UNIVERSITY OF PADOVA

DEPARTMENT OF MANAGEMENT AND ENGINEERING DTG

MASTER'S THESIS IN MANAGEMENT ENGINEERING

Investigating shared bike usage patterns in correlation with weather conditions through the application of machine learning algorithms

SUPERVISOR
PROF.SSA MARTA DISEGNA

MASTER CANDIDATE
MARZIEH JAVANMARDI JALALABADI

CO-SUPERVISOR
Elena Barzizza

STUDENT ID
2043103

Academic Year
2023-2024

`

Master Thesis
Master of science in Management Engineering

Investigating shared bike usage patterns in correlation with weather conditions through the application of machine learning algorithms

Author:
Marzieh Javanmardi Jalalabadi

Supervisor:
Marta Disegna

To my family تقدیم به خانواده ام

To my friends تقدیم به دوستانم

ABSTRACT(EN)

This Master's Thesis presents an analysis of shared bike usage based on weather conditions, utilizing machine learning algorithms capable of predicting future shared bike usage based on weather forecasts. As these services become increasingly integral to urban mobility, the reliability of supporting activities, such as accurate weather forecasts, is crucial to enhancing the overall efficiency of the system.

The machine learning algorithms employed in this Thesis belong to the category of supervised learning techniques. These algorithms learn to predict the desired parameter by analyzing a vast dataset containing numerous historical examples. The dataset, in this case, spans one year of records detailing the number of bikes rented in Vicenza, along with the corresponding weather conditions during that period.

Among the various algorithms explored, the random forest algorithm emerged as the most effective in providing accurate results.

This study identifies an opportunity for the municipality to formulate targeted strategies promoting year-round bike usage based on weather-related patterns. The positive correlation between mean temperature, solar radiation, and extended trip durations in summer suggests a propensity for heightened bike activity during warmer and sunnier conditions. In light of these findings, initiatives such as promoting bike-sharing programs, improving bike-friendly infrastructure, and organizing events during the summer months are recommended to capitalize on this observed trend, fostering increased community engagement and sustainable transportation habits.

ABSTRACT(IT)

Questa tesi di laurea presenta un'analisi dell'utilizzo condiviso delle biciclette in base alle condizioni meteorologiche, utilizzando algoritmi di apprendimento automatico capaci di prevedere l'utilizzo futuro delle biciclette condivise in base alle previsioni meteorologiche. Poiché questi servizi diventano sempre più fondamentali per la mobilità urbana, la affidabilità delle attività di supporto, come le previsioni meteorologiche accurate, è cruciale per migliorare l'efficienza complessiva del sistema.

Gli algoritmi di apprendimento automatico impiegati in questa tesi appartengono alla categoria delle tecniche di apprendimento supervisionato. Questi algoritmi imparano a prevedere il parametro desiderato analizzando un vasto set di dati contenente numerosi esempi storici. Il set di dati, in questo caso, copre un anno di registrazioni che dettagliano il numero di biciclette noleggiate a Vicenza, insieme alle condizioni meteorologiche corrispondenti durante quel periodo.

Tra i vari algoritmi esplorati, l'algoritmo random forest è emerso come il più efficace nel fornire risultati accurati.

Questo studio identifica un'opportunità per il comune di formulare strategie mirate a promuovere l'utilizzo delle biciclette durante tutto l'anno basandosi su modelli legati alle condizioni meteorologiche. La correlazione positiva tra temperatura media, radiazione solare e durata prolungata dei viaggi durante l'estate suggerisce una propensione per un'attività più intensa delle biciclette durante condizioni più calde e soleggiate. Alla luce di questi risultati, si raccomandano iniziative come la promozione di programmi di condivisione delle biciclette, il potenziamento dell'infrastruttura amica delle biciclette e l'organizzazione di eventi durante i mesi estivi per capitalizzare su questa tendenza osservata, favorendo un maggiore coinvolgimento della comunità e abitudini di trasporto sostenibili.

ABBREVIATION TABLE

Variable	Variable description	Type of variable
Ct	The city that the data was collected. All data was collected in Vicenza	nominal
B_N	Each bike is assigned a unique identification number	discrete
V_T	The Ridemovi application is designed for both bike and e-bike sharing, but this study specifically focuses on bike sharing	binary
U_I	Each user is given a unique user ID	discrete
U_T	There are two recorded user types: paying users and pass users	binary
S_T	The exact date and time the trip had started	temporal
E_T	The exact date and time the trip had ended	temporal
Dur	The duration of each trip is recorded in minutes	continues
Dis	The distance between the starting and ending points of trips is recorded in meters	continues
SLa	The system captures the latitude coordinate of the starting point per trip	continues
SLo	The system captures the longitude coordinate of the starting point per trip	continues
ELa	The system captures the latitude coordinate of the ending point per trip	continues
ELo	The system captures the longitude coordinate of the starting point per trip	continues
Mon	The recorded data includes the total payment amount per trip	discrete
Pro	Price difference to calculate payment amount	discrete
Pas	A binary variable to show that the user was pass or paying	binary
S_I	station ID	discrete
Day	day of the month	nominal
Mon	month of the year	nominal
Year	data has been collected in year 2022	discrete
Time	time of the day data has been obtained	temporal
T_M	Medium temperature at 2 m(°c)	continues
Pre	precipitation (mm)	continues
H_MI	minimum humidity at 2m (%)	continues
H_MA	maximum humidity at 2m (%)	continues
S_R	solar radiation (MJ/m2)	continues
W_S	average wind speed(m/s)	continues
M_G	maximum gust(m/s)	continues
D_R	direction prevailing	nominal

General Index

1 INTRODUCTION

Bike sharing stands as a crucial pillar in the realm of sustainability, offering a compelling solution to the environmental and urban challenges of our time. As cities grapple with issues like traffic congestion and air pollution, the significance of bike sharing becomes even more pronounced. By providing a green and efficient alternative to conventional transportation, bike sharing systems actively contribute to the reduction of carbon emissions, easing the strain on urban infrastructure.

Moreover, the importance of having a robust and user-satisfying bike sharing network is underscored by the changing dynamics of weather patterns and the escalating concerns about pollution. In the face of climate change, cities are experiencing more extreme weather events, making sustainable modes of transport like biking increasingly attractive. Bike sharing not only mitigates the impact of these changes but also encourages a healthier, more active lifestyle among citizens.

The utilization of bicycles and bike-sharing programs is inherently influenced by weather conditions and air quality. Weather can significantly impact the feasibility and comfort of cycling. Harsh weather, such as heavy rain, extreme heat, or severe cold, may deter individuals from choosing bikes as a mode of transport. Adverse weather conditions can affect both the safety and convenience of cycling, potentially reducing the uptake of bike-sharing services. Additionally, poor air quality resulting from pollution is a growing concern in many urban areas. Cyclists, especially those using bike-sharing services, may be reluctant to navigate through heavily polluted areas due to health concerns. Conversely, favorable weather conditions can enhance the appeal of cycling, making it a more attractive and enjoyable option for commuters, thereby positively influencing the usage of bike-sharing systems. Therefore, the interplay between weather conditions and air quality plays a significant role in shaping the practicality and popularity of biking and bike-sharing initiatives in urban environments.

This study has been undertaken due to the paramount importance of bike sharing in

fostering sustainability within urban environments. Recognizing the pivotal role weather conditions and air quality play in influencing the practicality and appeal of bike usage, the research aims to comprehensively analyze the relationship between shared bike usage and various weather parameters. Utilizing a machine learning algorithm, the study seeks to extract meaningful patterns from historical data and integrate weather forecasts to predict future shared bike usage. By addressing the dynamic nature of weather-related challenges, this research contributes valuable insights that can inform urban planners and policymakers in optimizing bike-sharing systems for changing weather conditions. Ultimately, the study aligns with the broader goal of promoting sustainable transportation by enhancing the adaptability and resilience of bike-sharing initiatives, particularly in the city of Vicenza, which is situated in northeastern Italy, about 60 kilometers west of Venice and 200 kilometers east of Milan.

The data for this study was sourced from the Municipality of Vicenza for the year 2022. Specifically, information pertaining to daily bike-sharing usage was obtained from municipal records. In tandem with this, comprehensive weather data, encompassing both weather conditions and air quality, was acquired from the ARPAV (Regional Environmental Protection Agency) website. The daily weather conditions were utilized to capture a nuanced understanding of the atmospheric context, including factors such as temperature, precipitation, and wind speed, etc. Simultaneously, air quality metrics were harnessed to gauge the environmental conditions influencing bike-sharing patterns throughout the specified period. This dual-sourced data approach ensures a robust and multifaceted analysis of the relationship between shared bike usage and the prevailing weather conditions in Vicenza during the year 2022.

In this study, our methodology involves the application of a machine learning algorithms to predict bike-sharing usage in Vicenza based on data collected from the Municipality of Vicenza for the year 2022 and daily weather information from the ARPAV website. The machine learning algorithms will be specifically tailored for regression tasks, utilizing historical data to discern patterns and relationships between weather variables and bike-sharing usage. Through rigorous model training, validation, and fine-tuning, we aim to develop a predictive tool capable of forecasting future bike-sharing demand, thus enhancing our understanding of how weather conditions influence the dynamics of bike-sharing patterns in Vicenza.

This study is organized into several chapters, with the first chapter dedicated to a comprehensive literature review of studies conducted on shared bikes across various fields. The second chapter focuses on a descriptive analysis of bike data sourced from the municipality of Vicenza city in northern Italy, utilizing information obtained through the Ridemovi application and preparing the data for analysis. The third chapter involves a literature review of various studies concerning weather data and the methodologies employed for their analysis. The fourth chapter is dedicated to a descriptive analysis of weather data obtained from the ARPAV website and preparing data for analysis. The fifth chapter delves into the analysis conducted in this study, employing various machine learning algorithms to predict the usage of shared bicycles based on different weather conditions and weather quality. The sixth and final chapter pertains to the conclusions derived from the analysis conducted in the fifth chapter.

1.1 Methodology

In this study, machine learning methods will be applied using three specific algorithms: linear regression, support vector machine, and random forest. These algorithms will be theoretically explained as follows:

1.1.1 Machine learning (1)

Machine learning is a branch of computer science that broadly aims to enable computers to “learn” without being directly programmed. It has origins in the artificial intelligence movement of the 1950s and emphasizes practical objectives and applications, particularly prediction and optimization. Computers “learn” in machine learning by improving their performance at tasks through “experience”. In practice, “experience” usually means fitting to data; hence, there is not a clear boundary between machine learning and statistical approaches. Indeed, whether a given methodology is considered “machine learning” or “statistical” often reflects its history as much as genuine differences, and many algorithms (e.g., least absolute shrinkage and selection operator (LASSO), stepwise regression) may or may not be considered machine learning depending on who you ask. Still, despite methodological similarities, machine learning is philosophically and practically distinguishable. At the liberty of (considerable) oversimplification, machine learning generally emphasizes predictive accuracy over hypothesis-driven inference, usually focusing on large, high-dimensional (i.e., having many covariates) data sets. Regardless of the precise distinction between approaches, in

practice, machine learning offers epidemiologists important tools. In particular, a growing focus on "Big Data" emphasizes problems and data sets for which machine learning algorithms excel while more commonly used statistical approaches struggle. This primer provides a basic introduction to machine learning with the aim of providing readers a foundation for critically reading studies based on these methods and a jumping-off point for those interested in using machine learning techniques in epidemiologic research. The "Concepts and Terminology" section of this paper presents concepts and terminology used in the machine learning literature. The "Machine Learning Algorithms" section provides a brief introduction to 3 common machine learning algorithms: linear regression , decision trees and support vector machines These are important and commonly used algorithms that epidemiologists are likely to encounter in practice, but they are by no means comprehensive of this large and highly diverse field. The following two sections, "Ensemble Methods" and "Epidemiologic Applications," extend this examination to ensemble-based approaches and epidemiologic applications in the published literature. "Brief Recommendations" provides some recommendations for incorporating machine learning into epidemiologic practice, and the last section discusses opportunities and challenges.

1.1.2 Concepts and terminology

For epidemiologists seeking to integrate machine learning techniques into their research, language and technical barriers between the two fields can make reading source materials and studies challenging. Some machine learning concepts lack statistical or epidemiologic parallels, and machine learning terminology often differs even where the underlying concepts are the same. Here we briefly review basic machine learning principles and provide a glossary of machine learning terms and their statistical/epidemiologic equivalents (Table 1).

Machine Learning Term(s)	Epidemiology Term(s)	Definition and Notes	Example
Attribute, feature, predictor, or field	Independent variable	Machine learning uses various terms to reference what epidemiologists would consider an "independent variable," including	In a data set with 4 independent variables (BMIa, age, race, and SES) and a dependent variable

		attribute, feature, predictor, and field.	(diabetes mellitus), BMI, age, race, and SES are attributes.
Domain	Range of possible variable values	The domain is the set of possible values of an attribute. It can be continuous or categorical/binary.	If race is recorded in a data set as "1 = Caucasian, 2 = African-American, and 3 = other," its domain is categorical and includes only the 3 referenced categories.
Input and output	Independent (exposure) and dependent (outcome) variables	In machine learning, "input" refers to all of the predictors or independent variables that enter the model, and "output" generally refers to the predicted value (whether a number, classification, etc.) of the dependent variable or outcome.	BMI, age, race, and SES are model input. In a binary classification algorithm, the model output is a prediction of whether a subject does (D = 1) or does not (D = 0) have diabetes.
Classifier, estimator	Model	"Classifiers" or "estimators" are used generally in the machine learning literature to refer to algorithms that perform a prediction or classification of interest. Their less common, though more technical, usage specifically refers to fully parameterized models that are used to predict or classify.	A decision tree is one type of machine learning classifier (general usage). The more specific usage of this term would refer only to a parameterized decision tree that has been fit in a data set (e.g., that predicts diabetes outcomes from BMI, age, sex, and SES).
Learner	Model-fitting algorithm	A learner inputs a training set and outputs a classifier. Usually, but not always, learner refers to the fitting algorithm, while classifier refers to the fitted model.	In decision tree learning, the classification and regression trees (CART) algorithm, developed by Breiman et al. (27)

			in 1984, is one of multiple available learners for developing a decision tree classifier.
Dimensionality	No. of covariates	No. of independent variables under consideration in a model.	A data set with 4 independent variables (BMI, age, race, and SES) and a dependent variable (diabetes) has 4 dimensions.
Label	Value of dependent variables, outcomes	A variable's label is its value for each observation (e.g., 0 or 1). Although labels can technically describe any variable, common shorthand is that "labeled data" refers to data in which the dependent variable assumes a value for all observations.	In a data set for which an investigator has collected information on diabetes status (outcome) for all subjects, this is "labeled" data. The label for diabetes is 0 or 1. Partially labeled data would have diabetes status missing for some subjects.
Imbalanced data	Data set in which some cases or risk categories occur much less frequently than the others	In imbalanced machine learning data sets, the outcome or another risk category of interest occurs much less frequently, either because of the intrinsic nature of the problem (e.g., a rare disease in a database of medical records) or because of the sampling strategy (e.g., prevalence of cases in the study population is much lower than that in the target/source population). Heavily imbalanced data may	Assume a hypothetical data set of pediatric, normal-weight patients in which the prevalence of diabetes is 2%. This data set is imbalanced because the outcome is very rare, which can lead to poor sensitivity of classification algorithms without parameter tuning or other corrective methods. This imbalance is due to the

		pose challenges in some classification algorithms and require tuning parameters in order to correct for or otherwise address this imbalance. One method for addressing imbalanced data sets is to "balance" them artificially, either by oversampling instances of the minority class or undersampling instances of the majority class.	intrinsic nature of the population we are evaluating (i.e., healthy children) and not due to the sampling strategy or other bias.
Loss function	Error measure	In machine learning, a loss function is generally considered a penalty for misclassification when assessing a model's predictive performance.	A simple loss function may be the absolute value of (predicted value minus true value). If a model predicts that a subject has diabetes (D = 1) and the subject does not (D = 0), the value of the loss function for this prediction is "1."

Table 1: Glossary of Machine Learning and Epidemiology Terminology

1.1.3 Supervised, unsupervised, and semi-supervised learning

Machine learning is broadly classifiable by whether the computer's learning (i.e., model-fitting) is "supervised" or "unsupervised." Supervised learning is akin to the type of model-fitting that is standard in epidemiologic practice: The value of the outcome (i.e., the dependent variable), often called its "label" in machine learning, is known for each observation. Data with specified outcome values are called "labeled data." Common supervised learning techniques include standard epidemiologic approaches such as linear and logistic regression, as well as many of the most popular machine learning algorithms (e.g., decision trees, support vector machines).

In unsupervised learning, the algorithm attempts to identify natural relationships and groupings within the data without reference to any outcome or the "right answer".

Unsupervised learning approaches share similarities in goals and structure with statistical approaches that attempt to identify unspecified subgroups with similar characteristics (e.g., “latent” variables or classes). Clustering algorithms, which group observations on the basis of similar data characteristics (e.g., both oranges and beach balls are round), are common unsupervised learning implementations. Examples may include k-means clustering and expectation-maximization clustering using Gaussian mixture models.

Semi-supervised learning fits models to both labeled and unlabeled data. Labeling data (outcomes) is often time-consuming and expensive, particularly for large data sets. Semi-supervised learning supplements limited labeled data with an abundance of unlabeled data with the goal of improving model performance (studies show that unlabeled data can help build a better classifier, but appropriate model selection is critical). For example, in a study of Web page classification, fit a naive Bayes classifier to labeled data and then used the same classifier to probabilistically label unlabeled observations (i.e., fill in missing outcome data). They then retrained a new classifier on the resulting, fully labeled data set, thereby achieving a 30% increase in Web page classification accuracy on data outside of the training set. Semi-supervised learning can bear some similarity to statistical approaches for missing data and censoring (e.g., multiple imputation), but as an approach that focuses on imputing missing outcomes rather than missing covariates.

1.1.4 Classification versus regression algorithms

Within the domain of supervised learning, machine learning algorithms can be further divided into classification or regression applications, depending upon the nature of the response variable. In general, in the machine learning literature, classification refers to prediction of categorical outcomes, while regression refers to prediction of continuous outcomes. We use this terminology throughout this primer and are explicit when referring to specific regression algorithms (e.g., logistic regression). Many machine learning algorithms that were developed to perform classification have been adapted to also address regression problems, and vice versa.

1.1.5 Generative versus discriminative algorithms

Machine learning algorithms, both supervised and unsupervised, can be discriminative or generative. Discriminative algorithms directly model the conditional probability of an outcome, Pr(y|x) (the probability of y given x), in a set of observed data—for example,

the probability that a subject has type 2 diabetes mellitus given a certain body mass index (BMI; weight (kg)/height $(m)^2$). Most statistical approaches familiar to epidemiologists (e.g., linear and logistic regression) are discriminative, as are most of the algorithms discussed in this primer.

In contrast, while generative algorithms can also compute the conditional probability of an outcome, this computation occurs indirectly. Generative algorithms first model the joint probability distribution, Pr(x, y) (the probabilities associated with all possible combinations of x and y), or, continuing our example, a probabilistic model that accounts for all observed combinations of BMIs and diabetes outcomes (Table 2). This joint probability distribution can be transformed into a conditional probability distribution in order to classify data, as Pr(y|x) = Pr(x, y)/Pr(x). Because the joint probability distribution models the underlying data-generating process, generative models can also be used, as their name suggests, for directly generating new simulated data points reflecting the distribution of the covariates and outcome in the modeled population. However, because they model the full joint distribution of outcomes and covariates, generative models are generally more complex and require more assumptions to fit than discriminative algorithms. Examples of generative algorithms include naive Bayes and hidden Markov models.

Table 2: Matrix of Joint Probabilities for Body Mass Indexa (x) and Diabetes Mellitus (y) in a Data Set With 4 Dichotomized Observations: (0, 1), (0, 1), (0, 1), and (0, 0)

Diabetes Status	BMI Status	
	Overweight BMI = 1	Overweight BMI = 0
D = 1	0/4	1/4
D = 0	2/4	1/4

Table 2: Abbreviation: BMI, body mass index= Weight (kg)/height (m)2

1.1.6 Reinforcement learning

In reinforcement learning, systems learn to excel at a task over time through trial and error. Reinforcement learning techniques take an iterative approach to learning by

obtaining positive or negative feedback based on performance of a given task on some data (whether prediction, classification, or another action) and then self-adapting and attempting the task again on new data (though old data may be reencountered). Depending on how it is implemented, this approach can be akin to supervised learning, or it may represent a semi-supervised approach (as in generative adversarial neural networks). Reinforcement learning algorithms often optimize the use of early, "exploratory" versions of a model—that is, task attempts—that perform poorly to gain information to perform better on future attempts, and then become less labile as the model "learns" more. Medical and epidemiologic applications of reinforcement learning have included modeling the effect of sequential clinical treatment decisions on disease progression (e.g., optimizing first- and second-line therapy decisions for schizophrenia management) and personalized, adaptive medication dosing strategies. For example, Nemati et al. used reinforcement learning with artificial neural networks in a cohort of intensive-care-unit patients to develop individualized heparin dosing strategies that evolve as a patient's clinical phenotype changes, in order to maximize the amount of time that blood drug levels remain within the therapeutic window.

1.1.7 Cross-validation

Cross-validation is a resampling technique that is often used to assess the adequacy of a statistical model. The idea is to randomly split the data into one set to fit the model and a second separate set to test the accuracy of the model for prediction. This approach came about in classification problems because the resubstitution method which tests the model on the same data used to fit the model is optimistically biased and the bias can be very large in small samples.

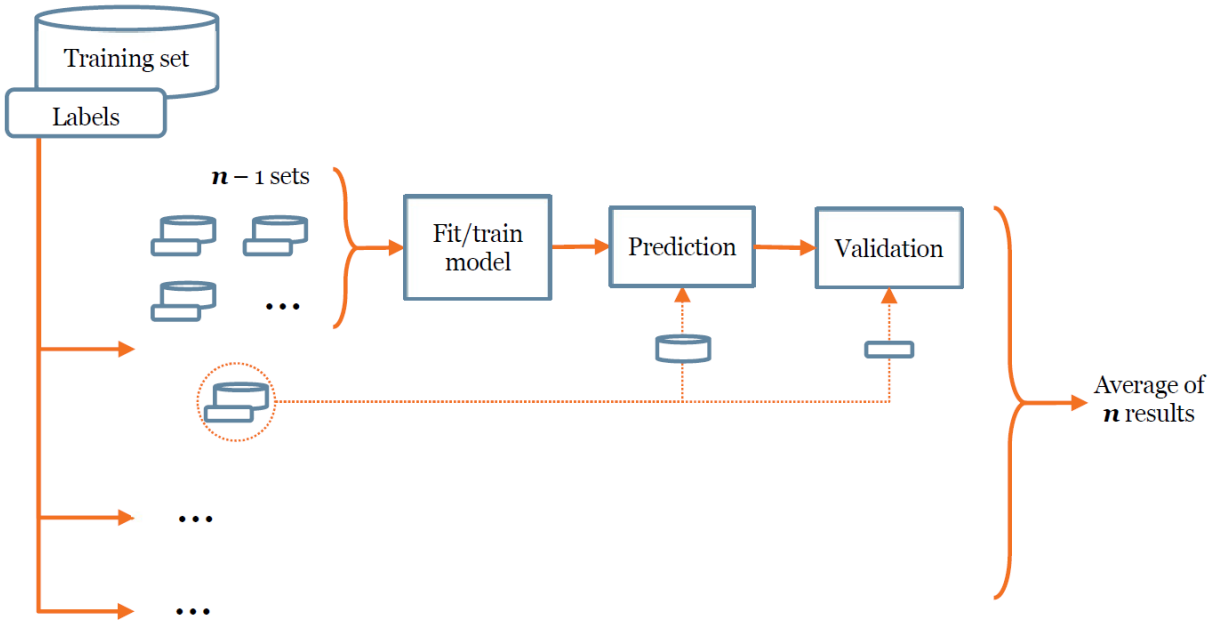


Figure 1 – Cross-validation

A special case of cross-validation is called leave-one-out. The leave-one-out method fits the model to all but one data point and then tests the accuracy of the model on the data point left out. Because this is such an extreme splitting of the data, the evaluation on just one data point is of course not enough to get a good estimate of the model's prediction accuracy. So for the leave-one-out estimate all n splits (leaving n − 1 points for fitting and keeping one for testing) are used to get an overall estimate of the model's prediction accuracy based on these n evaluations.

