billing purposes. While useful for estimating overall consumption, such data offered only a coarse understanding of water use and prevented the analysis of finer consumption behaviors or temporal trends. The introduction of smart metering technologies has significantly transformed

this scenario by enabling the acquisition of high-resolution water demand data at intervals as short as 15 minutes or hourly [124]. This technological shift has opened the way for the development of advanced analytical methods for profiling users, building predictive demand models, detecting anomalies, and supporting resource optimization.

Several studies have leveraged these new opportunities, especially in the residential domain. In [126], for example, the authors applied unsupervised learning techniques to develop a typology of residential water users in three cities in the Southwestern United States, combining social survey data with clustering methods to better understand behavioral patterns and conservation attitudes. Similarly, [127] examined a sample of 800 households using K-Means clustering, applying multiple pre-processing strategies to improve the accuracy of profile identification. Both contributions illustrate the potential of ML for segmenting residential users but remain confined to household-level consumption, neglecting the substantial share of demand attributed to commercial, industrial, and service users.

Recognizing this gap, [128] proposed a broader data-driven framework capable of profiling both residential and non-residential consumers. Their methodology was structured in two steps: first, a Fourier decomposition was applied to extract seasonal consumption patterns from smart meter data; then, two clustering strategies—K-Means and FReMix generative models—were employed to group users according to their behavioral characteristics. The study demonstrated the importance of capturing seasonality and heterogeneity in user behavior, while also emphasizing the qualitative interpretability of the resulting clusters.

Building on this foundation, the present thesis sought to refine clustering-based profiling through more advanced methodological pipelines. In particular, time-series decomposition methods, such as MSTL using LOESS, have been used to separate recurring patterns from noise in hourly water demand series [83]. This preprocessing step enhances the ability of clustering algorithms to focus on meaningful seasonal and trend components rather than being affected by short-term fluctuations. Following decomposition, optimized time-series clustering methods have been applied, with TSK-Means being a prominent example. To ensure robustness, these studies often adopt Bootstrap Sampling, testing different sets of hyperparameters across multiple clustering runs to evaluate the stability of the resulting profiles.

A further advancement has been the validation of consumption profiles using geo-referenced user data. By associating clusters with real-world residential and non-residential categories, recent research has been able to verify the practical significance of the clustering outcomes, ensuring they are not merely statistical artifacts but reflect actual patterns of water use across diverse consumer groups. This approach not only strengthens the interpretability of the results but also extends the utility of clustering for operational decision-making, for example in detecting irregular consumption that may indicate leaks, billing errors, or unauthorized water use.

Taken together, these contributions extend the reliability, interpretability, and applicability of clustering-based approaches, offering water utilities a novel decision-support tool for both consumer profiling and the early detection of irregularities such as leaks, billing errors, or unauthorized water use.

To implement these contributions, the next chapter presents the case study and details the methodological framework adopted. Here, the proposed approach is applied to real smart meter data from a WDS, illustrating step by step how clustering techniques and validation strategies were implemented and assessed in practice.

5.2 Case Study

The case study presented in this thesis concerns a real WDS located in a medium-sized city in central Italy. Consumption data were collected from six DMAs, monitored through smart meters that recorded hourly cumulative consumption values in l/s (litres per second) over a nine-month period, from September 5, 2023, to May 31, 2024.

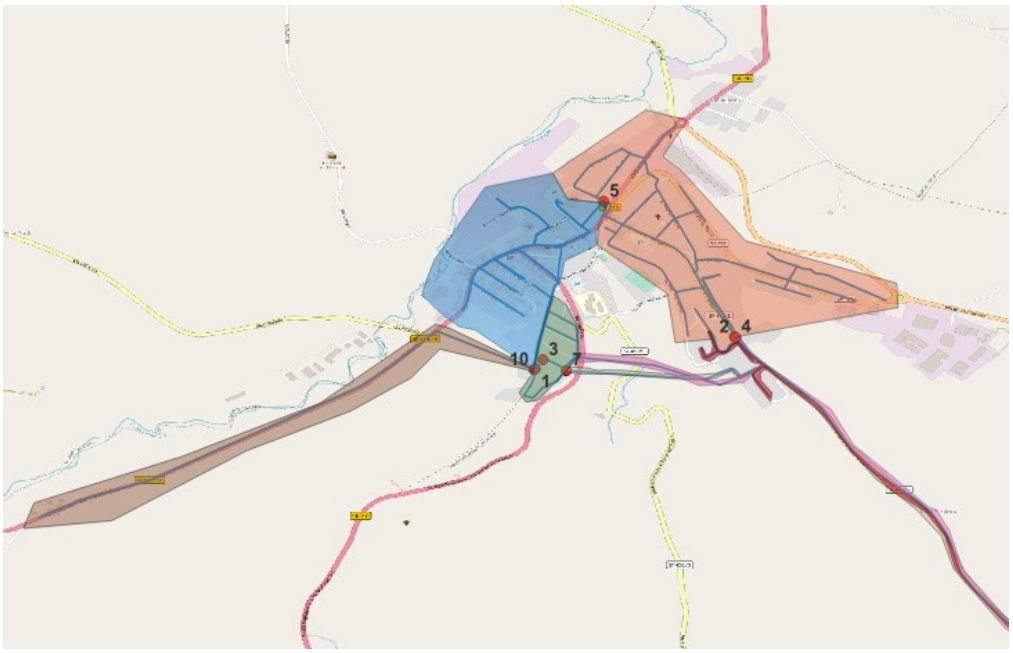


Figure 5.1 Map of the WDS.

Figure 5.1 shows a map of the WDS divided into its various DMAs, which included 955 users of different types and numbers, distributed as follows: 403 in DMA 1, 295 in DMA 2, 81 in DMA 3, 31 in DMA 4, 84 in DMA 5, and 61 in DMA 6.

As previously anticipated (*see Chapter 3),* after cleaning raw sensors data, the dataset reduced to 541 users. Since smart meters store cumulative values, it was necessary to transform them into effective hourly consumption by computing differences between consecutive cumulative values, enabling the analysis of daily and weekly behavioral patterns. To mitigate biases due to differences in household size or total demand, the data were normalized with Min–Max scaling, ensuring that clustering would capture similarities in usage patterns rather than consumption magnitudes. Preliminary data exploration was necessary to reconstruct user categories, since contractual information was unavailable. By exploiting geo-referenced data (longitude, latitude, and address), users were classified as residential or non-residential, with the latter accounting for only 60 cases, mainly related to commercial and service activities. This imbalance, while representative of real-world demand composition, required tailored strategies to ensure meaningful clustering.

Building on the methodological framework presented in Chapter 3, the analysis focused on extracting representative patterns of water consumption. Seasonal decomposition with MSTL was applied to highlight weekly variations between residential and non-residential users, and PCA further reduced dimensionality while preserving dominant structures.

Year	Month	Day	Hour	User1	…	UserN
2023	9	Monday	0	0.4774	…	0.5548
…	…	…	…	…	…	…
2024	5	Sunday	23	0.5995	…	0.4595

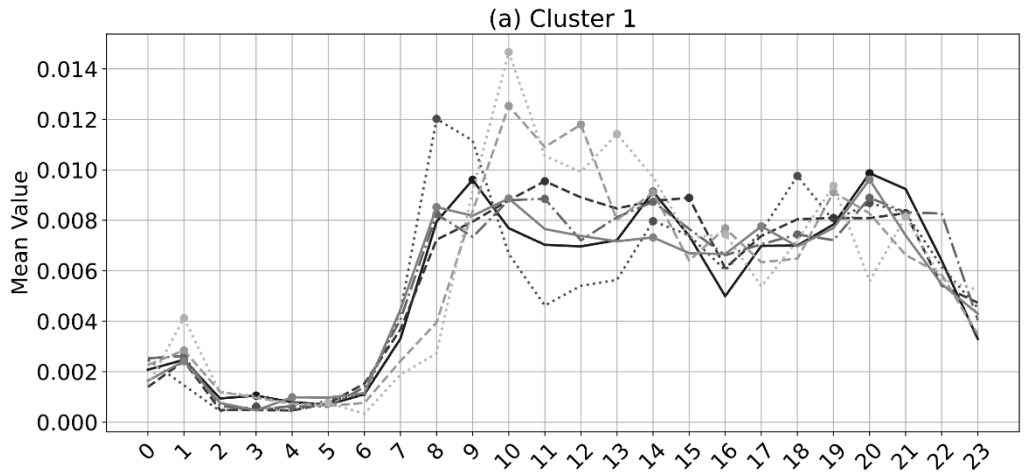
Table 5.III Example final dataset.

Table 5.III shows an example of the final dataset used as input for modelling, with dimensions 236 (*rows*) × 545 (*columns*). Clustering was then performed with optimized algorithms, leveraging bootstrap sampling to test stability and hyperparameter tuning to identify the best configurations. Two methods were compared: TSK-Means, which uses DTW to capture temporal distortions, and K-Shape, which relies on shape-based distance to align and cluster profiles by their overall structure.

In the following chapter, the results obtained from the clustering analysis are presented and critically discussed with reference to their implications for user profiling and anomaly detection in the WDS.

5.3 Results

The clustering experiments were first conducted on a balanced subset of 145 users, composed of 85 residential and 60 non-residential consumers randomly selected from the six DMAs [67]. After feature extraction with MSTL and dimensionality reduction through PCA, the optimized TSK-Means configuration yielded four stable clusters (K=4). Figure 5.2 shows the average weekly pattern of each cluster obtained.



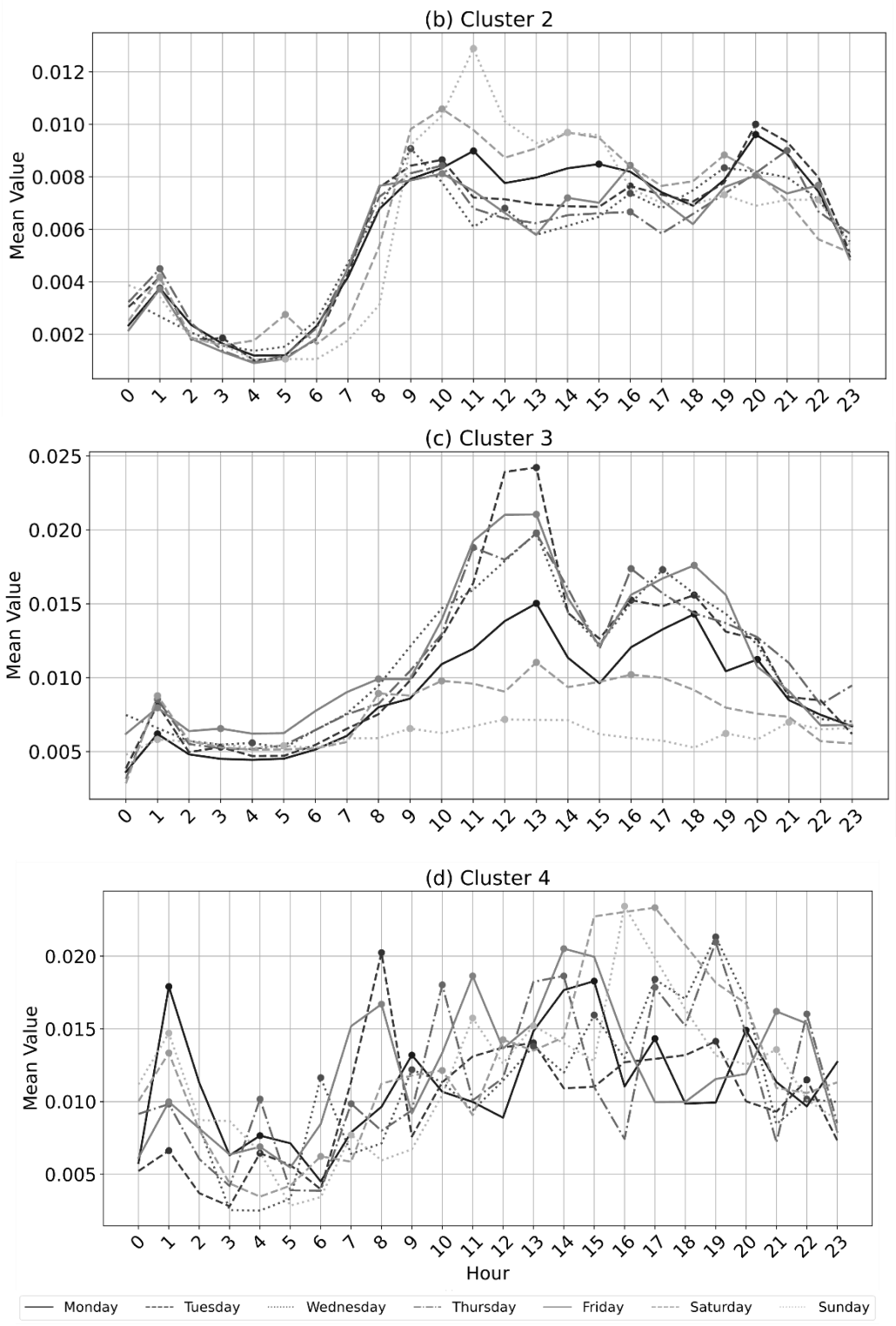


Figure 5.2 Average weekly pattern for each cluster.

Two of these clusters, Cluster 1 and Cluster 2, group residential users with characteristic daily peaks at lunchtime and in the evening, and with higher water demand on weekends, especially Sundays (*see Figure 5.2a and Figure 5.2b)*. The remaining two clusters, Cluster 3 and Cluster 4, represent non-residential consumers: one corresponding to commercial and service activities such as banks and garages, which showed concentrated weekday demand between morning and early evening (*see Figure 5.2c)*, and another grouping mixed or industrial users, whose

patterns diverged significantly from the others (*see Figure 5.2d)*. Validation with a new set of test users confirmed that the model was able to assign users to the correct category in most cases, while also flagging irregular consumption behaviors.

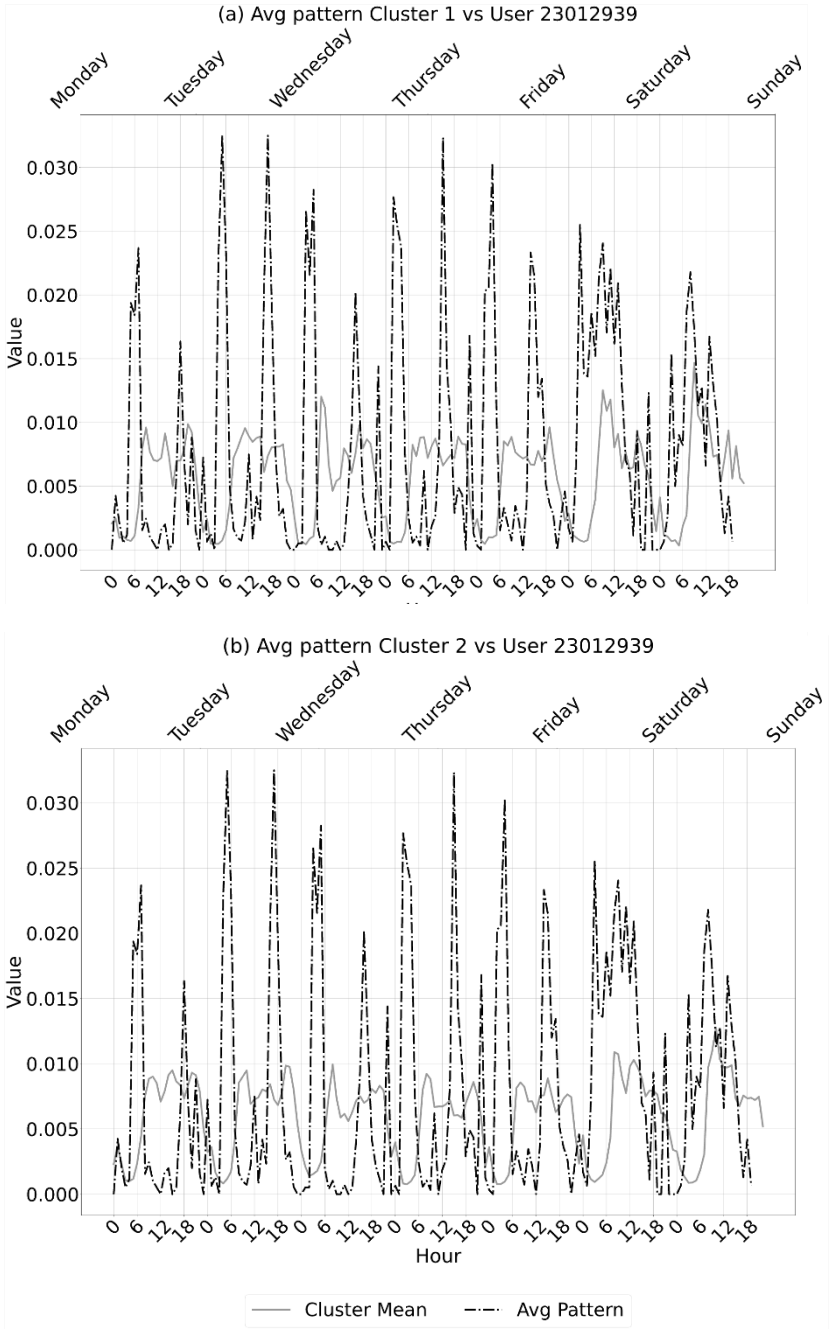
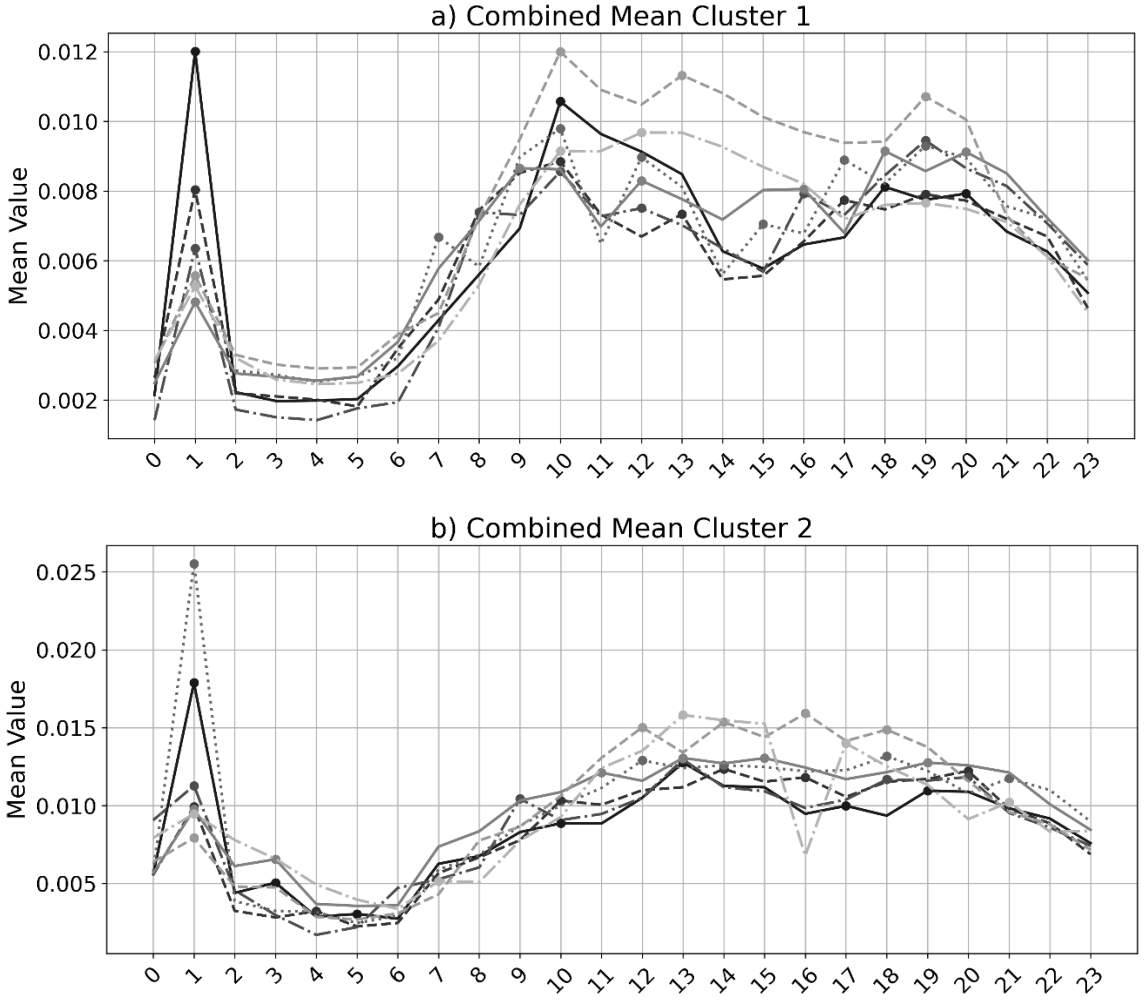


Figure 5.3 Example anomalous consumption pattern residential user.

