***ASSIGNMENT 3***

***Ottawa Bike Theft Analysis: Locating Hotspots and Trends***

**PROJECT BY: SAKSHIT SHARMA (041098605)**

**SRISHTI SRISHTI (041097754)**

**GROUP NO. 27**

**PROJECT TO: DR.ANU THOMAS**

Contents

[INTRODUCTION 3](#_Toc162648555)

[BUSINESS UNDERSTANDING 4](#_Toc162648556)

[DATA UNDERSTANDING 5](#_Toc162648557)

[Initial Data 5](#_Toc162648558)

[Describe Data 7](#_Toc162648559)

[EXPLORE DATA 8](#_Toc162648560)

[VERIFY DATA QUALITY 10](#_Toc162648561)

[DATA PREPARATION 11](#_Toc162648562)

[SELECT DATA: 11](#_Toc162648563)

[Clean Data: 12](#_Toc162648564)

[CONSTRUCT DATA: 13](#_Toc162648565)

[FORMAT DATA: 14](#_Toc162648566)

[MODELLING: 15](#_Toc162648567)

[OUTLIER DETECTION: 15](#_Toc162648568)

[CLASSIFICATION: 17](#_Toc162648569)

[DESCRIPTION OF RESULTS: 21](#_Toc162648570)

[ASSOCIATION 21](#_Toc162648571)

[PERFORMANCE VECTOR IN kNN. 22](#_Toc162648572)

[PERFORMANCE VECTOR IN DECISION TREES 24](#_Toc162648573)

[CROSS VALIDATION(Knn) 25](#_Toc162648574)

[Cross Validation(Decision Trees) 26](#_Toc162648575)

[Cross Validation for Random Forest 27](#_Toc162648576)

[LINEAR REGRESSION(R SQUARED) 28](#_Toc162648577)

[EVALUATION: 28](#_Toc162648578)

[Evaluation of Cross-Validation 28](#_Toc162648579)

[Evaluation of Association 31](#_Toc162648580)

[Local Deployment 32](#_Toc162648581)

[CONCLUSION 33](#_Toc162648582)

[References 34](#_Toc162648583)

# INTRODUCTION

In the busy city of Ottawa, a concern looms over the security of bicycles, a favored mode of transportation for many residents and visitors alike. With an aim to address this pressing issue, this data science project embarks on a journey to delve deep into the patterns of bike thefts across the city. Our objective is clear: to decipher the intricacies of bike theft occurrences, pinpointing the hotspots of activity and unraveling any discernible patterns in the types of bikes targeted. Through this analysis, we endeavor to offer actionable insights to various stakeholders, including law enforcement agencies, urban planners, and bicycle owners, with the ultimate goal of fortifying bike theft prevention measures and bolstering the overall security of bikes in Ottawa.

At the core of our project lies a fundamental question: Which locations in Ottawa bear the brunt of bike thefts most frequently, and what patterns emerge regarding the types of bikes stolen in these areas? To tackle this question, we employed the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, a robust framework encompassing six key phases. For this endeavor, our focus is directed towards the initial three phases: Business Understanding, Data Understanding, and Data Preparation. Leveraging tools such as **RapidMiner, Microsoft Excel, and datasets sourced from Ottawa's official website**, we embarked on a journey of meticulous analysis and strategic planning. Our aim is to unravel the intricate dynamics surrounding bike thefts in Ottawa, enabling stakeholders to make informed decisions and implement proactive measures to safeguard this vital mode of transportation.

***(Done by:SAKSHIT SHARMA)***

# BUSINESS UNDERSTANDING

The objective of the project is to analyze bike theft patterns in Ottawa to identify the locations with the highest frequency of bike thefts and understand any patterns regarding the types of bikes stolen in these areas. By gaining insights into these patterns, we aim to provide actionable information to stakeholders, such as law enforcement agencies, urban planners, and bicycle owners, to improve bike theft prevention measures and enhance overall bike security in Ottawa.