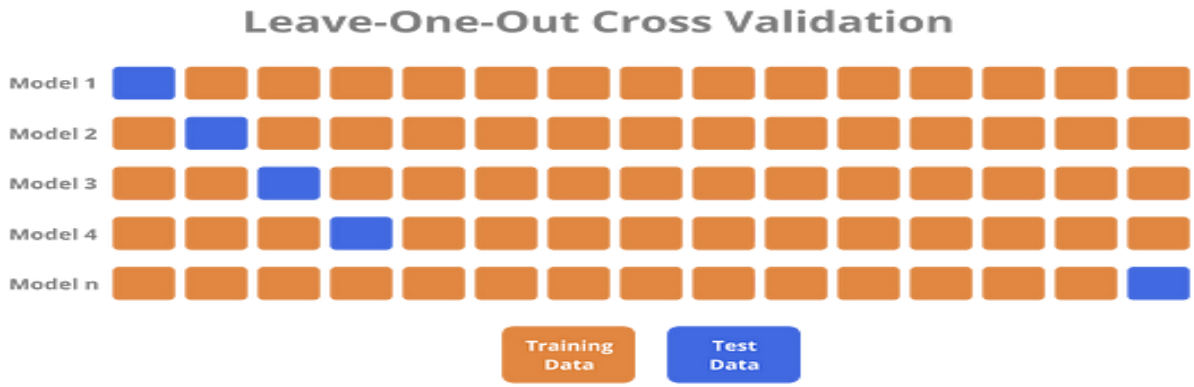


Figure 2 – Leave-one-out Cross Validation

For the problem of estimating the error rates for a classification algorithm (with two or

more classes), the leave-one-out approach was popular because it led to nearly unbiased estimates and fully exploits the data in contrast to a 50–50 split cross-validation which only uses half the data in the fit and the other half in the evaluation. However, Efron47 was the first to discover that for linear discriminant functions with multivariate normal class-conditional densities, a form of the bootstrap algorithm called the 632 estimator is superior to the leave-one-out estimator.

Other important applications of cross-validation have to do with smoothing density functions and splines. Here cross-validation is used to determine the appropriate degree of smoothing. Similarly, cross-validation can be used to determine the order of a model or the subset of variables to use in regression models to protect against overfitting.

It is similar to other methods, such as the Akaike information criterion (AIC), which penalizes the likelihood function for the number of variables included in the model. For AIC, you find the model that minimizes the penalized likelihood. For cross-validation, you look for the model that best predicts the observations that were left out of the fit.

Cross-validation is also used to determine how much to prune a classification or regression tree when using the CART procedure for constructing these types of trees.

For neural networks cross-validation has been used to make a proper choice of the number of nodes. So neural networks and classification trees, which are important data mining tools rely on cross-validation.

1.1.8 Machine learning algorithms

In this section, we introduce 3 common machine learning algorithms: linear regression, decision trees and support vector machines. For each, we include a brief description, summarize strengths and limitations, and highlight implementations available on common statistical computing platforms. This section is intended to provide a high-level introduction to these algorithms, and we refer interested readers to the cited references for further information.

1.1.9 Linear regression

Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. The goal is to find the best-fitting line (or hyperplane in the case of multiple independent variables) that minimizes the difference between the predicted values and the actual values of the dependent variable. The equation of a simple linear regression model with one independent variable is often written as:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

where:

- Y is the dependent variable.
- X is the independent variable.
- β_0 is the y-intercept (constant term).
- β_1 is the slope of the line.
- ε represents the error term.

1.1.9.1 Strengths and limitations

Strength of Linear regression

1.Simplicity: Linear regression is a simple and easy-to-understand method, making it a good starting point for analyzing relationships between variables.

2. Interpretability: The coefficients β_0 and β_1 have clear interpretations. β_0 represents the expected value of the dependent variable when the independent variable is zero, and β_1 represents the change in the dependent variable for a one-unit change in the independent variable.

3.Efficiency: Linear regression can be computationally efficient and works well with large datasets.

Limitations of Linear Regression:

1.Linearity Assumption: Linear regression assumes a linear relationship between the independent and dependent variables. If the relationship is not linear, the model may provide inaccurate predictions.

2. Sensitivity to Outliers: Linear regression is sensitive to outliers, which can significantly influence the regression equation and coefficients.

3. Assumption of Independence: The model assumes that the residuals (the differences between predicted and actual values) are independent. If this assumption is violated, it can lead to biased and inefficient parameter estimates.

4. Multicollinearity: When multiple independent variables are included in the model, multicollinearity (high correlation between independent variables) can cause issues in accurately estimating individual variable effects.

5. Homoscedasticity: Linear regression assumes homoscedasticity, meaning that the variance of the residuals is constant across all levels of the independent variable. Violations of this assumption can lead to inefficient parameter estimates.
Despite these limitations, linear regression is a valuable tool in many situations and serves as a foundation for more advanced modeling techniques. It is important to carefully assess the assumptions and limitations of the method before applying it to a particular dataset.

1.1.10 Decision trees

Decision trees (i.e., classification and regression trees (CART)) create a series of decision rules based on continuous and/or categorical input variables to predict an outcome. Classification trees predict categorical outcomes, and regression trees predict continuous outcomes. CART analysis has been popularized as an umbrella term for any decision tree learning method. However, "CART" is also a common implementation algorithm in the epidemiologic and medical literature, although a number of other decision tree algorithms have also been developed (e.g., ID3, CHAID).
Figure 3 presents a hypothetical classification tree for a binary outcome, diabetes. To derive a decision tree, the algorithm applies a splitting rule on successively smaller partitions of data, with each partition being a node on the tree. The partition consisting of all data is the root node; in Figure 3 this node is split on the basis of BMI. Splits are selected to minimize some measure of node impurity (i.e., diversity of classes) or heterogeneity (i.e., variance) in each resulting partition (the "daughter nodes"). The splitting process repeats on each branch of the tree until additional splits yield no further reductions in node impurity, or some other stopping criterion is reached (e.g., a specified minimum number of observations in terminal nodes or the value at which error is minimized in cross-validation). In many algorithms, this splitting is often followed by a "pruning" step in which partitions are remerged (i.e., some bottom nodes are removed, making the final tree smaller) based on some criterion designed to increase generalizability.

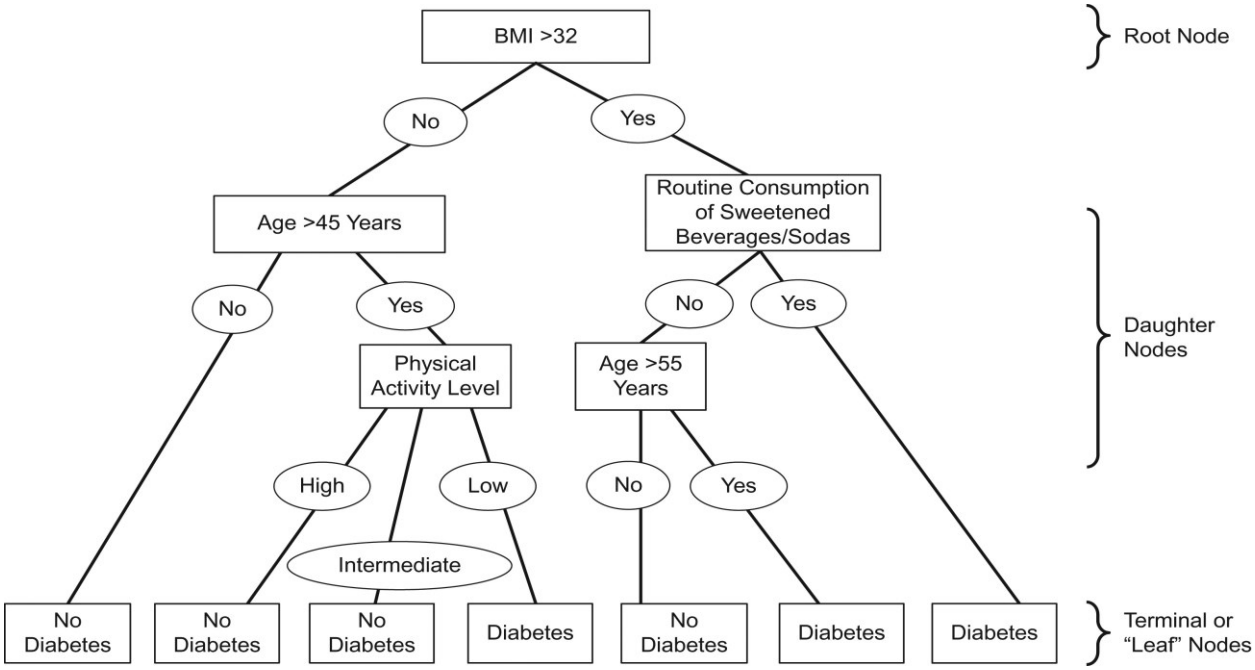


Figure 3: hypothetical decision tree to predict type 2 diabetes. BMI is the primary factor, with age, sweetened beverage consumption, and physical activity as subsequent factors.

Predictions are made in terminal nodes based on majority rule, using algorithm-derived cutpoints for BMI and age, suggesting interaction effects between age and diabetes based on BMI levels and sweetened beverage consumption.

A hypothetical classification decision tree for predicting a binary outcome, type 2 diabetes mellitus. Body mass index (BMI; weight (kg)/height $(m)^2$) occupies the root node (the most discriminatory variable in the data set); age, consumption of sweetened beverages, and physical activity occupy daughter nodes; and predicted diabetes status (yes/no) is reflected in the terminal or "leaf" nodes. Terminal node predictions proceed on the basis of simple majority rule (e.g., if 60% of patients in a terminal node are diabetes-positive, the entire terminal node will be classified as "Diabetes"). The cutpoints for the continuous variables, BMI and age, are algorithm-derived. The presence of age at different cutpoints in 2 different daughter nodes reflects likely interaction effects: The relationship between age and diabetes differs in patients with BMI ≤32 compared with patients with BMI >32 who

1.1.10.1 Strengths and limitations

Decision trees are generally easy to understand—its having been said that "$[o]_n$ interpretability, trees rate an A+" -making their output ideal for a range of target audiences. They are also flexible to nonlinear covariate effects and can incorporate

higher-order interactions between covariates. Trees may lose information by dichotomizing or categorizing variables where associations are continuous, and they can be unstable to even small data changes. Because most decision tree algorithms are "greedy" (splitting decisions are locally optimized at nodes), through a domino effect, dramatically different trees can result if even a single higher-level node shifts to a different variable. Hence, decision trees can be highly sensitive to small perturbations in data. Perhaps most fundamentally, decision trees are prone to overfitting, and their ultimate utility depends heavily on appropriately implemented pruning and/or stopping criteria. Ensemble-based decision trees (e.g., random forests) can address some of these concerns (see "Ensemble Methods" section), but they do not produce a single, easily interpretable tree.do not routinely consume sweetened beverages.

1.1.11 Support vector machines

Support vector machines (SVMs) are a set of supervised learning methods used for classification and regression problems. SVMs construct an optimal boundary, called a hyperplane, that best separates observations of different classes. In 1 dimension, this boundary is a point; in 2 dimensions, a line; and in 3, a plane (Figure 4). However, many observations often need to be transformed before they can be separated by a hyperplane. SVMs address this problem by applying a data transformation called a "kernel function" to the data. Kernel functions project the data into a higher-dimensional space where the input variables are separable (Figure 4). The optimal kernel function is usually chosen from a set of commonly used kernel functions selected through cross-validation. Popular kernel functions include polynomial kernel, gaussian kernel, and sigmoid kernel. Following kernel function transformation, the best hyperplane maximizes the separation between the different classes (i.e., the margin, defined as the distance from the hyperplane to the closest data point), while tolerating a specified level of misclassification. SVMs are traditionally used for binary classification, but multiple pairwise comparison can be applied for multiclass classification. Extensions to SVM techniques have also been developed that can be used to predict continuous outcomes (called support vector regression).

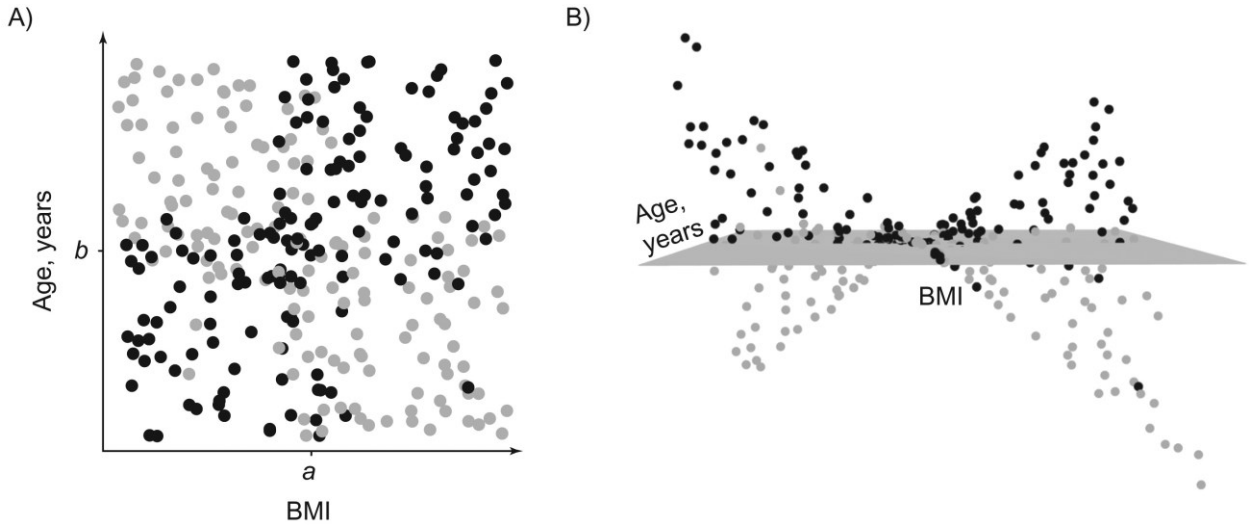


Figure 4: An illustration of data transformation with a support vector machine for predicting diabetes status. A) Hypothetical age and body mass index (BMI; weight (kg)/height (m)2) distribution of diabetic (black dots) and nondiabetic (gray dots) patie

An illustration of data transformation with a support vector machine for predicting diabetes status. A) Hypothetical age and body mass index (BMI; weight (kg)/height $(m)^2$) distribution of diabetic (black dots) and nondiabetic (gray dots) patients in 2-dimensional space. a and b are fixed parameters estimated from the data (see text). B) After transformation, these dots/patients who are not linearly separable in 2-dimensional space become linearly separable in 3-dimensional space. A hyperplane in 3-dimensional space is shown as a surface.

In Figure 4, persons with and without diabetes cannot be separated by a line in the 2-dimensional space based upon the predictors, age and BMI (Figure 4A). However, when we project the data into a 3-dimensional space by applying a kernel given by φ((age, BMI) = (age, BMI, (BMI − a) × (age − b)), where φ(.) is the feature mapping, a and b are fixed parameters estimated from the data, (age, BMI): the original two-dimensional input data representing age and body mass index. the data are now separable in the 3-dimensional space by a plane (Figure 4B).

1.1.11.1 Strengths and limitations

SVMs generally demonstrate low misclassification error and scale well to high-dimensional data. SVMs have reasonable interpretability, especially when a kernel function is not used. Where a kernel function is necessary, however, selecting the optimal kernel function typically requires experimenting with a set of standard functions. This approach can be time-consuming and does not guarantee that the set of

standard kernel functions that were evaluated included the optimal function, and in some cases hand-crafted kernel functions are used instead.

Additionally, in this study, Shapley values are employed to elucidate the output of machine learning models, especially in the context of black-box models. This method is applied using The Kernel SHAP Method.

1.1.12 The Kernel SHAP Method

Assume a predictive model f(x) for a response value y with features $x \in R^M$, trained on a training set, and that we want to explain the predictions for new sets of data. This may be done using ideas from cooperative game theory, letting a single prediction take the place of the game being played and the features the place of the players. Letting N denote the set of all M players, and $S \subseteq N$ be a subset of |S| players, the "contribution" function v(S) describes the total expected sum of payoffs the members of S can obtain by cooperation. The Shapley value ([Shapley ,1953]) is one way to distribute the total gains to the players, assuming that they all collaborate. The amount that player I gets is then

$$\phi i(v) = \phi i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(M - |S| - 1)!}{M!} (v(S \cup \{i\}) - v(S)),$$

f(x): Represents the predictive model, which takes a feature vector x ∈ ℝ^M and predicts a response value y.

N: Denotes the set of all M players (features in this context).

S: A subset of players (features), where S ⊆ N, and |S| represents the number of players in the subset.

v(S): The "contribution" function, which describes the total expected sum of payoffs that the members of subset S can obtain by cooperation.

ϕi(v): The Shapley value for player i. It represents the fair share or average contribution of player i to all possible combinations of players in N.

that is, a weighted mean over all subsets S of players not containing player i. Lundberg and Lee (2017) define the contribution function for a certain subset S of these features x_S as $v(S)=E[f(x)|x_S]$, the expected output of the predictive model conditional on the feature values of the subset. [Lundberg and Lee ,2017] names this type of Shapley values SHAP (SHapley Additive exPlanation) values. Since the conditional expectations can be written as

$$E[f(x)|x_s = x_S^*] = E[f(x_{S^-}, x_S)|x_S = x_S^*] = \int f(x_{S^-}, x_S^*)p(x_{S^-}|x_S = x_S^*)dx_{S^-},$$

the conditional distributions $p(x_{S^-}|x_S=x^*_S)$ are needed to compute the contributions. The Kernel SHAP method of [Lundberg and Lee ,2017] assumes feature independence, so that $p(x_{S^-}|x_S=x^*_S)=p(x_{S^-})$. If samples $x^k_{S^-}$,k=1,…,K, from $p(x_{S^-}|x_S=x^*_S)$ are available, the conditional expectation in above can be approximated by

$$\text{vKerSHAP}(S) = \frac{1}{K}\sum_{k=1}^{k} f(x_{S^-}^k, x_S^*).$$

In Kernel SHAP, $x^k_{S^-}$,k=1,…,K are sampled from the S^- -part of the training data, independently of x_S . This is motivated by using the training set as the empirical distribution of x_{S^-} , and assuming that x_{S^-} is independent of $x_S=x^*_S$. Due to the independence assumption, if the features in a given model are highly dependent, the Kernel SHAP method may give a completely wrong answer. This can be avoided by estimating the conditional distribution $p(x_{S^-}|x_S=x^*_S)$ directly and generating samples from this distribution. With this small change, the contributions and Shapley values may then be approximated as in the ordinary Kernel SHAP framework. [Aas, Jullum, and Løland ,2019] propose three different approaches for estimating the conditional probabilities. The methods may also be combined, such that e.g. one method is used when conditioning on a small number of features, while another method is used otherwise.

1.1.13 Multivariate Gaussian Distribution Approach

The first approach arises from the assumption that the feature vector x stems from a multivariate Gaussian distribution with some mean vector μ and covariance matrix Σ. Under this assumption, the conditional distribution $p(x_{S^-}|x_S=x^*_S)$ is also multivariate Gaussian $N|S^-|(\mu_{S^-|S},\Sigma_{S^-|S})$, with analytical expressions for the conditional mean vector $\mu_{S^-|S}$ and covariance matrix $\Sigma S^-|S$, see [Aas, Jullum, and Løland ,2019] for details. Hence, instead of sampling from the marginal empirical distribution of x_{S^-} approximated by the training data, we can sample from the Gaussian conditional distribution, which is fitted using the training data. Using the resulting samples $x^k_{S^-}$,k=1,…,K, the conditional expectations be approximated as in the Kernel SHAP.

1.1.14 Gaussian Copula Approach

If the features are far from multivariate Gaussian, an alternative approach is to instead represent the marginals by their empirical distributions and model the dependence

structure by a Gaussian copula. Assuming a Gaussian copula, we may convert the marginals of the training data to Gaussian features using their empirical distributions, and then fit a multivariate Gaussian distribution to these.

To produce samples from the conditional distribution $p(x_{S^-}|x_S=x^*_S)$, we convert the marginals of x_S to Gaussians, sample from the conditional Gaussian distribution as above, and convert the marginals of the samples back to the original distribution. Those samples are then used to approximate the sample from the resulting multivariate Gaussian conditional distribution. While other copulas may be used, the Gaussian copula has the benefit that we may use the analytical expressions for the conditionals $\mu_{S^-|S}$ and $\Sigma_{S^-|S}$.

Finally, we may convert the marginals back to their original distribution and use the resulting samples to approximate the conditional expectations as in the Kernel SHAP.

1.1.15 Empirical Conditional Distribution Approach

If both the dependence structure and the marginal distributions of x are very far from the Gaussian, neither of the two aforementioned methods will work very well. Few methods exist for the non-parametric estimation of conditional densities, and the classic kernel estimator [Rosenblatt ,1956] for non-parametric density estimation suffers greatly from the curse of dimensionality and does not provide a way to generate samples from the estimated distribution. For such situations, [Aas, Jullum, and Løland ,2019] propose an empirical conditional approach to sample approximately from $p(x_{S^-}|x^*_S)$. The idea is to compute weights $w_S(x^*,x^i)$, i=1,...,ntrain for all training instances based on their Mahalanobis distances (in the S subset only) to the instance x^* to be explained. Instead of sampling from this weighted (conditional) empirical distribution, [Aas, Jullum, and Løland ,2019] suggests a more efficient variant, using only the K instances with the largest weights:

$$\text{vcondKerSHAP}(S) = \frac{\sum_{k=1}^{K} w_S(x^*, x^{[k]}) f(x_{S^-}^{[k]}, x_S^*)}{\sum_{k=1}^{K} w_S(x^*, x^{[k]})},$$

The number of samples K to be used in the approximate prediction can for instance be chosen such that the K largest weights accounts for a fraction η, for example 0.9, of the total weight. If K exceeds a certain limit, for instance 5,000, it might be set to that limit. A bandwidth parameter σ used to scale the weights, must also be specified. This choice may be viewed as a bias-variance trade-off. A small σ puts most of the weight to a few

of the closest training observations and thereby gives low bias, but high variance. When $\sigma \to \infty$, this method converges to the original Kernel SHAP assuming feature independence. Typically, when the features are highly dependent, a small σ is typically needed such that the bias does not dominate. [Aas, Jullum, and Løland ,2019] show that a proper criterion for selecting σ is a small-sample-size corrected version of the AIC known as AICc. As calculation of it is computationally intensive, an approximate version of the selection criterion is also suggested. Details on this is found in [Aas, Jullum, and Løland ,2019].