Figure 5.3 shows an example of residential test user (23012939). Despite being labelled as residential through georeferenced information, the user was classified by the algorithm in Cluster 4. Indeed, the test user exhibits a completely different pattern compared to the typical residential patterns of Cluster 1 (*see Figure 5.3a*) and Cluster 2 (*see Figure 5.3b)*. Specifically, the water consumption of the test user never drops to zero at night, showing unusual peaks that may indicate irregular usage or the presence of a leakage.

Building on this first experiment, the analysis was extended to the full dataset of 541 users, including 481 residential and 60 non-residential consumers [68]. With the larger sample, both TSK-Means and K-Shape algorithms were optimized and compared.

For TSK-Means, the optimal configuration produced three clusters (K=3), each composed predominantly of residential users, though with varying proportions of non-residential ones. To mitigate the influence of this imbalance, different aggregation strategies were applied to derive representative cluster patterns, including simple mean, weighted mean, geometric mean, and up/down-sampling means, with the final representative profile obtained as a combined mean of all methods. In Figure 5.4 the final combined weekly patterns for each cluster are shown.



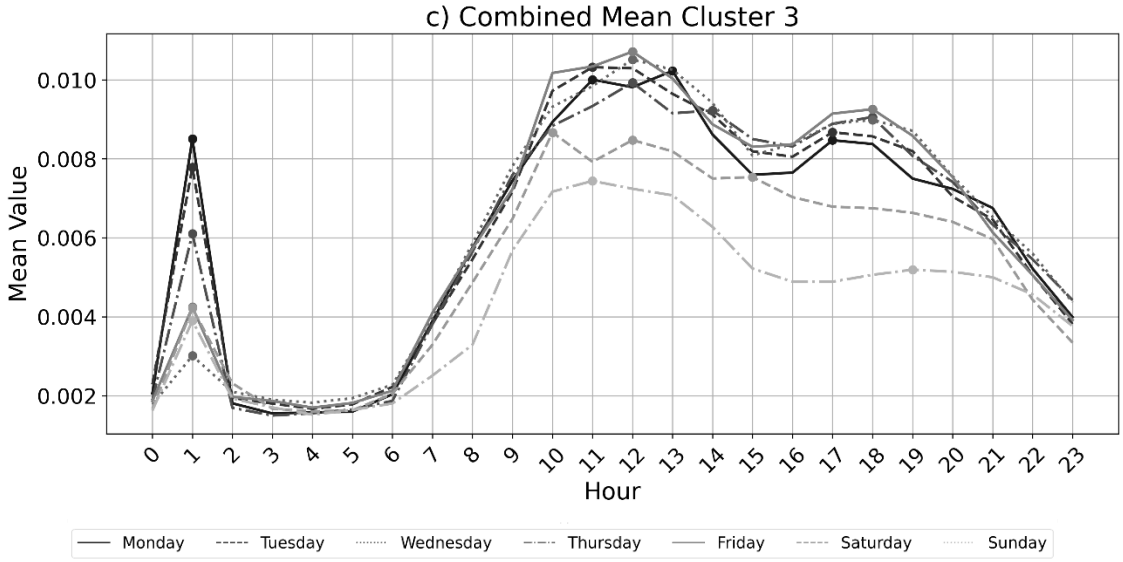


Figure 5.4 TSK-Means: Final combined average weekly pattern for each cluster.

The three resulting clusters captured distinct weekly dynamics. Figure 5.5 compares the average weekly patterns of residential vs non-residential users belonging to each cluster.

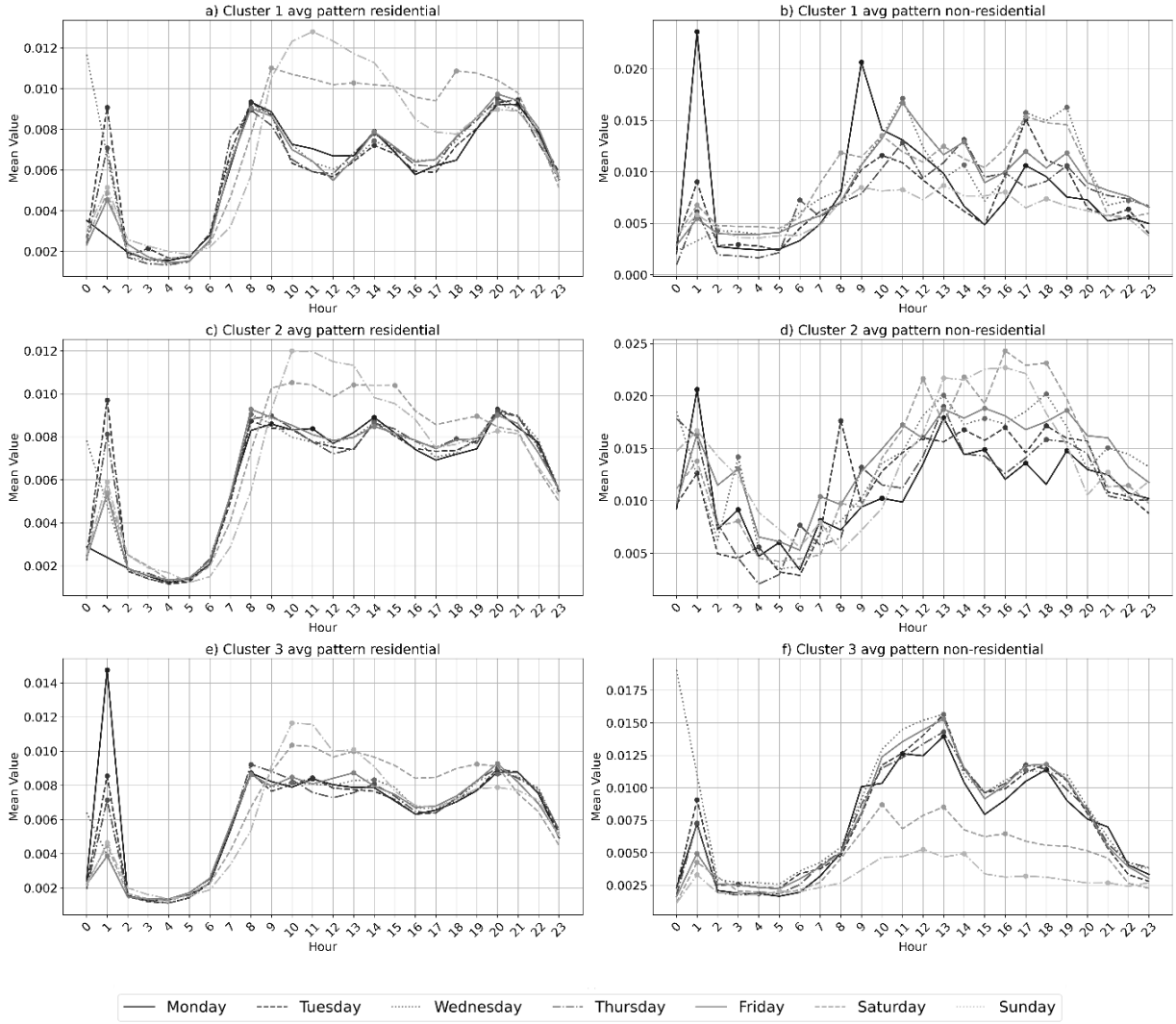
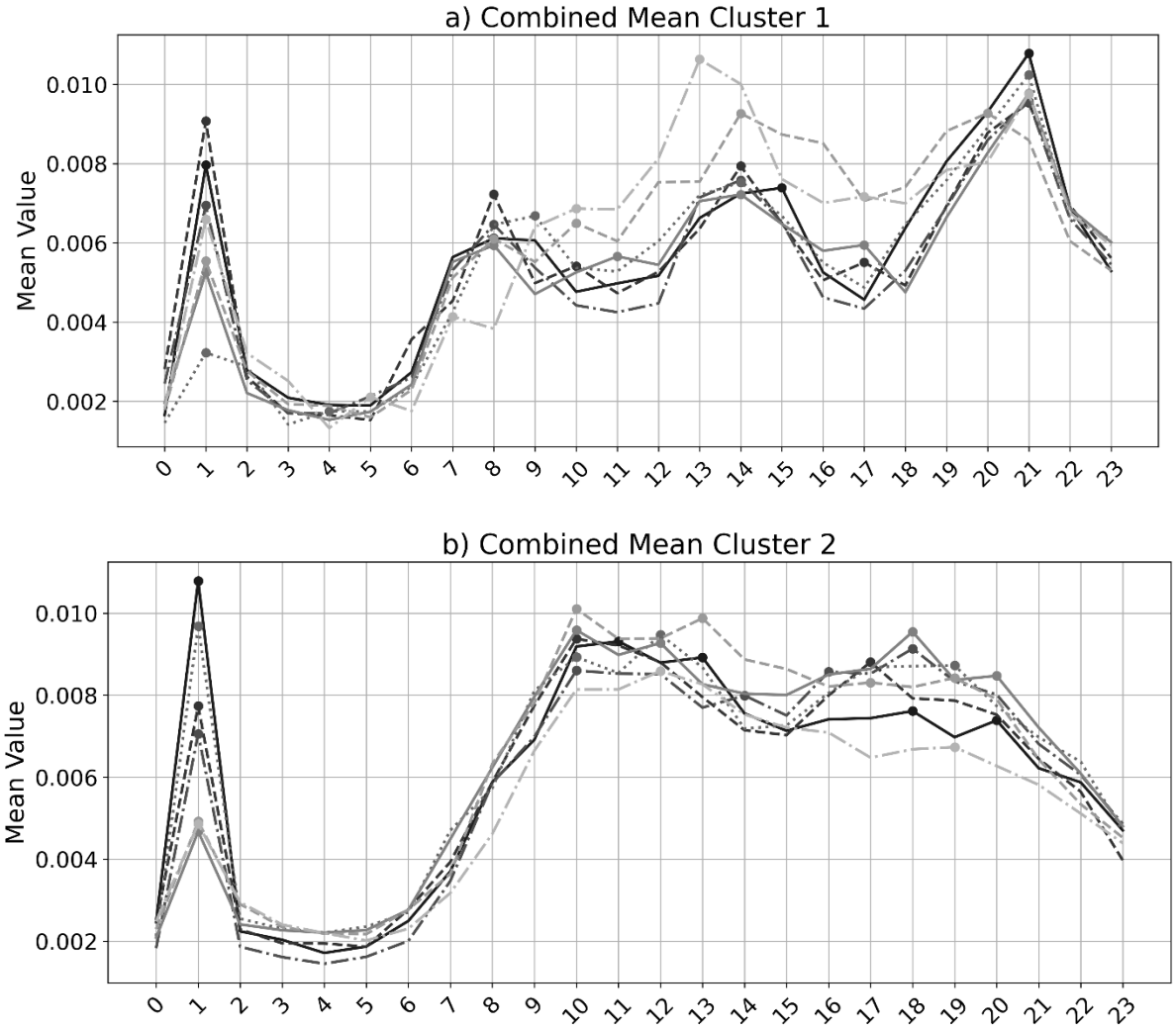


Figure 5.5 TSK-Means: Residential vs Non-residential average weekly pattern for each cluster.

As depicted in the plot, Cluster 1 is composed of a total of 89 users of which 10 non-residential and 79 residential. The water pattern of users belonging to this cluster is characterized by higher weekend consumption, with residential peaks in the morning, afternoon, and evening, and non-residential users showing steady use across all days (*see Figure 5.5a/b).* Next, Cluster 2 is made up by 216 users, 12 non-residential and 204 residential. In concerns, water has a similar demand on weekdays and weekends, where residential users consumed slightly more on weekdays, while non-residential users exhibited uniform activity driven by continuous industrial operations (*see Figure 5.5c/d).* Finally, the third cluster has 127 users, 26 non-residential and 101 residential. Its water demand pattern reflects a stronger weekday orientation, with residential users showing fewer but more intense peaks, and non-residential users active mainly during working hours with little weekend consumption (*see Figure 5.5e/f)*.

The K-Shape algorithm, applied within the same framework, also identified three clusters with structures broadly consistent with TSK-Means, though with subtle differences in the characterization of residential and non-residential demand. Figure 5.6 shows the final combined weekly patterns for each cluster yielded by K-Shape.



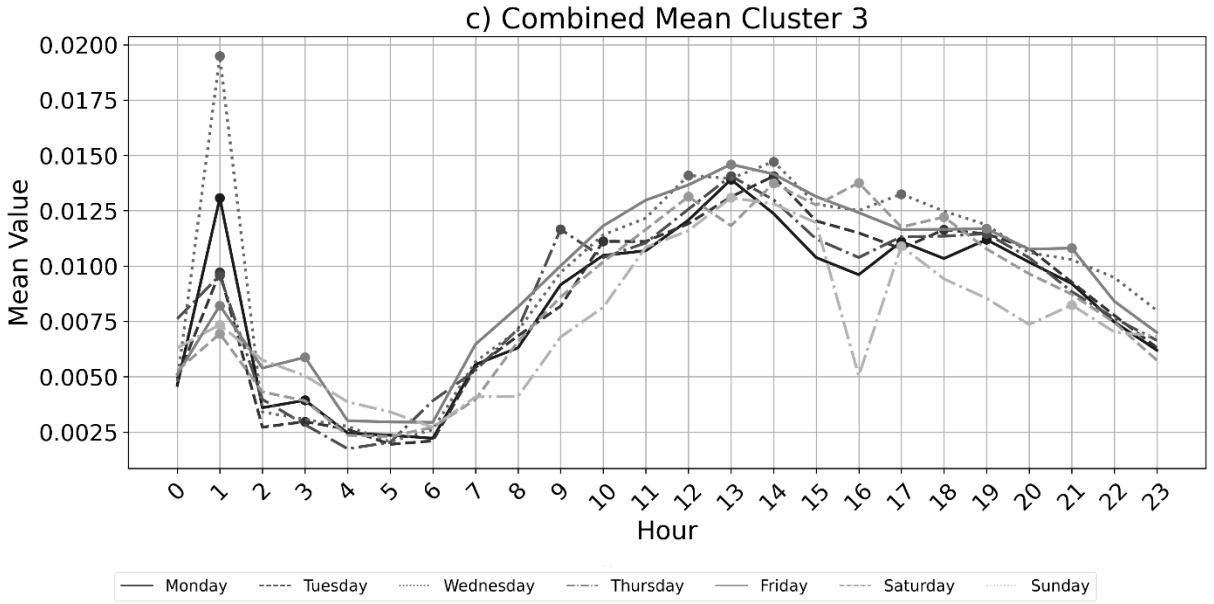


Figure 5.6 K-Shape: Final combined average weekly pattern for each cluster.

In Figure 5.7 the average weekly patterns of residential vs non-residential users belonging to each cluster are compared.

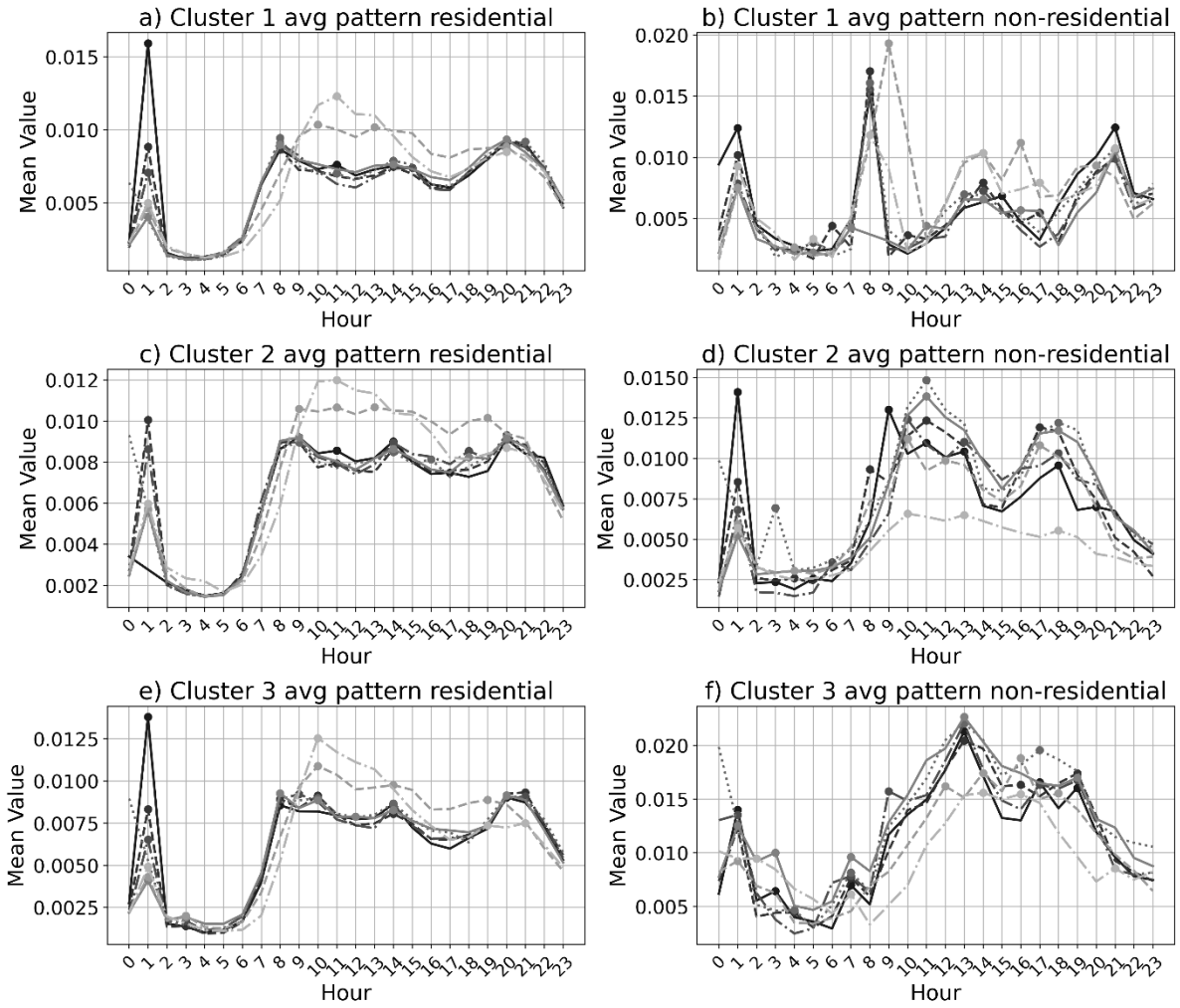


Figure 5.7 K-Shape: Residential vs non-residential average weekly pattern for each cluster.