**Project Question:**

The central question driving this data science project is: "Which locations in Ottawa experience the highest frequency of bike thefts, and are there any patterns in terms of the types of bikes stolen in these areas?"

**Business Goals: (personal)**

Identify High-Risk Areas: Determine the geographical locations in Ottawa that experience the highest frequency of bike thefts.

Understand Bike Theft Patterns: Analyze the types of bikes stolen, what speed they were in, when they were reported, which year, what color in these high-risk areas and identify any recurring patterns or trends.

**Project Plan:**

To achieve the project objectives and answer the central question, we will follow the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, which consists of six phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. For this assignment, we will focus on the first three phases: Business Understanding, Data Understanding, and Data Preparation.

***(Done By: SRISHTI)***

# DATA UNDERSTANDING

Initial Data***:***We start off bydownloading the dataset from the Ottawa’s official website where we got the [Bike Thefts](https://open.ottawa.ca/datasets/ottawa::bike-thefts/about) dataset. There were several formats to download it and we chose the most convenient one that is the csv file where we could easily open the dataset in Microsoft Excel. The website gives information about the attributes used, etc. For loading the Bike thefts dataset, we open rapidminer, then a blank process and then we use the Read CSV operator, in which we load the CSV file which contains the dataset and then we store it in the Repository in Rapid Miner under the processes section. Once we have the dataset saved in the Repository section we can drag it to a blank database section as the Retrieve Final\_Assignment and this is the way we collect the initial data.

A computer screen shot of a computer

Description automatically generated A screenshot of a calendar

Description automatically generated

***(Done by:SAKSHIT SHARMA)***

Describe Data***:***

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Description** | **Data Type** |
| X | X coordinate of the bike theft location | Polynominal |
| Y | Y coordinate of the bike theft location | Polynominal |
| FID | Feature ID | Numeric |
| ID | Unique identifier | Numeric |
| Year | Year of the reported incident | Numeric |
| Reported\_D | Date when the bike theft incident was reported | Date/Time |
| Reported\_T | Time when the bike theft incident was reported | Date/Time |
| Reported\_W | Day of the week when the theft incident was reported | Nominal |
| Occurrence | Date of the occurrence of the theft | Date/Time |
| Occurren\_1 | Time of the occurrence of the theft | Date/Time |
| Occurren\_2 | Day of the week of the occurrence of theft | Nominal |
| Location\_T | Type of location where the theft occurred | Nominal |
| Primary\_Of | Primary offense category associated with the theft | Nominal |
| Bicycle\_St | Status of the stolen bicycle | Nominal |
| Bicycle\_Va | Value of the stolen bicycle | Nominal |
| Bicycle\_Ma | Brand of the stolen bicycle | Nominal |
| Bicycle\_Mo | Model of the stolen bicycle | Nominal |
| Bicycle\_Ty | Type of the stolen bicycle | Nominal |
| Bicycle\_Fr | Sex of the owner of the stolen bicycle | Nominal |
| Bicycle\_Co | Color of the stolen bicycle | Nominal |
| Bicycle\_Sp | Speed of the stolen bicycle | Numeric |
| Ottawa\_Nei | Ottawa neighborhood name | Nominal |
| Ottawa\_N\_1 | Ottawa neighborhood sector | Nominal |
| Ottawa\_N\_2 | Ottawa neighborhood region | Nominal |
| Census\_Tra | Census tract | Nominal |
| Census\_T\_1 | Census tract sector | Nominal |
| X\_Coordina | X-coordinate | Numeric |
| Y\_Coordina | Y-coordinate | Numeric |

*X,Y coordinate have been repeated for consistency.*

***(Done By:SRISHTI)***

## EXPLORE DATA

A screenshot of a graph

Description automatically generated

This histogram plot visualizes the frequency distribution of different types of bicycles .The X-axis represents the speed of the bicycle. Each point on the X-axis corresponds to a specific speed with a gap of 20 .The Y-axis represents the frequency or count of bicycle occurrences.The height of each bar indicates how many times a specific type of bicycle appears in the dataset.