1.1.16 Conditional Inference Tree Approach

The previous three methods can only handle numerical data. This means that if the data contains categorical/discrete/ordinal features, the features first have to be one-hot encoded. When the number of levels/features is large, this is not feasible. An approach that handles mixed (i.e numerical, categorical, discrete, ordinal) features and both univariate and multivariate responses is conditional inference trees [Hothorn, Hornik, and Zeileis ,2006].

Conditional inference trees is a special tree fitting procedure that relies on hypothesis tests to choose both the splitting feature and the splitting point. The tree fitting procedure is sequential: first a splitting feature is chosen (the feature that is least independent of the response), and then a splitting point is chosen for this feature. This decreases the chance of being biased towards features with many splits [Hothorn, Hornik, and Zeileis ,2006].

We use conditional inference trees (ctree) to model the conditional distribution, $p(x_{S^{-}}|x^{*}_{S})$, found in the Shapley methodology. First, we fit a different conditional inference tree to each conditional distribution. Once a tree is fit for given dependent features, the end node of x^{*}_{S} is found. Then, we sample from this end node and use the resulting samples, $x^{k}_{S^{-}}, k=1,\ldots,K$, when approximating the conditional expectations as in Kernel SHAP. See [Redelmeier, Jullum, and Aas ,2020] for more details.

The conditional inference trees are fit using the party and partykit packages (Hothorn and Zeileis (2015)).

2 Bike Literature Review

With the growing emphasis on sustainability in the contemporary world, numerous research studies have explored various facets of bicycle usage and bike sharing. These investigations have delved into diverse aspects, including the influence of weather conditions, air quality, calendar events, and more. The data for such research endeavors has been drawn from a range of sources, encompassing surveys, GPS data, direct observations, and other relevant means.

Furthermore, researchers have employed a variety of analytical techniques to decipher the insights within the collected data. Among these approaches, machine learning algorithms have gained prominence for yielding the highest predictive accuracy and effectiveness in analysis. This underscores the pivotal role of machine learning in extracting meaningful patterns and correlations from the complex datasets associated with bicycle usage and bike sharing. In recent years a lot of research have been done in this topic.

[Huthaifa I. Ashqar et al, 2019] conducted a study on bike counts in a bike-sharing system, investigating the influence of weather conditions. To achieve this, a dataset was utilized. The dataset was collected from August 2023 to August 2025 and comprised essential information such as station ID, the number of available bikes, number of available docks, and the precise time of recording. The time data included details like the year, month, day-of-the-month, time-of-the-day, and minutes at which each incident was recorded. As each minute was documented for 70 stations in San Francisco over the span of two years, the dataset contained a substantial number of recorded incidents. To identify instances when there were changes in bike counts at each station, the data was subjected to a change detection process. The similar research was done by [Joost de Kruijf et al., 2021], the researchers conducted a segmentation of all GPS data into journeys and stages (segments). They used a tool to impute the specific travel purpose

for each journey based on the location of origin and destination. This imputation was done by considering the proximity to various facilities and using information from self-reported data about facility locations. The identified travel purposes included work, shopping groceries, social, and recreational activities.

After segmenting the data, the researchers determined the number of different modes of transport (stages) that were used during one relocation. They also identified the specific mode of transport used for each stage.

Focused on using daily commuting. GPS data was collected from January 2014 until mid-September 2014. The dataset consisted of a total of 242,179 journeys and 355,996 stages. From these records, the researchers selected 71,772 journeys made by 573 participants, specifically those that were commutes from their "home" to their "work" destinations.

To account for trip chains, where participants made stops at certain locations (e.g., drinks after work) on their way home, these stops were treated as separate journeys in the analysis. However, in [Huthaifa I. Ashqar et al, 2019], through pre-processing, specific features were extracted from the dataset, including the station ID, number of available bikes, month, day-of-the-week, and time-of-the-day. Time-of-the-day was transformed to a time resolution of 0:23 (representing hours in a day) and used as a feature in the study. One year later a study conducted by [Jan Wessel ,The University of Münster, 2020] and hourly bicycle count data from 188 bicycle counting stations located in 37 different cities and regions across Germany was collected. Among these stations, 140 provided hourly bicycle counts for the entire sample period, spanning from January 1, 2017, to December 31, 2018. Additionally, 175 stations offered data for at least 365 consecutive days.

On average, each bicycle counting station in the sample provided hourly data for approximately 668.2 days, covering around 91.5% of the total sample period.

For their regression analysis, the researchers decided to utilize data from all 188 stations. They made this choice because the missing observations appeared to be randomly distributed across the entire sample period. By including all stations in the analysis, they ensured the data's representativeness and minimized any potential biases in the results. Similar to this study a 6-year dataset was collected by [Craig Morton, 2020] to analyze data on cycle hires, weather conditions, and local air pollutant

concentrations at daily intervals from January 1st, 2012, to January 1st, 2018. The number of cycle hires on the London Bicycle Sharing Scheme (LBSS) was obtained from Transport for London's system management platform, which offers public access to disaggregated trip data for research purposes. Two distinct demand levels were recorded: one for LBSS members (individuals with annual subscriptions) and the other for individuals paying the £2 daily charge by debit or credit card. These separate records were used to distinguish user types, with LBSS members representing regular cyclists and individuals paying by debit or credit card representing casual cyclists. Three years later a study titled Investigating the temporal differences among bike sharing users through comparative analysis based on count, time series, and data mining models conducted by [Ahmed Jaber et al., 2023]. They collected bike usage statistics from Citi Bike's website for the months of April, May, June, and July 2014. The dataset comprises information on start station id, end station id, station latitude, station longitude, and trip time for each bike trip. Among the 332 bike stations with at least one originating bike trip, 253 were located in Manhattan, and the remaining 79 were situated in Brooklyn. The researchers then conducted data processing to determine the number of bike trips between each station pair specifically during morning rush hours.

In their study, [Ahmadreza Faghih-Imani et al, 2017] developed an information systems infrastructure with a web crawler to collect real-time snapshot data of bike sharing systems from the programs' websites. The dataset covers the period from May 1 to September 20, 2009 and captures the state information of all bike stations in the city at 5-minute intervals (due to crawler restrictions). However, intermittent errors in the information systems infrastructure resulted in some missing data for certain stations and time points. The researchers performed data cleaning to obtain 34 days and 21 days of 5-minute state data for each station in Barcelona and Seville, respectively. Trip rate information was derived from this collected state data. Additionally, the latitude and longitude of each bike station in the city and the total number of bike stations in each Spatial Contextual Division (SCD) were recorded, creating a unique longitudinal dataset on usage at each individual station and SCD.

To convert this data into the dependent variables used in their models, the researchers computed the total arrival and departure rates at each station at a 5-minute level. They noted that total arrivals and departures of bikes could be influenced by both customer

usage and rebalancing operations by the operator. To distinguish between these influences, they divided the apparent total arrival rate and total departure rate into four components: (i) arrival rate due to customer usage, (ii) departure rate due to customer usage, (iii) refilling rate due to operator rebalancing, and (iv) removal rate due to operator rebalancing. This separation was achieved using a heuristic approach.
The heuristic approach is based on the assumption that when the operator rebalances bikes at a station, there will usually be a significant change in the total number of bikes at the station (either refilling or removal) in a short span of time, compared to the rate at which users borrow and return bikes. Therefore, when the researchers observed a 5-minute total arrival (or total departure) rate that exceeded the 99th percentile of the arrival (or departure) rate for that station, they assumed that a rebalancing operation (refilling or removal) was performed by the operator. Specifically, the heuristic assumed that when the total arrival (or total departure) rate exceeded the 99th percentile of the arrival (or departure) rate for that station, the arrival (or departure) rate due to public demand was approximated as the average rate of the last two 5-minute arrivals (or departures) for that station, and the remainder was attributed to refilling (or removal) by the operator. The 5-minute level data of the arrival rate, departure rate, refilling rate, and removal rate were further aggregated temporally and spatially to create their corresponding hourly metrics at the SCD-hour level. Four years later a study was conducted by [Hongtai Yang et al., 2021], and the same data was gathered. The researchers gathered an extensive dataset comprising over 2,870,000 bike sharing trips spanning from March 2019 to October 2019. This dataset encompasses valuable trip-related details such as the start and end times, originating and concluding stations, trip duration, user category (including annual members, 15-day members, and non-members), and demographic information for annual and 15-day members, covering age and gender (the demographic attributes of non-members remain undisclosed). Alongside trip data, the bike sharing dataset also provides the geographic coordinates and capacity information for each individual bike station.
Five years before what is done by [Ahmed Jaber et al., 2023], an analysis was done by [Yongping Zhanga,2018], the data was sourced from Mobike, a prominent provider of bike-sharing services that leverages IoT (Internet of Things) technology to facilitate short urban trips with convenient parking options. As of March 2017, Mobike boasted a

fleet of over 4 million red-framed bicycles distributed across nearly 80 cities globally. Their operational scale was substantial, processing around 20 million daily orders, equivalent to 56.56% of the total market share. This marked them as the largest dockless bike-sharing company in the world.

The dataset employed, generously shared by Mobike, encapsulates approximately 56.62% of the total trip orders from August 2016. Within this dataset, a total of 1,023,603 orders were made by 306,936 users, encompassing 17,688 individual bikes. Each order entry comprises essential trip details, including the order ID, user ID, bike ID, start time, origin's longitude and latitude, end time, destination's longitude and latitude, and the track. These attributes are represented as columns in the dataset. The 'track' attribute entails a sequence of longitude and latitude pairs between the start and end points. In cases of N-location tracks, the 'track' column's format is exemplified as 'longitude1, latitude1# longitude2, latitude2# ... longitudeN, latitudeN#'. It's noteworthy that all bikes were GPS-tracked, effectively rendering a bike trip as a sequence of GPS points ordered chronologically, such as $p1 \rightarrow p2 \rightarrow \cdots \rightarrow pn$, where each point is described by geospatial coordinates and a timestamp, denoted as $p = (x, y, t)$. However, to address privacy concerns, Mobike preprocessed the tracks within the dataset. Consequently, each track contains solely an assortment of chronologically disordered spatial locations, devoid of temporal data, implying that the actual trip routes of users remain undisclosed.

Upon extracting the sequentially ordered trips, the researchers proceeded to compute the distances for all trips. Subsequently, they filtered out trips exceeding a distance of 50 km or a time duration surpassing 10 hours. Following this rigorous data cleaning process, a refined dataset containing 1,023,529 trips was obtained.

In analyzing the data, the calculated mean trip distance stood at 2.4 km, accompanied by a standard deviation of 2.2 km. Cumulatively, the summation of distances for all bike trips amounted to an impressive 2.4 million kilometers.

As for travel time, the average duration was determined to be 16.8 minutes, with a corresponding standard deviation of 18.5 minutes. Remarkably, the cumulative travel time for all bike trips translated to an astonishing 32.7 years.

Authors	Bicycle variables	Target variable
[Huthaifa I. Ashqar et al, 2019]	station ID, the number of available bikes, number of available docks, and the precise time of recording	Bike counts at different stations
[The University of Münster, 2020]	hourly bicycle counts	
[Joost de Kruijf et al., 2021]	Longitude and latitude of the stops	Reduction in car use
[Craig Morton, 2020]	cycle hires	demand for cycling
[Ahmed Jaber et al., 2023]	on start station id, end station id, station latitude, station longitude, and trip time for each bike trip	Members' bike-sharing use
[Ahmadreza Faghih-Imani et al, 2017]	arrival rate due to customer usage, departure rate due to customer usage, refilling rate due to operator rebalancing, and removal rate due to operator rebalancing	Demand (consisting of customer arrivals and departures), and Rebalancing (consisting of the frequency and quantity of operator refills and removals)
[Hongtai Yang et al., 2021]	the start and end times, originating and concluding stations, trip duration, user category, and demographic information for annual and 15-day members, covering age and gender, geographic coordinates and capacity information for each individual bike station	Usage of bike sharing
[Yongping Zhanga,2018]	order identification (ID), user ID, bike ID, start time, the longitude and latitude of the origin, end time, the longitude and latitude of the destination, and track. Each attribute is a column in the dataset	Petrol saving, reducing CO_2 and NO_x emission

Table 3: summary of articles, bike variables and target variables

3 BIKE DESCRIPTIVE ANALYSIS AND DATA PREPARATION

The objective of this study is to analyze the usage of shared bikes based on weather conditions utilizing machine learning algorithms able to predict the future shared bike usage based on the weather forecasts.

3.1 Descriptive analysis and data preparation

The dataset utilized for analysis was sourced from the municipality of Vicenza city in northern Italy through the Ridemovi application, which is specifically designed for bike and e-bike sharing. The dataset comprises 12 separate Excel files, each corresponding to a month of the year 2022. In total, the dataset contains 30,378 rows of data. As it is expressed in Excel file: the first row includes 16 parameters, while the subsequent rows provide the
corresponding parameter values for each individual trip.

Ct	B_N	V_T	U_I	U_T	S_T	E_T	Dur	Dis	Sla	Slo	Ela	Elo	Mon	Pro	Pas
Vicenza	IB12A007	bike	555.314	PAYG	01/01/	01/01	0.583300	6	45.545911	11.547220	45.55	11.547002	100	0	0
Vicenza	IB12A010	bike	154.219	PAYG	01/01/	01/01	4.683300	12	45.545391	11.550808	45.54	11.541374	100	0	0
Vicenza	IB12A044	bike	44.586	Pass	01/01/	01/01	15.966700	1,996	45.560380	11.529404	45.55	11.547403	100	100	1
Vicenza	IB12A008	bike	544.095	PAYG	01/01/	01/01	25.950000	3,016	45.544987	11.523768	45.53	11.508011	200	0	0
Vicenza	IB12A003	bike	287.271	PAYG	02/01/	02/01	10.900000	1,689	45.542053	11.540565	45.56	11.544154	100	100	0
Vicenza	IB12A016	bike	449.899	PAYG	02/01/	02/01	12.433300	0	45.508005	11.561179	45.51	11.561179	100	0	0

Table 4: dataset obtained from Ridemovi application

3.2 Recorded parameters:

Variable	Variable description	Type of variable
Ct	City: The city that the data was collected. All data was collected in Vicenza	nominal
B_N	Bike number: Each bike is assigned a unique identification number	discrete
V_T	Vehicle type: The Ridemovi application is designed for both bike and e-bike sharing, but this study specifically focuses on bike sharing	binary
U_I	User id: Each user is given a unique user ID	discrete
U_T	User type:There are two recorded user types: paying users	binary

	and pass users	
S_T	Start time:The exact date and time the trip had started	temporal
E_T	End time: The exact date and time the trip had ended	temporal
Dur	Duration: The duration of each trip is recorded in minutes	continues
Dis	Distance: The distance between the starting and ending points of trips is recorded in meters	continues
SLa	Start latitude: The system captures the latitude coordinate of the starting point per trip	continues
SLo	Start longitude: The system captures the longitude coordinate of the starting point per trip	continues
ELa	End latitude: The system captures the latitude coordinate of the ending point per trip	continues
ELo	End longitude: The system captures the longitude coordinate of the starting point per trip	continues
Mon	Amount: The recorded data includes the total payment amount per trip	discrete
Pro	Promotion: Price difference to calculate payment amount	discrete
Pas	Pass: A binary variable to show that the user was pass or paying	binary

Table 5: Recorded parameter description

In order to fulfill the objective of analyzing shared bike usage in relation to daily weather conditions, the data was transformed to a daily basis. This was achieved by merging the data using SQL techniques. Consequently, a dataset was created, consisting of 365 rows, each representing a unique day throughout the year 2022.

Given that the city remains consistent across all data entries and the bike number and user type do not significantly impact our analysis and this study's focus is solely on bikes. Consequently, five attributes were deemed irrelevant and eliminated from the dataset, leaving a total of 11 remaining attributes for analysis.

3.3 Univariate Analysis

	dur	dis	U_I	mon	pro	pas	Sla	Slo	Ela	Elo
average	1197.097	106439.718	83.195	10063.014	4885.496	5.617	45.548	11.544	45.548	11.544
max	4608.100	235522.000	176.000	22800.000	12900.000	19.000	45.581	11.554	45.581	11.556
min	47.183	5030.000	4.000	500.000	100.000	0.000	45.541	11.536	45.526	11.495
std-dv	631.647	48867.966	34.701	4362.502	2525.125	4.643	0.002	0.003	0.003	0.004

Table 6: average, maximum, minimum and standard deviation of dataset

As can be seen in the table the longest distance traveled was 235522 meters and the standard deviation is 48867.96 showing the high variability within the dataset distance values.

Moreover, a significant variation exists in trip durations, resulting in a substantial disparity between the maximum and minimum durations observed.

To gain deeper insights into the evolution of trip duration throughout the year 2022, a

comprehensive plot has been created. The plot reveals distinct fluctuations in bike usage across various days of the year, highlighting the dynamic nature of trip durations.
In the plot, it is evident that the trip duration experienced an upward trend from May to October after a downward in January and February, followed by a subsequent decline in November, December.

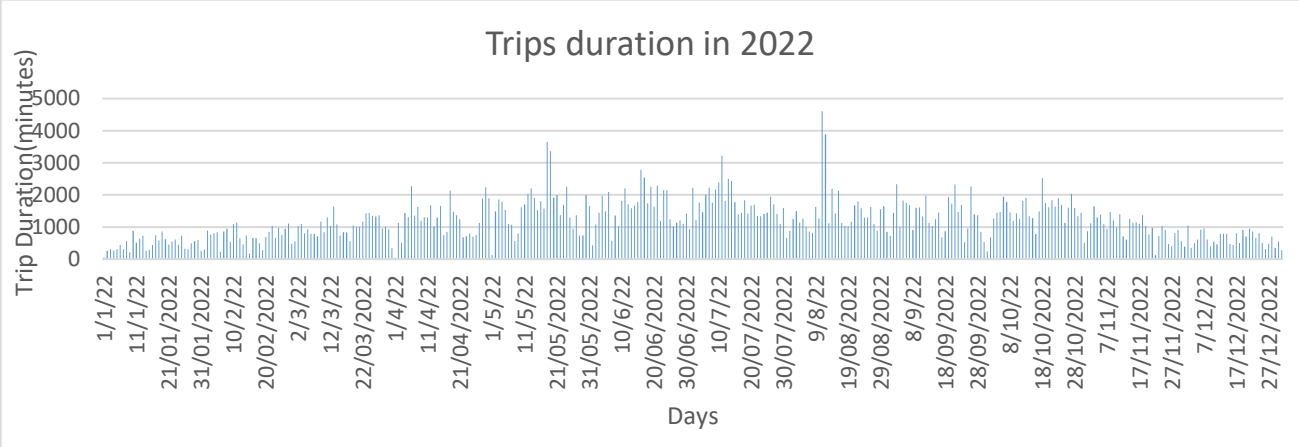


Chart 1: Trip duration based on the days of 2022

It is also possible to see the difference in demand for shared bike between weekdays and weekends.

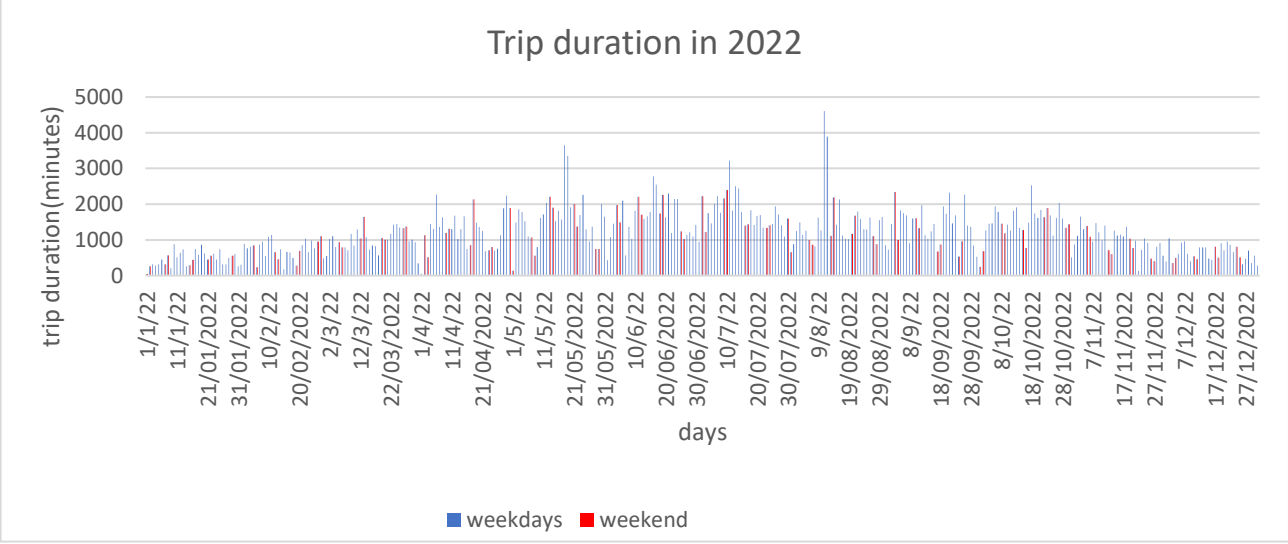


Chart 2: Trip duration in weekdays and weekends

A statistical hypothesis has been done on the data of year 2022, in order to assess the difference trips' duration between weekdays and weekend. Firstly, a new variable has been defined to show the weekdays and weekend by assigning the value 0 to weekdays and 1 to weekend. The result obtained out of this analyze is as follows:
A small p-value (usually less than 0.05) suggests that the difference between the groups

is statistically significant. In this case, the p-value is 0.01086, which indicates some evidence against the null hypothesis.

The 95% confidence interval (41.38988 to 313.97530). The mean value of weekdays (1247.725) is higher than the mean value of weekends (1070.042), but the difference is not statistically significant.

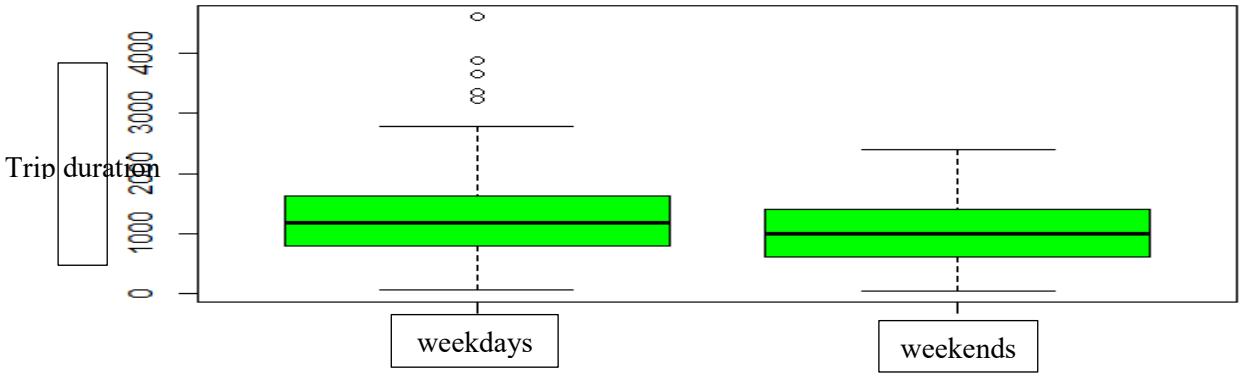


Chart 3: box plot of weekdays and weekends

3.4 Data visualization

3.4.1 Normality

Normality typically refers to the distribution of data. A normal distribution, also known as a Gaussian distribution or bell curve, is a symmetrical probability distribution where the majority of observations cluster around the mean, and the data points are evenly distributed on both sides. In a normal distribution, the mean, median, and mode are all equal.