Looking at the georeferenced information, it was found that each group is composed by the following users: (i) cluster 1 by 131 users – 2 non-residential, 129 residential; (ii) cluster 2 by 202 users – 29 non-residential, 173 residential; (iii) cluster 3 by 99 users – 17 non-residential and 82 residential.

Residential users in the first cluster displayed strong evening and morning peaks, slightly shifting between weekdays and weekends, while non-residential users maintained steady industrial consumption across all days (*see Figure 5.7a/b)*. In the second cluster, residential users consumed more on weekends compared to weekdays, with three modest peaks, while non-residential users reduced their activity on Sundays (*see Figure 5.7c/d)*. The third cluster highlighted weekday-dominant consumption, with residential users showing three well-defined peaks on working days but shifting to morning-dominated demand during weekends, and non-residential users displaying minimal variation across the week (*see Figure 5.7e/f)*.

Overall, both algorithms consistently identified three major profiles that aligned with the residential/non-residential divide and revealed finer distinctions in consumption behaviors. The comparison demonstrated that while TSK-Means effectively captured temporal distortions using DTW, K-Shape better emphasized the overall shape of consumption patterns. In both cases, the integration of geo-referenced validation confirmed that clusters corresponded to real user categories and activities, strengthening the interpretability and practical relevance of the results and making the proposed approach an effective decision-support tool for water utilities able not only to profile the water consumption patterns of user types but also to detect irregularities, possibly due to billing errors, leakages or even potential fraud in water consumption. By scaling the analysis to the full user base and comparing two advanced clustering algorithms, the extended study provided more robust and generalizable insights into water demand profiling, reinforcing the potential of these methods as decision-support tools for utilities to manage consumption diversity and detect anomalies such as leaks or irregular behaviors.

5.4 Conclusions

The results of this chapter confirm the potential of clustering methods for identifying meaningful consumption profiles in WDSs. By combining decomposition and dimensionality reduction with robust clustering techniques, it was possible to distinguish residential from non-residential demand patterns and to validate these groupings with geo-referenced data, ensuring practical significance. The comparative analysis of TSK-Means and K-Shape further showed how methodological choices influence the characterization of consumption behavior, with both algorithms capturing complementary aspects of water demand dynamics. Beyond methodological contributions, these findings demonstrate the value of clustering as a decision-support tool, enabling utilities to better understand user heterogeneity, detect anomalies, and design targeted management strategies.

Having established the effectiveness of clustering for profiling water consumption, the thesis now moves to a different but related application of ML in smart cities. The next chapter focuses on traffic flow forecasting, where predictive frameworks are applied to model and anticipate

mobility patterns. This extension highlights the versatility of ML in addressing diverse urban challenges, from the management of water demand to the optimization of transportation systems.

Chapter 6

Machine Learning for Urban Traffic Flow Estimation and Forecasting

Urban mobility is one of the central challenges of modern cities, as traffic congestion, air pollution, and inefficient transport systems significantly impact both quality of life and economic productivity. With the growth of urban populations and the increasing demand for sustainable transportation systems, accurate estimation and forecasting of traffic flows have become essential for effective traffic management, infrastructure planning, and real-time control strategies. Traditional approaches rely primarily on fixed sensors installed at strategic locations to monitor vehicle counts and speeds. While highly reliable, such infrastructures are costly to deploy and limited in spatial coverage, leaving many areas of the city unmonitored [73].

In recent years, the emergence of FCD, collected from GPS-enabled vehicles and mobile devices, has opened new opportunities for city-wide traffic monitoring. FCD provides broad spatial coverage but requires calibration against physical sensor data to ensure accuracy and reliability. By combining these two complementary sources—localized but precise fixed sensors, and large-scale but noisier FCD—cities can move toward a more holistic and data-driven view of urban traffic dynamics.

In this chapter, we introduce a ML–based framework for traffic flow estimation and forecasting that integrates fixed sensor data with FCD. The proposed approach consists of two phases. The first phase, Urban Flow Estimation, focuses on scaling and calibrating FCD against real sensor measurements to produce accurate city-wide traffic estimates. The second phase, Urban Flow Forecasting, leverages these enriched datasets to train predictive models capable of anticipating future traffic conditions across the urban network. By correlating heterogeneous data sources with spatio-temporal features such as time of day, road characteristics, and traffic patterns, our system aims to improve predictive performance while maintaining interpretability for decision-makers.

Building on the methodological principles established in the previous chapter—where clustering techniques were applied to uncover hidden patterns in water demand—this chapter extends the use of ML to another critical smart city domain: urban mobility. The ability to predict traffic flows with high accuracy not only supports traffic operators in preventing congestion and improving road safety but also provides valuable insights for long-term transport planning and sustainable mobility policies.

The remainder of this chapter is structured as follows: first, we describe approaches present in the current literature focusing on the novelty of the proposed ML-methodology. Next, a

description of the datasets and features employed in the case study, including details about the fixed sensors deployed in Catania and the FCD used for model calibration is given. We continue with the description of the architecture of the proposed methodology and the discussion of its implementation for both estimation and forecasting tasks. Finally, we evaluate the results, comparing the performance of the models and analyzing their implications for urban traffic management.

6.1 State of the Art

Traffic monitoring and forecasting have traditionally relied on fixed sensors, such as inductive loops or microwave counters, which provide highly reliable and precise vehicle counts. However, these infrastructures are costly to install and maintain, and their deployment is limited to selected road segments, leaving large parts of urban networks without direct observation [129]. To overcome this limitation, FCD, derived from GPS traces of vehicles or mobile devices, has gained prominence due to its broad spatial coverage and cost-effectiveness [130]. Nonetheless, FCD does not directly measure traffic volumes and requires careful calibration against sensor data to ensure it faithfully reflects actual conditions [131]. This has motivated a growing body of research on both the calibration of FCD and the forecasting of traffic flows.

Several methodologies have been developed to handle incomplete, noisy, or geographically sparse traffic data. Early approaches focused on interpolation-based methods, which estimate missing values using nearby temporal or pattern-based observations [132]. These are simple to implement but limited in their ability to capture random fluctuations and irregular events. Statistical learning-based approaches, relying on probabilistic assumptions about data distribution, generally improve accuracy but are difficult to apply when sensor coverage is sparse or historical datasets are insufficient [133]. More recent prediction-based methods have reframed missing data as a forecasting problem, applying models such as ARIMA [134], feed-forward neural networks [135], or other machine learning algorithms [136]. While effective in certain contexts, many of these approaches still require dense sensor networks or supplementary devices like cameras, leading to high costs and potential reliability issues [137].

A key advance has been the integration of limited sensor data with FCD. By leveraging the precision of sensor counts and the extensive spatial availability of FCD, these hybrid approaches provide scalable city-wide traffic flow estimation without installing new infrastructure [83]. Calibrating FCD against fixed sensors allows researchers to extrapolate traffic conditions to unmonitored road segments, capturing broader traffic dynamics and improving representativeness [131]. Similar concerns are addressed by [73], who emphasize the role of ML prediction models in extending monitoring capabilities to sensor-less roads. Compared to earlier methods, this integration avoids the costs and technical dependencies associated with deploying additional devices, while still delivering accurate and actionable insights [138].

Parallel to this, traffic flow forecasting has become central to urban mobility management. Forecasting models in literature generally fall into three categories. Statistical methods, such as the History Average Model [139] or ARIMA [134], remain useful for stable and predictable time

series but fail in non-linear, dynamic contexts. Machine learning methods, including regression models and boosting algorithms like XGBoost [138], LightGBM [136], and CatBoost [131], offer more flexibility by extracting patterns from historical data and handling complex relationships. DL techniques, particularly LSTM networks [140], ST-ResNet [141], and spatio-temporal graph convolutional networks [142], push the boundary further by modeling long-term dependencies and spatial interactions. These models achieve state-of-the-art accuracy but demand substantial amounts of training data and computational resources, which can be a barrier in ma ny real-world scenarios [135].

Despite their promise, many forecasting frameworks in the literature remain heavily dependent on dense sensor installations, using spatio-temporal correlations between monitored road segments to infer flows in unmonitored ones [129]. This assumption limits their scalability to cities where only a subset of roads is equipped with sensors. To address this, two-level machine learning approaches have emerged. In these frameworks, clustering algorithms are used to group roads or demand profiles with similar traffic behaviors, while supervised forecasting models are trained for each cluster, improving precision and robustness [137]. Although promising, many such studies remain constrained to highways or simplified networks and do not fully integrate heterogeneous data sources [83].

The novelty of the present work lies in bridging these two research streams—data calibration and traffic flow forecasting—within a unified machine learning framework. First, unlike most existing approaches that either rely solely on sensors or treat FCD as auxiliary information, the proposed methodology directly integrates FCD and fixed sensor data, calibrating and scaling them to ensure coherence and reliability across the urban network. This step allows the exploitation of the extensive spatial coverage of FCD data without sacrificing the precision of sensor counts. Second, the forecasting phase adopts a two-level learning strategy: clustering identifies roads or patterns with similar behaviors, while tailored prediction models are trained for each cluster, enhancing interpretability and predictive accuracy. Importantly, the approach is not limited to a specific road type (e.g., highways) but has been designed to handle heterogeneous urban scenarios, including downtown and suburban roads, thereby ensuring greater generalizability and transferability across cities.

In summary, the reviewed literature highlights both the progress and the limitations of existing calibration and forecasting methods, particularly their dependence on dense sensor networks, additional devices, or scenario-specific features. The framework introduced in this work advances the state of the art by combining the precision of fixed sensors with the coverage of FCD, while embedding them into a robust two-level machine learning architecture. In the following section, we describe in detail the methodology developed and applied to the case study of Catania, illustrating how this integrated approach enables accurate urban traffic estimation and forecasting across heterogeneous road networks.

6.2 Case Study

Urban mobility represents one of the most pressing challenges for modern cities, particularly in medium-sized Mediterranean contexts such as Catania, Italy, where congestion, air pollution,

and infrastructural limitations converge to create a critical demand for intelligent management of traffic flows. The metropolitan area of Catania, home to roughly 300,000 residents and over 750,000 inhabitants when considering its surrounding municipalities, is characterized by an increasing dependence on private vehicles and a sprawling settlement pattern extending beyond the municipal boundaries. This socio-spatial configuration has generated high traffic demand directed toward the central areas, resulting in recurrent daily congestion, delays, and amplified environmental externalities. As highlighted by [73], one of the most pressing challenges in urban traffic management is the lack of direct monitoring on large portions of city road networks, often referred to as *sensor-less* roads. Without reliable traffic counts, decision-makers struggle to obtain accurate information for both real-time control and long-term planning. This issue is particularly evident in Catania, where only 14 microwave counters are deployed across a complex and congested metropolitan road network. Against this background, the integration of ML methodologies into traffic monitoring and forecasting represents a promising approach for enabling data-driven decision-making in the pursuit of sustainable urban mobility.

The case study under consideration proposes an innovative ML-based framework for urban flow estimation and forecasting, leveraging heterogeneous data sources to provide reliable short-term predictions of traffic volumes across the city.

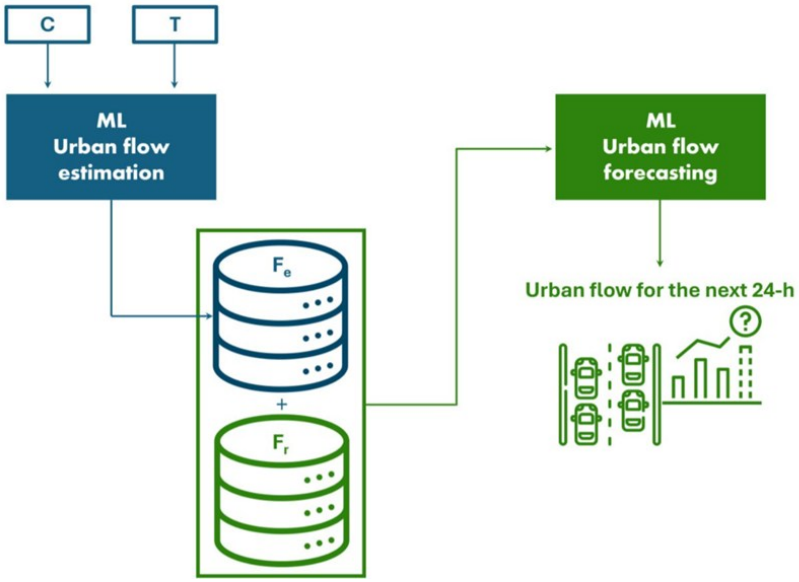


Figure 6.1 Architecture Urban Flow Estimation/Forecasting.

The system architecture, schematically represented in Figure 6.1, integrates two main stages:

i. an urban flow estimation module, which combines data from fixed microwave traffic sensors *(C)* with FCD *(T)* to extrapolate accurate vehicle counts across the network, including segments without sensor coverage, and
ii. an urban flow forecasting module, which employs clustering and supervised learning models to predict traffic conditions for the following 24 hours.

This two-tiered methodology bridges the gap between limited sensor-based observations and the need for large-scale, high-resolution traffic information essential for effective planning and control.

The estimation stage draws upon two complementary datasets. The first consists of sensor-based vehicle counts, collected through MobilTraf300 microwave counters strategically deployed across 14 zones within the city.

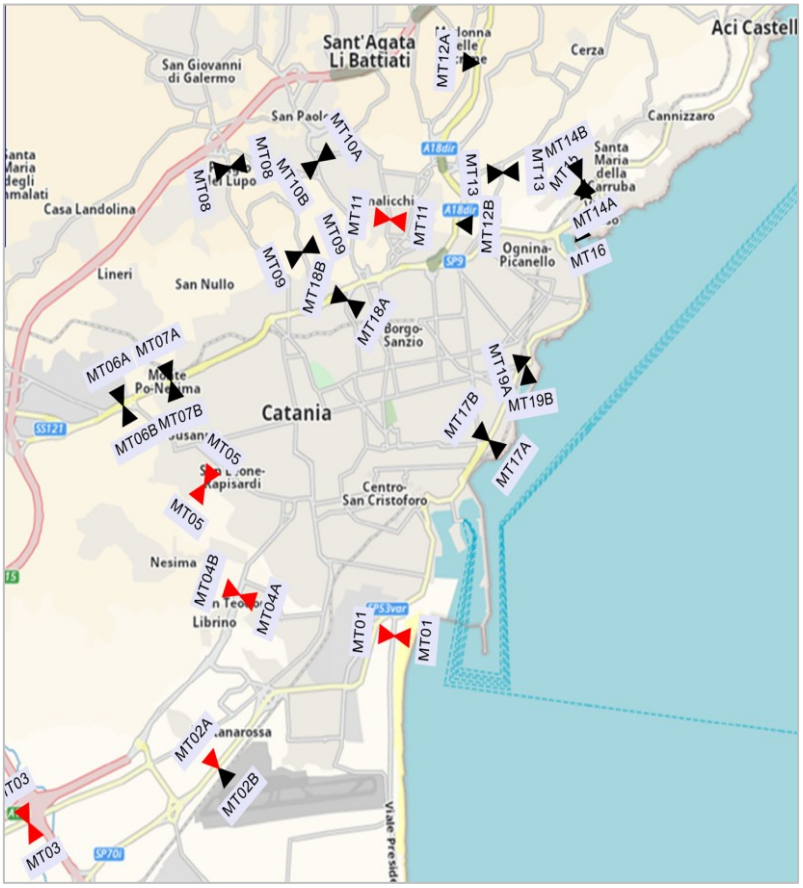


Figure 6.2 Urban area of Catania - in black operational traffic counters.

Figure 6.2 shows a map of the operational traffic counters located across the urban area of Catania. Traffic counters operate autonomously and internally, comprise sensors, controlling electronics and a hybrid power supply system consisting of batteries and photovoltaic panels. These devices, integrated into the Laboratory of Sustainable Mobility at the University of Catania, monitor up to two lanes per direction, capturing precise passage counts and traffic directions at five-minute intervals. Their main advantage lies in the accuracy and reliability of direct traffic measurement, although their spatial coverage is limited to the fixed points where sensors are installed.

The second dataset derives from FCD acquired through the TomTom Traffic Stats portal [143]. This source aggregates anonymized GPS traces from connected vehicles, providing large-scale, georeferenced information on average travel times, speed percentiles, congestion indices, and directional flows. Road segments in the FCD dataset are designed to ensure homogeneity in traffic conditions, with paired inbound and outbound segments allowing the inclusion of

contextual features in the estimation and forecasting models. Together, these datasets offer a unique opportunity: the sensors anchor the analysis with precise counts, while FCD extends spatial granularity to areas beyond direct measurement.

To ensure comparability between these heterogeneous sources, a rigorous data preprocessing phase was implemented. As briefly described in Chapter 3, sensor readings were aggregated by road category (single-lane, two-lane same direction, two-lane opposite direction), normalized to 15-minute intervals to align with FCD resolution, and subsequently aggregated into hourly counts for forecasting tasks. Missing values, due to sensor malfunctions or abnormal events such as extreme weather conditions, were treated through time-based imputation methods, while outliers were identified via boxplots and the IQR method. Furthermore, the preprocessing integrated auxiliary contextual variables—including weather conditions, temporal indicators (weekday, day of month, time of day), geometric features (lane width, number of lanes, parking presence), and dynamic traffic descriptors (harmonic average speed, 15th and 85th percentile speeds). This enriched dataset provided the foundation for robust model training by incorporating both infrastructural and temporal dimensions of urban traffic dynamics.

Building on this data foundation, the forecasting stage adopted a two-level machine learning strategy.