A screenshot of a graph

Description automatically generated

The X-axis represents the years from 2014 to 2024.Each point on the X-axis corresponds to a specific year.The Y-axis represents the probability of a specific type of bicycle occurrence.The probability values range from 0 to 0.5.The colorful curve represents a bell-shaped distribution.It shows how the probability of different bicycle types changes over time.

*A screenshot of a computer

Description automatically generated*

The scatter plot represents data points categorized by the “Location\_T” attribute on the X-axis and Bicycle\_Sp(Y-axis).Different colors represent different categories or groups of data points.The X-axis represents the “Location\_T” attribute, with categories such as “West,” “East,” and “Central.”The Y-axis represents the “Bicycle\_Sp” attribute, ranging from 0 to 120.

***(Done by:SAKSHIT SHARMA)***

## VERIFY DATA QUALITY

For verifying the data quality,we matched the data type of each attribute to their respective types.We converted the X and Y coordinates to polynomial which was originally real.Then, we checked for missing values in the dataset and used the replace missing value operator.Also, we used the remove duplicates operator to identify and remove the duplicates. There were total 22 missing values in the Bicycle\_Ma attribute. A screenshot of a computer

Description automatically generated

***(Done by:SAKSHIT SHARMA)***

# DATA PREPARATION

## SELECT DATA:

We select the data, by using the operator Select Attribute and drag it onto the process window,and select the relevant attributes. According to us, these certain attributes are the most useful in determing the type of bicycle stolen, and the location, so I included these attributes for better results.

***A screenshot of a computer

Description automatically generated***

***(Done By:SRISHTI)***

## Clean Data:

Now to handle the missing values, we have placed an operator Replace missing values in which the values with ? have been replaced by a value which is the maximum among all other values.

***A screenshot of a computer

Description automatically generated***

***(Done By:SRISHTI)***

## CONSTRUCT DATA:

For construct data, we used the Generate Attribute operator which will generate two new attributes, called Bicycle\_Type\_Count and Day\_Type with the expression statement as if([Location\_Type] == "SINGLE HOME, HOUSE", if([Bicycle\_Type] == "REGULAR", 1, 0),

if([Location\_Type] == "PARKING LOTS", if([Bicycle\_Type] == "MOUNTAIN", 1, 0),

if([Location\_Type] == "OTHER COMMERCIAL / CORPORATE PLACES", if([Bicycle\_Type] == "HYBRID", 1, 0),

if([Location\_Type] == "UNIVERSITIES / COLLEGES", if([Bicycle\_Type] == "RACER", 1, 0),

if([Location\_Type] == "OTHER PUBLIC TRANSPORTATION AND CONNECTED FACILITIES", if([Bicycle\_Type] == "TOURING", 1, 0),

if([Location\_Type] == "SCHOOLS DURING SUPERVISED ACTIVITY", if([Bicycle\_Type] == "OTHER", 1, 0),

if([Location\_Type] == "OPEN AREAS", if([Bicycle\_Type] == "OTHER", 1, 0),

if([Location\_Type] == "PHARMACY", if([Bicycle\_Type] == "RACER", 1, 0),

if([Location\_Type] == "PRIVATE PROPERTY STRUCTURE", if([Bicycle\_Type] == "MOUNTAIN", 1, 0),

if([Location\_Type] == "DWELLING UNIT", if([Bicycle\_Type] == "HYBRID", 1, 0),

if([Location\_Type] == "CONVENIENCE STORES", if([Bicycle\_Type] == "REGULAR", 1, 0), 0))))))))))) and

if([Reported\_Day] == "Monday" || [Reported\_Day] == "Tuesday"|| [Reported\_Day] == "Wednesday"|| [Reported\_Day] == "Thursday"|| [Reported\_Day] == "Friday", "weekday", "weekend") respectively..