Visual inspection of the distribution may be used for assessing normality, although this approach is usually unreliable and does not guarantee that the distribution is normal. However, when data are presented visually, readers of an article can judge the distribution assumption by themselves. The frequency distribution (histogram), stem-and-leaf plot, boxplot, P-P plot (probability-probability plot), and Q-Q plot (quantile-quantile plot) are used for checking normality visually. The frequency distribution that plots the observed values against their frequency, provides both a visual judgment about whether the distribution is bell shaped and insights about gaps in the data and outliers outlying values. The boxplot shows the median as a horizontal line inside the box and

the interquartile range (range between the 25th to 75 the percentiles) as the length of the box. The whiskers (line extending from the top and bottom of the box) represent the minimum and maximum values when they are within 1.5 times the interquartile range from either end of the box.

In order to assess the normality of variables, RStudio programming language was utilized to visually represent the distribution of each variable through box plots, which can be observed as depicted below.

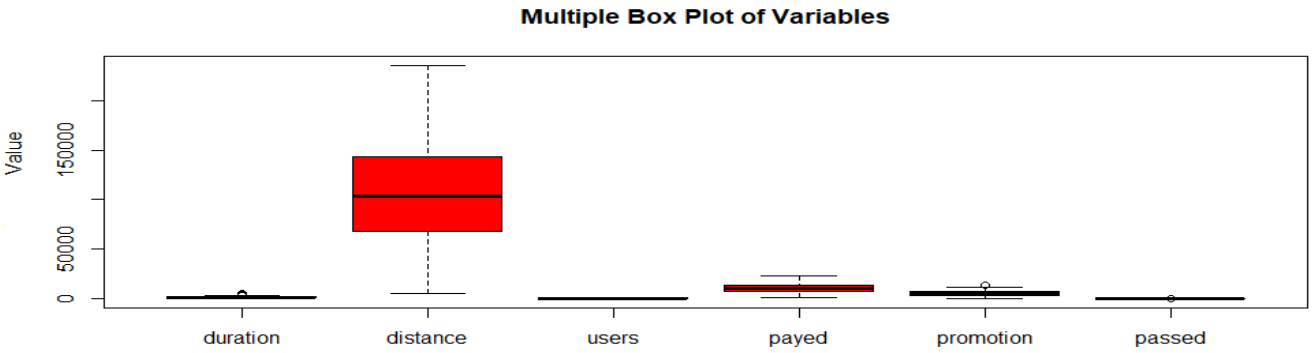


Chart 4:box plot of distribution of each variable

As evident from the boxplots, the variables exhibit different ranges. Consequently, a scaling function was applied to the data in order to standardize them, resulting in a new boxplot shown below.

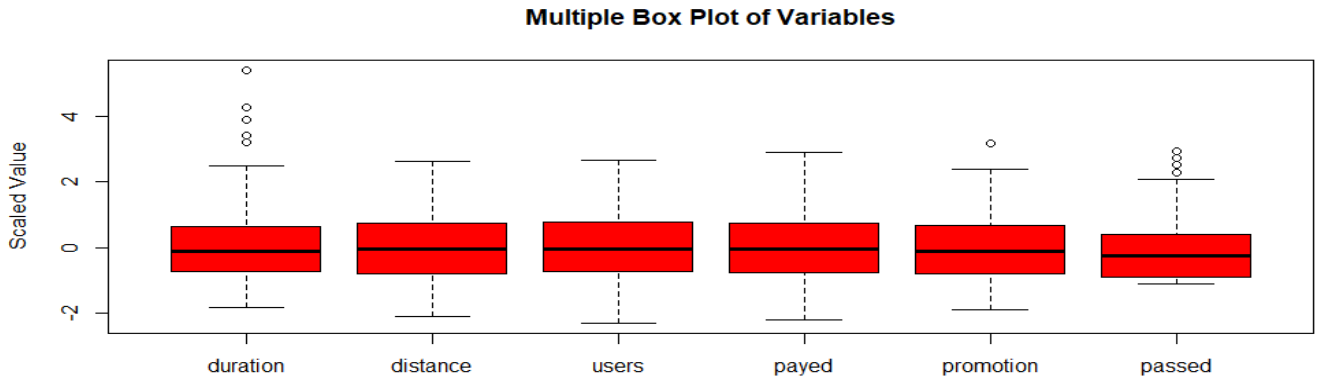


Chart 5: boxplot of normalized variables

As can be seen almost all of boxplots are symmetric and the distribution of variables are normal.

To examine the normality of the data distribution, histogram plots were utilized for the

variables. The outcomes of this analysis are presented below.

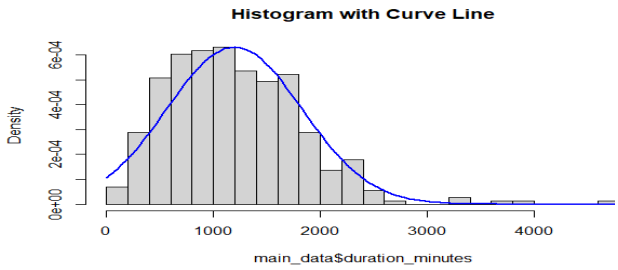


Chart 6:histogram plot of trip duration

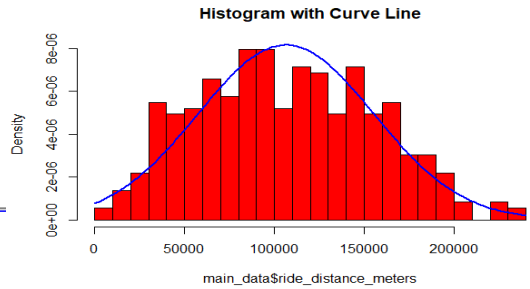


Chart 7: histogram plot of ride_distance

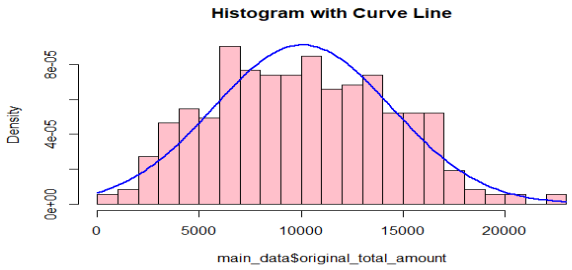


Chart 8: histogram plot of original_total_amount

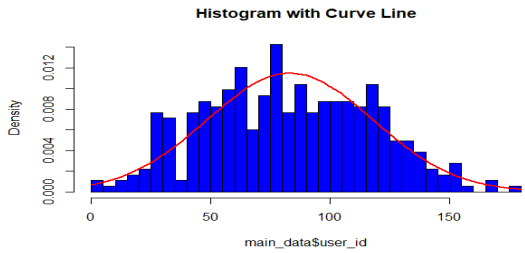


Chart 9: histogram plot of user_id

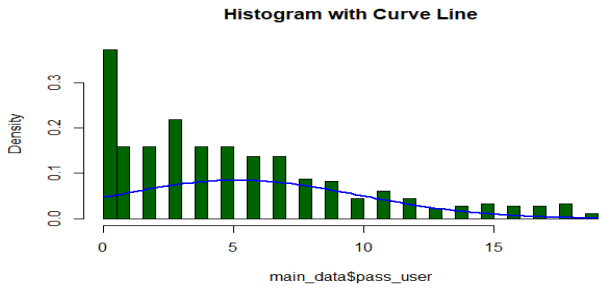


Chart 10: histogram plot of pass_user

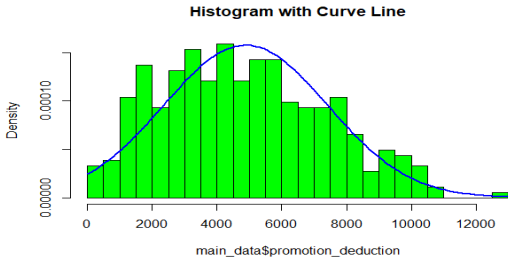


Chart 11: histogram plot of promotion_deduction

In the histogram plots, it is evident that the distance, original amount, promotion deduction, and user ID follow a normal distribution, characterized by a bell-shaped curve. However, the pass user data does not exhibit a normal distribution. Similarly, the distribution of trip time is not completely normal and tends to skew towards the left side.

When assessing bike usage based on each season, the following results are observed:

	spring	summer	fall	winter
dur	118531.9	147942.7	116393.7	54072.22
dis	10160614	12290400	11410554	4988929

Table 7: bike usage in different seasons

The table clearly indicates that trip durations during winter are the shortest, while trip durations substantially increase during the summer season. This observation holds true for the distance traveled by bike as well.

3.5 Main points of the trips

Data has been merged by using the SQL based on the User_id. This merging process resulted in a dataset comprising 4626 rows.

U_I	Sla	Slo	Ela	Elo
1.073.603	45.56467	11.53926	45.5689	11.52446
784.348	45.56161	11.53042	45.54741	11.54615
1.060.599	45.54607	11.5571	45.55231	11.54846
588.143	45.54486	11.52781	45.556	11.53377
573.691	45.54608	11.55893	45.54603	11.55907
942.258	45.54193	11.55821	45.54382	11.56305
540.5	45.54322	11.51911	45.54318	11.5175
947.6	45.54744	11.54487	45.54746	11.54485

Table 8: dataset based on the user_id

The starting and ending points are visualized on the Vicenza map using the R programming language by instalation the leaflet package. The leaflet library is employed for this purpose. The plot clearly illustrates that some points overlap, as each ending point can be a starting point for another trip.

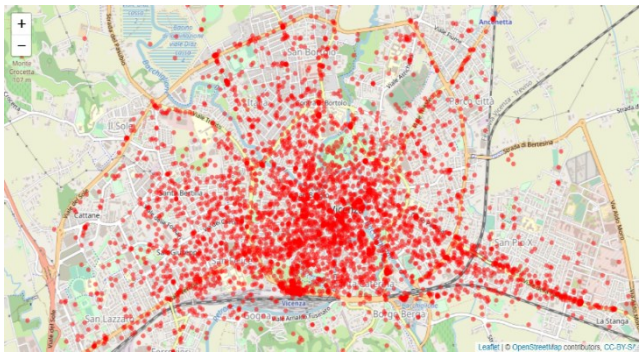


Figure 6: starting points

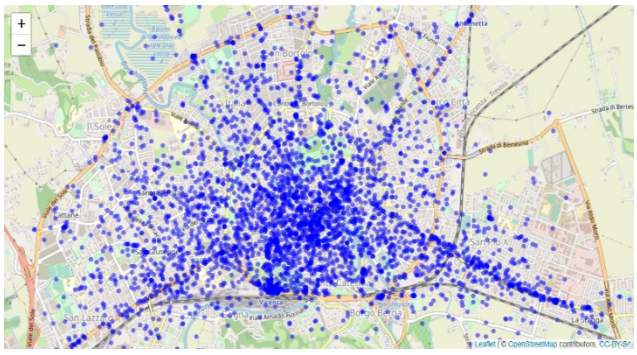


Figure 5: ending points

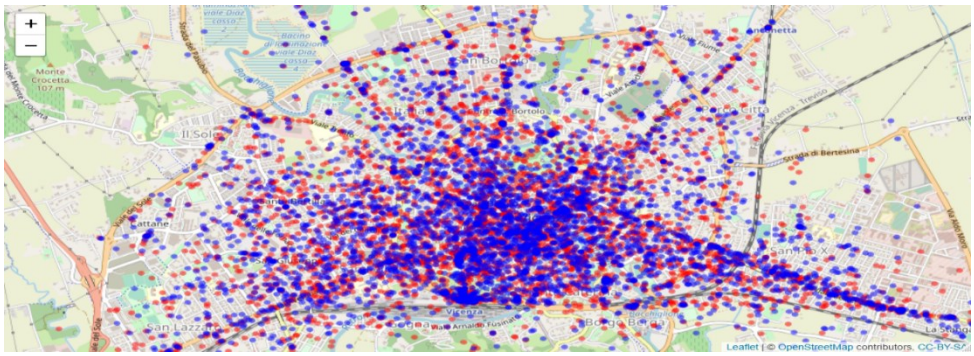


Figure 7: starting and ending points

To determine the main starting and ending points, R Studio was utilized to calculate the frequency of each location. The most frequently occurring location was then chosen as the main point for both starting and ending points.

	longitude	latitude
Starting point	11.540927	45.541862
Ending point	11.54055	45.541638

Table 9: main starting and ending point

As was expected the main starting and ending points are located at train station of Vicenza. The red point indicates the starting point, and the blue point indicates the ending point.

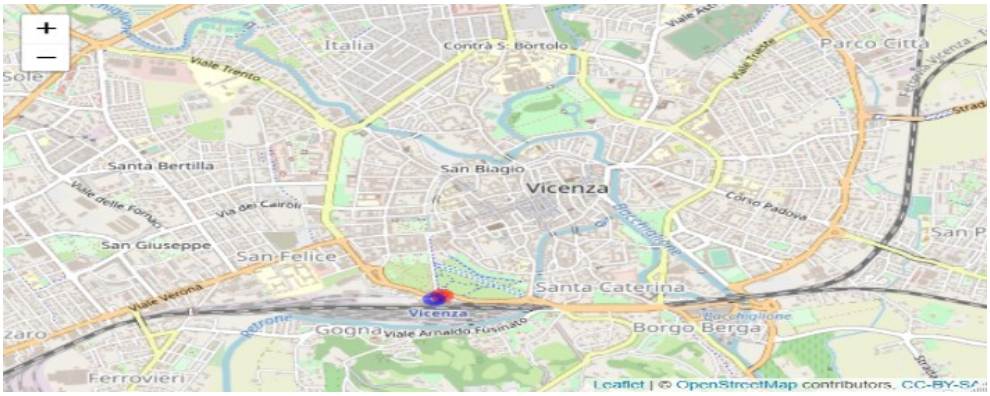


Figure 8: location of starting and ending point

4 WEATHER DATA METHOD LITERATURE REVIEW

Weather conditions and weather quality data are typically collected from various sources, including weather stations, satellites, and weather balloons. Weather condition data typically encompasses parameters such as temperature, humidity, and wind speed, among others. On the other hand, weather quality data incorporates essential information about harmful airborne particles like NO, CO, O3, and PM10, making it critical for understanding air quality and its impact on human health and the environment. The data collected includes various types, such as categorical and numerical data. For facilitating a comprehensive analysis of weather patterns and air pollution levels the appropriate preparation is required.

In recent years, the application of machine learning (ML) and artificial intelligence (AI) in weather data analysis has witnessed significant growth. ML algorithms are effectively utilized for tasks like pattern recognition, anomaly detection, and enhancing the accuracy of weather forecasts. Additionally, AI-driven decision support systems play a crucial role in interpreting complex weather data, enabling timely warnings for extreme events, such as hurricanes and storms. As a result, adequate preparation and preprocessing of data for ML algorithms become paramount, and researchers have explored various approaches to achieve this.

Different weather condition data including: 1. Real temperature in °C, 2. Apparent temperature in °C, 3. Humidity in percentage (%), 4. Wind speed in km/h, 5. Category of weather (e.g., rainy, foggy, cloudy, etc.) gathered by [El Arbi Abdellaoui Alaoui et al, 2021]. In order to manage the bike sharing in smart cities by using machine learning and internet of things.

To facilitate the usage of this weather data in machine learning algorithms, the researchers employed a process of data preparation. Categorical data was codified, with each weather condition being assigned a numerical value. This transformation allowed the incorporation of weather categories into the machine learning model. Additionally, all the features of the weather data were normalized using the following formula:

$$Vij \leftarrow \frac{Vij - \min j(Vij)}{\max j(Vij) - \min j(Vij)}$$

This normalization process was carried out to minimize the impact of varying scales among the different features, ensuring that no single variable dominated the predictor performance. By

conducting this comprehensive analysis and data preparation, [El Arbi Abdellaoui Alaoui et al,2021] aimed to enhance the accuracy and effectiveness of their machine learning-based intelligent bike sharing management system.

The same variables plus solar intensity were also assessed by [Mahmoud Elgendi et al, 2023]. They improved the accuracy of predictions, by eliminated outliers from the dataset (Outliers are data points that deviate significantly from the rest of the data and can adversely impact the performance of machine learning models). Like what has been done by [El Arbi Abdellaoui Alaoui et al,2021], to further enhance the accuracy and training speed of the machine learning model, the features including solar intensity, temperature, humidity, wind speed were scaled to a specific range. Feature scaling is essential because it ensures that all features contribute equally to the model, preventing one dominant feature from overshadowing others. The researchers used the same formula used by [El Arbi Abdellaoui Alaoui et al,2021] for scaling the features:

$$\text{XScaled} = \frac{\text{X} - \text{Xmin}}{\text{Xmax} - \text{Xmin}}$$

In this formula:

- XScaled represents the scaled value of a feature.
- X represents the original value of the feature.
- Xmin represents the minimum value of the feature.
- Xmax represents the maximum value of the feature

By performing feature scaling, the researchers allowed the gradient descent algorithm to converge faster to the optimal minima during the training process. Scaling the features to similar and small ranges aids in the optimization process, leading to more accurate predictions.

In contrast, [Jan Wessel, 2020] to prepare the weather condition data, used dummy variables as following:

During this study two weather variables, average monthly temperature and monthly precipitation have been collected and three different types of regression models were applied to analyze the average monthly temperature and monthly precipitation data. Non-linear impacts were observed for actual and forecasted air temperature, as well as precipitation levels. Consequently, so, six dummy variables indicating light drizzle (precipitation<0.5 mm/h), strong drizzle (0.5 mm/h precipitation <1 mm/h), light rain (1 mm/h precipitation <2 mm/h), moderate rain (2 mm/h precipitation <5 mm/h), heavy rain (5 mm/h precipitation <10 mm/h), and very heavy rain (10 mm/h precipitation) have been replaced the continuous data. Moreover, another dummy variable is used in the regression model to control for actual snowfall. It takes the value 1 if we have precipitation and sub-zero air temperatures, and otherwise 0.

Two years before what is done by [El Arbi Abdellaoui Alaoui et al, 2021], snow data was treated as categorical, where a value of 1 indicated the occurrence of snow and 0 represented

non-occurrence of snow on a given day by [Ran An et al, 2019]. Other weather variables were analyzed as follows:

Rainfall Data: Rainfall data was analyzed by calculating the total daily rainfall. Each day's rainfall amount was recorded and used for further analysis.

Temperature, Wind, and Humidity Data: For temperature, wind, and humidity, the average daily values were calculated. This approach allowed the researchers to obtain representative figures for these weather conditions and use them in their subsequent analysis.

In 2018 and 2020 two research were done to analyze weather condition data namely: temperature, humidity, wind speed and precipitation by [Kyoungok Kim, 2018] and [Craig Morton, 2020] however different methodology was applied on data.

[Kyoungok Kim, 2018] collected weather data at various time intervals, ranging from 1 hour to a year. Recognizing that temperature and humidity may not independently impact people's activities, the researchers considered the Temperature Humidity Index (THI) as an explanatory variable, which accounts for the interaction between these two variables. The THI serves as a combined indicator of environmental temperature and relative humidity, and it was employed as a discomfort index to assess the risk of heat stress. THI is defined as follows:

$$THI = \frac{9}{5}T - 0.55(1 - RH)(\frac{9}{5}\text{T-26}) + 32$$

where T represents temperature in degrees Celsius (°C) and RH represents relative humidity in percentage (%). When the THI exceeds 80, it indicates discomfort for most people, while a THI of less than 70 implies no discomfort. To establish the relationship between the THI and discomfort, the THI was further transformed using the logit function as follows:

$$f(x) = \frac{1}{1 + \exp\{-0.8(x - 75)\}}$$

Through this transformation, the final THI values were scaled within the range of [0, 1], with values close to 0 indicating low discomfort (THI < 70) and values close to 1 representing high discomfort (THI > 80).

To ensure comparability and standardization of variables, the researchers converted the unit of relative humidity from percentage to ratio and adjusted the unit of precipitation from millimeters (mm) to centimeters (cm). Subsequently, daily weather variations were calculated based on meteorological observations at one-hour intervals. Mean values were utilized for temperature, relative humidity, wind speed, and THI, while the total sum of precipitation was employed as the representative value for each day. But [Craig Morton, 2020] utilized weather data for study examining the demand for cycle sharing and its association with weather conditions, air quality levels, and cycling patterns among regular and casual users in London, including maximum air temperature (Temp), mean wind speed (Wind), mean relative humidity

(Humid), and precipitation, sourced from the Centre for Environmental Data Analysis, which stores information from the UK Met Office weather stations. As mentioned, [Craig Morton, 2020] analyzed the weather quality data as well.

Air quality data, specifically the mean concentration levels of ozone (O_3), nitrogen oxides (NOX), and particulate matter 10 (PM10), were obtained from the monitoring station in London Bloomsbury, managed by the Department for Food, Environment, and Rural Affairs.

To process the weather data, a two-step transformation procedure was followed. Firstly, a 9-term moving average was calculated using a weekly index (τ). This index allowed for a comparison of the observed value (Dt) on a given day (e.g., a Tuesday) with the same day in the preceding and succeeding 4 weeks, as summarized in Eq:

$$D_t^{MA\pm4} = \frac{\sum_{\tau=-9}^{4} D_{t+7\tau}}{9}$$

Secondly, the residuals were computed to capture deviations from the moving average, as shown in Eq:

$$\Delta D_t = \frac{D_t - D_t^{MA\pm4}}{D_t^{MA\pm4}}$$

However, for precipitation, daily residuals were deemed unsuitable due to occurrences of days with no recorded rainfall. Instead, precipitation was included in the analysis as two dummy variables: light rainfall (LPrecip) represented by precipitation between 0.1 and 4.9 millimeters, and heavy rainfall (HPrecip) classified as precipitation of 5 millimeters or higher.

Two different researchers conducted in 2019 and 2021 and researchers used dummy variable for data preparation method.

[Anik Das et al, 2019] employed qualitative-based measures extracted from NDS (Naturalistic Driving Study) videos to categorize fog into two types: heavy fog and distant fog. This classification was based on multiple factors such as the visibility of road markings, road signs, roadside surroundings (e.g., delineators, guardrails, New Jersey barriers), and the horizon. A foggy condition was labeled as "heavy fog" when the majority of these elements could not be recognized, indicating severe visibility impairment. On the other hand, "distant fog" was defined as a foggy weather condition where all the above-mentioned elements could still be distinguished, except for the horizon.

To account for the high variability of weather conditions within a single trip, the researchers divided each trip into one-minute time chunks. This division allowed them to create homogeneous segments with similar traffic and environmental conditions. Manual observation templates were provided to video observers, who reported the traffic and environmental conditions for each one-minute segment, facilitating accurate data collection. In addition, [Ying

Wang et al, 2021] used dummy variables for weather quality features and ranked air quality into six levels based on AQI: excellent (0 = AQI≤50), good (50 < AQI≤100), lightly polluted (100 < AQI≤150), moderately polluted (150 < AQI≤200), heavily polluted (200 < AQI≤300) and severely polluted (300 < AQI≤500). As previous studies have observed, tourists may react to air quality according to the air quality standards established by the government (Yoon, 2019). We therefore used the government air quality levels (LEVEL) as an alternative measure for air quality. they designed six dummy variables (EXCELLENT, GOOD, LIGHTLY, MODERATELY, HEAVILY, SEVERELY) to record the six levels.

They also included weather conditions as independent variables in their model. A number of studies have shown that temperature, precipitation, wind and cloudiness affect tourist perception and tourism activity. Thus, daily average temperature (TEMPERATURE), wind level (WIND), precipitation (RAIN) and cloudiness (CLOUD) in the tourist attraction were considered.