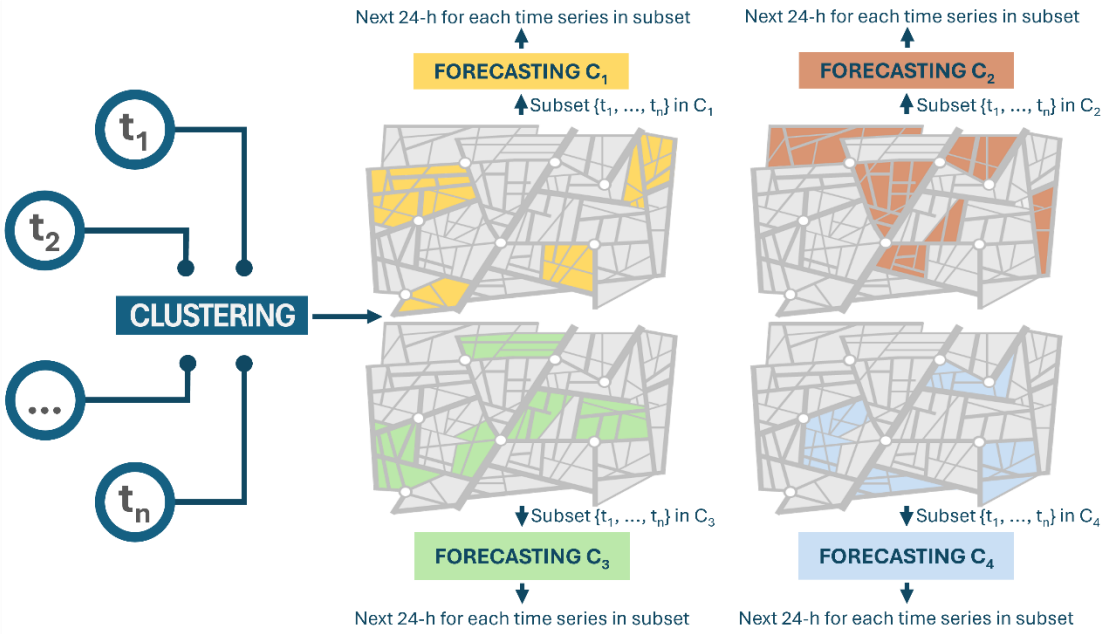


Figure 6.3 Two-level ML forecasting strategy schema.

Figure 6.3 illustrates the two-level ML strategy adopted. On the left, individual time series $(t_1,t_2,\ldots, t_n)$ represent traffic flow data collected from different road segments. Instead of training a single forecasting model for all series, in the first level, unsupervised learning techniques were applied to extract structural patterns within the traffic time series. Specifically, the TSK-Means was employed, an adaptation of the traditional K-means that incorporates temporal aspects into the clustering process by using DTW as a distance metric. This clustering step enabled the grouping of road segments exhibiting similar traffic profiles, irrespective of their absolute volume scales. By normalizing the series through Min-Max scaling, clustering was driven by

shape similarity rather than magnitude, ensuring that patterns in traffic dynamics (e.g., rush-hour peaks, weekend troughs) were preserved. Determination of the optimal number of clusters was achieved via silhouette scoring, providing a quantitative evaluation of clustering quality.

Once clusters were identified, as shown on the right of Figure 6.3, the second level involved training supervised forecasting models within each cluster. The rationale behind this hierarchical structure lies in the ability to exploit transfer learning: once a model is trained on a group of roads sharing similar dynamics, it can be effectively applied to forecast traffic on new or scarcely monitored segments belonging to the same cluster. This approach not only improves generalization but also reduces the reliance on extensive training data from every single road segment, a significant advantage in contexts where sensor infrastructure is sparse.

The novelty of this approach resides in its hybridization of data sources and methods. By merging fixed sensor data with FCD, the framework overcomes the intrinsic limitations of each source, creating a richer and more representative dataset. The clustering-based transfer learning strategy introduces scalability, enabling reliable forecasts even for roads with minimal observations, while the use of advanced machine learning models ensures strong performance across diverse contexts. This methodological combination not only addresses the challenges of traffic forecasting in Catania but also provides a replicable framework for other cities seeking to improve their traffic management under constrained sensor deployments.

In conclusion, the proposed ML-based system demonstrates how integrating heterogeneous data and advanced predictive techniques can produce accurate and scalable forecasts of urban flows. By combining the precision of fixed sensors with the spatial coverage of FCD and embedding them into a two-level machine learning framework, the methodology effectively addresses the limitations of traditional approaches that rely exclusively on dense sensor networks or scenario-specific features. This hybrid strategy ensures robust traffic estimation and forecasting across heterogeneous urban contexts, from arterial roads to suburban streets, offering a decision-support tool for both real-time traffic management and long-term planning.

The following section presents the results of the forecasting experiments, where the performance of the proposed approach is quantitatively assessed. Through comparative evaluation against alternative models, we highlight the predictive accuracy, robustness, and practical utility of the framework when applied to the urban traffic network of Catania.

6.3 Results

The preliminary study [70] served as a proof of concept, showing how clustering roads with similar traffic patterns and training cluster-specific forecasting models could improve short-term traffic prediction. With a dataset covering three months data — January 1, 2023 to March 31, 2023 — and only a limited number of sensors, the clustering procedure yielded three groups of time series with a silhouette score of 0.48, considered reasonably good. Forecasting was then performed using several machine learning algorithms, with CatBoost emerging as the best performer across all evaluation metrics (e.g., SMAPE ≈ 25.4, MAE ≈ 0.065, RMSE ≈ 0.094).

Despite the reduced dataset, the approach demonstrated the potential of combining clustering with transfer learning to generalize across unobserved roads.

The extended works [71][72] significantly scaled this analysis. Using a full year of data — January 1, 2022 to December 31, 2022 — from 14 sensors installed across the city of Catania, the clustering procedure achieved a silhouette score of 0.52. This outcome is considered quite favorable, suggesting that the clusters are adequately separated, though not entirely distinct. It is also noteworthy that a team of experts in the field has verified the efficacy of the clustering algorithm in grouping roads with similar structural characteristics.

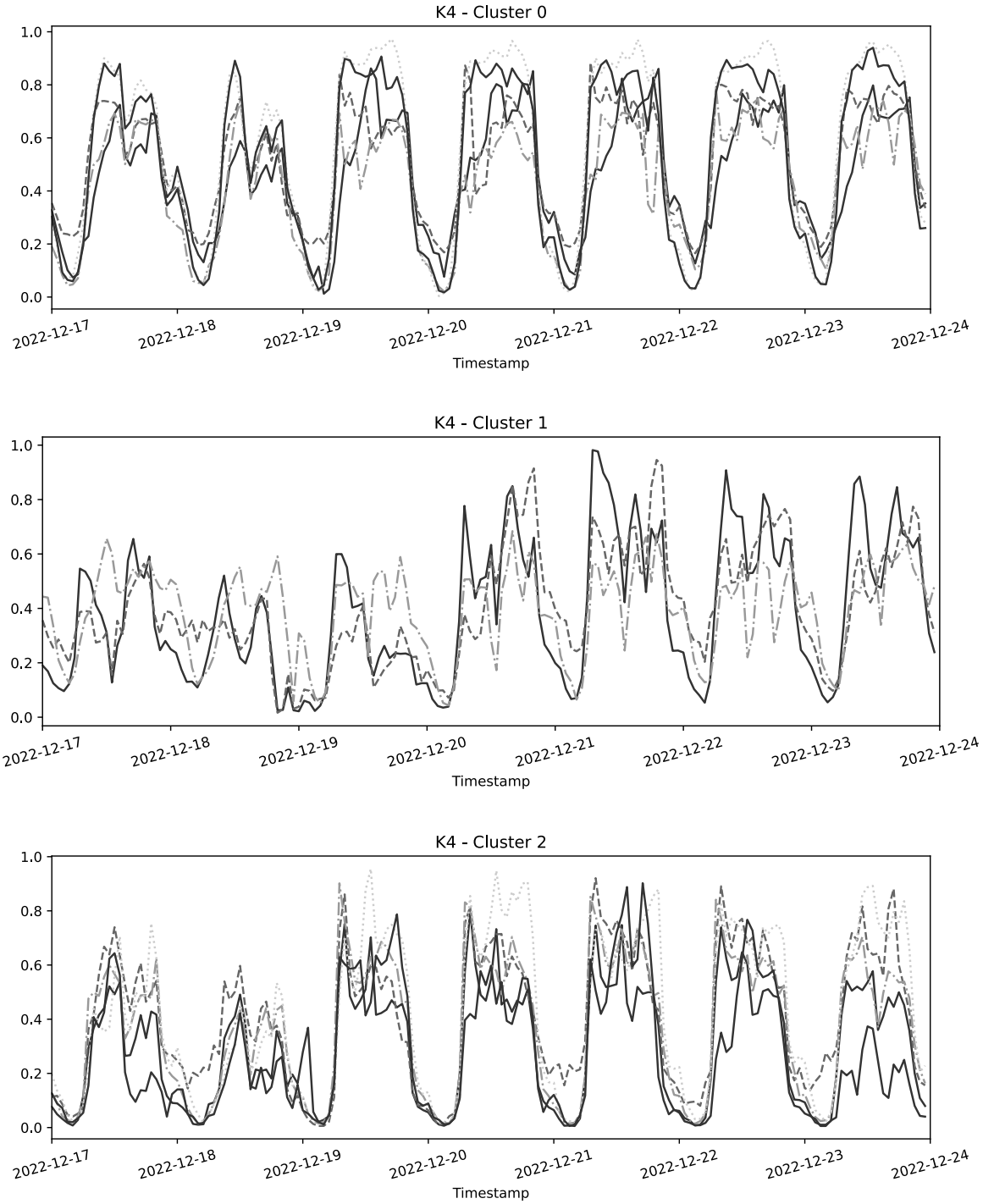


Figure 6.4 Example one-week pattern sensors divided in three clusters.

Figure 6.4 provides a visual representation of the time series and their patterns for each cluster. Each curve refers to a sensor of a particular road in Catania. Please note that, due to space constraints, the figure only represents data from last week, even though an entire year of data as input. Upon observation, it becomes apparent that the first cluster comprises 5 sensors, the second cluster consists of 3sensors, and the third cluster encompasses 5 sensors. The fourth cluster, comprising only one time series, was omitted from the visual representation due to its singular nature.

Keeping in mind the outcomes of the clustering step, forecasting was implemented. First, the dataset was divided into two sets: a training set going from January 1st, 2022, to December 16th, 2022, and a test set consisting of two weeks of observations, spanning from December 17th, 2022, to December 31st, 2022. This choice was influenced by the fact that in Italy last week of December is Christmas week. The traffic patterns in Catania during this period deviate from the usual, particularly impacting on certain roads that are notably affected by the onset of the Christmas holidays.

The forecasting phase expanded the comparative evaluation of models. A suite of ML algorithms—including gradient boosting (CatBoost, XGBoost, LightGBM), Random Forests, and DL models— was tested. Boosting methods consistently achieved the best results. During the regular week of December 17th –23rd, 2022, the best-performing models obtained average SMAPE values of ≈13.2 for Cluster 1, ≈18.2 for Cluster 2, and ≈28.5 for Cluster 3, confirming strong predictive accuracy, particularly for the first group. CatBoost and XGBoost both ranked as top performers, with XGBoost showing slightly greater resilience when abnormal patterns emerged. Indeed, during the Christmas week December 24th–31st, when universities closed and travel dynamics shifted substantially, error rates increased across all clusters, with SMAPE average values equals to ≈18.2 for Cluster 1, ≈27.0 for Cluster 2, and ≈33.6 for Cluster 3. The impact was particularly pronounced on sensors near university-related traffic, where demand fluctuations were sharper.

Another novel contribution of the extended works lies in the leave-one-sensor-out validation strategy. For each cluster, models were trained excluding one sensor and tested on the omitted time series. Despite being exposed only to similar but not identical traffic profiles, the models achieved relatively low errors: for example, during the normal week, excluded sensors in Cluster 1 showed SMAPE ≈ 15.6, while Cluster 2 sensors reached ≈ 33.3, and Cluster 3 sensors ≈ 32.9. As expected, errors increased during the holiday week (e.g., Cluster 1 ≈ 21.1, Cluster 3 ≈ 39.6), but the forecasts still captured general dynamics, even without having seen such atypical patterns during training. This demonstrates the generalization capability of the framework, as it requires only a minimal number of new observations (as low as ~15 days of input data) to generate credible forecasts for unseen roads.

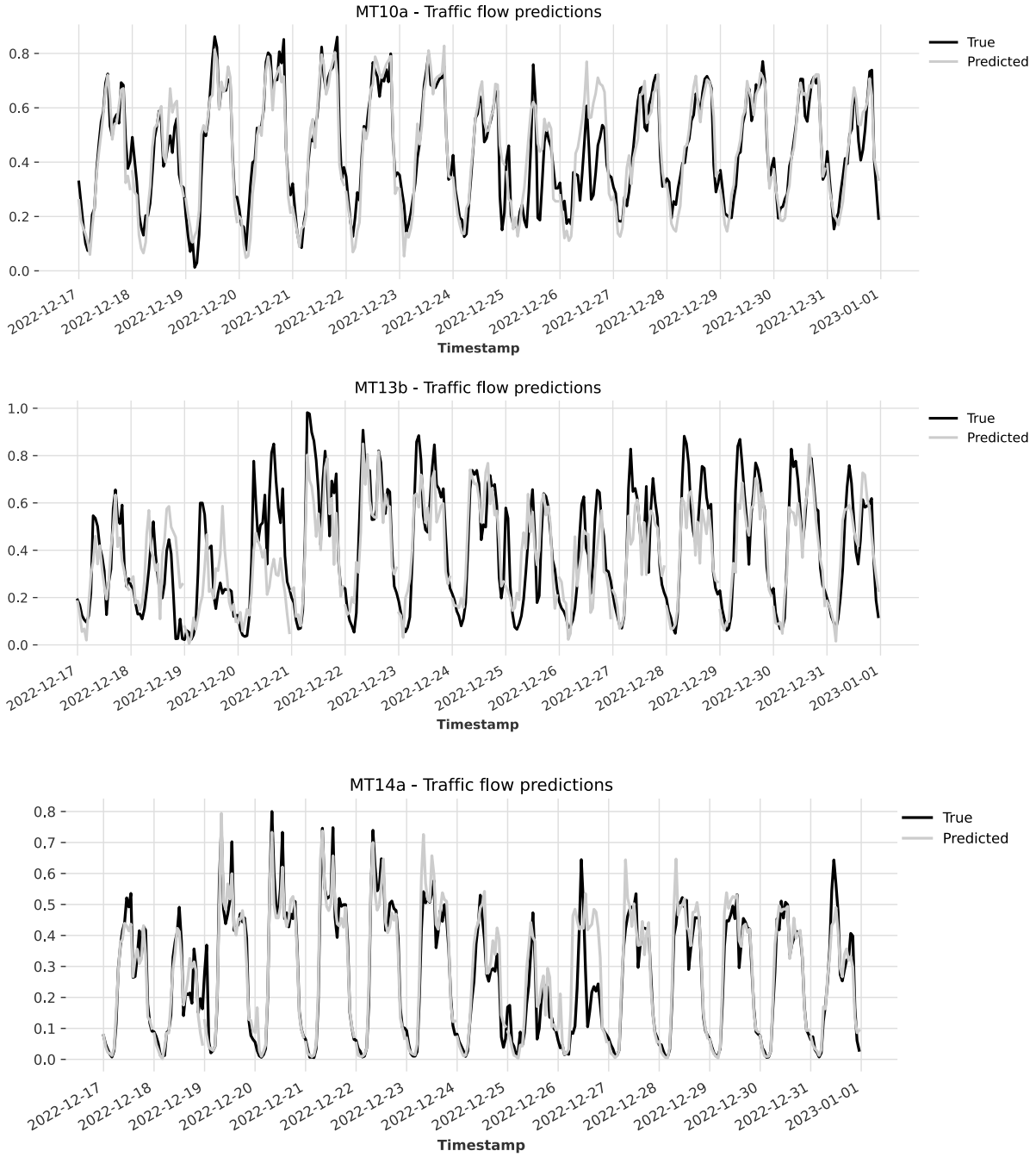


Figure 6.5 Example test on excluded time series for the two weeks.

Figure 6.5 presents the actual and predicted values of traffic flow in some time series excluded by each cluster on the two test weeks spanning from 17th to 23rd December, 2022, and on Christmas week.

As can be seen from the graphs, for these three tested excluded sensors the model was able to predict quite accurately the traffic flow registered in the week from 17th to 23rd December 2022. Instead, as concerns the holiday week for the majority of the sensors, the model accuracy slightly decreased when predicting the traffic flow on the Christmas days (25th and 26th December, 2022).

These findings resonate with the work in [73], who also emphasized the importance of providing accurate predictions for “sensor-less” roads in order to strengthen urban traffic management. While their study highlighted the potential of machine learning models to compensate for limited sensor infrastructure, the present work extends these contributions by introducing two methodological advances: first, the calibration and scaling of FCD against fixed sensors, ensuring data coherence and reliability across the network; and second, a two-level strategy combining clustering with boosting-based forecasting, which improves both predictive accuracy and robustness.

Taking together, these results demonstrate how the framework not only reproduces traffic dynamics on monitored roads but also generalizes effectively to unmonitored segments, even under irregular demand conditions such as holiday periods. Indeed, clustering enhances interpretability and reduces variance, while boosting-based forecasting models deliver state-of-the-art accuracy in both normal and abnormal conditions. Compared with the preliminary study, the extended analyses confirm the robustness of the approach on larger datasets, highlight its resilience to unusual temporal dynamics, and show its practical utility in scaling traffic forecasting beyond sensor-covered roads. These findings underscore the value of the framework for smart city applications, offering a cost-effective and transferable methodology for traffic management and urban mobility planning.

6.4 Conclusions

The results of this chapter highlight the effectiveness of integrating fixed sensors with FCD through a ML framework that combines calibration with cluster-based forecasting. By merging the precision of ground sensors with the spatial coverage of FCD, the proposed approach achieved accurate and robust traffic flow predictions across diverse urban scenarios, even under irregular conditions such as holiday-related demand shifts.

The two-level learning strategy, in which clustering precedes forecasting, enhanced both interpretability and generalization, allowing the framework to adapt to diverse traffic dynamics without requiring dense sensor installations. These contributions underline the scalability and cost-effectiveness of the methodology, pointing to its potential as a reliable tool for future intelligent transportation systems.

Building on this, the next chapter extends the investigation of predictive analytics in smart cities from traffic flow forecasting to air quality prediction, focusing on how domain-adversarial learning can enable cross-city generalization. This approach addresses the challenge of transferring predictive models across urban areas with different monitoring infrastructures and pollution dynamics, aiming to improve scalability and applicability of air quality forecasting systems.

Chapter 7

Towards Transferable Air Quality Prediction: A Domain-Adversarial Framework for Cross-City Generalization

Urban air quality has emerged as a critical concern in the pursuit of sustainable and livable cities, as rising urbanization intensifies the pressure on public health, mobility systems, and the environment. Pollutants such as nitrogen dioxide (NO_2), fine particulate matter ($PM_{2.5}$), and coarse particulate matter (PM_{10}) are closely linked to adverse health outcomes and ecological impacts, including climate change and excessive greenhouse gas emissions. Given that road traffic remains a primary contributor to deteriorating air quality, the issue is inseparably connected to broader urban sustainability challenges such as congestion, energy consumption, and mobility planning. Tackling these intertwined problems requires advanced predictive frameworks capable of modeling the complex, nonlinear interactions among emissions, meteorological factors, and urban morphology.

ML has become a powerful tool in this domain, enabling the extraction of patterns from large-scale, heterogeneous urban data collected via fixed monitoring stations, remote sensing technologies, and mobile devices. Despite these advances, most studies have focused on building highly accurate models tailored to individual cities. While effective in localized contexts, such city-specific models face a fundamental scalability issue: training and maintaining separate models for every urban area is both costly and impractical, particularly in cities with sparse monitoring infrastructure or limited historical data.

