**Homeless Ireland 2019 to 2021**

Student Name: Denise Martins

Student Number: sbs24065

Machine Learning

Lecturer: Dr. Muhammad Iqbal

https://github.com/Denisbs24065/MachineLearning

CCT College, Dublin

March, 2024

**Table of content:**

[Introduction](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250006) 3

Problem Definition and [Objectives](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250005) 3

[Scope](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250003) 3

Figure 1: The data mining life cycle 4

Business Understanding 4

Data Understanding 4

Data Preparation 4

Modeling 4

Evaluation 4

Deployment 5

[Potential data for the project](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250002) 5

Data sources 5

[Ethical considerations](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250001) 5

[References](file:///C:\Users\denis\Downloads\CA%201%20Example%202.docx#_TOC_250000) 6

### Introduction

### Scope

I will analysis this data, about occurrence of homeless people in the years between 2019 and 2021, including region of occurrence, total of adults, and with year for example, to analyse and insights that can drive some strategy and decrease the occurrence of homeless and understand the cases of this numbers.

I load a CSV file named Homeless\_Ireland\_2019\_to\_2021.csv and apply .head() function to understand the size of the dataset and with information I have for this analyse.

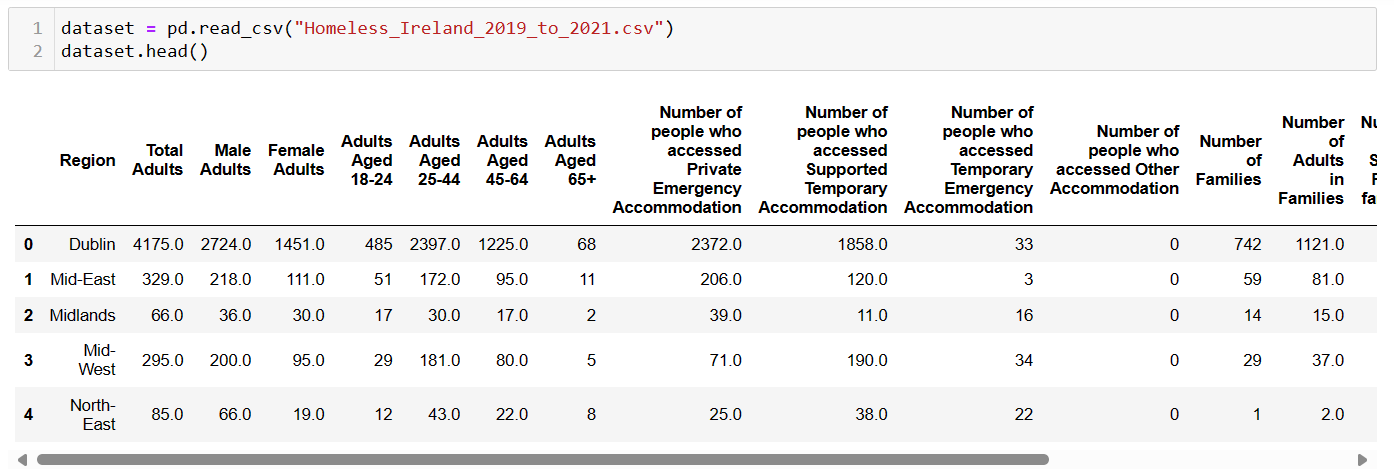


Figure 1

I use two search functions, searching for Non-Null values or missing data. The function isnull().sum() checks for missing values in each column of the dataset and sums up the count of missing values for each column. In answer I had one column that had Non-Null values, but I drop this column that was with same content as another column.

And the function .info() provides a concise summary of the DataFrame's information including the number of entries (288 rows) and the data type of each column. It also shows the total memory usage of the DataFrame.

Using the option replace(), I replace qualitative value for quantitative in two important columns that I will be using in my analysis.

According with Geeks for Geeks “Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units. If [feature scaling](https://www.geeksforgeeks.org/python-how-and-where-to-apply-feature-scaling/) is not done, then a [machine learning](https://www.geeksforgeeks.org/machine-learning/) algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values”.

In my analyse I applied MinMaxScaler, Normalization and Standardization to improve the results.

I split my dataset in test and training, X\_train, X\_test, y\_train, y\_test using cross validation techniques for the variation in the accuracy across using three trainings splits (20%, 25% and 30%).

I decide to apply Linear Regression because, I the goal of this analyse is understand the relationship between a dependent variable (Total of Adults) and one more independent variable (in my case year), and this relationship can be approximated by a linear equation.

In the Classification It is provided the class of the dataset based on the independent input variable, which although it is a good idea to analyse the occurrence in certain regions, it is this also important to know the total per year. Regression, on the other hand, predicts the continuous output variables based on the independent input variable, the total number of adults per year.