4.1 Summary of weather condition and weather quality data:

Authors	Weather condition variables	Weather quality variables	Model	Target variable
El Arbi Abdellaoui Alaoui et al, 2021	Real temperature, Apparent temperature, Humidity, Wind speed, Category of weather		Linear regression, random forest, XGBoost, SVR, AdaBoost, bagging regressor	number of bikes shared
Mahmoud Elgendi et al, 2023	solar intensity, temperature, humidity, wind speed		machine learning algorithms: LR and ANN	solar still locations
Jan Wessel, 2020	Light drizzle, Strong drizzle, Light rain, Moderate rain, Heavy rain, Very heavy rain, Snowfall		Log-linear regression	bike ridership
Ran An et al, 2019	Snow, Rainfall, Temperature, Wind, Humidity		multilevel regression	number of cycling trips
Kyoungok Kim, 2018	Temperature, Humidity, wind speed, precipitation		Clustering	number of bike rentals
Craig Morton, 2020	Max Temperature, Mean wind speed, Mean relative humidity, Mean relative precipitation	O_3 ,NOX , PM10	ADL regression models	cycling demand
Anik Das et al, 2019	Heavy fog, Distant fog		logistic regression	significant driver behavior and performance differences between driving in foggy conditions and clear weather conditions
Ying Wang et al, 2021	AQI excellent, AQI good, AQI lightly polluted, AQI moderately polluted, AQI heavily polluted, AQI severely polluted, average temperature, wind level, precipitation, cloudiness		Sentiment analysis	tourists' emotional experience

Table 10: Summary of weather condition and weather quality data

5 WEATHER DATA DESCRIPTION AND DATA PREPARATION

During this study, weather data has been obtained from ARPAV website,

comprising two distinct types of information: data pertaining to weather quality and data concerning weather conditions, encompassing parameters such as temperature, humidity, wind speed, and other relevant factors. These data sets have been recorded in Excel files, facilitating organization and analysis for the research.

5.1 Weather condition data:

The data collected encompasses 13 variables, and dataset contains 8,760 rows of information. Each row corresponds to a specific time of day during the entire year 2022, providing a comprehensive representation of the weather conditions throughout the year. These variables capture various aspects of the weather, allowing for detailed analysis and insights into the meteorological patterns and trends during the specified period.

S_I	day	mon	year	time	T_M	pre	H_MI	H_MA	S_R	W_S	M_G	D_R
451	1	1	2022	1	1.3	0	99	99	0	0.5	157.5	SSE
451	1	1	2022	2	1.2	0	99	99	0	0.5	135	SE
451	1	1	2022	3	1	0	99	99	0	0.6	157.5	SSE
451	1	1	2022	4	0.7	0	99	99	0	0.7	202.5	SSO
451	1	1	2022	5	0.8	0.2	99	99	0	0.4	112.5	ESE
451	1	1	2022	6	0.5	0	99	99	0	0.5	180	S
451	1	1	2022	7	0.3	0	99	99	0	0.7	180	S
451	1	1	2022	8	-0.1	0	99	99	0	0.5	90	E

Table 11: Dataset obtained from ARPAV website

5.1.1 Recorded attributes:

Variables	Variables' description	Type of variable
S_I	station ID	discrete
Day	day of the month	nominal
Mon	month of the year	nominal
Year	data has been collected in year 2022	discrete
Time	time of the day data has been obtained	temporal
T_M	Medium temperature at 2 m(°c)	continues

Pre	precipitation (mm)	continues
H_MI	minimum humidity at 2m (%)	continues
H_MA	maximum humidity at 2m (%)	continues
S_R	solar radiation (MJ/m2)	continues
W_S	average wind speed(m/s)	continues
M_G	maximum gust(m/s)	continues
D_R	direction prevailing	nominal

Table 12: Recorded parameters description

Utilizing SQL, we merged the dataset based on the day of the year 2022 and subsequently computed the weather conditions.

day	AVG(t_m)	AVG(pre)	Min(h_mi)	Max(h_ma)	AVG(s_r)	AVG(w_s)	Max(m_g)
1/1/22	2.041667	0.008333	94	99	0.1875	0.566667	270
2/1/22	-0.08333	0.008333	99	99	0.128583	0.429167	315
3/1/22	4.329167	0.008333	90	99	0.056458	0.3	337.5
4/1/22	5.966667	0.1	90	99	0.0295	0.3	337.5
5/1/22	6.8375	0.291667	60	99	0.016792	0.970833	337.5
6/1/22	4.054167	0.108333	44	99	0.23725	0.95	225
7/1/22	0.683333	0.008333	46	99	0.22225	0.3125	337.5

Table 13: Merged dataset

Certain variables, namely S_I, day, mon, year, time, and D_R, were excluded from the analysis, while eight variables were retained.

	T_M	Pre	H_MI	H_MA	S_R	W_S	M_G
Average	14.36525	0.093311	48.85753	97.42192	0.595962	0.862135	272.8664
Max	29.20417	2.783333	99	99	1.231625	3.195833	337.5
Min	-0.28333	0	15	66	0.0055	0.129167	67.5
Std_dv	8.300384	0.280486	17.46949	4.309147	0.349135	0.446818	55.30656

Table 14: average, maximum, minimum and standard deviation of variables

The table reveals that the average temperature in Vicenza is approximately 14°C, while the humidity spans from 48% to 97%, accompanied by an average wind speed of 0.8 m/s.

To assess the normality of the variables, we utilized the RStudio programming language to generate box plots, visually depicting their distributions. Given the variability in ranges among these variables, we applied data standardization to alleviate this disparity. The resultant presentation is illustrated in the subsequent box plot.

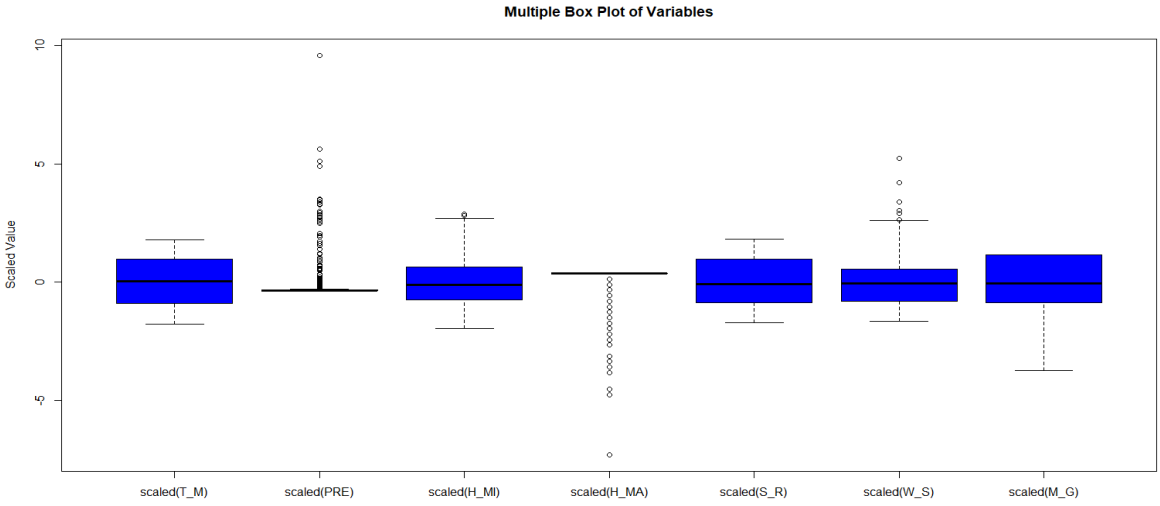
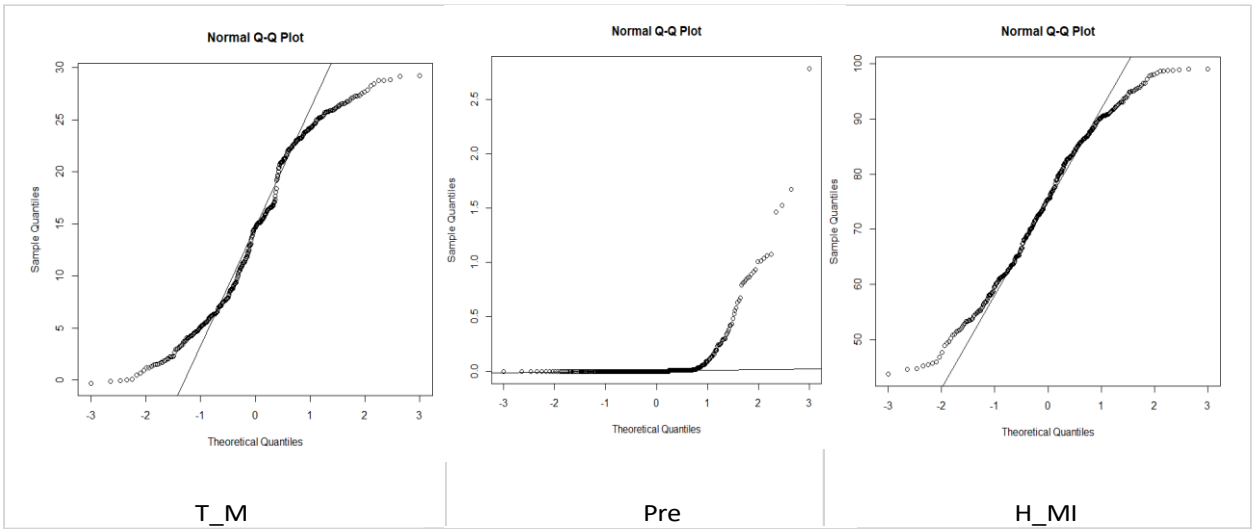


Chart 12: Box plot of weather condition variables

As can be seen all variables except precipitation and maximum humidity, have symmetric distribution and are normal.

We can also evaluate normality using a Q-Q plot, as depicted below. All variables display normal distribution characteristics, with the exception of precipitation.



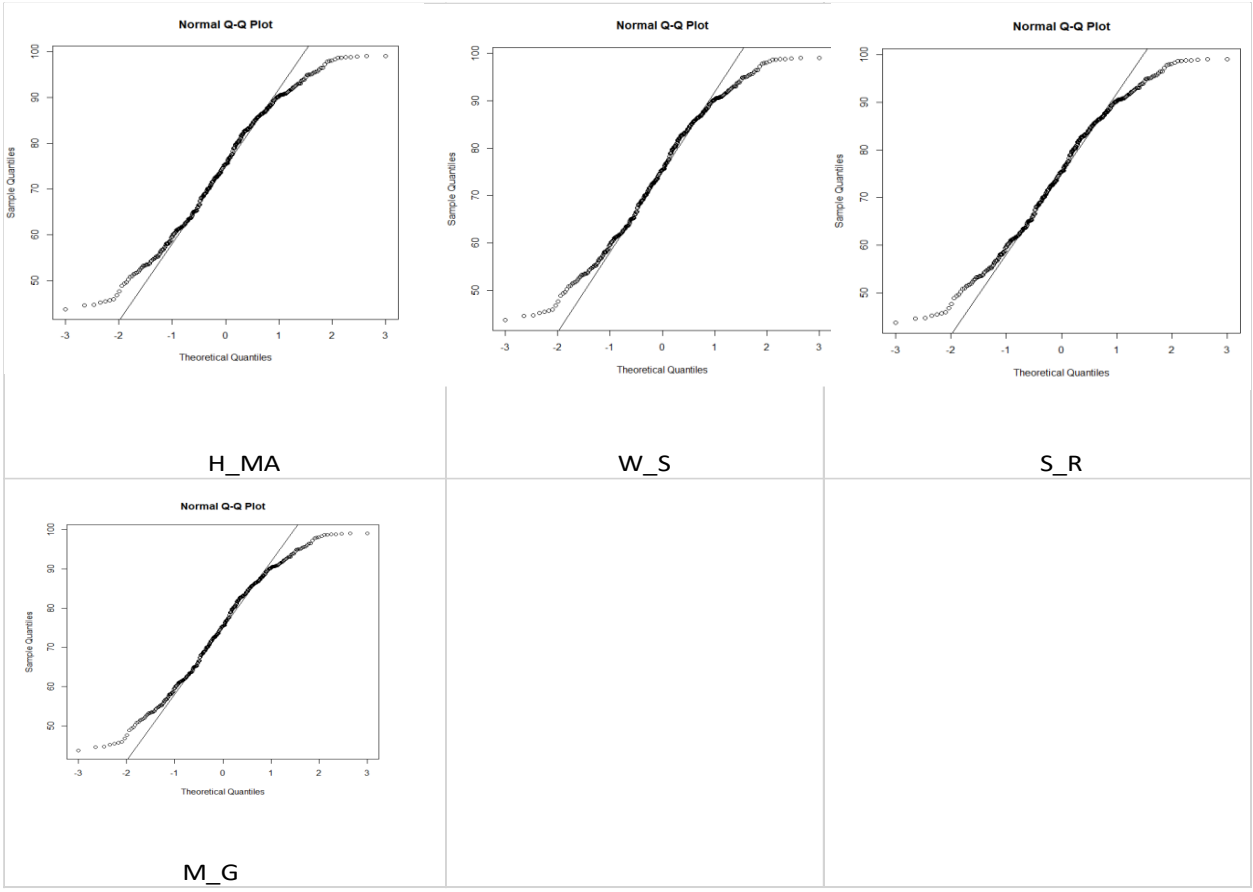


Chart 13: Q-Q plot of attributes

5.2 Weather quality data:

The collected data is comprised of information stored within four distinct Excel files.

The variables encompass measurements of CO, NO2, O3, PM10, and PM2.5.

Dati Rete Qualità dell'aria

Attenzione : I dati tengono conto dei limiti di rilevabilità.
Nota : I dati provengono direttamente dalle centraline automatiche e possono subire parziali modifiche anche dopo la validazione.

Periodo da	**Jan-22**	**Dec-22**
	Quartiere Italia	**Quartiere Italia**
	PM10	**PM2.5**
Giorno	µg/m3	µg/m3
1/1/22	62	53
1/2/22	43	40
1/3/22	50	40
1/4/22	58	52

Figure 9: weather quality data provided by ARPAV website

The map illustrates the data acquired from two distinct stations: Quartiere Italia and San Felice.

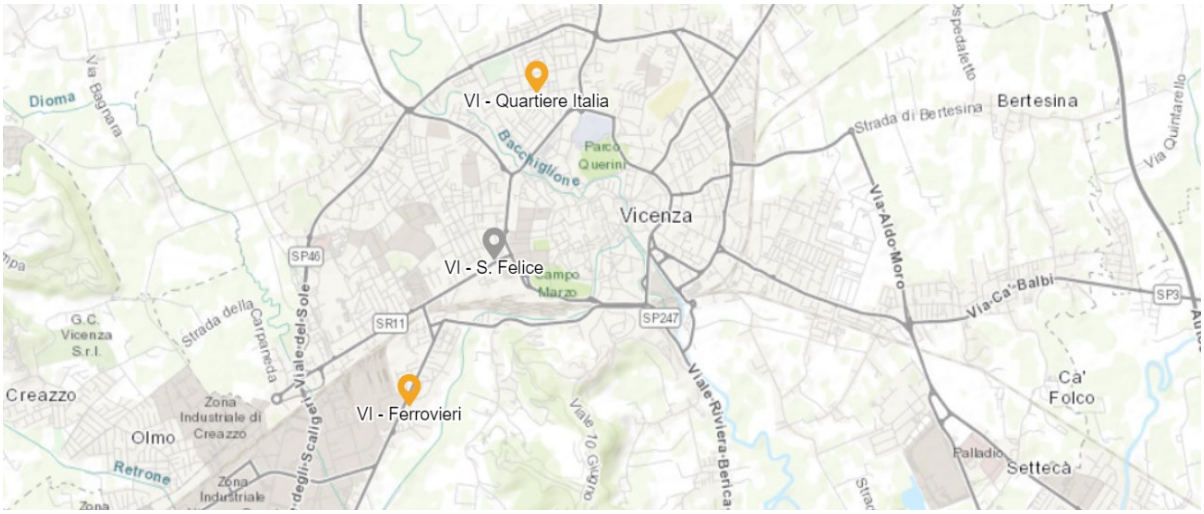


Figure 10 :Stations for obtaining weather quality data

The NO_2 data from Quartiere Italia was excluded from consideration due to a significant prevalence of null values within the dataset. Instead, the focus of the analysis was directed towards the data collected in San Felice. Additionally, in the case of PM10, the average values from both station datasets were employed, as the null value occurrences were consistent between the two stations. The data for PM2.5 and O3 was sourced from the Quartiere Italia station, whereas the CO data was acquired from the San Felice station.

The dataset containing 365 rows and encompassing six variables, delineated as follows.

day	CO	NO2	O3	PM10	PM2.5
1/1/22	1.049005	23.19024	8.798593	60	53
2/1/22	1.016667	25.625	6.5	43.5	40
3/1/22	0.744838	35.85691	9.048593	47	40
4/1/22	0.8375	29.5	7	57.5	52
5/1/22	0.561505	24.27357	19.92359	31.5	27
6/1/22	0.429167	28.41667	30.45833	15	15
7/1/22	0.528171	41.02357	15.75693	26.5	24

Table 15: weather quality data

PM10($\mu g/m^3$): stands for "Particulate Matter 10," which refers to airborne particles with a diameter of 10 micrometers or smaller. They can come from various sources and are inhalable, potentially impacting respiratory health. Monitoring and controlling PM10

levels are essential for maintaining air quality and protecting public health.

PM 2.5($\mu g/m^3$): stands for "Particulate Matter 2.5," which refers to fine airborne particles with a diameter of 2.5 micrometers or smaller. These particles are smaller than PM10 and can penetrate deep into the lungs, posing significant health risks when inhaled. Sources include vehicle emissions, industrial processes, and combustion. Monitoring and limiting PM2.5 levels are crucial to safeguard public health and maintain good air quality.

CO(mg/m^3): carbon monoxide, a colorless, odorless gas formed from incomplete combustion. It can be harmful when inhaled, leading to symptoms like headache and dizziness. Monitoring and controlling CO levels are crucial for indoor air quality and public health.

NO_2 ($\mu g/m^3$): nitrogen dioxide, a reddish-brown gas from vehicle and industrial emissions. It causes respiratory irritation and affects air quality. Reducing emissions is crucial for public health.

O_3($\mu g/m^3$): ozone, a harmful gas in smog formed by sunlight and pollutants. High levels can harm respiratory health. Reducing emissions is crucial for air quality and public health.

	CO	NO2	O3	PM10	PM2.5
Average	0.376112	25.56578	49.16624	31.63826816	22.98886
Max	1.079167	63.875	121.0486	97	78
Min	0.0685	8.708333	2.291667	3	2
Std_dv	0.179045	11.58488	32.62447	18.08241728	15.46264

Table 16 : average, maximum, minimum, standard deviation of variables

The table illustrates significant variability in the ranges of the variables, with substantial gaps between the maximum and minimum values for each variable.

In order to evaluate the normality of the data, the box plots were constructed using the standardized values of the variables. This approach was chosen due to the substantial disparity in the magnitude of the values.

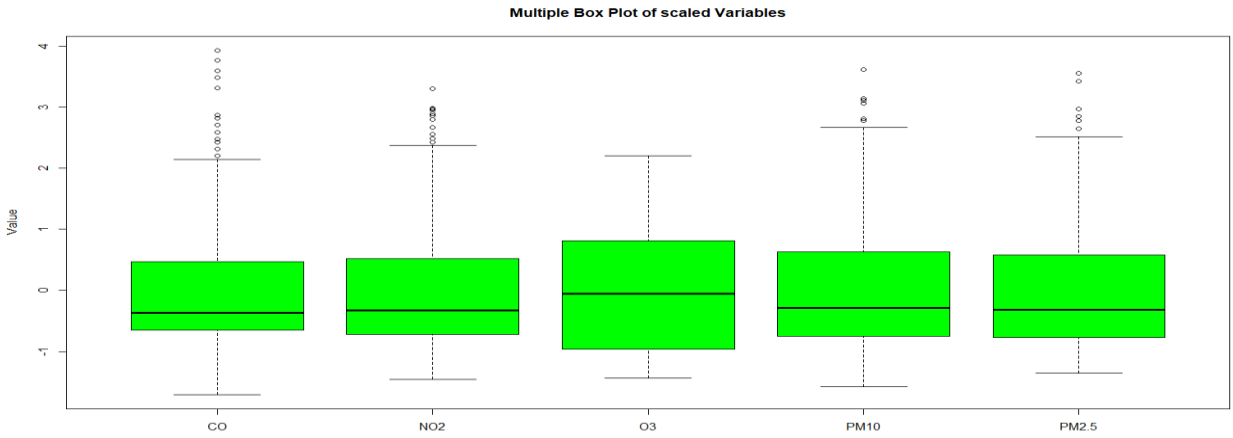


Chart 14: Boxplot of weather quality data

As can be seen in the plot all variables are normal and distributed symmetrically.

6 METHODOLOGY

6.1 Training set and testing set

Before proceeding with further transformations, it is imperative to allocate a portion of the data for evaluating the performance of the developed model. To achieve this, a percentage of the data has been reserved for testing purposes. Specifically, 80% of the data is designated for training the models, while the remaining 20% is dedicated to the testing set for performance assessment.

The dataset comprises 365 rows corresponding to the year 2022. In the context of training and testing, four-fifth of the data, which equates to 292 rows, has been allocated to the training set, while the remaining 73 rows constitute the testing set.

6.2 Ride distance case study

6.2.1 Correlation Analysis of Weather Data Variables with Ride Distance

Utilizing the RStudio programming language, an analysis was conducted to evaluate the correlations between various variables, encompassing weather conditions, weather quality, and ride distance.

Results of correlation analysis:

	dis	t_m	pre	h_mi	h_ma	s_r	w_s	m_g	co	no_2	O_3	pm10	pm2.5
dis	1	0.5811862	-0.22517	-0.351	-0.0103	0.5335	0.079761	-0.15508	-0.48	-0.326	0.458	-0.347656	-0.4217
t_m	0.5812	1	-0.02854	-0.3683	-0.0728	0.7685	0.363302	-0.23272	-0.68	-0.71	0.846	-0.565152	-0.661
pre	-0.225	-0.028537	1	0.36314	0.11938	-0.236	0.205561	0.093908	-0.1	-0.194	-0.06	-0.174052	-0.1835
h_mi	-0.351	-0.368325	0.36314	1	0.36544	-0.743	-0.34661	0.245588	0.357	-0.08	-0.6	0.2642296	0.2889
h_ma	-0.01	-0.072771	0.11938	0.36544	1	-0.223	-0.3575	0.203635	0.1	-0.051	-0.25	0.0206459	0.0296
s_r	0.5335	0.7685317	-0.23571	-0.7435	-0.2228	1	0.491775	-0.27008	-0.61	-0.379	0.897	-0.50786	-0.5484
w_s	0.0798	0.3633017	0.20556	-0.3466	-0.3575	0.4918	1	-0.26166	-0.49	-0.359	0.592	-0.420377	-0.4607
m_g	-0.155	-0.232719	0.09391	0.24559	0.20364	-0.27	-0.26166	1	0.258	0.1678	-0.31	0.2086432	0.2314
co	-0.484	-0.675198	-0.10011	0.35656	0.09976	-0.614	-0.49093	0.257705	1	0.656	-0.65	0.7687008	0.8348
no2	-0.326	-0.709565	-0.19418	-0.0802	-0.0506	-0.379	-0.35904	0.167804	0.656	1	-0.54	0.6604687	0.702
o3	0.4577	0.8462223	-0.05962	-0.5964	-0.2543	0.8967	0.592178	-0.31055	-0.65	-0.543	1	-0.559144	-0.613
pm10	-0.348	-0.565152	-0.17405	0.26423	0.02065	-0.508	-0.42038	0.208643	0.769	0.6605	-0.56	1	0.9654
pm2.5	-0.422	-0.661009	-0.18347	0.28888	0.02961	-0.548	-0.4607	0.231445	0.835	0.702	-0.61	0.9653592	1

Table 17: correlation between different variables

The table demonstrates several noteworthy correlations among the variables. Specifically:

- There is a high correlation between two independent variables, PM2.5 and PM10. Due to multicollinearity concerns, it is advisable to remove one of them. In this case, PM10, which has the lower correlation with the dependent variable (ride distance), will be eliminated.
- Strong correlations are observed between O_3 and mean temperature, as well as between O_3 and solar radiation.
- There is a notable correlation between PM2.5 and CO.
- Mean temperature and solar radiation exhibit the highest correlations with the distance of trips conducted within a given day of the year 2022.
- Conversely, the maximum gust and minimum humidity variables display the lowest reverse correlations with ride distance.