To address this gap, this chapter introduces a domain-adversarial learning framework designed to enhance cross-city generalization of air quality prediction. The proposed methodology combines a DANN model with ML regressors to extract domain-invariant features that capture structural similarities across cities. By transferring knowledge from a source city to target cities with analogous urban characteristics, the framework reduces the need for extensive retraining and ensures scalability. The approach is evaluated using real-world sensor data from Paris, Madrid, Berlin, and Helsinki, focusing on the prediction of NO_2, $PM_{2.5}$, and PM_{10} concentrations. Results demonstrate that the proposed method achieves reliable performance across different urban environments, offering a transferable, data-efficient, and sustainable solution for urban air quality management.

7.1 State of the Art

This section reviews existing approaches that address the complex, nonlinear relationship between traffic and air pollution, with a particular focus on methods for transferring predictive models across different contexts. Monitoring urban pollution based on traffic dynamics remains a significant challenge. Traditional approaches—ranging from deterministic atmospheric dispersion models to statistical time series techniques such as ARIMA [145] and SARIMA [146]—have contributed to scientific rigor but are hindered by strong simplifying assumptions, high computational requirements, and limited ability to capture the inherent complexity of environmental systems. To overcome these drawbacks, ML and DL methods have increasingly been applied, showing strong performance in modeling spatio-temporal dependencies. Classical ML algorithms such as Random Forests [147] and XGBoost [138] have consistently outperformed traditional statistical models, while DL techniques further improved accuracy by learning intricate patterns from large-scale environmental datasets [49].

Despite these advances, the majority of predictive models are still developed for individual pollutants or tailored to specific cities, which limits their scalability and adaptability. Transfer learning has emerged as a promising strategy to address these limitations by reusing knowledge across locations, pollutants, or tasks. For instance, [148] proposed a multi-source ensemble-based transfer learning framework, where models trained on data from several monitoring stations are combined to improve both accuracy and stability. Similarly, RegionTrans, developed by [149] is a deep spatio-temporal transfer learning framework that aligns regional features across cities using auxiliary data and adversarial domain adaptation, effectively transferring knowledge from data-rich to data-scarce environments. Other approaches have explored pollutant-level transfer. For example, [51] introduced a neural architecture first trained on $PM_{2.5}$ prediction and then fine-tuned for NO_2 and CO, leveraging shared pollutant dependencies to enhance performance under data scarcity. Cross-city transfer learning has also been studied: [150] evaluated strategies for PM_{10} prediction across Graz and Zagreb, showing that temporal and seasonal alignment between datasets plays a critical role in achieving robust model adaptability, even when cities differ in climatic conditions.

While these works highlight the potential of transfer learning to address challenges such as data sparsity, pollutant coverage, and regional heterogeneity, important gaps remain. Many approaches are limited to a single pollutant or location and require fine-tuning before deployment in new contexts, restricting their applicability in real-world zero-shot scenarios. In such cases, models must generalize directly to unseen cities or pollutants without retraining, which requires robust generalization across geographic, temporal, and pollutant dimensions. Critically, no unified framework has yet emerged that can simultaneously handle multi-pollutant forecasting across multiple cities in zero-shot settings.

To bridge this gap, in [69] the authors of the thesis introduced a structured transfer learning pipeline designed for zero-shot, multi-city, and multi-pollutant prediction. Unlike prior studies, the proposed approach explicitly incorporates city-level similarity through diverse urban features and employs a domain-adversarial neural network to learn invariant, transferable representations. These representations are then leveraged by a multi-target predictor for NO_2,

$PM_{2.5}$, and PM_{10} concentrations, enabling scalable and generalizable air quality forecasting across heterogeneous urban environments.

7.2 Case Study

The case study investigates the transferability of air quality prediction models across urban environments by applying a structured methodology that leverages similarities between cities.

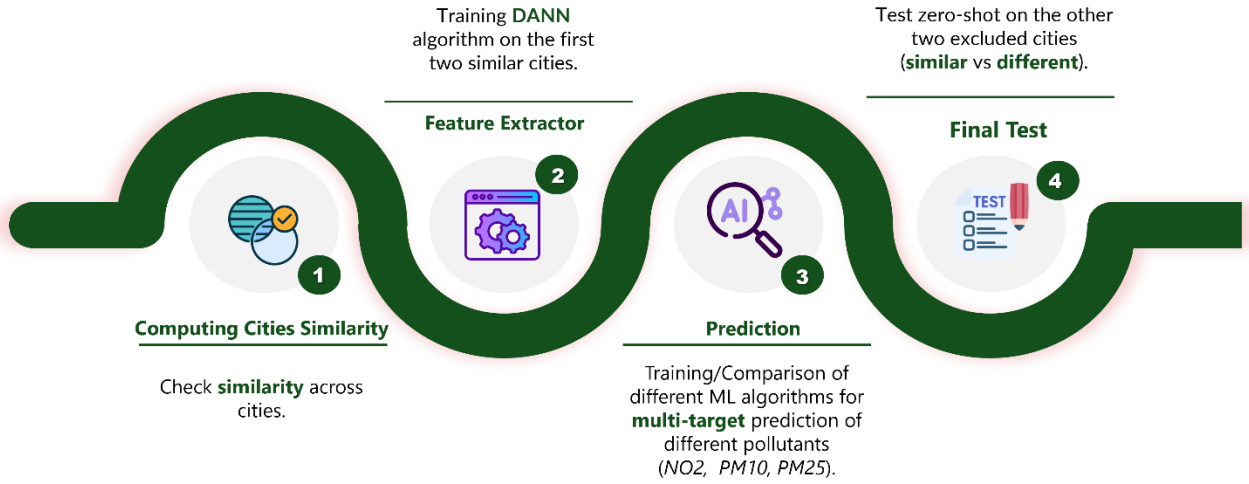


Figure 7.1 Cross-City Generalization approach.

The overall pipeline, illustrated in Figure 7.1, is organized into four main steps: computing inter-city similarity, extracting transferable features, developing predictive models, and performing zero-shot testing on unseen cities. This workflow emphasizes the importance of selecting structurally comparable urban contexts rather than attempting to design a universal model, thereby improving generalization and scalability in multi-city air quality forecasting.

The process begins with the computation of city similarity, which plays a central role in identifying pairs of structurally and environmentally comparable cities to be used for training. To this end, open-access datasets were collected from Paris, Madrid, Berlin, and Helsinki, covering traffic, meteorological, road infrastructure, and air quality information. Urban indicators such as population density, road density, modal split, and pollutant emission levels were derived from publicly accessible sources and analyzed using statistical tools including radar plots, Principal Component Analysis (PCA), and hierarchical clustering. This similarity analysis allowed the grouping of cities into comparable clusters, forming the foundation for knowledge transfer experiments.

In the second stage, DANN was employed as a feature extractor to learn domain-invariant representations from the two most similar source cities. The DANN architecture combines three components: a feature extractor, a label predictor, and a domain classifier. While the label predictor estimates pollutant concentrations (NO_2, $PM_{2.5}$, and PM_{10}), the domain classifier attempts to identify the city of origin. Through adversarial training, the feature extractor simultaneously minimizes prediction error and maximizes the confusion of the domain classifier, thereby enforcing the extraction of generalizable, city-independent features.

Once domain-invariant representations were obtained, the third stage consisted of training and comparing several machine learning models for multi-target pollutant prediction. Tree-based regressors implemented in scikit-learn [151] were applied to the transformed dataset, aiming to capture the nonlinear and monotonic relationships between traffic conditions, meteorological variables, and pollutant emissions. The models were trained chronologically on two similar cities and validated within the same contexts, with pollutant concentrations predicted simultaneously rather than separately.

Finally, the last step focused on zero-shot evaluation to assess the robustness of the methodology. The best-performing model, combining the DANN feature extractor and the most effective regressor, was applied to the two cities excluded from training. This setup was specifically designed to evaluate whether the framework could generalize pollutant predictions in structurally similar but previously unseen cities, while failing in dissimilar ones. Such a strategy highlights both the strengths and limitations of the approach, showing that instead of pursuing a one-size-fits-all solution, domain-adversarial transfer learning provides a viable pathway toward scalable, generalizable air quality prediction in structurally related urban contexts.

7.3 Results

The evaluation of the proposed domain-adversarial framework was carried out in two complementary trials, corresponding to similarity assessments performed with and without public transport indicators to account for cities lacking tram or metro systems.

Figures from 7.2 to 7.4 show the results of the performed similarity analysis with public transport indicators.

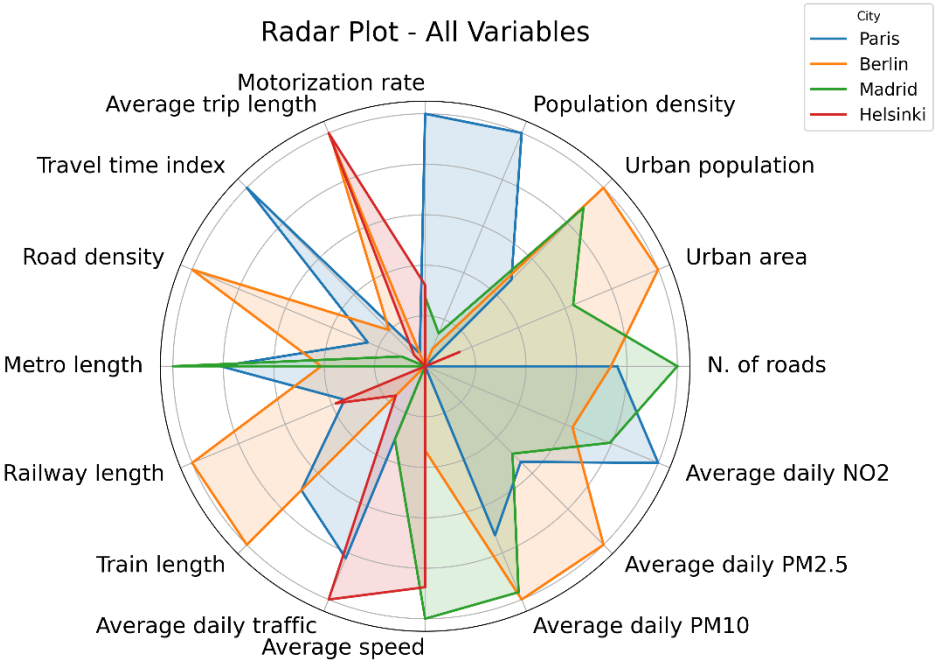


Figure 7.2 Radar plot with public transport indicators.

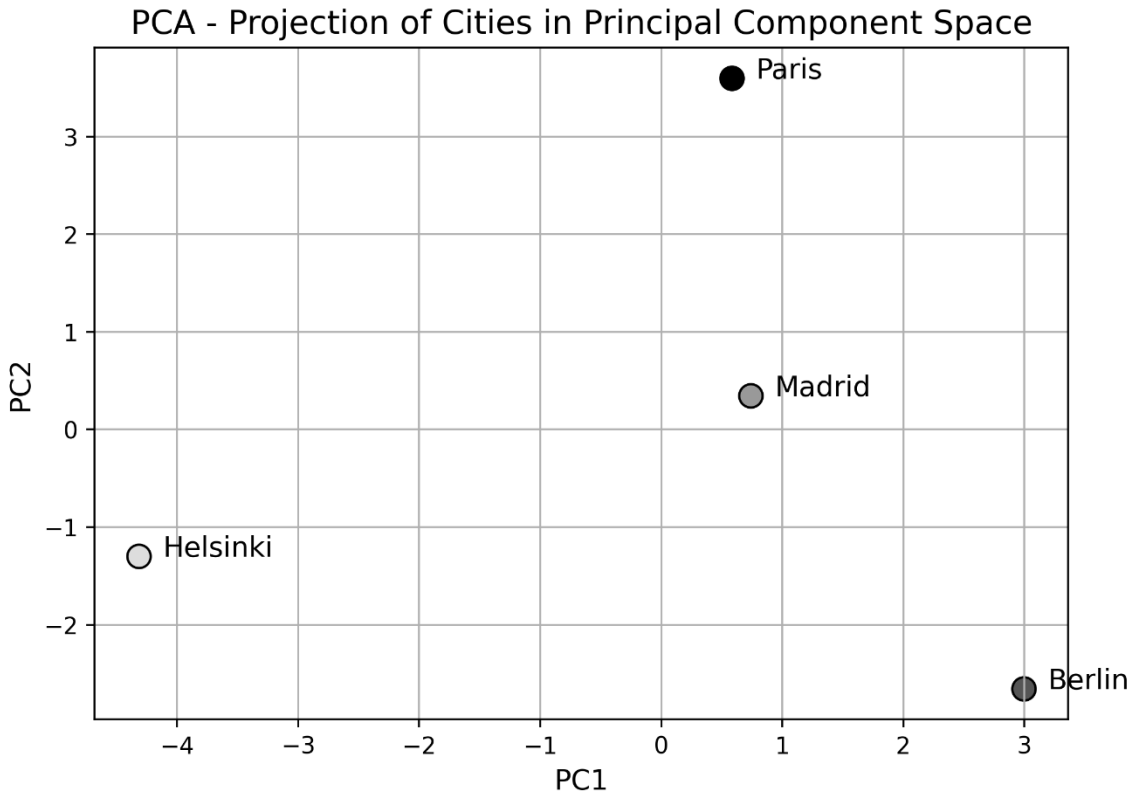


Figure 7.3 PCA with public transport indicators.

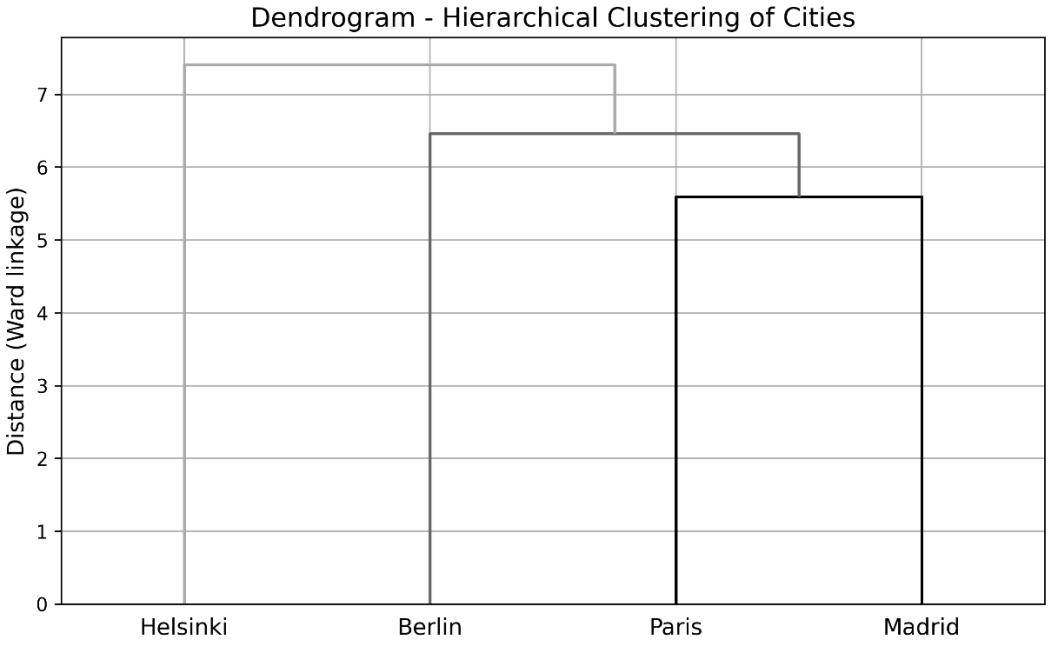


Figure 7.4 Dendrogram with public transport indicators.

In the first case, the similarity analysis highlighted strong structural and mobility-based relationships among the four cities. Radar plots revealed Paris as the most congested and dense urban environment, while Helsinki consistently ranked lowest across population, traffic, and pollution dimensions. Madrid displayed a profile closely aligned with Paris, particularly in traffic and pollution, whereas Berlin shared more similarities with Helsinki in public transport

infrastructure. Principal Component Analysis (PCA) and hierarchical clustering reinforced these findings: Paris and Madrid formed the closest pair, Berlin occupied an intermediate position, and Helsinki appeared clearly dissimilar. Based on this analysis, Paris and Madrid were selected as training cities, with Berlin used as the moderately similar test case and Helsinki as the dissimilar one.

Across 20 DANN models tuned using the Optuna library [94], the best configuration achieved a label loss of 0.0505 and a domain loss of 0.6983, striking an effective balance between predictive accuracy and domain invariance. Among the tested regressors, XGBoost emerged as the most effective, achieving the lowest average errors and the highest rank-based correlations. When applied to Berlin, the DANN + XGBoost combination delivered accurate predictions for NO_2, PM_{10}, and $PM_{2.5}$, with SMAPE values around 8.1, 6.2, and 4.3, respectively. These results confirm the ability of the model to generalize effectively to structurally related cities. Figures from 7.5 to 7.7 present the test results for Berlin across various pollutants, potting the actual values against those predicted.

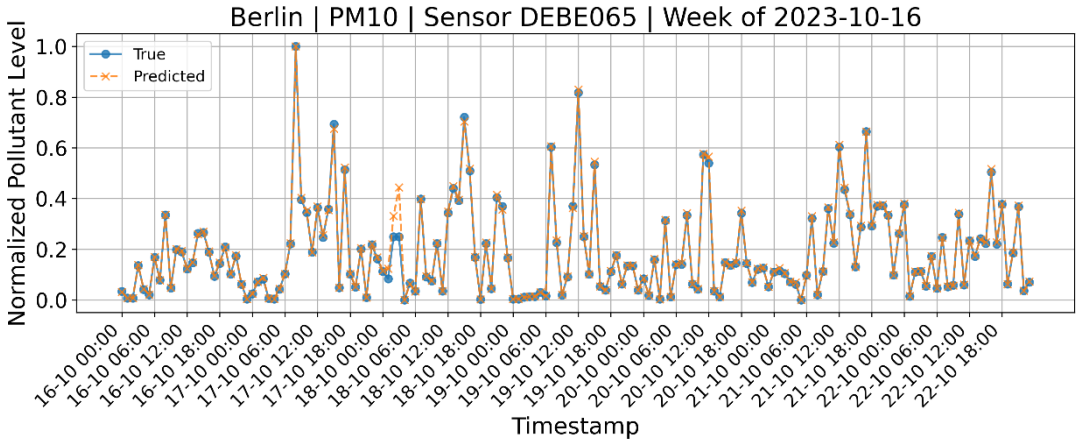


Figure 7.5 Test results for Berlin PM_{10}.

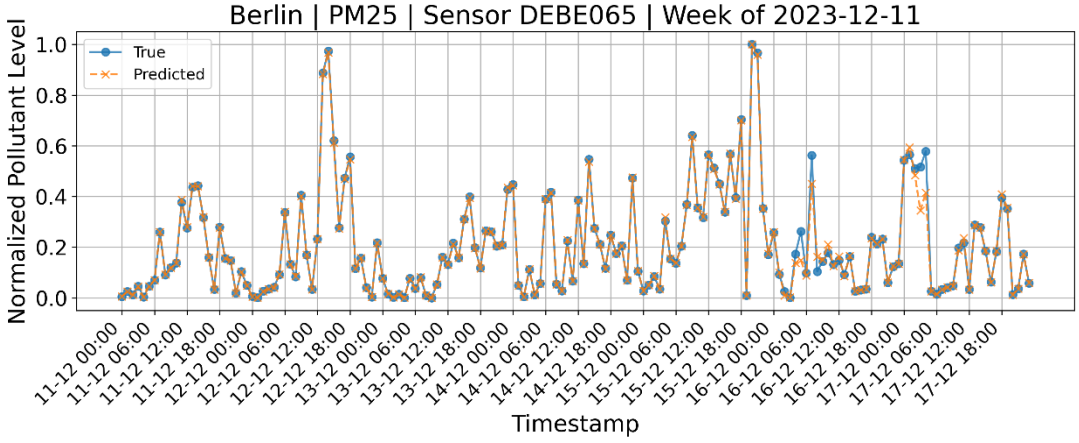


Figure 7.6 Test results for Berlin $PM_{2.5}$.