A screenshot of a computer

Description automatically generated

***(Done by:SAKSHIT SHARMA)***

## FORMAT DATA:

We used an operator called "nominal to numerical" operator,in which the format of the data has been transformed from a categorical or nominal format to a numerical format. This means that the previously categorical variables, have been encoded into numerical values.

We renamed the attributes so they are better understandable according to our question based on bike thefts and is focused on organizing and clarifying the structure of your data, rather than integrating new data sources. Also changed the data types of some of the attributes like the X and Y coordinates from real to polynominal.

A screenshot of a computer

Description automatically generated

***(Done By:SRISHTI)***

# MODELLING:

OUTLIER DETECTION: We applied 2 techniques to detect outliers in our dataset. One of them was the Local Outlier Factor and another one, Isolation Forest.

**Local Outlier Factor**: LOF is a density-based algorithm that measures the local deviation of a data point with respect to its neighbors. Points with significantly lower LOF scores are considered outliers, indicating they are less dense than their neighbors.

For using this technique we dragged the detect outlier(LOF) on the dataset and connected it to the Join operator which was used to combine the results of LOF and ISF using the Id attribute.

**Isolation Forest (ISF):** ISF is an algorithm that isolates outliers by recursively partitioning the dataset into subsets. Outliers are identified as instances that require fewer partitions to be isolated. This algorithm is particularly effective for high-dimensional data.

For applying Isolation Forest, we first drag the Select Attributes operator and exclude the Occurrence and reported Date, then we replace all the missing values in the dataset and then apply the ISF operator which will help us detect the outliers. Then we use the outer join and set the right and left key attributes. In the end we apply a set role operator to set the role of the outlier and prediction and connect it to the result.

A diagram of a computer system

Description automatically generated

**A screenshot of a computer

Description automatically generated**

## CLASSIFICATION:

For classification we used three algorithms, k-NN, Decision Tree and Random Forest. These models were trained to predict the likelihood of future bicycle theft incidents based on various features.

**k-Nearest Neighbors (kNN) Modeling**

1.Generate Test Design:

**Data Preparation:**

First we split the training and the testing data then, apply the algorithm , apply model and check the performance. 70% of the data is the training data which is used to train the data set and the rest 30% is the testing data on which we test the dataset.

**Split dataset:** Divide the dataset into training and testing sets using a suitable split ratio (e.g., 70% training, 30% testing).

2. Build Model: kNN Algorithm:

We apply the kNN algorithm to the training data using the "k-Nearest Neighbors" operator.For kNN we used different values of k to use the algorithm, and what we figured out was that at k=25, the dataset showed the maximum accuracy of 26.00%.Hence we kept k as 25, then we used the apply model, The testing data is fed into the "apply model" step through another port, indicating that the trained kNN model is also being applied to the testing data.

3. Test Model: Prediction:

We use the trained kNN model to make predictions on the test data.

4. Evaluation:

Then we applied the Performance operator on the trained data to check its accuracy.

**Operators Used:**

"Split Data": to divide the dataset into training and testing sets.

"k-Nearest Neighbors": to build the kNN model.

"Apply Model": to apply the trained model to test data.

"Performance": to evaluate the model's performance.

A diagram of a computer

Description automatically generated

**Decision Trees (DT) Modeling**

1. Generate Test Design: Data Preparation:

Select features: Firstly, we choose relevant features for building the decision tree model. For decision trees, after splitting the data in the ratio of 7:3, we use an operator decision tree, we train the decision tree model using the training dataset, where the model recursively partitions the feature space based on attribute values to create a tree-like structure.

Split dataset: Divide the dataset into training and testing sets.

2. Build Model: Decision Tree Algorithm:

Next, we apply the decision tree algorithm to the training data using the "Decision Tree" operator.

3. Test Model: Prediction:

We use the trained decision tree model to make predictions on the test data.

Evaluation:

We evaluate the model's performance using metrics like accuracy, precision, recall, and F1-score using the performance operator. We can then visualize the tree and check for the results.

Operators Used:

"Split Data": to divide the dataset into training and testing sets.