6.2.2 Training models for predicting the distance traveled by shared bike

6.2.2.1 Linear regression model

By applying the linear model on the data, we can derive the following results:

Min	1Q	Median	3Q	Max
-145833	-24413	-771	24573	94344

Table 18: analyzing the symmetricity of residuals

- Residual standard error: 36860 on 281 degrees of freedom
- Multiple R-squared: 0.4752
- Adjusted R-squared: 0.4546
- F-statistic: 23.13 on 11 and 281 DF
- p-value: < 2.2e-16

As can be seen in the results the distribution of residuals is not symmetric, indicating that the model may have limitations in capturing certain patterns in the data. The average difference between the actual data points and the predicted value by model is 36860. R square is equal to 0.47 so approximately 47.52% of the variability in the dependent variable is accounted for by independent variables. The adjusted R-squared

value, which is 0.4546, is a modified version of R-squared that accounts for the number of independent variables in model. It penalizes the addition of unnecessary variables. In this case, suggested that the model explains 45.46% of the variance while considering the model's complexity.

The analysis reveals that the most significant coefficients are associated with medium temperature and carbon dioxide. Hence, these two factors assume a pivotal role in predicting the distance traveled. It is notable that variables such as precipitation, solar radiation, wind speed, and ozone exhibit relatively lower importance in the predictive model.

The F-statistic is 23.13, and it has associated degrees of freedom (DF) values.

The p-value associated with the F-statistic tests the null hypothesis that all coefficients (independent variables) are equal to zero (i.e., the model has no predictive power). A very low p-value, as indicated by "< 2.2e-16" (essentially zero), suggests that at least one of the independent variables is significant in explaining the variation in the dependent variable. Therefore, the model, as a whole, is statistically significant.

6.2.2.1.1 Fine tuning the linear model

- **Feature Selection/Engineering**

  Some features that have the highest value of Pr, have been eliminated from the model and the best model that can be obtained is as bellow,

Min	1Q	Median	3Q	Max
-148984	-24414	-742	25694	96279

Table 19: analyzing the symmetricity of residuals

- Residual standard error: 36760 on 286 degrees of freedom
- Multiple R-squared: 0.4686
- Adjusted R-squared: 0.4575
- F-statistic: 42.03 on 6 and 286 DF
- p-value: < 2.2e-16

It is apparent that several variables, notably medium temperature, solar radiation, and carbon dioxide, play pivotal roles in predicting the traveled distance.

The R-squared values (Multiple R-squared and Adjusted R-squared) suggest that the

independent variables in the model explain approximately 46.86% of the variance in the dependent variable. This implies that the model has moderate explanatory power.

The residual standard error is now 36,760, which is an improvement of 100 compared to the previous model. The adjusted R-squared value is 0.45, implying that the model explains 45.75% of the variance while considering the model's complexity. The F-statistic is 42.03, indicating that at least one independent variable in the model has a statistically significant effect on the dependent variable.

6.2.2.1.2 Shapley values:

By employing the kernel SHAP (SHapley Additive exPlanations) method within the linear regression framework, Shapley values have been derived as follows:

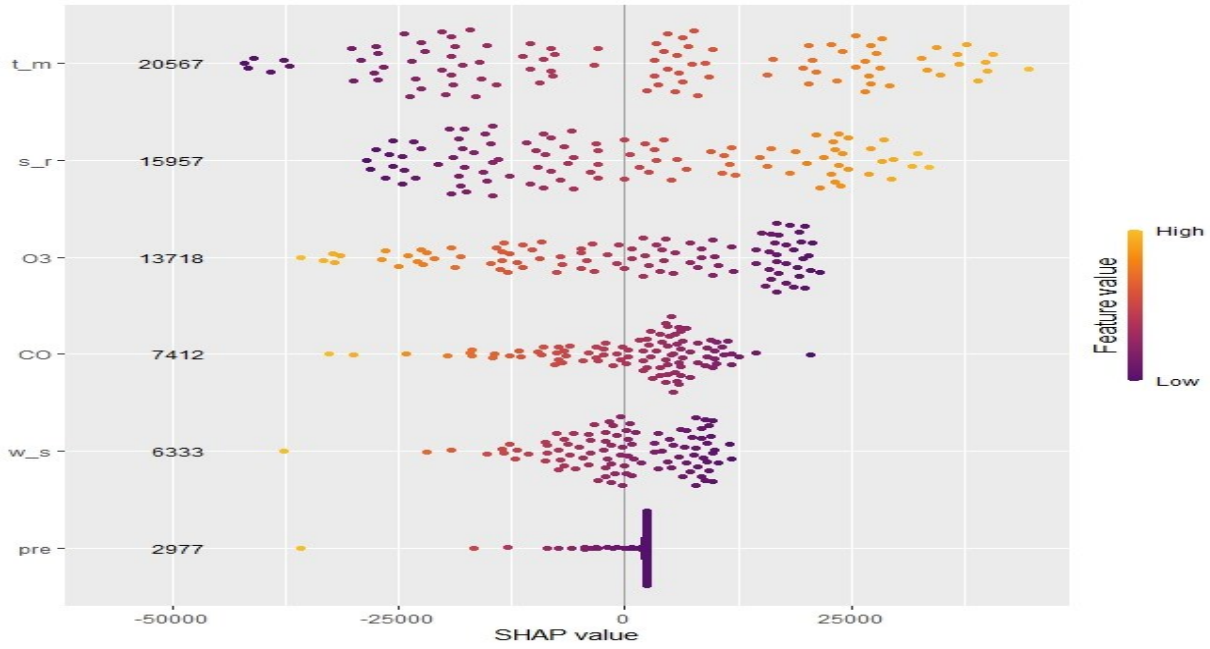


Chart 15: Shapley values for linear regression

The graph illustrates that mean temperature and solar radiation positively influence the distance traveled by shared bikes. Conversely, ozone, carbon monoxide, wind speed, and precipitation exhibit a negative impact on the distance traveled. In simpler terms, higher mean temperature is associated with a greater distance traveled.

6.2.2.1.3 Cross-Validation

The model has been refined using cross-validation techniques to provide a more

accurate assessment of its generalization performance. This approach enhances the model's ability to make predictions on unseen data.

RMSE	R-squared	MAE
37382.93	0.4427339	30345.87

Table 20: results of using cross validation

The RMSE of approximately 37,382.93 suggests that the model's predictions have some variability from the true values. Similarly, the MAE of approximately 30,345.87 represents the average absolute error.

The R-squared value of approximately 0.4427 indicates that the linear regression model explains 44.27% of the variance in the dependent variable. While this suggests a moderate level of explanatory power, there is still a significant portion of unexplained variance.

6.2.2.2 Support vector machine

When applying the Support Vector Machine (SVM) with a five-fold cross-validation approach to predict the distance traveled by bike based on various variables, the following results have been obtained:

C	RMSE	Rsquared	MAE
0.25	37641.29	0.437308	30348.38
0.5	37473.23	0.439156	29989.25
1	37615.87	0.437603	29739.91
2	38126.46	0.428617	29672.37
4	39191.12	0.407241	30225.07
8	40030.92	0.398058	30762.69
16	42710.12	0.355171	32589.7
32	44306.86	0.342901	34039.71
64	46099.36	0.327732	35027.72
128	49367.32	0.306216	37888.94

Table 21: support vector machine results

In the obtained results, it is evident that when the regularization parameter ‘C’ is set to small values (e.g., 0.25, 0.50, 1.00), the model exhibits a decrease in both RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error). This reduction signifies enhanced predictive accuracy, but it raises concerns of potential

overfitting.

Nevertheless, it is important to note that the R-squared values remain relatively low across these ‘C’ values. This observation implies that the model may not effectively account for a substantial portion of the variance in the target variable. In other words, it may not adequately capture the underlying relationships and patterns in the data.

With the increase in ‘C’ values (e.g., 16.00, 32.00, 64.00, 128.00), a noteworthy trend emerges in the model’s performance. Specifically, the RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error) values progressively rise, signifying a decline in predictive accuracy. Concurrently, the R-squared values decline, pointing to a reduction in the model’s explanatory power.

This pattern underscores the trade-off between model complexity, governed by the ‘C’ parameter, and the model’s ability to generalize effectively. As ‘C’ grows, the model becomes less prone to overfitting but may sacrifice predictive accuracy and explanatory capability. Hence, the selection of ‘C’ necessitates careful consideration to strike an optimal balance between these competing factors.

Sigma’ being held constant suggests that the width of the RBF kernel was determined to be optimal at 0.1280465 for my dataset, and further variations in ‘sigma’ did not significantly improve the model’s performance.

‘C’ was chosen as 2, which represents the regularization parameter. A smaller ‘C’ typically results in a larger margin with more support vectors, while a larger ‘C’ can lead to a narrower margin with fewer support vectors. The value of 2 suggests a moderate level of regularization.

6.2.2.2.1 Shapley values:

By employing the kernel SHAP (SHapley Additive exPlanations) method within the Support Vector Machine framework, Shapley values have been derived as follows:

Chart 16: Shapley value for support vector machine

The graph clearly indicates that mean temperature and solar radiation have a direct impact on the target variable (distance traveled), with higher temperatures associated with greater ride distances. In contrast, PM2.5, CO, and wind speed exert a negative impact on the distance, signifying that higher PM2.5 levels and wind speed are linked to lower distances traveled. Notably, minimum humidity appears to have no discernible effect on the target variable. However, for maximum humidity, the distance traveled reaches its maximum around zero.

6.2.2.3 Random Forest

Random Forest was applied to the dataset using a five-fold cross-validation approach, yielding the following results:

mtry	RMSE	Rsquared	MAE
2	35859.37	0.491164	27770.5
3	35585.3	0.496719	27356.52
4	35689.3	0.493135	27294.35
5	35721.4	0.49178	27321.85
6	35750.55	0.491136	27273.8
7	35994.14	0.485027	27453.12
8	36078.77	0.482076	27529.04
9	35725.46	0.492086	27414.89
10	36082.12	0.482481	27608.94
11	35979.53	0.485422	27460.93

Table 22: random forest results

The results indicate that a larger value for "mtry" (Number of Variables Randomly Selected at Each Split), specifically a value of 6, resulted in improved model performance. This improvement is evident in both predictive accuracies, as indicated by lower values of RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error), and increased explanatory power, as reflected in a higher R-squared.

It's worth highlighting that increasing "mtry" tends to make individual trees in the Random Forest more decorrelated, which can effectively reduce overfitting. However, the choice of the optimal "mtry" value should be made judiciously, considering both cross-validation results and domain knowledge. This ensures that the model generalizes effectively to new, unseen data.

In this context, the selection of "mtry = 6" represents a well-balanced choice, striking the optimal trade-off between predictive accuracy and model complexity based on the MAE criterion. This finding underscores the importance of thoughtful hyperparameter tuning in Random Forest modeling to achieve the best possible performance.

6.2.2.3.1 Shapley values:

Utilizing the kernel SHAP (SHapley Additive exPlanations) method within the framework of Random Forest, Shapley values have been computed as follows:

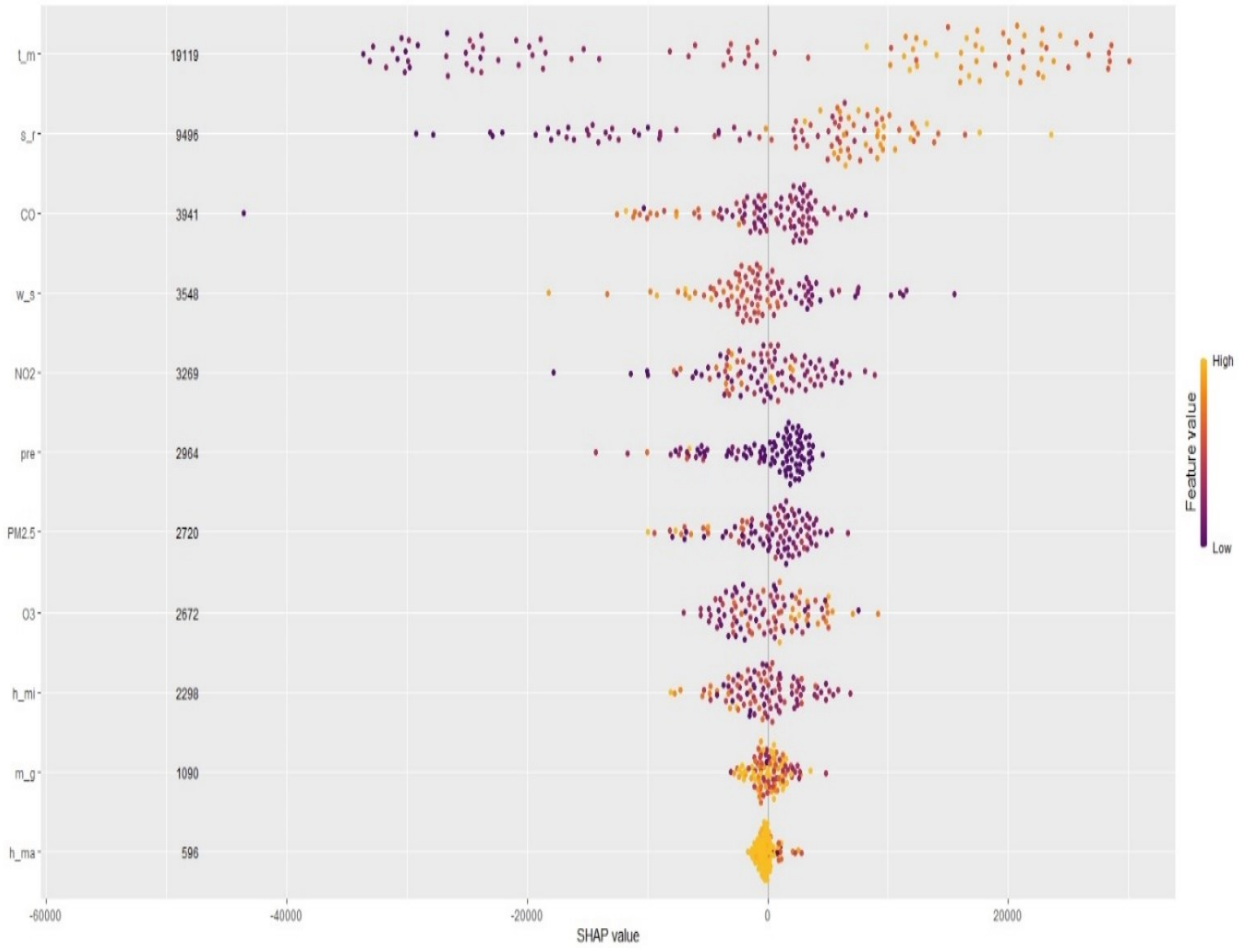


Chart 17: Shapley value for Random Forest

The graph reveals that mean temperature and solar radiation positively influence the distance traveled by shared bikes. Conversely, carbon monoxide, wind speed, precipitation, and PM2.5 exhibit a negative effect on the distance traveled. Notably, when max humidity is approximately zero, the distance traveled is high. Other variables do not show a clear impact on the target variable.

6.2.3 Comparing the results of trained models

Results obtained by three models containing linear regression, support vector machine and random forest are as following:

	RMSE	R-squared	MAE
Linear Regression	37382.93	0.442734	30345.87
Support Vector Machine	38126.46	0.428617	29672.37
Random Forest	35750.55	0.491136	27273.8

Table 23: comparing the results of different algorithms on training data

Among the three models, Random Forest performs the best in terms of predictive accuracy, as indicated by the lowest RMSE and MAE values. It also has the highest R-squared value, indicating a relatively better fit to the data.

The Support Vector Machine (SVM) model has the second-best performance in terms of RMSE and R-squared. While its RMSE is slightly higher than that of Linear Regression, it has the lowest MAE, suggesting better accuracy.

Linear Regression performs the least favorably in terms of RMSE and MAE, indicating relatively higher prediction errors. Its R-squared value also suggests it explains less variance in the target variable compared to the other models.

6.2.4 Testing the models applied for predicting the distance traveled by shared bike

When applying three different models—namely, Linear Regression, Support Vector Machine, and Random Forest to the testing dataset, the following results have emerged:

	RMSE	R-squared	MAE
Linear Regression	30779.0225	0.531819	24044.03
Support Vector Machine	3.31E+04	4.65E-01	2.61E+04
Random Forest	3.15E+04	5.12E-01	2.55E+04

Table 24: comparing the results of different algorithms on test data

Linear Regression has the lowest RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error), which indicates better predictive accuracy and smaller prediction errors on the testing dataset. So, in terms of RMSE and MAE, Linear Regression outperforms both Support Vector Machine (SVM) and Random Forest on the testing dataset. Additionally, the higher R-squared value for Linear Regression suggests that it explains a larger proportion of the variance in the target variable compared to the other models.

Therefore, based on these specific metrics (RMSE, MAE, and R-squared), Linear Regression is the best-performing model among the three for this dataset.

Concluding that consistent model performing well on both the training and test sets and indicating learning meaningful patterns in the data is random forest so this algorithm can be used to predict the traveled distance based on the weather variables.

Furthermore, according to the results derived from the Shapley values, mean temperature, carbon monoxide, and solar radiation emerge as the most influential factors in bike usage. This suggests that bike users exhibit a higher inclination to utilize shared bikes in conditions characterized by sunny weather, warmth, and clean air. As a recommendation to the municipality of Vicenza, it is advised to allocate more bikes on days with such favorable weather conditions. Additionally, creating incentives, such as discounts, for colder or rainy days could encourage bike usage during less favorable weather. Given the frequent rainy days in Vicenza, especially in winter, the suggestion extends to providing rain covers, as depicted in the attached picture, to enhance the convenience and appeal of bike-sharing services on such days.

Figure 11: bicycle with cover

6.3 Trip duration case study

6.3.1 Correlation Analysis of Weather Data Variables with Trip duration

The correlation between various variables, encompassing weather conditions, weather quality, and ride distance, has been evaluated using RStudio as the programming language.

Results are as following:

	dur	t_m	pre	h_mi	h_ma	s_r	w_s	m_g	co	no_2	O_3	pm10	pm2.5
dur	1	0.61	-0	-0.39	-0	0.61	0.1	-0.17	-0.5	-0.35	0.572	-0.4	-0.4
t_m	0.61	1	-0	-0.34	-0	0.75	0.3	-0.21	-0.7	-0.71	0.832	-0.6	-0.7
pre	-0.17	-0	1	0.34	0.12	-0.2	0.2	0.11	-0.1	-0.2	-0.049	-0.2	-0.2
h_mi	-0.39	-0.3	0.3	1	0.34	-0.7	-0.4	0.22	0.36	-0.09	-0.59	0.25	0.28
h_ma	-0.04	-0	0.1	0.34	1	-0.2	-0.3	0.22	0.08	-0.1	-0.195	-0	0.01
s_r	0.61	0.75	-0	-0.73	-0.2	1	0.4	-0.26	-0.6	-0.35	0.893	-0.5	-0.5
w_s	0.14	0.3	0.2	-0.35	-0.3	0.43	1	-0.24	-0.4	-0.29	0.542	-0.4	-0.4
m_g	-0.17	-0.2	0.1	0.22	0.22	-0.3	-0.2	1	0.26	0.174	-0.281	0.21	0.24
co	-0.5	-0.7	-0	0.36	0.08	-0.6	-0.4	0.26	1	0.655	-0.65	0.77	0.84
no2	-0.35	-0.7	-0	-0.09	-0.1	-0.4	-0.3	0.17	0.65	1	-0.52	0.66	0.69
O3	0.57	0.83	-0	-0.59	-0.2	0.89	0.5	-0.28	-0.6	-0.52	1	-0.6	-0.6
pm10	-0.37	-0.6	-0	0.25	-0	-0.5	-0.4	0.21	0.77	0.664	-0.556	1	0.97
pm2.5	-0.44	-0.7	-0	0.28	0.01	-0.5	-0.4	0.24	0.84	0.694	-0.609	0.97	1

Table 25: correlation between different variables

- The table illustrates the correlation among variables. Notably, a substantial correlation exists between PM10 and PM2.5. To mitigate multicollinearity, a decision was made to remove one of these variables. Specifically, PM10, which exhibited a lower correlation with the dependent variable (trip duration), was eliminated.
- Additionally, noteworthy correlations were observed between O3 and mean temperature, as well as O3 and solar radiation.
- The correlation between CO and PM2.5 is high.

6.3.2 Training models for predicting the trip duration traveled by shared bike

6.3.2.1 Linear regression model

Upon applying the linear regression model to the training dataset to predict trip duration, the ensuing results are as follows:

Min	1Q	Median	3Q	Max
-1370.9	-288.35	-15.86	254.69	2429.59

Table 26: analyzing the symmetricity of residuals

- Residual standard error: 458.1 on 281 degrees of freedom
- Multiple R-squared: 0.4631
- Adjusted R-squared: 0.4421
- F-statistic: 22.03 on 11 and 281 DF
- p-value: < 2.2e-16

As evident from the results, the distribution of residuals does not exhibit symmetry. Several variables, notably medium temperature, solar radiation, wind speed, and carbon dioxide, have stronger correlation with the target variable, trip duration.

The RSE of 458.1 indicates the average absolute difference between the observed and predicted values in the model.

The multiple R-squared value of 0.4631 suggests that the independent variables in the model collectively explain approximately 46.31% of the variance in the dependent variable.

The adjusted R-squared value of 0.4421, while slightly lower than the multiple R-squared value, accounts for the number of predictors in the model and provides a more conservative estimate of the model's goodness of fit.