In Linear Regression I plot comparing the actual values (y\_test) with the predicted values (y\_pred) for different trainings as was mentioned before. Bellow all three results of the plot:

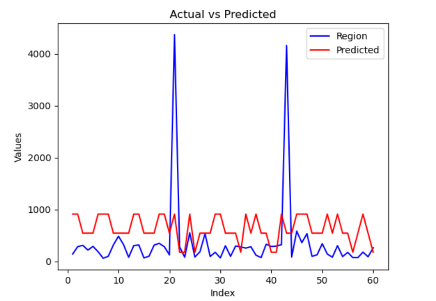
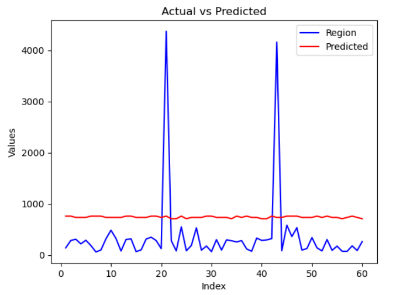
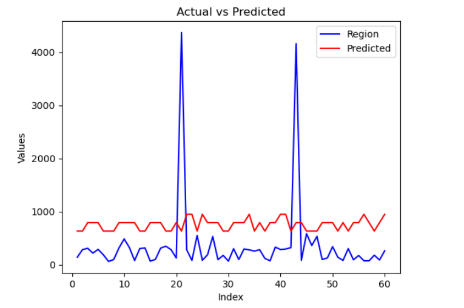
  

Figure 2 (20%) Figure 3 (25%) Figure 3 (30%)

Here in this figures we can check the blue line represents the actual values (y\_test) and the red line (y\_pred), represents the predicted values, both corresponds to a specific index, but the red line its associated predicted values. By comparing the blue and red lines on the chart, we can visually assess how well my predictive model is performing. Ideally, the red line (predicted values) should closely follow the blue line (actual values). However, as we can see the discrepancies between the two lines, they indicate areas where the model's predictions deviate from the actual values. This visualization helps to visually evaluate the performance of the predictive model, which despite the aforementioned discrepancy still presents a good visualization.

And for all three trainings I checked MSE (Mean Squared Error), that measure of the error between predicted and actual values, and R2 (R-squared), that provides a relative measure of how well the regression model fits the data. In my analyses, the MSE is between 634149 and 724461, this means that, on average, the squared difference between the predicted value and actual value (for each point) is between 634149 and 724461. In my R2 all the results is negative, between -0.16 and -0.33 that suggests the model does not explain any variance in the target variable beyond that of the mean of the target variable.

I also presented some graphs such as Histogram, showing of the Total Adults variable in this dataset, as well as the relationship between the Total Adults and the Year of occurrence, and adaptivity level column, between moderate: 0, low: 1 and high: 2.

As you can check in my analyse in python, at first, I had chosen the classification method, because I thought I would be able to explain those futures as my target. Implementing two observations in Y and the rest of my dataset as X, I checked the accuracy and the Standard Deviation was really low, and I could not explain why. However, as you can see bellow, this dataset just classifies for class 0, and didn't work for others classes. Even so, I thought it was important to finish the sampling.

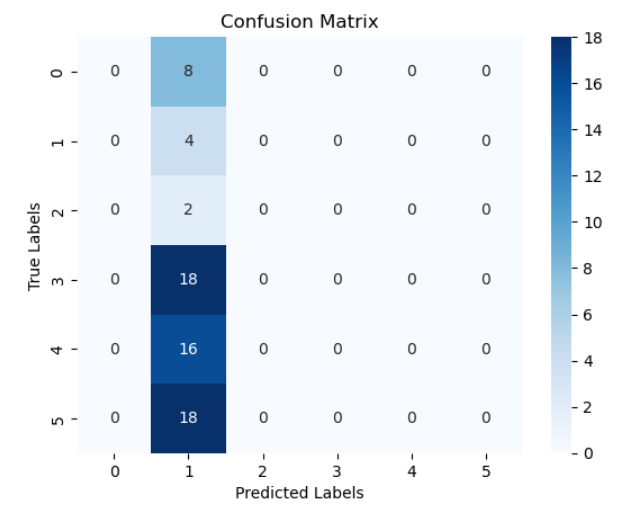


Figure 4

In Confusion Matrix Geeks for Geeks explain “confusion matrix is a matrix that summarizes the performance of a machine learning model on a set of test data. It is a means of displaying the number of accurate and inaccurate instances based on the model’s predictions. It is often used to measure the performance of classification models, which aim to predict a categorical label for each input instance”.

After understand that I changed to Linear Regression.

To optimal parameters, I applied decision tree, that according with IBM “A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes.” The accuracy achieved as a response is 0.16, which means that the model correctly classified only 16% of the examples. This suggests that the model is not performing well on the classification task.

The mean Accuracy is 0.14 and the Standard Deviation is 0.061, low numbers for this model.

Another technique used in this model was the k-means, that shows in output clusters labels (numbers indicating which cluster each data point belongs to), and cluster centroids (average feature values for each cluster, representing the centre of the cluster).

Support Vector Classifier SVC mode, “is a widely used supervised learning method and we can use it in classification problems. It uses the C regularization parameter to optimize the margin in hyperplane and it is also called C-SVC”, said the website DataTechNotes. Bellow the results of the SVC classifier, with precision 0.2 indicating only 20% of the sample predicted as class 3 were actually class 3. Recall is 0.45, indicating that the model correctly identified 45% of the samples that were actually class 3. F1-score it range from 0 to 1, with higher values indicating better performance. Support is the number of occurrences of each class in the test dataset. Accuracy, in this case is 0.19, indicating that the model correctly classified around 19% of the samples.

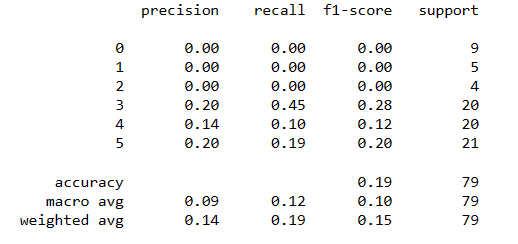


Figure 5

And the last one was grid search, evaluate the model`s performance on the test data, and visualizes the results using a heatmap. I could print the accuracy for the grid and the result is 0.24, that’s mean approximately 24% of the samples in the test dataset is correctly predicted the class label.

### Conclusion

### References