"Decision Tree": to build the decision tree model.

"Apply Model": to apply the trained model to test data.

"Performance": to evaluate the model's performance.

A diagram of a light bulb

Description automatically generated

A white background with text overlay

Description automatically generated

**Random Forest (RF) Modeling**

1. Generate Test Design: Data Preparation:

Split dataset: Firstly, we divide the dataset into training and testing sets.

2. Build Model: Nextly, we train the Random Forest model using the training dataset, where multiple decision trees are constructed and combined through voting to improve predictive performance.

Random Forest Algorithm:

We apply the random forest algorithm to the training data using the "Random Forest" operator .And set parameters such as the number of trees, maximum depth, and minimum samples per leaf.

3. Test Model: Prediction:

We also use the trained random forest model to make predictions on the test data.

Evaluation:

We can evaluate the results using the Performance operator. And visualize feature importance and ensemble predictions.

**Operators Used:**

"Split Data": to divide the dataset into training and testing sets.

"Random Forest": to build the random forest model.

"Apply Model": to apply the trained model to test data.

"Performance": to evaluate the model's performance.

A computer screen shot of a diagram

Description automatically generated

# DESCRIPTION OF RESULTS:

## ASSOCIATION

***A screenshot of a computer

Description automatically generated***

This confusion matrix generated by these association rules provides insights into the strength, reliability, and predictive power of the rules in predicting the occurrence of specific outcomes based on given conditions or premises. Each metric contributes to the understanding of the performance of the association rules in capturing patterns and relationships in the dataset.

## PERFORMANCE VECTOR IN kNN.

***A screenshot of a computer

Description automatically generated***

***A screenshot of a graph

Description automatically generated***

The kNN algorithm performed well in predicting instances belonging to the "SINGLE HOME, HOUSE" class, with a precision of 25.15% and a recall of 53.95%.

However, for other classes such as "PARKING LOTS" and "OTHER COMMERCIAL / CORPORATE PLACES," the precision and recall values are relatively low, indicating that the algorithm struggled to correctly classify instances in these categories.

The most important thing about this classification is that the precision and recall values vary across different classes, suggesting that the performance of the kNN algorithm may be influenced by the characteristics of each class and the distribution of data within the dataset.

## PERFORMANCE VECTOR IN DECISION TREES

A screenshot of a computer

Description automatically generated

The decision tree model achieved high precision (28.63%) for predicting the class "SINGLE HOME, HOUSE" but lower recall (98.68%), indicating that while it correctly predicted instances of this class, it missed some instances.

For other classes such as "PARKING LOTS," "OTHER COMMERCIAL / CORPORATE PLACES," and "DWELLING UNIT," the precision and recall values vary, suggesting varying degrees of performance for different classes.

## CROSS VALIDATION(Knn)

***A screenshot of a computer

Description automatically generated***

In this cross validation, KNN was used for classification, and the model's performance was evaluated using cross-validation. Each cell in the confusion matrix represents the count of instances where the actual class (row) matches the predicted class (column) for a specific fold of the cross-validation process. The precision and recall values for each class are calculated based on the aggregated counts across all folds.

## Cross Validation(Decision Trees)

A screenshot of a graph

Description automatically generatedEach cell represents the count of instances where the actual class (row) matches the predicted class (column) for a specific fold or iteration of the cross-validation process.The class recall values represent the percentage of instances of each class that were correctly classified by the decision tree model across all folds.

Cross Validation for Random Forest A screenshot of a chart

Description automatically generated

Each cell in the confusion matrix represents the count of instances where the actual class (row) matches the predicted class (column) for a specific fold or iteration of the cross-validation process.The class recall values represent the percentage of instances of each class that were correctly classified by the random forest model across all folds.

## LINEAR REGRESSION(R SQUARED)

***A screenshot of a computer program

Description automatically generated***

ThIS linear regression model,Regression, predicts the label Location\_Type based on features such as Bicycle\_State and Bicycle\_Type\_Count. The model's intercept represents the baseline prediction when all features are zero. By multiplying feature values by coefficients, summing them, and adding the intercept, the model generates predictions.