6.3.2.1.1 Fine tuning the linear model

- **Feature Selection/Engineering**

After removing certain features with the highest p-values from the dataset, the following results were obtained:

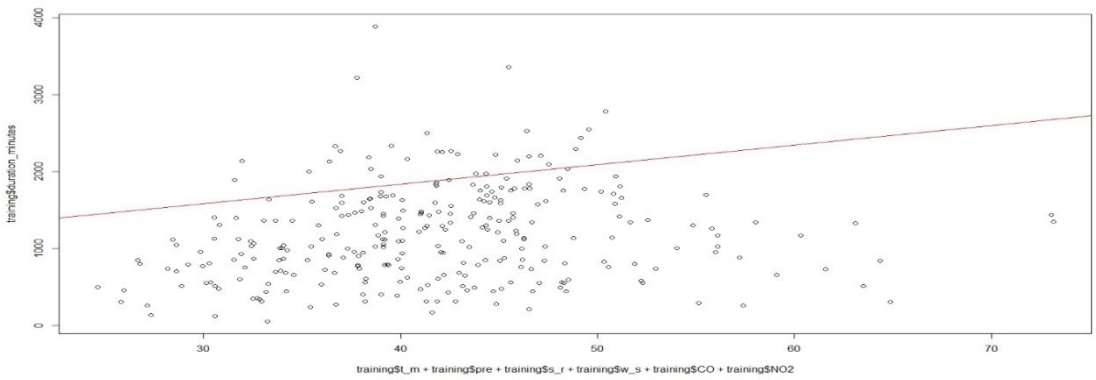


Chart 18: linear regression

Min	1Q	Median	3Q	Max
-1388.77	-279.22	-18.68	261.09	2439.22

Table 27 : analyzing the symmetricity of residuals

- Residual standard error: 455 on 285 degrees of freedom
- Multiple R-squared: 0.4628
- Adjusted R-squared: 0.4496
- F-statistic: 35.08

As can be seen some variables, notably medium temperature, solar radiation, wind speed, and carbon dioxide, play pivotal roles in predicting the trip duration.
The RSE of 455 indicates the average absolute difference between the observed and predicted values in the model.
The multiple R-squared value of 0.4628 suggests that the independent variables in the model collectively explain approximately 46.28% of the variance in the dependent variable.
The adjusted R-squared value of 0.4496, while slightly lower than the multiple R-squared value, accounts for the number of predictors in the model and provides a more conservative estimate of the model's goodness of fit.
The significant F-statistic of 35.08 indicates that the overall model is statistically significant, suggesting that at least some of the independent variables are relevant in explaining the variance in the dependent variable.

6.3.2.1.2 shapely value:

Utilizing the kernel SHAP (SHapley Additive exPlanations) method within the framework of linear regression model, Shapley values have been computed as follows:

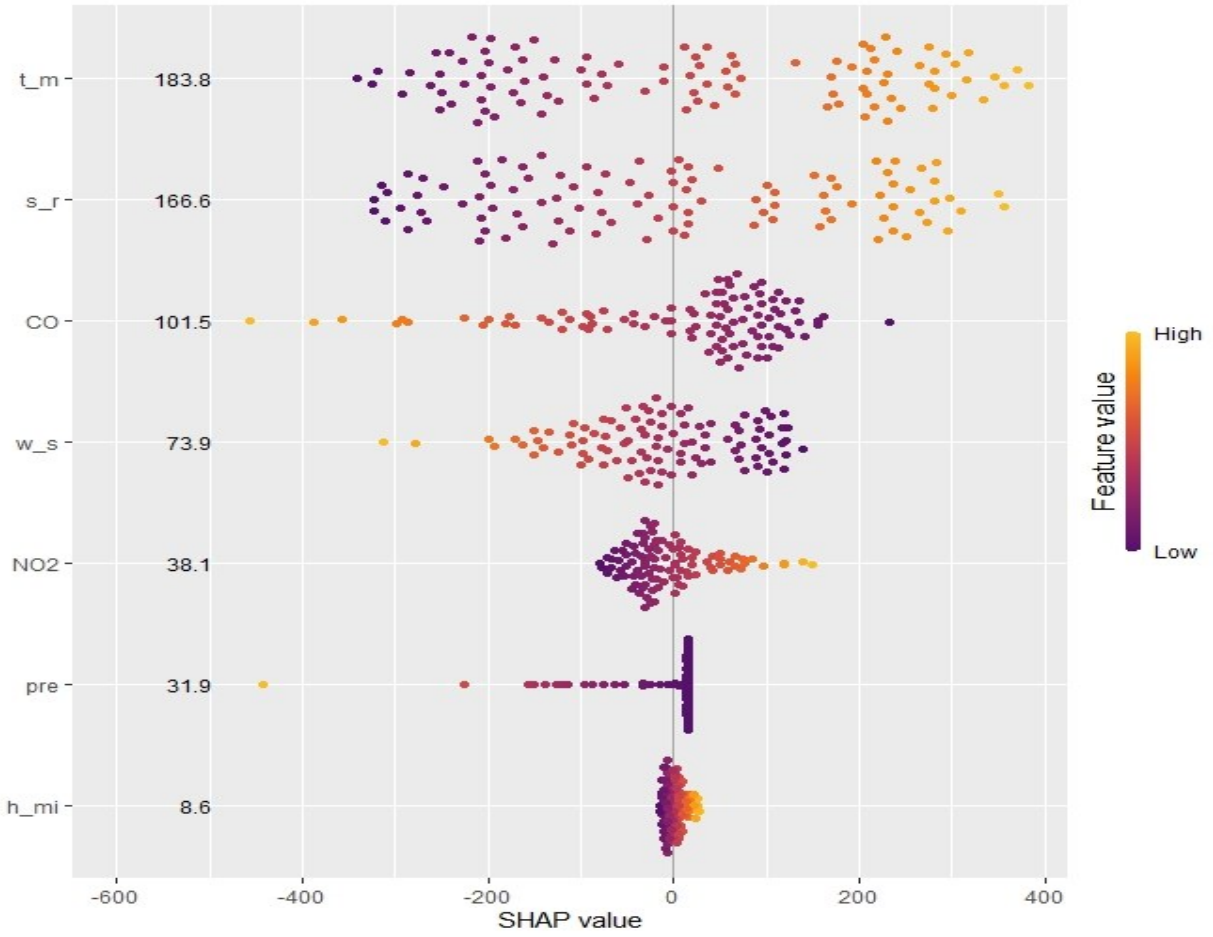


Chart 19: Shapley value for linear regression

The graph illustrates that mean temperature and solar radiation have a positive impact on the target variable, indicating that higher temperatures result in longer trip durations. Additionally, lower levels of carbon monoxide (CO) and wind speed are associated with extended trip durations.

6.3.2.1.3 Cross validation

The model has undergone refinement through the application of cross-validation techniques, aiming to offer a more precise evaluation of its generalization performance. This iterative process significantly augments the model's capability to make accurate predictions on previously unseen data.

RMSE	R-squared	MAE
456.2802	0.456366	337.8266

Table 28: cross validation results

The RMSE value of 456.2802 indicates that, on average, the linear regression model's predictions have an absolute error of approximately 456.28 units.

The R-squared value of 0.456366 suggests that the model explains about 45.64% of the variance in the dependent variable. This indicates a moderate level of predictive power.

The MAE value of 337.8266 represents the average absolute prediction error, which is the average absolute difference between the actual and predicted values.

6.3.2.2 Support vector machine

Upon implementing the Support Vector Machine (SVM) model and employing a five-fold cross-validation approach, the following results have been achieved:

C	RMSE	R-squared	MAE
0.25	460	0.45	346
0.5	454	0.46	339
1	450	0.47	335
2	453	0.45	336
4	462	0.43	340
8	476	0.41	349
16	500	0.37	367
32	545	0.32	393
64	598	0.26	426
128	638	0.24	458

Table 29: support vector machine results

The final SVM regression model with 'sigma' = 0.08886073 and 'C' = 1 achieved the lowest MAE, indicating that it has the smallest average absolute prediction error among the tested models.

The R-squared values indicate how well the selected model explains the variance in the target variable. An R-squared value of 0.4651 suggests that this model explains

approximately 46.51% of the variance in the target variable.

RMSE values show the model's prediction accuracy. Smaller RMSE values indicate better accuracy, and in this case, the selected model has a relatively low RMSE.

6.3.2.2.1 shapely value:

Utilizing the kernel SHAP (SHapley Additive exPlanations) method within the framework of support vector machine, Shapley values have been computed as follows:

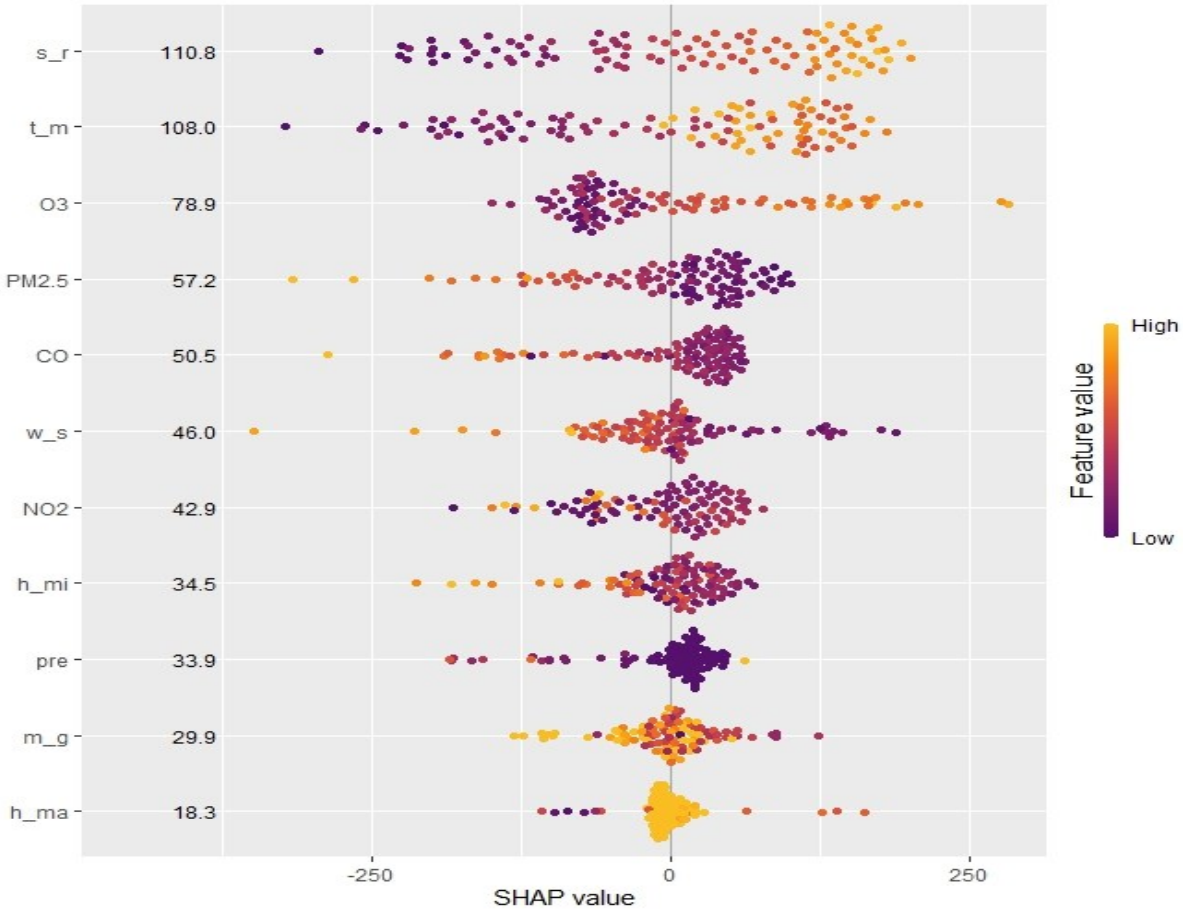


Chart 20: Shapley value for support vector machine

As evident in the graph, solar radiation, mean temperature, and ozone exhibit a direct positive impact on trip duration. Consequently, on sunny days characterized by higher temperatures and stronger solar radiation, the trip duration by shared bike tends to be longer. Conversely, precipitation, PM2.5, and carbon monoxide show a negative impact on trip duration. Notably, concerning maximum humidity, the graph

illustrates that humidity around zero is associated with longer trip durations.

6.3.2.3 Random forest

When applying the Random Forest model and utilizing a five-fold cross-validation technique, the ensuing results have been obtained:

mtry	RMSE	R-squared	MAE
2	439.1396	0.48846	324.2596
3	437.9219	0.490296	324.721
4	437.7012	0.490982	323.4035
5	437.836	0.491031	324.6855
6	439.3314	0.487287	325.0474
7	439.1621	0.487955	324.409
8	440.3393	0.485417	325.9317
9	439.5299	0.487639	327.124
10	439.3902	0.487605	326.1184
11	440.9964	0.484334	326.8041

Table 30: random forest results

The Random Forest regression model with mtry = 4 achieved the lowest MAE, indicating that it has the smallest average absolute prediction error among the tested models. This suggests that it provides the most accurate predictions.

The R-squared values indicate how well the selected model explains the variance in the target variable. An R-squared value of 0.4909815 suggests that this model explains approximately 49.10% of the variance in the target variable.

RMSE values show the model's prediction accuracy. Smaller RMSE values indicate better accuracy, and in this case, the selected model with mtry = 4 has a relatively low RMSE.

6.3.2.3.1 shapely value

Utilizing the kernel SHAP (SHapley Additive exPlanations) method within the framework of random forest model, Shapley values have been computed as follows:

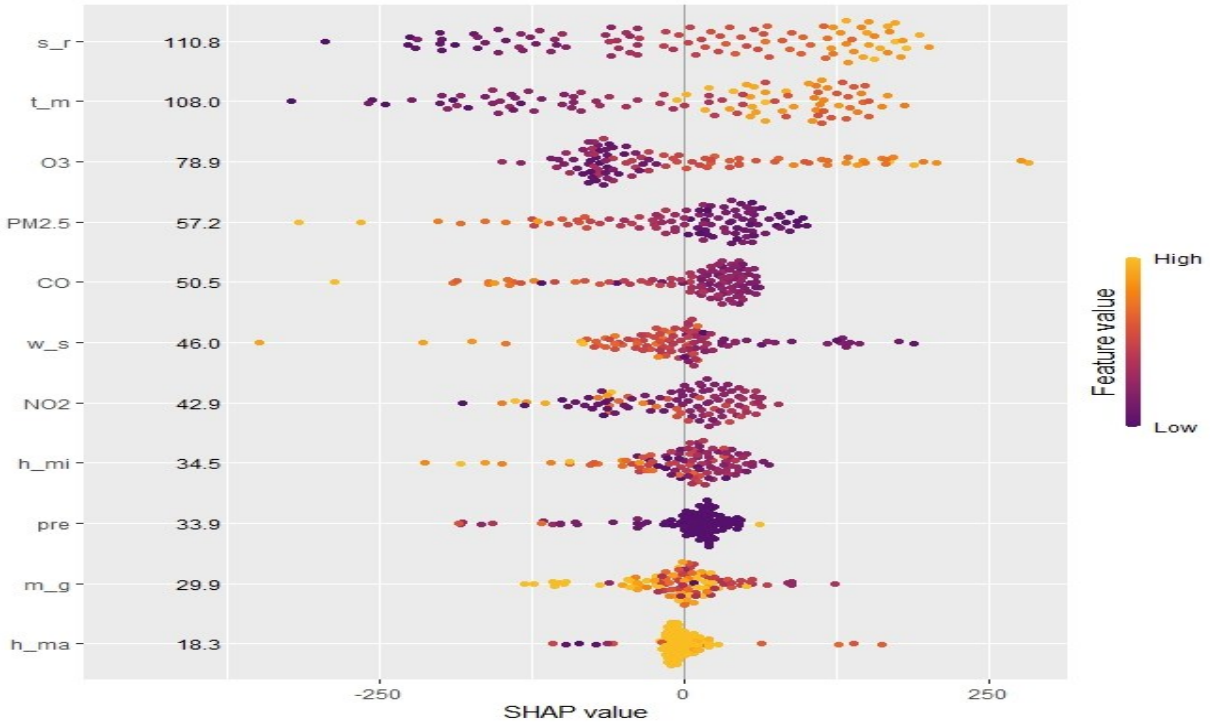


Chart 21: Shapley value for Random Forest

As observed in the graph, solar radiation and mean temperature positively impact trip duration. Conversely, PM2.5, carbon monoxide, and wind speed exhibit a negative impact on trip duration. This implies that on days characterized by windiness and higher pollution levels, the duration of trips tends to be shorter.

6.3.3 Comparing the results of trained models

Three distinct models including linear regression, support vector machine and random forest were applied to the training dataset, yielding the following results:

	RMSE	R-squared	MAE
linear regression	456.2802	0.456366	337.8266
support vector machine	449.9948	0.465095	335.1681
random forest	437.7012	0.490982	323.4035

Table 31: comparing the results of different algorithms on training data

Random Forest appears to be the best-performing model among the three, with the lowest RMSE, highest R-squared, and lowest MAE. It offers better predictive accuracy and explains a larger portion of the variance in the target variable compared to Linear Regression and Support Vector Machine.

6.3.4 Testing the models applied for predicting the trip duration traveled by shared bike

When subjecting various regression models, including Linear Regression, Support Vector Machine, and Random Forest, to the testing dataset, the following outcomes have been observed:

	RMSE	R-squared	MAE
linear regression	515.91	0.477097	294.5791
support vector machine	554.3282	0.403741	312.0009
random forest	510.6293	0.48583	306.5038

Table 32:comparing the results of different algorithms on test data

As can be seen, Random Forest appears to be the best-performing model among the three, with the lowest RMSE, highest R-squared, and lowest MAE. It offers better predictive accuracy and explains a larger portion of the variance in the target variable compared to Linear Regression and Support Vector Machine.
Concluding that the random forest is the consistent algorithm that has the best performance in both testing and training data set. So for the prediction of the travel duration by bikes based on the weather variables random forest can be used.
Given the positive impact of mean temperature and solar radiation on trip duration, it becomes evident that during summer, characterized by more sunny and warm days, there is likely to be increased bike usage. As a strategic initiative, it is recommended that the municipality of Vicenza consider investing more in providing bikes during the summer season. Furthermore, organizing winter cycling events or challenges could serve as a motivational factor for residents to continue biking during colder months. Collaborating with local businesses to sponsor winter biking initiatives can prove beneficial in fostering community engagement.
To enhance the winter biking experience, the municipality can conduct workshops on winter biking techniques and safety. These workshops would educate cyclists on navigating winter conditions and dressing appropriately for the weather. Such educational efforts contribute not only to the safety of cyclists but also to the promotion of biking as a viable transportation option throughout the year.

8 CONCLUTION

This study aimed to analyze shared bike usage patterns in relation to weather conditions in Vicenza, a city situated in the north of Italy. Machine learning algorithms, including linear regression, support vector machine, and random forest, were employed for this analysis.

The results obtained from machine learning algorithms indicated that the random forest model outperformed others in predicting bike shared usage in various weather conditions, showcasing superior performance in both testing and training datasets. Upon closer examination of the model, it was revealed that variables such as mean temperature and solar radiation had a positive impact on bike usage. In contrast, carbon monoxide and PM2.5 showed a negative impact, suggesting that residents exhibit greater inclination to use shared bikes in sunny, warm weather with lower pollution levels. Conversely, in rainy, cold, and polluted conditions, residents prefer alternative means of transportation.

Based on these findings, the municipality has an opportunity to tailor strategies to encourage year-round bike usage. The positive correlation observed between mean temperature, solar radiation, and increased trip duration during summer underscores the potential for heightened bike usage in warmer and sunnier weather. Initiatives such as promoting bike-sharing programs, enhancing bike-friendly infrastructure, and organizing events during the summer months can capitalize on this trend.

Considering the prevalence of rainy days, particularly in winter, specific measures are recommended to address challenges posed by wet weather. Implementing rain-ready infrastructure, promoting weather-appropriate gear, offering promotions on rainy days, and integrating biking with alternative transportation during adverse weather conditions can mitigate the impact of rain on biking and create a more resilient biking culture.

However, it is essential to acknowledge some limitations in this study, such as not considering users' age, gender, and standard of living. Future research could extend the

study by incorporating additional variables like cycling patterns, safety perceptions, user characteristics, and the influence of seasonal tourism on bike-sharing usage in the city.

Moreover, conducting a similar study in larger cities like Milan, Rome, or Berlin would provide insights into how factors like longer distances between starting and ending points can affect bike usage, contributing to a more comprehensive understanding of shared bike utilization in urban environments.

LIST OF TABLES

LIST OF CHARTS

LIST OF FIGURES

REFERENCES

1. Qifang Bi, Katherine E Goodman, Joshua Kaminsky, Justin Lessler
   American Journal of Epidemiology, Volume 188, Issue 12, December 2019, Pages 2222–2239, What is Machine Learning? A Primer for the Epidemiologist
   https://academic.oup.com/aje/article/188/12/2222/5567515

2. Efron B. Estimating the error rate of a prediction rule: improvements on cross validation. J Am Stat Assoc 1983, 78: 316–331.
   https://www.jstor.org/stable/2288636

3. Stone M. Cross-validatory choice and assessment of statistical predictions. J R Stat Soc B 1974, 36: 111–147.
   https://www.jstor.org/stable/2984809

4. Good IJ, Gaskins RA. Density estimation and bump hunting by the penalized likelihood method exemplified by scattering and meteorite data. J Am Stat Assoc 1980, 75: 42–56.
   https://www.jstor.org/stable/2287377

5. Aas, Kjersti, Martin Jullum, and Anders Løland. 2019. "Explaining Individual Predictions When Features Are Dependent: More Accurate Approximations to Shapley Values." arXiv Preprint arXiv:1903.10464.
   https://arxiv.org/abs/1903.10464

6. Hothorn, Torsten, Kurt Hornik, and Achim Zeileis. 2006. "Unbiased Recursive Partitioning: A Conditional Inference Framework." Journal of Computational and Graphical Statistics 15 (3): 651–74.
   https://www.tandfonline.com/doi/abs/10.1198/106186006X133933

7. Hothorn, Torsten, and Achim Zeileis. 2015. "partykit: A Modular Toolkit for Recursive Partytioning in R." Journal of Machine Learning Research 16: 3905–9.
   https://www.jmlr.org/papers/v16/hothorn15a.html

8. Lundberg, Scott M, and Su-In Lee. 2017. "A Unified Approach to Interpreting Model Predictions." In Advances in Neural Information Processing Systems, 4765–74.
   https://www.semanticscholar.org/paper/A-Unified-Approach-to-Interpreting-Model-Lundberg-Lee/442e10a3c6640ded9408622005e3c2a8906ce4c2

9. Redelmeier, Annabelle, Martin Jullum, and Kjersti Aas. 2020. "Explaining Predictive Models with Mixed Features Using Shapley Values and Conditional Inference Trees." Submitted.
   https://www.researchgate.net/publication/343750768_Explaining_Predictive_Models_with_Mixed_Features_Using_Shapley_Values_and_Conditional_Inference_Trees

10. Rosenblatt, Murray. 1956. "Remarks on Some Nonparametric Estimates of a Density Function." The Annals of Mathematical Statistics 27: 832–37.
    https://www.jstor.org/stable/2237390

11. Shapley, Lloyd S. 1953. "A Value for N-Person Games." Contributions to the Theory of Games 2: 307–17.
https://books.google.it/books?hl=it&lr=&id=Pd3TCwAAQBAJ&oi=fnd&pg=PA307&dq=Shapley,+Lloyd+S.+1953.+%E2%80%9CA+Value+for+N-Person+Games.%E2%80%9D+Contributions+to+the+Theory+of+Games+2:+307%E2%80%9317.&ots=gunVLbcisZ&sig=I3rEfj3_X4ANRnz1WSKufBRju0w&redir_esc=y#v=onepage&q&f=false

12. Dennis Wackerly, William Mendenhall, and Richard L. Scheaffer, Mathematical Statistics with Applications

13. Huthaifa I. Ashqar[a], Mohammed Elhenawy[b], Hesham A. Rakha (2019), Modeling bike counts in a bike-sharing system considering the effect of weather conditions.
https://www.sciencedirect.com/science/article/pii/S2213624X16301018

14. Joost de Kruijf [a], Peter van der Waerden [b], Tao Feng [b], Lars Böcker [d], Dea van Lierop [a], Dick Ettema [a], Martin Dijst [c],(2021) Integrated weather effects on e-cycling in daily commuting: A longitudinal evaluation of weather effects on e-cycling in the Netherlands
https://www.sciencedirect.com/science/article/pii/S0965856421000951