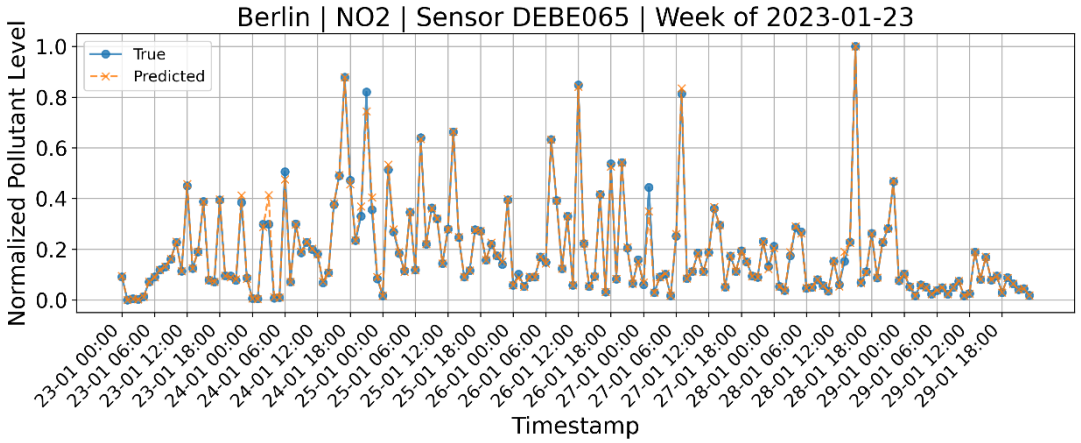


Figure 7.7 Test results for Berlin NO_2.

However, its transfer to Helsinki demonstrated marked deterioration, with prediction errors increasing substantially across all pollutants, highlighting the limited ability of the framework to handle contexts where structural and environmental conditions diverge significantly. Figure 7.8 depicts an example of the worst results obtained for the dissimilar city Helsinki for pollutant $PM_{2.5}$.

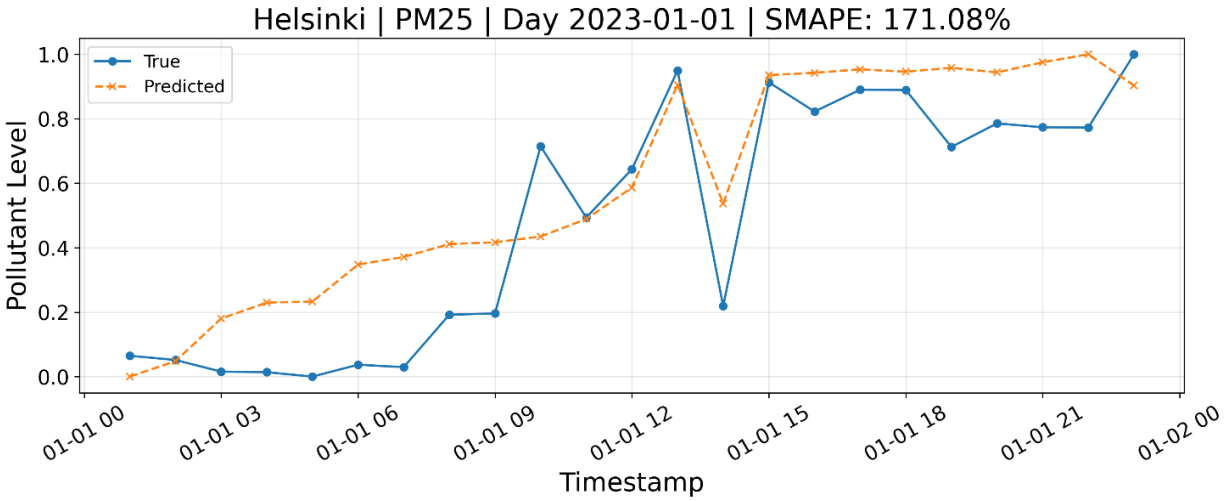


Figure 7.8 Test results Helsinki $PM_{2.5}$.

When public transport indicators were excluded from the similarity analysis, the city relationships shifted. Figures from 7.9 to 7.11 show the results obtained from the similarity analysis without public indicators.

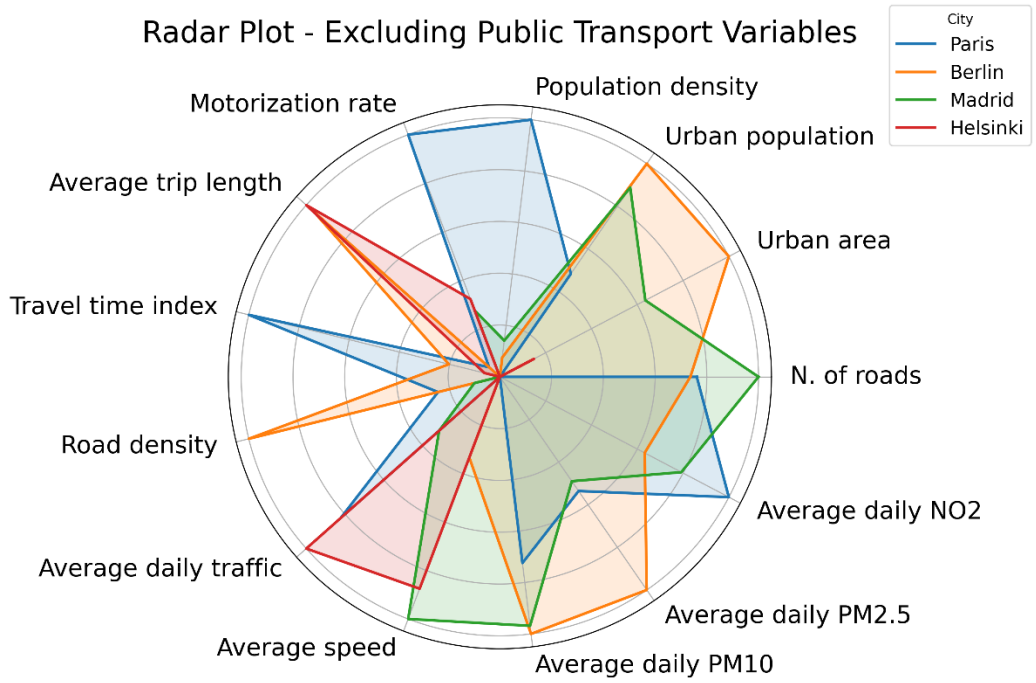


Figure 7.9 Radar plot without public transport indicators.

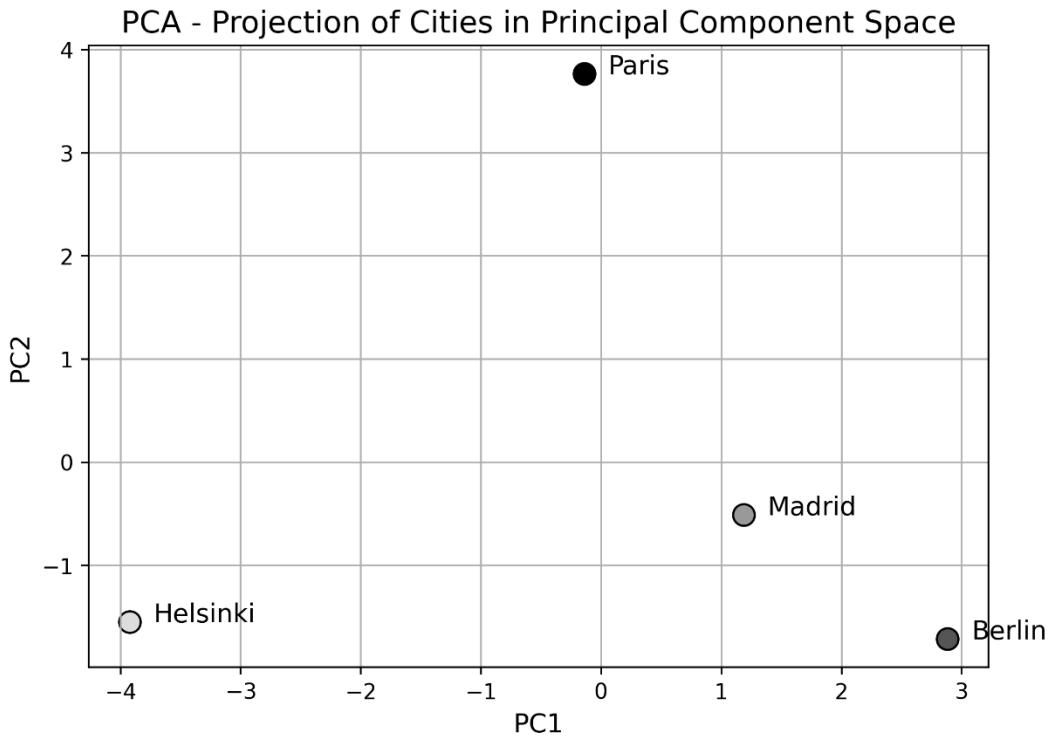


Figure 7.10 PCA without public transport indicators.

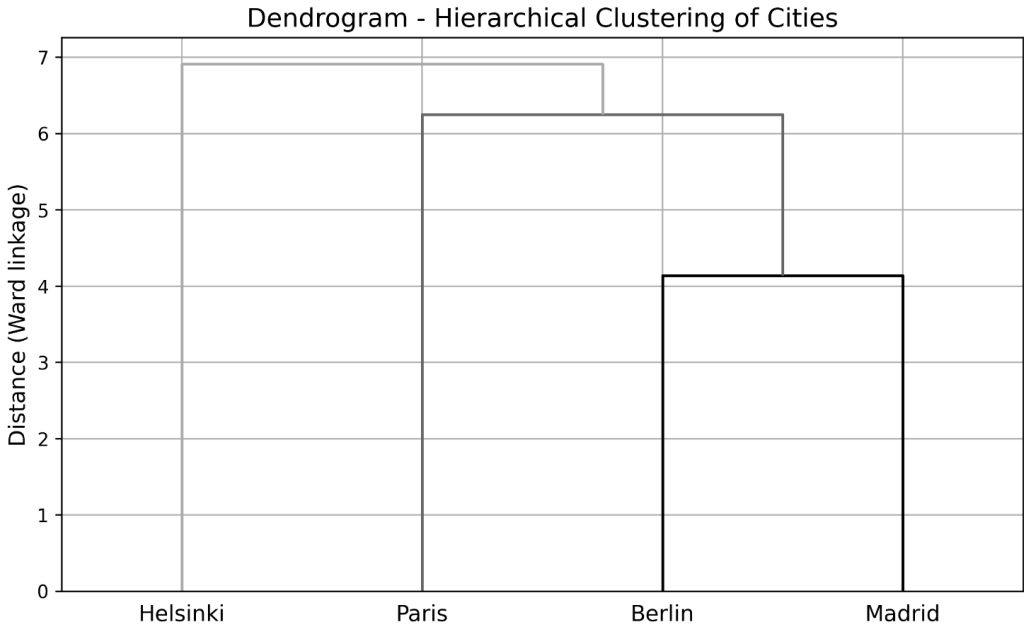


Figure 7.11 Dendrogram without public transport indicators.

Madrid and Berlin emerged with balanced profiles across traffic, pollution, and infrastructure, positioned between the extremes of Paris and Helsinki. PCA and hierarchical clustering confirmed this grouping, with Madrid and Berlin forming a cohesive cluster, Paris appearing moderately distinct, and Helsinki once again standing out as the most structurally dissimilar city.

On this basis, Madrid and Berlin were selected for training, while Paris and Helsinki were retained for testing. In this configuration, the optimal DANN setup yielded a label loss of 0.0899 and a domain loss of 0.6987, with LightGBM identified as the best-performing regressor. Applied to Paris, the DANN + LightGBM model produced robust pollutant forecasts across most sensors, with SMAPE values ranging between 11.1% and 20.6% depending on the pollutant and location. Figures from 7.12 to 7.14 show the test results for Paris across various sensors and for different pollutants, plotting the actual against the predicted values.

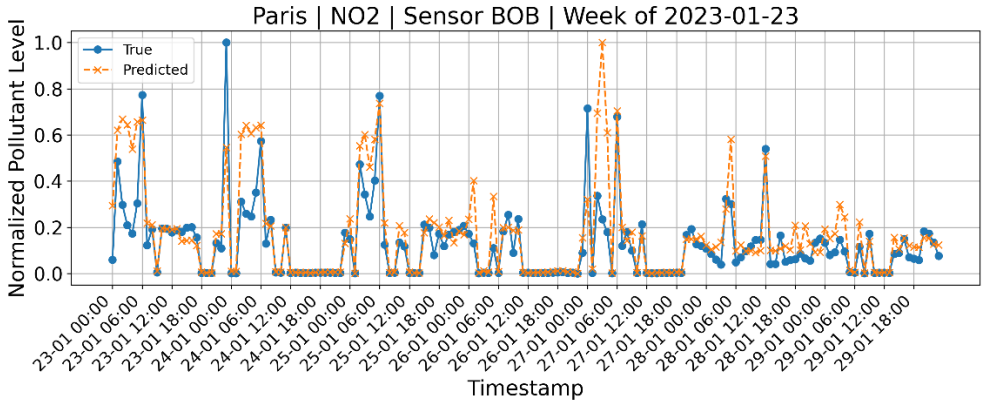


Figure 7.12 Test results Paris NO_2.

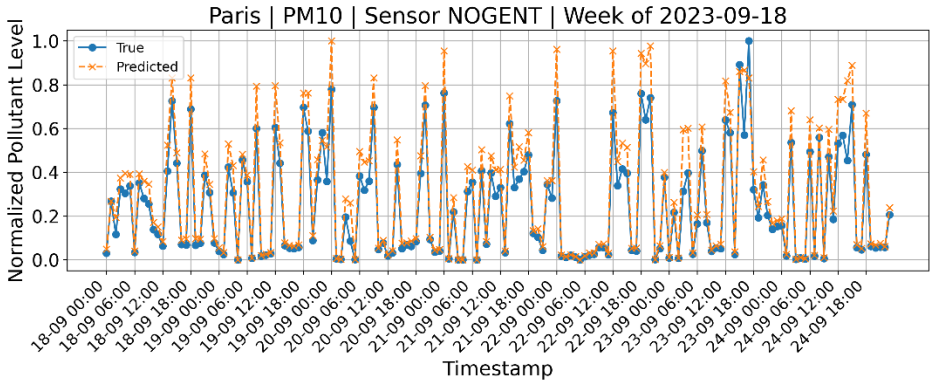


Figure 7.13 Test results Paris PM_{10}.

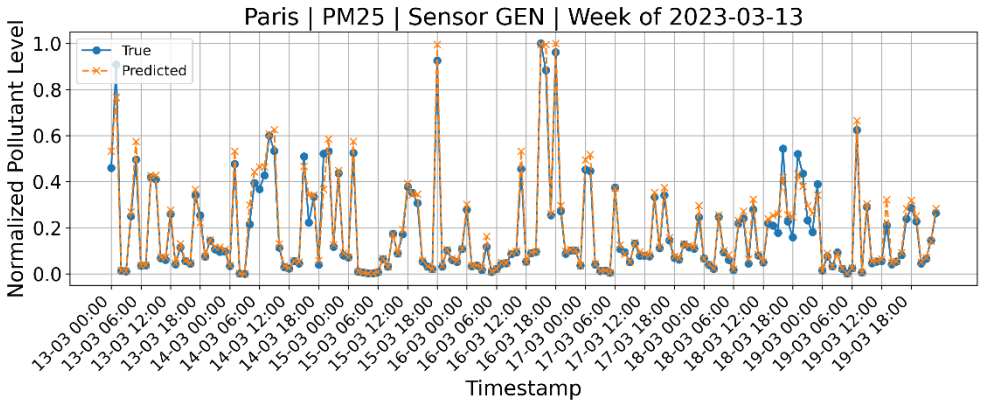


Figure 7.14 Test results Paris $PM_{2.5}$.

Across the various sensors for Paris, the only notable exception was the STDEN sensor.

Figure 7.15 Mapping Paris air quality sensors.

As shown in Figure 7.15, STDEN air quality sensor is in La Plaine Saint-Denis, a heavily industrialized area characterized by warehouses, logistics hubs, and construction zones. This setting likely contributes to a distinct pollution profile, potentially accounting for the reduced performance of the model at this site. Indeed, the model struggled to capture pollutant variability, likely due to atypical local emission patterns.

Finally, when applied to Helsinki, performance again declined sharply, confirming that successful knowledge transfer depends strongly on structural similarity between source and target domains.

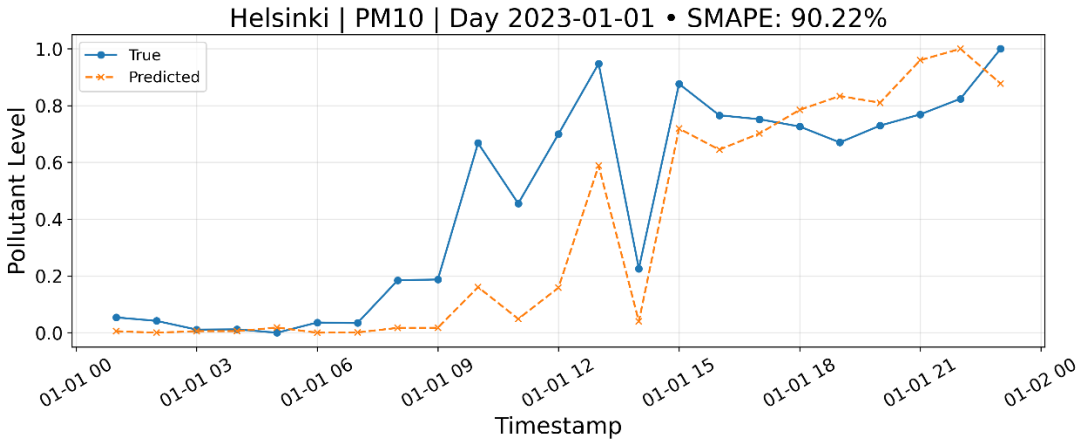


Figure 7.16 Test results for Helsinki PM_{10}.

Figure 7.16 shows an example of model deterioration for PM_{10} pollutant concentration recorded in Helsinki, the dissimilar city.

Overall, the results demonstrate both the promise and the limitations of the proposed approach. The DANN-based framework, coupled with tree-based regressors such as XGBoost and LightGBM, proved capable of effectively generalizing pollutant predictions across structurally similar cities and sensors. At the same time, the experiments revealed that its predictive power decreases significantly in dissimilar environments, underscoring the critical importance of robust similarity assessment to guide cross-city transfer learning strategies in air quality prediction.