# EVALUATION:

## Evaluation of Cross-Validation

1. Cross-Validation Process:

Here, we implemented k-fold cross-validation to assess the performance of the models. And divided the dataset into 10 folds and trained the model 10 times, each time using a different fold as the test set and the remaining folds as the training set. Then, we averaged the performance metrics across all folds to obtain an overall estimation of the model's performance.

2. Operators Used:

"Cross-Validation": Employed the Cross-Validation operator to perform k-fold cross-validation(kNN/Decision Trees/Random Forest)

“Apply Model”: After splitting the dataset into k folds, each fold is treated as a test set while the remaining k-1 folds are used for training the model.

Within each fold, the model is trained using the training data and then applied to the test data using the "Apply Model" operator.

"Performance": Utilized to calculate and aggregate performance metrics across all folds.

4. Results:

The cross-validation results provided a robust estimation of the models' performance by evaluating them on multiple subsets of the data.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Cross Validation for kNN

A screenshot of a computer

Description automatically generated

Cross Validation for Decision Trees

A screenshot of a computer

Description automatically generated

Cross Validation for Random Forest

## Evaluation of Association

1. Association Analysis Process:

We have also applied the Association algorithm to discover frequent item sets and association rules among different attributes of bicycle theft incidents.

For this, we identified patterns of co-occurrence or association between categorical variables such as location type, primary offense, bicycle type, etc.

2. Operators Used:

“Select Attributes”: the attributes selected for association analysis using the "Select Attributes" operator.

“Nominal to Binominal”: We transform of nominal attributes to binomial format using the "Nominal to Binomial" operator.

“FP Growth”: Then, we the apply the FP-Growth algorithm for mining frequent itemsets from the dataset.

"Association Rule": Utilized to evaluate and visualize association rules and their metrics.

4. Results:

The association analysis revealed interesting patterns and correlations between different attributes of bicycle theft incidents.High-confidence association rules provided actionable insights into the relationships between location type, primary offense, bicycle type, and other categorical variables.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

# Local Deployment

For this project, deployment has been conducted locally on our machine. The deployment process involves transferring the trained models, scripts, and necessary dependencies from the development environment to the production environment on our local machine.

# CONCLUSION

In conclusion, this data science project focused on analyzing bike theft patterns in Ottawa with the aim of identifying high-risk areas and understanding patterns related to the types of bikes stolen. Through the application of the CRISP-DM methodology, we meticulously navigated through all the phases.

Our analysis revealed valuable insights into the dynamics of bike theft occurrences in Ottawa. We identified specific locations, such as "SINGLE HOME, HOUSE" and "PARKING LOTS," as high-risk areas for bike thefts. Additionally, we uncovered patterns regarding the types of bikes targeted in these areas, providing stakeholders with actionable information to enhance bike theft prevention measures.Furthermore, our modeling efforts using techniques like k-Nearest Neighbors, Decision Trees, Random Forest, and linear regression enabled us to predict future bike theft incidents with reasonable accuracy.

Ultimately, through this comprehensive analysis, we aim to contribute to the enhancement of bike security measures in Ottawa, ensuring a safer environment for cyclists and promoting sustainable transportation in the city.

**NOTE: BOTH CLASSIFICATION AND OUTLIER DETECTION WAS DONE IN COLLABORATION BY BOTH THE TEAM MATES (SRISHTI AND SAKSHIT)**

A screenshot of a computer

Description automatically generated

# References

*Cross-industry standard process for data mining*. (2024, 03 23). Retrieved from wikipedia: https://en.wikipedia.org/wiki/Cross-industry\_standard\_process\_for\_data\_mining

Ottawa, C. O. (2023, 06 20). *Bike Thefts*. Retrieved from Ottawa: https://open.ottawa.ca/datasets/ottawa::bike-thefts/about