15. Jan Wessel ,University of Münster, Institute of Transport Economics, Am Stadtgraben 9, 48143 Münster, Germany (2020), Using weather forecasts to forecast whether bikes are used
https://www.sciencedirect.com/science/article/pii/S0965856420306157

16. Craig Morton,School of Architecture, Building, and Civil Engineering, Loughborough University, Loughborough LE11 3TU, United Kingdom (2020), The demand for cycle sharing: Examining the links between weather conditions, air quality levels, and cycling demand for regular and casual users
https://www.sciencedirect.com/science/article/pii/S0966692319306167

17. Ahmed Jaber *, Ba´ lint Csonka, Department of Transport Technology and Economics, Faculty of Transportation Engineering and Vehicle Engineering, Budapest University of Technology and Economics, M}uegyetem rkp. 3., H-1111 Budapest, Hungary,(2023) Investigating the temporal differences among bikesharing users through comparative analysis based on count, time series, and data mining models
https://www.sciencedirect.com/science/article/pii/S1110016823005641

18. Ahmadreza Faghih-Imani a, Robert Hampshire b, Lavanya Marla c,, Naveen Eluru d a Department of Civil Engineering and Applied Mechanics, McGill University, Canada b Transportation Research Institute, University of Michigan, United States c Department of Industrial and Systems Engineering, University of Illinois at Urbana-Champaign, United States d Department of Civil, Environmental and Construction Engineering, University of Central Florida, United States,(2017) An empirical analysis of bike sharing usage and rebalancing:
Evidence from Barcelona and Seville

https://www.sciencedirect.com/science/article/pii/S0965856416311648

19. Hongtai Yang a, Jinghai Huo b,*, Yongxing Bao b, Xuan Li b, Linchuan Yang c, Christopher R. Cherry d a School of Transportation and Logistics, National Engineering

Laboratory of Integrated Transportation Big Data Application Technology, National United Engineering Laboratory of Integrated and Intelligent Transportation, Institute of System Science and Engineering, Southwest Jiaotong University, Chengdu, Chinab School of Transportation and Logistics, National Engineering Laboratory of Integrated Transportation Big Data Application Technology, National United Engineering Laboratory of Integrated and Intelligent Transportation, Southwest Jiaotong University, Chengdu, China c School of Architecture and Design, Department of Urban and Rural Planning, Southwest Jiaotong University, Chengdu, China d Civil and Environmental Engineering, University of Tennessee-Knoxville, Knoxville, TN 37996-2313, USA, (2021) Impact of e-scooter sharing on bike sharing in Chicago

https://www.sciencedirect.com/science/article/pii/S0965856421002445

20. Yongping Zhanga,1, Zhifu Mib,⁎,1 a The Bartlett Centre for Advanced Spatial Analysis, University College London, 90 Tottenham Court Road, London W1T 4TJ, UK
b The Bartlett School of Construction and Project Management, University College London, 1-19 Torrington Place, London WC1E 7HB, UK,(2018), Environmental benefits of bike sharing: A big data-based analysis
https://www.sciencedirect.com/science/article/pii/S0306261918304392

21. El Arbi Abdellaoui Alaoui a,b,*, Stephane Cedric Koumetio Tekouabou c,d
a EIGSI-Casablanca, 282 Route of the Oasis, Casablanca, Morocco b My Ismail University, Morocco c Center of Urban Systems (CUS), Mohammed VI Polytechnic University (UM6P), Hay Moulay Rachid, 43150 Ben Guerir, Morocco d Laboratory LAROSERI, Department of Computer Science, Faculty of Sciences, B.P. 20, 24000 El Jadida, Morocco (2021)
Intelligent management of bike sharing in smart cities using machine learning and Internet of Things
https://www.sciencedirect.com/science/article/pii/S2210670720309161

22. Mahmoud Elgendi , Mohamed Atef , a Department of Mechanical and Aerospace Engineering, United Arab Emirates University, Al Ain City, United Arab Emirates b National Water and Energy Center, United Arab Emirates University, Al Ain, P.O. Box 15551, United Arab Emirates c Department of Mechanical Power Engineering and Energy, Faculty of Engineering, Minia University, Minia, Egypt d Electrical and Communications Engineering Department, United Arab Emirates University, 15551, Al Ain, United Arab Emirates(2023), Calculating the impact of meteorological parameters on pyramid solar still yield using machine learning algorithms
https://www.sciencedirect.com/science/article/pii/S2666202723000605

23. Ran Ana, Renee Zahnowb, Dorina Pojania, Jonathan Corcorana, a School of Earth and Environmental Sciences, The University of Queensland, Brisbane, Australia b School of Social Sciences, (2019)The University of Queensland, Brisbane, Australia, Weather and cycling in New York: The case of Citibike
https://www.sciencedirect.com/science/article/pii/S0966692318307282

24. Kyoungok Kim, Information Technology Management Programme, International Fusion School, Seoul National University of Science & Technology (SeoulTech), 232 Gongreungno, Nowongu, Seoul 139-743, Republic of Korea,(2018) Investigation on the effects of weather and calendar events on bike-sharing according to the trip patterns of bike rentals of stations

https://www.sciencedirect.com/science/article/pii/S0966692317304659

25. Anik Das, Ali Ghasemzadeh, ∗ Mohamed M. Ahmed, University of Wyoming, Department of Civil & Architectural Engineering, 1000 E University Ave, Dept. 3295, Laramie, WY 82071, United States(2019), Analyzing the effect of fog weather conditions on driver lane-keeping
performance using the SHRP2 naturalistic driving study data
https://www.sciencedirect.com/science/article/pii/S0022437518300744

26. Ying Wang a, Yang Yang b,*, Songshan (Sam) Huang c, Li Huang b, Weijie Sun b
a Business School, Sichuan University, Chengdu, Sichuan, 610065, China
b College of History and Culture (Tourism), Sichuan University, Chengdu, Sichuan, 610065, China c School of Business and Law, Edith Cowan University, Joondalup, WA, 6027, Australia, (2021) Effects of air quality and weather conditions on Chinese tourists'
emotional experience
https://www.sciencedirect.com/science/article/pii/S1447677021000814

APPENDIX

Bike descriptive analysis

```
library(readr)
install.packages("rlang")
packageVersion("rlang")

install.packages("ggplot2")
library(ggplot2)

data <- read_csv("main_data.csv")
View(data)

bar_chart <- ggplot(data, aes(x = data$start_time, y = data$duration_minutes)) +
 geom_bar(stat = "identity", fill = "blue") +
 labs(title = "Bar Chart with Trend Line",
     x = "Category",
     y = "Value") +
 theme_minimal()

# Add a linear trend line using a second y-axis
trend_line <- bar_chart +
 geom_smooth(method = "lm", se = FALSE, color = "red") +
 scale_y_continuous(
  sec.axis = sec_axis(~ ., name = "Trend Line", breaks = NULL)
 )

# Display the chart with the trend line
print(trend_line)
#####################################
library(readr)
library(dplyr)
library(corrplot)
library(car)
library(stringi)
library(ggplot2)
library(plyr)
library(readxl)
library(lattice)
library(reshape2)
main_data <- read_excel("main_data.xlsx")
View(main_data)
hist(residuals(duration_minutes),col="red4",prob=TRUE,breaks=10)
boxplot(residuals(duration_minutes), col = "green")
hist(main_data , breaks = 10, main = "Histogram of Data", xlab = "Value")
boxplot(duration_minutes)
p<- ggplot(main_data, aes(x=duration_minutes, y=user_id))
p
boxplot(main_data$duration_minutes,main_data$ride_distance_meters,
```

```
main_data$user_id,main_data$promotion_deduction,main_data$pass_user,col = "red")
data_long<- melt(main_data)
ggplot(data_long,aes(x=variable, y=value))
boxplot(value~variable,data_long)


scaled_duration<- scale(main_data$duration_minutes)
scaled_ride<- scale(main_data$ride_distance_meters)
scaled_user<- scale(main_data$user_id)
scaled_amount<- scale(main_data$original_total_amount)
scaled_promotion<- scale(main_data$promotion_deduction)
scaled_pass<- scale(main_data$pass_user)
my_table<-
data.frame(scaled_duration,scaled_ride,scaled_user,scaled_amount,scaled_promotion,scale
d_pass)
boxplot(my_table ,names = c("duration","distance",
"users","payed","promotion","passed"), col="red",
      main = "Multiple Box Plot of Variables", ylab = "Scaled Value")
View(my_table)
data.stand<- scale(main_data$duration_minutes)
View(data.stand)
boxplot(scaled_user)
boxplot(scaled_duration, scaled_ride, scaled_user, names = c("duration_minutes",
"ride_distance_meters", "user_id"),
      main = "Multiple Box Plot of Variables")

boxplot(duration_minutes,ride_distance_meters,user_id,original_total_amount,promotion_
deduction,pass_user,names = c("duration","distance",
"users","payed","promotion","passed"), col="red",
      main = "Multiple Box Plot of Variables", ylab = "Value")
View(duration_minutes)
hist(main_data$duration_minutes, main = "Histogram of duration_minutes", xlab =
"Value", col="green")
curve(dnorm(x, mean(main_data$user_id), sd(main_data$user_id)), add = TRUE, col =
"red", lwd = 2)

hist(main_data$ride_distance_meters, breaks = 365, freq = FALSE,col = "blue", main =
"Histogram with Curve Line")

hist(main_data$ride_distance_meters, main = "Histogram of ride_distance_meters",
xlab="Days",ylab = "value", col="red" ,breaks = 365)
hist(main_data$original_total_amount, main = "Histogram of original_total_amount", xlab =
"Value", col="blue")
hist(main_data$promotion_deduction, main = "Histogram of promotion_deduction", xlab =
"Value", col="violet")
hist(main_data$pass_user, main = "Histogram of pass_user", xlab = "Value", col="orange")
hist(main_data$user_id, main = "Histogram of user_id", xlab = "Value", col="brown")

##################################

library(readr)
install.packages("ggplot2")
```

```
library(ggplot2)
data <- read_csv("main_data.csv")
View(data)
datta<- data[,c("duration_minutes","weekend")]
weekdays <- subset(datta, weekend == 0)
weekends <- subset(datta, weekend == 1)
result <- t.test(weekdays$duration_minutes, weekends$duration_minutes)
print(result)
View(weekdays)
View(weekends)
boxplot(weekdays$duration_minutes,weekends$duration_minutes, col = "green")

mean(weekdays$duration_minutes)
mean(weekends$duration_minutes)

dataa<- data[,c("ride_distance_meters","season","duration_minutes")]
spring<-  subset(dataa,season==1)
summer<- subset(dataa,season==2)
fall<- subset(dataa,season==3)
winter<- subset(dataa,season==4)
sum(spring$ride_distance_meters)
sum(spring$duration_minutes)
sum(summer$ride_distance_meters)
sum(summer$duration_minutes)
sum(fall$ride_distance_meters)
sum(fall$duration_minutes)
sum(winter$ride_distance_meters)
sum(winter$duration_minutes)

#################################################################
Weather data descriptive analysis

library(readr)
library(dplyr)
library(corrplot)
library(car)
library(stringi)
library(ggplot2)
library(plyr)
library(readxl)
library(lattice)
library(reshape2)
library(readxl)
library(dbscan)
library(ggplot2)
library(maps)
library(sf)
library(leaflet)

install.packages("dbscan")
install.packages("factoextra")
install.packages("ggplot2")
```

```
install.packages("maps")
install.packages("sf")
install.packages("leaflet")
library(readr)
start_points <- read_csv("start_end_points.csv")
View(start_end_points)
end_point <- read_csv("end_point.csv")



ending <- read_excel("ending.xlsx")
starting <- read_excel("starting.xlsx")

start <- read_csv("start.csv")

end <- read_csv("end.csv")
frequency_start<- table(ending$longitude)
frequency_start
max(frequency_start)
most_frequent_data <- names(frequency_start)[which.max(frequency_start)]
most_frequent_data

print(frequency_start)
View(end)
final.f=ending
View(final.f)

results<kmeans(final.f,1)
results$centers
table(final$type,results$cluster)
plot(final.f,type="p",col=results$cluster+1)

db<- dbscan(final.f,eps=0.01, minPts = 2)
plot(final.f, col=db$cluster , main="dbscan")
centers <- final.f %>%
 filter(db$cluster != 0) %>%
 group_by(db$cluster) %>%
 summarize(center_longitude = mean(longitude), center_latitude = mean(latitude))

clustered_points <- final.f[db$cluster != -1, ]

# Calculate centroid using group_by and summarize
centers <- clustered_points %>%
 group_by(cluster = factor(db$cluster[db$cluster != -1])) %>%
 summarize(center_longitude = mean(longitude),
        center_latitude = mean(latitude))

centers
```

```
plot(start$latitude,start$longitude, type = "p")

db <- dbscan(cordination, eps = 0.5, MinPts = 5)

# Access the cluster labels
cluster_labels <- db$cluster
print(core_points)


print(startpoint)
#plotting the poits on the map

m <- leaflet() %>% addTiles()
m <- m %>% addCircleMarkers(
  data = start_points,
  lng = ~longitude,
  lat = ~latitude,
  color = "red",
  radius = 5
)
n <- m %>% addCircleMarkers(
  data = end_point,
  lng = ~longitude,
  lat = ~latitude,
  color = "blue",
  radius = 5
)
print(n)

#####################################################################

Algorithm


library(caret)
library(mlbench)
library(dplyr)
library(parallel)
library(doParallel)
library(pROC)
library(caTools)
library(readr)
library(corrplot)
library(car)

data <- read_csv("data.csv")
View(data)
set.seed(1234)
ids<-createDataPartition(data$duration_minutes,p=0.80,list=FALSE)
```

```
training<-data[ids,]
testing<-data[-ids,]
View(training)
dim(training)
dim(testing)
# Calculate correlations for multiple variables
correlations <- cor(training[, c('t_m', 'pre', 'h_mi', 'h_ma',
's_r','w_s','m_g','co','no2','o3','pm10','pm2.5')], training$duration_minutes)
print(correlations)
corrplot(correlations, method = "color")
correlations <- cor(training[, c('duration_minutes','t_m', 'pre', 'h_mi', 'h_ma',
's_r','w_s','m_g','co','no2','o3','pm10','pm2.5')])
print(correlations)
ctrl<-trainControl(method="cv",number=5,summaryFunction =
defaultSummary,allowParallel = TRUE)
cv_model <- train(ride_distance_meters ~
t_m+pre+h_mi+h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5, data = training, method = "lm",
trControl = ctrl)
cv_model
mod<- lm(ride_distance_meters~
t_m+pre+h_mi+h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5  , data= training    )
summary(mod)
mod1<- lm(ride_distance_meters~ t_m+pre+s_r+w_s+CO+O3  , data= training    )
summary(mod1)
abline(coef=coef(mod1), col="red")


plot(training$t_m+training$pre+training$s_r+training$w_s+training$CO+training$m_g+tr
aining$NO2+training$O3+training$PM2.5 , training$ride_distance_meters, type='p')

plot(training$t_m+training$pre+training$s_r+training$CO+training$O3+training$w_s ,
training$ride_distance_meters, type='p')



#////////////////////////////////////////////////////////////////////////////
/////
ctrl<-trainControl(method="cv",number=5,summaryFunction =
defaultSummary,allowParallel = TRUE)
set.seed(1234)
svm_mod<-train(
 form=ride_distance_meters~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
 data=training,
 method="svmRadial",
 trControl=ctrl,
 tuneLength=10,
 metric="MAE"
)
svm_mod

vif(svm_mod)
```

```
set.seed(123)
rf_mod<- train(
  ride_distance_meters ~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
  data = training,
  method = "rf",
  tuneLength = 10,
  trControl = ctrl,
  metric = "MAE"
)
rf_mod
#Evaluate using the test set
postResample(pred=predict(mod1,testing),obs=testing$ride_distance_meters)
postResample(pred=predict(svm_mod,testing),obs=testing$ride_distance_meters)
postResample(pred=predict(rf_mod,testing),obs=testing$ride_distance_meters)

#-----------------------------------------------------------------
 #duration_minutes


correlations <- cor(training[, c('duration_minutes','t_m', 'pre', 'h_mi', 'h_ma',
's_r','w_s','m_g','CO','NO2','O3','PM10','PM2.5')])
correlations
ctrl<-trainControl(method="cv",number=5,summaryFunction =
defaultSummary,allowParallel = TRUE)
cv_model <- train(duration_minutes ~
t_m+pre+h_mi+h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5, data = training, method = "lm",
trControl = ctrl)
cv_model
mod<- lm(duration_minutes~ t_m+pre+h_mi+h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5  ,
data= training    )
summary(mod)
plot(training$t_m+training$pre+training$s_r+training$w_s+training$CO+training$m_g+tr
aining$NO2+training$O3+training$PM2.5 , training$duration_minutes, type='p')

abline(coef=coef(mod), col="red")
mod1<- lm(duration_minutes~ t_m+pre+h_mi++s_r+w_s+CO+NO2  , data= training    )
summary(mod1)
plot(training$t_m+training$pre+training$s_r+training$w_s+training$CO+training$NO2 ,
training$duration_minutes, type='p')

abline(coef=coef(mod1), col="red")
ctrl<-trainControl(method="cv",number=5,summaryFunction =
defaultSummary,allowParallel = TRUE)
cv_model <- train(duration_minutes ~ t_m+pre+h_mi++s_r+w_s+CO+NO2, data = training,
method = "lm", trControl = ctrl)
cv_model
ctrl<-trainControl(method="cv",number=5,summaryFunction =
defaultSummary,allowParallel = TRUE)
set.seed(1234)
svm_mod<-train(
  form=duration_minutes~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
  data=training,
```

```
 method="svmRadial",
 trControl=ctrl,
 tuneLength=10,
 metric="MAE"
)
svm_mod
set.seed(123)
rf_mod<- train(
 duration_minutes ~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
 data = training,
 method = "rf",
 tuneLength = 10,
 trControl = ctrl,
 metric = "MAE"
)
rf_mod
postResample(pred=predict(mod1,testing),obs=testing$duration_minutes)
postResample(pred=predict(svm_mod,testing),obs=testing$duration_minutes)
postResample(pred=predict(rf_mod,testing),obs=testing$duration_minutes)
#//////////////////////

install.packages("kernelshap")
library(ggplot2)
library(kernelshap)
library(shapviz)

diamonds <- transform(
 training,
 log_ride_distance_meters = log(ride_distance_meters),
 log_t_m = log(t_m)
)
mod1<- lm(log_ride_distance_meters~ log_t_m+pre+s_r+w_s+CO+O3  , data= diamonds    )
fit_lm <- lm(log_price ~ log_carat + clarity + color + cut, data = diamonds)

# 1) Sample rows to be explained
set.seed(10)
xvars <- c("log_t_m","pre","s_r","w_s","CO","O3")
X <- diamonds[sample(nrow(diamonds), 100), xvars]

# 2) Select background data
bg_X <- diamonds[sample(nrow(diamonds), 200), ]

# 3) Crunch SHAP values for all 1000 rows of X (~7 seconds)
system.time(
 shap_lm <- kernelshap(mod1, X, bg_X = bg_X)
)
shap_lm

sv_lm <- shapviz(shap_lm)
sv_importance(sv_lm)
sv_dependence(sv_lm, "log_carat", color_var = NULL)
#//////////////////////////////////////
```

```
library(ranger)

set.seed(10)
xvars <- c("t_m","pre","h_mi","h_ma","s_r","w_s","m_g","CO","NO2","O3","PM2.5")
X <- training[sample(nrow(training), 100), xvars]

# 2) Select background data
bg_X <- training[sample(nrow(training), 200), ]
shap_rf <- kernelshap(mod1, X, bg_X = bg_X)
shap_rf
#////////////////////////////////////
#linear model for distance traveled
library(caret)
library(kernelshap)
library(shapviz)
library(ranger)

mod1<- lm(ride_distance_meters~ t_m+pre+s_r+w_s+CO+O3  , data= training    )
summary(mod1)
set.seed(10)
xvars <- c("t_m","pre","s_r","w_s","CO","O3")
X <- training[sample(nrow(training), 100), xvars]
bg_X <- training[sample(nrow(training), 200), ]
shap_rf <- kernelshap(mod1, X, bg_X = bg_X)
shap_rf
sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "CO")
dade<-c(t_m,pre,s_r,w_s,training$CO,training$O3,training$ride_distance_meters)

s <- kernelshap(mod1, dade[, -1], predict, bg_X = dade)
sv <- shapviz(s)
sv_waterfall(sv, 1)
#////////////////////////////////////
#random forest for distance traveled
library(caret)
library(kernelshap)
library(shapviz)
library(ranger)

rf_mod<- train(
  ride_distance_meters ~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
  data = training,
  method = "rf",
  tuneLength = 10,
  trControl = ctrl,
  metric = "MAE"
)

s <- kernelshap(rf_mod, training[, -1], predict, bg_X = training)

sv <- shapviz(s)
```

```
sv_waterfall(sv, 1)

sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "t_m")
#//////////////////////////////////////////
#support vector machine for distance traveled
library(caret)
library(kernelshap)
library(shapviz)
set.seed(1234)
svm_mod<-train(
 form=ride_distance_meters~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
 data=training,
 method="svmRadial",
 trControl=ctrl,
 tuneLength=10,
 metric="MAE"
)
svm_mod
shap_rf <- kernelshap(svm_mod, X, bg_X = bg_X)
shap_rf
sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "CO")

s <- kernelshap(svm_mod, training[, -1], predict, bg_X = training)
sv <- shapviz(s)
sv_waterfall(sv, 1)
#//////////////////////////////////////
#linear model for trip duration
library(caret)
library(kernelshap)
library(shapviz)
library(ranger)

mod1<- lm(duration_minutes~ t_m+pre+h_mi++s_r+w_s+CO+NO2  , data= training    )
xvars <- c("t_m","pre","s_r","w_s","CO","h_mi","NO2")
X <- training[sample(nrow(training), 100), xvars]
bg_X <- training[sample(nrow(training), 200), ]
s <- kernelshap(mod1, training[, -1], predict, bg_X = training)
sv <- shapviz(s)
sv_waterfall(sv, 1)
bg_X <- training[sample(nrow(training), 200), ]
shap_rf <- kernelshap(mod1, X, bg_X = bg_X)
shap_rf
sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "t_m")
#//////////////////////////////////////
#SUPPORT VECTOR MACHINE for trip duration
library(caret)
```

```
library(kernelshap)
library(shapviz)
set.seed(1234)
svm_mod<-train(
  form=duration_minutes~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
  data=training,
  method="svmRadial",
  trControl=ctrl,
  tuneLength=10,
  metric="MAE"
)

s <- kernelshap(svm_mod, training[, -1], predict, bg_X = training)
sv <- shapviz(s)
sv_waterfall(sv, 1)
shap_rf <- kernelshap(svm_mod, X, bg_X = bg_X)
shap_rf
sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "CO")
#////////////////////////////////////////////////////////
#random forest for trip duration
library(caret)
library(kernelshap)
library(shapviz)

set.seed(123)
rf_mod<- train(
  duration_minutes ~ t_m+ pre+ h_mi+ h_ma+s_r+w_s+m_g+CO+NO2+O3+PM2.5,
  data = training,
  method = "rf",
  tuneLength = 10,
  trControl = ctrl,
  metric = "MAE"
)

s <- kernelshap(rf_mod, training[, -1], predict, bg_X = training)
sv <- shapviz(s)
sv_waterfall(sv, 1)
shap_rf <- kernelshap(rf_mod, X, bg_X = bg_X)
shap_rf
sv_rf <- shapviz(shap_rf)
sv_importance(sv_rf, kind = "bee", show_numbers = TRUE)
sv_dependence(sv_rf, "t_m")
```