7.4 Conclusions

In conclusion, this chapter highlighted the role of domain-adversarial learning in advancing cross-city air quality prediction, with particular attention to the conditions under which transfer learning can be effectively applied. The methodology demonstrated that predictive knowledge can be transferred across cities sharing comparable structural, mobility, and environmental characteristics, yielding accurate pollutant forecasts in moderately similar urban contexts.

However, the experiments also underscored important limitations: when applied to structurally dissimilar cities, such as Helsinki, the predictive capacity of the framework deteriorates markedly, suggesting that the effectiveness of domain-invariant feature extraction is strongly

dependent on the degree of similarity between source and target domains. This finding calls into question the feasibility of universal predictive models and instead supports the development of more context-specific approaches, where transfer learning is selectively applied to clusters of cities with demonstrable structural and environmental affinity. Beyond technical performance, this work also raises critical questions regarding data availability, sensor coverage, and the robustness of urban similarity measures, all of which significantly influence model reliability.

Taken together, the results position domain-adversarial learning not as a definitive solution, but as a promising direction within a broader research agenda that must address heterogeneity in urban systems, refine measures of inter-city comparability, and explore hybrid modeling strategies that can better balance generalizability with local specificity.

Chapter 8

Conclusions

The research presented in this dissertation set out to investigate the potential of machine learning to address a set of critical challenges in the development of smart cities, spanning water distribution systems, urban traffic management, and air quality monitoring. The thesis has explored these domains not as isolated problems but as interconnected components of complex urban ecosystems, where data availability, computational methodologies, and contextual variability strongly shape the feasibility and impact of predictive models. Across the different case studies, ranging from leak detection in water networks to clustering of water demand profiles, from traffic flow forecasting to transferable air quality prediction, the work has sought to advance methodological solutions, provide empirical evidence of their effectiveness, and critically reflect on their implications for future urban intelligence.

One of the central contributions of this work lies in demonstrating how machine learning can significantly improve the detection of leaks in water distribution systems. By designing pipelines that combine careful preprocessing, feature engineering, and predictive modeling, it has been possible to identify anomalous patterns with higher precision than traditional statistical approaches. This is particularly relevant in contexts where data is sparse, noisy, or incomplete, as is often the case in infrastructure monitoring. Beyond the technical improvements in accuracy, the findings highlight the role of machine learning in supporting the sustainability agenda of cities, as timely leak detection directly contributes to water conservation, operational efficiency, and resilience against climate-related stresses. Closely related is the exploration of clustering techniques for modeling water demand, which provided an unsupervised means of uncovering recurring consumption profiles across different urban contexts. These profiles revealed heterogeneity in usage behaviors that could not have been easily detected through aggregate statistics, thereby offering utilities new opportunities to refine resource allocation, identify anomalies, and support long-term infrastructure planning. The emphasis on pattern discovery and behavioral segmentation illustrates how machine learning can extend beyond prediction to provide explanatory insights that enrich domain knowledge.

In parallel, the thesis addressed the increasingly urgent problem of urban mobility through traffic flow forecasting. Here, spatio-temporal variability posed a significant challenge, as traffic data is inherently heterogeneous across regions and time horizons. By integrating clustering methods as a preprocessing step, it was possible to reduce variability and enable more robust short-term forecasting. The results demonstrate that hybrid approaches, which combine

unsupervised structure discovery with supervised predictive models, can offer tangible improvements in accuracy compared to single-step methods. Importantly, the traffic forecasting work also underscores the societal relevance of machine learning: accurate predictions have direct implications for congestion management, incident response, and sustainable mobility planning. However, the results also remind us that prediction alone is not sufficient unless coupled with interpretability and integration into decision-making processes that account for human and institutional factors.

The most ambitious and novel part of the thesis concerned the problem of transferable air quality prediction across cities. By adopting a domain-adversarial neural network (DANN) framework, the research aimed to test the extent to which models trained in one city could generalize to others with varying degrees of structural similarity. The empirical analysis, which carefully compared configurations with and without transport-related indicators, revealed both the promise and the limitations of current domain adaptation techniques. On the one hand, the results showed that generalization is feasible when the source and target cities share comparable characteristics, as seen in the successful transfer between Paris and Madrid or between Berlin and Madrid. On the other hand, the dramatic drop in performance when transferring to a structurally different city such as Helsinki illustrates the fragility of adversarial adaptation methods. These findings have broader implications for the field of urban AI: while the vision of developing generalizable, transferable models across heterogeneous cities is compelling, its realization requires richer contextual information, more flexible adaptation mechanisms, and possibly multimodal data integration that goes beyond static indicators.

Taken together, the different case studies offer a set of methodological insights that cut across domains. The first is the centrality of data preprocessing and feature engineering. Regardless of whether the application concerned leaks, water demand, traffic, or air quality, the quality of the raw data and the strategies used to enhance, normalize, and transform it proved decisive in determining model success. A second insight is the value of hybrid approaches that combine unsupervised and supervised learning. Clustering, in particular, emerged as a powerful tool for structuring data in ways that enhanced predictive performance in subsequent stages. A third lesson concerns the trade-off between accuracy and interpretability: while complex models such as ensemble learners or adversarial networks deliver higher predictive power, their opacity poses challenges for adoption in critical infrastructures where trust, accountability, and explainability are paramount. Finally, the work underscores the limitations of current transfer learning frameworks. While DANN represents an important step towards cross-city generalization, its failure in dissimilar contexts suggests that new paradigms, possibly rooted in meta-learning, graph-based adaptation, or causal inference, are needed to move the field forward.

The implications of these findings for smart city development are significant. Improved leak detection and demand modeling have direct consequences for resource sustainability, helping utilities reduce losses and anticipate consumption peaks. Traffic forecasting contributes to the

optimization of mobility systems, enabling more efficient and resilient transport infrastructures. Transferable air quality prediction, even in its current imperfect state, points towards a future in which cities can share knowledge and models to reduce redundant data collection and accelerate environmental monitoring. At the same time, the limitations observed in this research—data constraints, computational overhead, lack of generalizability in highly dissimilar contexts, and limited attention to end-user integration—highlight the challenges that must be addressed before machine learning can be fully embedded in urban management systems.

Looking ahead, several avenues for further research emerge from this thesis. Future efforts should focus on enhancing the adaptability of transfer learning methods, potentially by incorporating multi-source and multimodal data such as satellite imagery, social media signals, or Internet of Things (IoT) streams. There is also a pressing need for interpretable AI frameworks that balance predictive accuracy with transparency and usability for non-technical stakeholders, especially given the critical importance of trust in urban governance. Computational challenges could be addressed through lightweight models and edge computing architectures capable of supporting real-time deployment. Perhaps most ambitiously, the results of this thesis suggest that cross-domain learning—where traffic, air quality, water demand, and other urban systems are modeled jointly rather than in isolation—could provide a holistic view of urban dynamics, opening the way to integrated decision support systems that better reflect the interdependencies of modern cities.

In conclusion, this thesis has shown that machine learning holds significant promise for tackling some of the most pressing challenges faced by contemporary cities. By addressing domains as diverse as water management, traffic forecasting, and air quality, it has provided methodological advances, empirical evidence, and critical reflections that contribute both to academic scholarship and to practical applications. At the same time, the research has highlighted the importance of contextual awareness, data quality, interpretability, and adaptability as preconditions for the successful deployment of AI in urban environments. The road towards intelligent and sustainable cities will require not only continued technical innovation but also the capacity to bridge diverse data sources, adapt to heterogeneous contexts, and embed machine learning systems within broader frameworks of social, institutional, and environmental governance. The contributions of this thesis represent a step in this direction, while the challenges identified set the stage for future work at the intersection of artificial intelligence, data science, and urban systems.

Bibliography

[1] Caragliu, A., Del Bo, C., and Nijkamp, P. (2011). "Smart Cities in Europe". In *Journal of Urban Technology*, 18 (2), pp. 65-82.

[2] Nam, T., and Pardo, T. A. (2011). "Conceptualizing smart city with dimensions of technology, people, and institutions". In *Proceedings of the 12th Annual International Conference on Digital Government Research* (*DG. O 2011*). College Park, MD, USA, June 12-15, 2011, pp. 282-291.

[3] Lee, E. A. (2008). "Cyber physical systems: Design challenges". In *2008 11th IEEE International symposium on object and component-oriented real-time distributed computing* (*ISORC*). Orlando, Florida, USA, May 5-7, 2008, pp. 363-369.

[4] Zanella, A., Bui, N., Castellani, A., Vangelista, L., and Zorzi, M. (2014). "Internet of things for smart cities." *IEEE Internet of Things Journal,* 1 (1), pp. 22-32.

[5] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). "Internet of Thinkg (IoT): A vision, architectural elements and future directions". *Future generation computer systems,* 29 (7), pp. 1645-1660

[6] Perera, C., Zaslavsky, A., Christen, P., and Georgakopoulos, D. (2014). "Context aware computing for the Internet of Things: A survey". *IEEE Communications Surveys & Tutorials*, 16 (1), pp. 414–454.

[7] Mohanty, S. P., Choppali, U., and Kougianos, E. (2016). "Everything you wanted to know about smart cities". *IEEE Consumer Electronics Magazine*, 5(3), pp. 60–70.

[8] Allam, Z., and Dhunny, Z. A. (2019). "On big data, artificial intelligence and smart cities". *Cities*, 89, pp. 80–91.

[9] Liu, G.X., Shi, H. Kiani, A., et al. (2022). "Smart traffic monitoring system using computer vision and edge computing". *IEEE Transactions on Intelligent Transportation Systems,* 23 (8), pp. 12027-12038.

[10] Lakshmi, S., and Krishnamoorthy, A. (2023). "Deep Learning Techniques for Air Quality Prediction: A Focus on $PM_{2.5}$ and Periodicity". *Migration Letters,* 20 (S13), pp. 468-484.

[11] Batty, M. (2013). "Big Data, smart cities and city planning". *Dialogues in Human Geography,* 3 (3), pp. 272-279.

[12] ISO/IEC 30182:2017 – *Smart city concept model – Guidance for establishing a model for data interoperability*.

[13] Ahn, J. Y., Lee, J. S., Kim, H. J., and Hwang, D. J. (2016). "Smart city interoperability framework based on city infrastructure model and service prioritization". In Proceeding

of *2016 8th International Conference on Ubiquitous and Future Networks* (*ICUFN*), IEEE. Vienna, Austria, July 5-8, 2016, pp. 337-342.

[14] Mohammadi, M., Al-Fuqada, A., Sorour, S., and Guizani, M. (2018). "Deep Learning for IoT Big Data and Streaming Analytics: A Survey." *IEEE Communications Surveys & Tutorials,* 20 (4), pp. 2923-2960.

[15] Allam, Z., and Dhunny, Z. A. (2019). "On big data, artificial intelligence and smart cities." *Cities,* 89, pp. 80-91.

[16] Kitchin, R. (2014). "The real-time city? Big data and smart urbanism." *GeoJournal,* 79 (1), pp. 1-14.

[17] Dou, X., Chen, W., Zhu, L., Bai, Y., Li, Y., and Wu, X. (2023). "Machine learning for smart cities: A comprehensive review of applications and opportunities." *International Journal of Advanced Computer Science and Applications (IJACSA),* 14 (9), pp. 999-1016.

[18] Ahmad, K., Maabreh, M., Ghaly, M., Khan, K., Qadir, J., and Al-Fuqaha, A. (2021). "Developing future human-centered smart cities: Critical analysis of smart city security, data management, and ethical challenges." *Computer Science Review,* 43, p. 100452.

[19] Ullah, A., Anwar, S.M., Li, J., Nadeem, L., Mahmood, T., Rehman, A., and Saba, T. (2024). "Smart cities: the role of Internet of Things and machine learning in realizing a data-centric smart environment." *Complex & Intelligent Systems,* 10, pp. 1607–1637.

[20] Hashem, I.A.T., Chang, V., Anuar, N.B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., and Chiroma, H. (2016). "The role of big data in smart city." *International Journal of Information Management,* 36 (5), pp. 748–758.

[21] Luusua, A., Ylipulli, J., Foth, M., and Aurigi, A. (2022). "Urban AI: understanding the emerging role of artificial intelligence in smart cities." *AI & Society*, 38, pp. 1039–1044.

[22] Batty, M. (2018). "Digital twins." *Environment and Planning B: Urban Analytics and City Science*, 45(5), pp. 817–820.

[23] Sarker, I.H. (2022). "Data-driven Smart Cities: Leveraging IoT, Machine Learning, and Analytics for Intelligent Urban Systems." *Internet of Things,* 19, p. 100528.

[24] Romano, M., Kapelan, Z., and Savic, D. (2014). "Automated detection of pipe bursts and other events in water distribution systems." *Journal of Water Resources Planning and Management*, 140 (4), pp. 457–467.

[25] Alzubaidi, L., Al-Amidie, M., Alsharif, N., Farhan, L., Deng, J., Alazeb, M., and Duan, Y. (2023). "A survey on deep learning tools dealing with data scarcity." *Journal of Big Data*, 10 (46), pp. 1-82.

[26] Albaseer, A., Ciftler, B. S., Abdallah, M., and Al-Fuqaha, A. (2020). "Exploiting unlabeled data in smart cities using federated edge learning." In *2020 16th International Wireless Communications and Mobile Computing (IWCMC2020),* Limassol, Cyprus, June 15-19, 2020, IEEE, pp. 1666-1671.

[27] Chen, G., Su, Y., Zhang, X., Hu, A., Chen, G., Feng, S., et al. (2022). "A cross-city federated transfer learning framework: A case study on urban region profiling." *arXiv preprint arXiv:2206.00007*.

[28] Zou, X., Yan, Y., Hao, X., Hu, Y., Wen, H., Liu, E., et al. (2025). "Deep learning for cross-domain data fusion in urban computing: Taxonomy, advances, and outlook." *Information Fusion*, 113, p. 102606.

[29] Jain, A. K. (2010). “Data clustering: 50 years beyond K-means.” *Pattern Recognition Letters*, 31 (8), pp. 651–666.

[30] Aghabozorgi, S., Shirkhorshidi, A. S., and Wah, T. Y. (2015). “Time-series clustering – A decade review.” *Information Systems*, 53, pp. 16–38.

[31] Ullah, F., Raza, M., Malik, A. W., Qayyum, A., and Khan, M. A. (2024). “AI and IoT for sustainable smart water grids: A case study on data-driven consumer profiling and anomaly detection.” *Sustainable Cities and Society*, 96, p. 104660.

[32] Zheng, Y., Capra, L., Wolfson, O., and Yang, H. (2015). “Urban computing: Concepts, methodologies, and applications.” *ACM Transactions on Intelligent Systems and Technology*, 5(3), p. 38.

[33] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). “A density-based algorithm for discovering clusters in large spatial databases with noise.” In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96),* Portland, Oregon, USA, August 2-4, 1996. AAAI Press, pp. 226–231.

[34] Hashem, I. A. T., Chang, V., Anuar, N. B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., and Chiroma, H. (2016). The role of big data in smart city.” *International Journal of Information Management*, 36(5), pp. 748–758.

[35] Zhang, X., Yang, X., and Li, Y. (2021). “Data-driven analysis of bike-sharing systems using clustering and prediction methods.” *Applied Sciences*, 11(4), p. 1576.

[36] Jolliffe, I. T., and Cadima, J. (2016). “Principal component analysis: A review and recent developments.” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374 (2065), p. 20150202.

[37] van der Maaten, L., and Hinton, G. (2008). “Visualizing data using t-SNE.” *Journal of Machine Learning Research*, 9, pp. 2579–2605.

[38] Yuan, Y., Raubal, M., and Liu, Y. (2021). “Correlating human mobility and air quality in urban areas using spatial-temporal visual analytics.” *Computers, Environment and Urban Systems*, 86, p. 101584.

[39] McInnes, L., Healy, J., and Melville, J. (2018). “UMAP: Uniform Manifold Approximation and Projection for dimension reduction.” *arXiv preprint arXiv:1802.03426*.

[40] Chen, W., Dou, X., Zhu, L., Bai, Y., Li, Y., and Wu, X. (2022). “Data-driven smart cities: Leveraging IoT, machine learning, and analytics for intelligent urban systems.” *Internet of Things*, 19, p. 100528.

[41] Albaseer, A., Khan, M. A., Ahmad, A., and Salah, K. (2020). “Artificial intelligence-based anomaly detection in smart cities: A review.” *Sustainable Cities and Society*, 61, p. 102328.

[42] Chandola, V., Banerjee, A., and Kumar, V. (2009). “Anomaly detection: A survey.” *ACM Computing Surveys*, 41(3), pp. 1–58.

[43] Pimentel, M. A., Clifton, D. A., Clifton, L., and Tarassenko, L. (2014). “A review of novelty detection.” *Signal Processing*, 99, pp. 215–249.

[44] Li, Y., and Guo, Q. (2018). “Traffic anomaly detection based on reconstruction error using autoencoder.” In *Proceedings of the 2018 IEEE International Conference on Big Data (IEEE Big Data 2018),* Seattle, WA, USA, December 10-13, 2018. IEEE.

[45] Erfani, S. M., Rajasegarar, S., Karunasekera, S., and Leckie, C. (2016). "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning." *Pattern Recognition*, 58, pp. 121–134.

[46] Liu, F. T., Ting, K. M., and Zhou, Z. H. (2008). "Isolation forest." In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining,* Pisa, Tuscany, Italy, December 15-19, 2008. IEEE. pp. 413–422.

[47] Zhang, X., Xu, Z., Zhao, S., and Wang, Z. (2022). "Deep learning-based anomaly detection: A survey." *ACM Transactions on Intelligent Systems and Technology*, 13 (3), pp. 1–41.

[48] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., and Farhan, L. (2023). "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions." *Journal of Big Data*, 10(1), pp. 1–74.

[49] LeCun, Y., Bengio, Y., and Hinton, G. (2015). "Deep learning." *Nature*, 521(7553), pp. 436–444.

[50] Schmidhuber, J. (2015). "Deep learning in neural networks: An overview." *Neural Networks*, 61, pp. 85–117.

[51] Zhang, J., Zheng, Y., and Qi, D. (2017). "Deep spatio-temporal residual networks for citywide crowd flows prediction." In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, *AAAI-17*, 31(1), San Francisco, California, USA, February 4-9, 2017, pp. 1655-1661.

[52] Zou, H., Yang, Y., Lv, T., and Lu, H. (2024). "Intelligent environmental monitoring in smart cities: Machine learning methods and applications." *Environmental Modelling & Software*, 169, p. 105796.

[53] Hu, Y., Yuan, M., Wu, J., and Li, L. (2021). "Deep learning for high-resolution urban remote sensing: A review." *International Journal of Applied Earth Observation and Geoinformation*, 103, p. 102465.

[54] Ullah, S., Abbas, H., Yaqoob, I., Gani, A., and Khan, M. K. (2024). "Machine learning for smart cities: Applications, challenges, and future directions." *Sustainable Cities and Society*, 97, p. 104673.

[55] Yu, B., Yin, H., and Zhu, Z. (2018). "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting." In *Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI-18,* Stockholm, Sweden, July 13-19, 2018, pp. 3634–3640.

[56] Ryu, S., Noh, J., and Kim, H. (2017). "Deep neural network based demand side short-term load forecasting." *Energies*, 10(1), 3, pp. 1-20.

[57] Kühnert, C., Gonuguntla, N. M., Krieg, H., Nowak, D., and Thomas, J. A. (2021). "Application of LSTM Networks for Water Demand Prediction in Optimal Pump Control." *Water*, 13(5), p. 644.

[58] Li, Y., Yu, R., Shahabi, C., and Liu, Y. (2018). "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting." In *Proceedings of the 6th International Conference on Learning Representations, ICLR 2018,* Vancouver, Canada, April 30-May 3, 2018, pp. 1-16.

[59] Xu, C., Chen, H., Wei, Z., and Li, Y. (2020). "Multimodal data fusion for urban disaster management: A deep learning perspective." *IEEE Access*, 8, pp. 224103–224115.

[60] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). "Attention is all you need." *Advances in Neural Information Processing Systems*, 30, pp. 1-15.

[61] Pan, S. J., and Yang, Q. (2010). "A survey on transfer learning." *IEEE Transactions on Knowledge and Data Engineering, 22*(10), pp. 1345–1359.

[62] Zhang, Y., Zhang, X., and Wang, Y. (2019). "Transfer learning for air quality prediction: A deep learning approach." *Journal of Cleaner Production, 237*, 117729, pp. 1-11.

[63] Chen, C., Li, K., Teo, S. G., Zou, X., and Zhang, J. (2021). "Gated Residual Recurrent Graph Neural Networks for Traffic Prediction." In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, *AAAI-21*, 35(5), *virtually*, February 2-9, 2021, pp. 3520–3527.

[64] Lundberg, S. M., and Lee, S.-I. (2017). "A unified approach to interpreting model predictions." *Advances in Neural Information Processing Systems, 30*, pp. 4765–4774.

[65] Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why Should I Trust You? Explaining the Predictions of Any Classifier." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 13-17, 2016, San Francisco, California, USA, pp. 1135–1144.

[66] Berlotti, M., Di Grande, S., Cavalieri, S., and Gueli, R. (2023). "Detection and prediction of leakages in water distribution networks." In *Proceedings of the 12th International Conference on Data Science, Technology and Applications, DATA 2023,* July 11-13, 2023, Rome, Italy, SCITEPRESS – Science and Technology Publications, pp. 436–443.

[67] Berlotti, M., Di Grande, S., Cavalieri, S., and Gueli, R. (2025). "Modelling and clustering patterns from smart meter data in water distribution systems." In *Proceedings of the 27th International Conference on Enterprise Information Systems (ICEIS 2025)*, April 1-6, 2025, Porto, Portugal, SCITEPRESS – Science and Technology Publications, pp. 691-698

[68] Berlotti, M., Di Grande, S., Cavalieri, S., and Gueli, R. (2025). "Smart meter data analysis: Modelling and clustering patterns in water distribution systems." In *Proceedings of the 39th European Conference on Modelling and Simulation (ECMS 2025)*. June 24-27, 2025, Catania, Sicily, Italy, ECMS, pp. 691- 698.

[69] Berlotti, M., Di Grande, S., Cavalieri, S., and Costa, D. G. (2025). "Cross-city generalization of air quality prediction: A domain-adversarial learning approach." In *Proceedings of the IEEE International Smart Cities Conference, ISC2 2025,* October 6-9, Patras, Greece, IEEE.

[70] Berlotti, M., Di Grande, S., Cavalieri, S., Torrisi, V., and Inturri, G. (2023). "Proposal of an AI-based approach for urban traffic prediction from mobility data." In *2023 IEEE International Conference on Big Data (IEEE BigData 2023),* December 15-18, 2023, Sorrento, Naples, Italy, IEEE. pp. 2570–2577.

[71] Berlotti, M., Di Grande, S., and Cavalieri, S. (2024). "Proposal of a machine learning approach for traffic flow prediction." *Sensors, 24*(7), 2348, pp. 1-19.

[72] Di Grande, S., Berlotti, M., and Cavalieri, S. (2024). "AI-powered urban mobility analysis for advanced traffic flow forecasting." In *Proceedings of the 13th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS 2024),* May 2-4, 2024, Angers, France. SCITEPRESS – Science and Technology Publications. pp. 57–64.

[73] De Souza Oliveira, T., Di Grande, S., Berlotti, M., Cavalieri, S., Torrisi, V., Calabrò, G., and Inturri, G. (2024). "Enhancing urban traffic management through machine learning prediction models for sensor-less roads." In *Emerging cutting-edge applied research and development in intelligent traffic and transportation systems.* IOS Press. pp. 102–111.

[74] Laney, D. (2001). "3D data management: Controlling data volume, velocity and variety." *META group research note*, *6* (70), 1.

[75] Klise, K. A., Murray, R., and Haxton, T. (2018). "An overview of the water network tool for resilience (WNTR)." In *Proceedings of the 1st International WDSA/CCWI 2018 Joint Conference,* July 23-25, 2018, Kingston, Ontario, Canada, pp. 1-8.

[76] Boeing, G. (2017). "OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks." *Computers, Environment and Urban Systems*, 65, pp. 126–139.

[77] Sahoo, S., and Ghose, M. K. (2022). "An efficient imputation method for missing data in IoT environment." *Journal of King Saud University - Computer and Information Sciences,* 34(6), pp. 2653–2664.

[78] Pedregosa, F., et al., "StandardScaler". In *Scikit-learn documentation* (Version 1.5.0). Accessed on August 15, 2025, from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

[79] Kuhn, M., and Johnson, K. (2019). "Feature engineering and selection: A practical approach for predictive models." *CRC Press.*

[80] Tukey, J. W. (1977). "Exploratory data analysis", *Reading, MA: Addison-wesley.* 2, pp. 131-160.

[81] Shmueli, G., and Polak, J. (2024). "Practical time series forecasting with r: A hands-on guide." *Axelrod schnall publishers.*

[82] E. Clifford, S. Mulligan, J. Comer, and L. Hannon. (2018). "Flow-Signature Analysis of Water Consumption in Nonresidential Building Water Networks Using High-Resolution and Medium-Resolution Smart Meter Data: Two Case Studies." *Water Resources Research,* 54 (1), pp. 88-106.

[83] Bandara, K., Hyndman, R. J., and Bergmeir, C. (2021). "MSTL: A Seasonal-Trend Decomposition Algorithm for Time Series with Multiple Seasonal Patterns." (arXiv:2107.13462). arXiv. http://arxiv.org/abs/2107.13462

[84] Yang, L., Wen, Q., Yang, B., and Sun, L. (2021). "A Robust and Efficient Multi-Scale Seasonal-Trend Decomposition." In *Proceedings of the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, June 6-11, 2021, Toronto, Ontario, Canada, pp. 5085–5089.

[85] Guyon, I., and Elisseeff, A. (2003). "An introduction to variable and feature selection." *Journal of Machine Learning Research,* 3, pp. 1157–1182.

[86] Rosner, B. (1983). "Percentage points for a generalized ESD many-outlier procedure." *Technometrics,* 25(2), pp. 165–172.

[87] Tukey, J. W. (1977). "Exploratory data analysis." Addison-Wesley.

[88] Yang, Y., and Shami, A. (2020). "On hyperparameter optimization of machine learning algorithms: Theory and practice." *Neurocomputing,* 415, pp. 295–316.

[89] Brockwell, P. J., and Davis, R. A. (2016). "Introduction to time series and forecasting". (3rd ed.), Springer.

[90] Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. (2000). "LOF: Identifying density-based local outliers." *ACM SIGMOD Record*, 29(2), pp. 93–104.

[91] Liu, F. T., Ting, K. M., and Zhou, Z.-H. (2008). "Isolation forest." In *2008 Eighth IEEE International Conference on Data Mining,* December 15-19, Pisa, 2008, Tuscany, Italy, pp. 413–422.

[92] Frey, B. J., and Dueck, D. (2007). "Clustering by passing messages between data points." *Science,* 315(5814), pp. 972–976.

[93] Jain, A. K., and Moreau, J. V. (1987). "Bootstrap technique in cluster analysis." *Pattern Recognition*, 20(5), pp. 547–568.

[94] Optuna. "A hyperparameter optimization framework." *Optuna 4.0.0 documentation. (n.d.).* Retrieved 17 August 2025, from https://optuna.readthedocs.io/en/stable/index.html

[95] Paparrizos, J., and Gravano, L., "K-Shape: Efficient and Accurate Clustering of Time Series." In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, (SIGMOD/PODS'15),* May 31- June 4, 2015, Vicotria, Australia, pp. 1855-1870.

[96] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). "Domain-adversarial training of neural networks." *Journal of machine learning research*, 17(59), pp. 1-35.

[97] Sokolova, M., and Lapalme, G. (2009). "A systematic analysis of performance measures for classification tasks." *Information Processing & Management, 45*(4), pp. 427–437.

[98] Rousseeuw, P. J. (1987). "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis." *Journal of Computational and Applied Mathematics, 20*, pp. 53–65.

[99] Burnham, K. P., and Anderson, D. R. (2004). "Multimodel inference: Understanding AIC and BIC in model selection." *Sociological Methods & Research, 33*(2), pp. 261–304.

[100] Hyndman, R. J., and Koehler, A. B. (2006). "Another look at measures of forecast accuracy." *International Journal of Forecasting*, 22(4), pp. 679–688.

[101] Chicco, D., Warrens, M. J., and Jurman, G. (2021). "The coefficient of determination R^2 and bias in artificial intelligence." *Computational and Mathematical Methods in Medicine*, pp. 1–11.

[102] Benesty, J., Chen, J., Huang, Y., and Cohen, I. (2009). "Pearson correlation coefficient." *Noise reduction in speech processing*, Springer, pp. 1–4.

[103] Hunaidi, O., Chu, W., Wang, A., and Guan, W. (1999). "Leak detection methods for plastic water distribution pipes." *American Water Works Association Research Foundation (AWWARF) Report*. Denver, CO: AWWA.

[104] Cody, R. J., Wylie, S. R., and El-Shafie, A. (2020). "Deep learning hydroacoustic spectrograms for leakage detection in water pipelines." *Journal of Hydroinformatics,* 22(5), pp. 1122–1136.

[105] Wang, F., Li, Y., and Sun, J. (2021). "Acoustic-based leakage detection in water pipelines using artificial neural networks." *Measurement,* 171, p. 108777.

[106] Bentoumi, M., Laouar, L., and Triki, A. (2017). "Leak detection in water distribution systems using wavelet transform and vibration analysis." *International Journal of Advanced Computer Science and Applications,* 8(5), pp. 381–387.

[107] Yu, X., Li, M., Chen, Y., and Zhang, K. (2023). "Leak detection in real-world water distribution networks using piezoelectric accelerometers and machine learning." *Water Research,* 234, p. 119734.
[108] Wan, H., Zhang, J., and Gong, J. (2022). "A review of transient-based methods for leak detection in water distribution systems." *Journal of Water Supply: Research and Technology – AQUA,* 71(5), pp. 576–589.
[109] Colombo, A. F. (2009). "Transient-based leak detection: Theory and practice." *Journal of Hydroinformatics,* 11(4), pp. 281–296.
[110] Perez, R., Puig, V., Pascual, J., Quevedo, J., Landeros, E., and Peralta, A. (2014). "Methodology for leakage isolation using pressure sensitivity analysis in water distribution networks." *Control Engineering Practice,* 30, pp. 1–13.
[111] Kammoun, M., Beirami, A., and Karimi, I. (2022). "Machine learning approaches for leakage detection in water distribution networks: A review." *Journal of Water Resources Planning and Management,* 148(5), p. 04022014.
[112] Barros, F. M. (2023). "Machine learning methods for water quality anomaly detection: A semi-supervised approach." *Environmental Modelling & Software*, 162, p. 10562.
[113] NAIADES Project. (2022). "Next-generation integrated urban water management." *H2020 European Project*. Retrieved from https://naiades-project.eu/
[114] Lijuan, Z., Hong, L., and Changming, M. (2012). "Pipe leakage prediction based on RBF neural network." *Procedia Engineering,* 28, pp. 143–148.
[115] Leu, J.-S., and Bui, V. (2016). "A Bayesian network learning model for leak prediction in water distribution systems." *Expert Systems with Applications,* 65, pp. 305–316.
[116] Wang, Z., Liu, J., and Zhang, H. (2022). "Digital twin-based leakage prediction and fault diagnosis for hydraulic systems." *Mechanical Systems and Signal Processing,* 162, p. 107991.
[117] ADTK. (2023). "Anomaly Detection Toolkit (ADTK)." *Computer software*. GitHub. https://github.com/arundo/adtk
[118] Gopali, R., and Namin, A. S. (2022). "Anomaly detection in time-series data: A survey." *Journal of Big Data*, 9(1), p. 23.
[119] Otte, S., Aust, C., and Hochreiter, S. (2022). "Anomaly detection in multivariate time series with robust recurrent autoencoders." *Machine Learning*, 111(3), pp. 1099–1125.
[120] Gutschi C., Furian N., Suschnigg J., Neubacher D., and Voessner S. (2018). "Log-based predictive maintenance in discrete parts manufacturing." *Procedia CIRP, vol.79,* ELSEVIER, pp. 528-533.
[121] Salvaggio, M., Futrell, R., Batson, C.D., Brents, and B.G. (2014). "Water scarcity in the desert metropolis: How environmental values, knowledge and concern affect Las Vegas residents' support for water conservation policy." *Journal of Environmental Planning and Management*, 57 (4), pp.588–611.
[122] Pearson, D. (2019). "Standard Definitions for Water Losses." *IWA Publishing,* pp. 1-78.
[123] Cardell-Oliver, R., Wang, J. and Gigney. H. (2016). "Smart Meter Analytics to Pinpoint Opportunities for Reducing Household Water Use." *Journal of Water Resources Planning and Management*, 142 (6), p. 04016007.

[124] Cominola, A., Nguyen, K., Giuliani, M., Stewart, R.A, Maier, H.R., and Castelletti, A. (2019). "Data Mining to Uncover Heterogeneous Water Use Behaviors From Smart Meter Data." *Water Resources Research*, 55 (11), pp. 9315–9333.

[125] Aksela, K., and Aksela, M. (2011). "Water consumption analysis and prediction in Finnish households." *Water Science and Technology: Water Supply,* 11(3), pp. 348–357.

[126] Obringer, R., and White, D. D. (2023). "Clustering residential water consumers with unsupervised learning: Insights from the Southwestern United States." *Water Resources Research*, 59(1), e2022WR032067.

[127] Arsene, C. T., Eremia, M., and Toma, L. (2021). "Profiling water consumers using clustering techniques and smart meter data." *Applied Sciences*, 11(16), p. 7360.

[128] Cheifetz, N., Taormina, R., Galelli, S., and Ostfeld, A. (2017). "Clustering water consumers based on smart meter data for detecting irregularities and profiling demand patterns." *Journal of Water Resources Planning and Management,* 143(10), p. 04017060.

[129] Guo, J., Huang, W., and Williams, B. M. (2019). "Real time traffic flow forecasting using spatio-temporal data." *Transportation Research Part C: Emerging Technologies,* 105, pp. 197–211.

[130] Salvo, G., Biancardo, S. A., and Caroppo, A. (2017). "Floating car data for traffic monitoring and forecasting: An overview." *Procedia Computer Science,* 109, pp. 273–280.

[131] Liu, Y., Chen, C., and Wang, H. (2020). "CatBoost for traffic flow prediction: A machine learning approach." *Applied Sciences*, 10 (7), p. 2401.

[132] Li, Y., Zheng, Y., and Yang, Q. (2015). "Traffic prediction in a bike-sharing system." *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, (*ACM SIGSPATIAL 2015)*, November 3-6, 2015, Seattle, Washington, USA, pp. 1–10.

[133] Smith, B. L., Williams, B. M., and Oswald, R. K. (2012). "Comparison of parametric and nonparametric models for traffic flow forecasting." *Transportation Research Part C: Emerging Technologies,* 10(4), pp. 303–321.

[134] Zhou, M., Zhang, W., and Li, X. (2020). "Short-term traffic flow prediction using ARIMA and machine learning approaches." *Procedia Computer Science,* 174, pp. 500–509.

[135] Vijayalakshmi, V., Sharma, A., and Kumar, R. (2021). "A deep learning approach for short-term traffic forecasting using back propagation and LSTM models." *International Journal of Intelligent Transportation Systems Research,* 19(3), pp. 512–523.

[136] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T. Y. (2017). "LightGBM: A highly efficient gradient boosting decision tree." *Advances in Neural Information Processing Systems,* 30, pp. 3146–3154.

[137] Zhang, Y., Zhao, P., and Wu, X. (2023). "Spatio-temporal graph neural networks for traffic flow forecasting." *IEEE Transactions on Intelligent Transportation Systems,* 24(5), pp. 5152–5163.

[138] Chen, T., and Guestrin, C. (2016). "XGBoost: A scalable tree boosting system." *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (*KDD 2016),* August 13-17, 2016, San Francisco, CA, USA, pp. 785–794.

[139] Liu, J., and Guan, W. (2004). "Traffic flow prediction with history average models in urban networks." *Journal of Transportation Engineering,* 130(6), pp. 687–693.

[140] Li, Z., Li, Q., and Li, J. (2020). "Traffic flow prediction with LSTM neural networks." *Neural Computing and Applications,* 32(12), pp. 9713–9725.

[141] Ma, X., Tao, Z., Wang, Y., Yu, H., and Wang, Y. (2015). "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data." *Transportation Research Part C: Emerging Technologies,* 54, pp. 187–197.

[142] Zhang, Y., Zheng, Y., and Qi, L. (2017). "Deep spatio-temporal residual networks for citywide crowd flows prediction." *Proceedings of the AAAI Conference on Artificial Intelligence, (AAAI-17),* 31(1), February 4-9, San Francisco, CA, USA, pp. 1655–1661.

[143] TomTom Traffic Stats. (2023). *TomTom Traffic Statistics Portal*. Retrieved from https://www.tomtom.com

[144] Herzen, J., Januschowski, T., Rangapuram, S. S., Gasthaus, J., and Flunkert, V. (2023). "Darts: User-friendly modern machine learning for time series." *Journal of Machine Learning Research,* 24(146), pp. 1–6.

[145] Box, G. E. P., and Jenkins, G. M. (1976). "Time series analysis: Forecasting and control." *Holden-Day.*

[146] Hipel, K. W., and McLeod, A. I. (1994). "Time series modelling of water resources and environmental systems." *Elsevier*.

[147] Breiman, L. (2001). "Random forests." *Machine Learning, 45*(1), pp. 5–32.

[148] De Souza, R. M., Batista, B. G., Pires, R. M., and Delbem, A. C. B. (2020). "Multi-source ensemble transfer learning for air pollution prediction." *Environmental Modelling & Software,* 132, p. 104796.

[149] Yao, H., Tang, X., Wei, H., Zheng, G., and Li, Z. (2019). "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction." In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence, (AAAI-19),* January 27- February 1, Honolulu, Hawaii, USA, pp. 5668–5675.

[150] Ma, X., Ma, H., and Jin, W. (2020). "Cross-city air quality prediction using transfer learning." *Knowledge-Based Systems,* 191, p. 105217.

[151] scikit-learn developers. (n.d.). *scikit-learn: Machine learning in Python*. Retrieved August 21, 2025, from https://scikit-learn.org/stable/