ENEL 645

Data Mining & Machine Learning

Assignment 1

Author:

Steven Duong
(30022492)

Affiliation
Department of Electrical and Software Engineering
University of Calgary
Calgary, Alberta

Lab Block: B01
Date of Report: May 18, 2023

Introduction

This assignment involves implementing linear regression for a real-world machine learning problem. The dataset provided was sourced from Open Calgary and contains Community Crime and Disorder Statistics. The objective was to develop a model that can predict the number of crimes in each community center based on various input features. The model was trained using 70% of the data, and the remaining 30% was used for evaluation. The performance of the model was evaluated using the mean-squared-error (MSE) cost function.

Data Preprocessing

1. **Data Loading:** First, the dataset was loaded using pandas library. The 'Community Name' field was considered as the index column of the dataset.

2. **Data Preprocessing:** The next step was data preprocessing. Here, the categorical variables were converted into dummy/indicator variables using the get_dummies() function from pandas library. This step is crucial as it converts categorical data into a format that could be provided to ML algorithms to improve prediction results.

3. **Data Splitting:** The dataset was then divided into input features (X) and the target variable (y). The input features included various crime-related data, and the target variable was 'Crime Count.' After defining these, the dataset was split into a training set (70% of the data) and a testing set (30% of the data).

Algorithm Description

1. **Model Training:** The Linear Regression model was trained using the fit method of the LinearRegression class from the sklearn library. The fit method takes two arguments: X_train and y_train. X_train is a 2D array-like structure of shape (n_samples, n_features), where n_samples is the number of samples (in this case, 70% of the total samples), and n_features is the number of features. y_train is a 1D array-like structure of shape (n_samples), which contains the target values, i.e., the number of crimes (Crime Count). The fit method adjusts the internal parameters of the model in order to minimize the cost function (in this case, the mean-squared-error function). This is done using the Ordinary Least Squares (OLS) method, which aims to minimize the sum of the squares of the differences between the observed (actual) and predicted values.

2. **Model Prediction:** Once the model has been trained, it can be used to make predictions on unseen data. In this case, the model was used to predict the number of crimes for the testing data (which is 30% of the total samples). The predict method of the LinearRegression class was used for this purpose.

3. **Model Evaluation:** The final step was to evaluate the model. This was done using the mean-squared-error (MSE) cost function from sklearn's metrics. MSE was computed between the actual and predicted values of the testing set.

Results

The model showed a Mean Squared Error (MSE) of 525.355, indicating the average squared difference between the estimated values and the actual value.

```
[ ]  predictions = model.predict(X_test)
     mse = mean_squared_error(y_test, predictions)
     print('MSE:', mse)

     MSE: 525.3552882121902
```

Figure 1: Evaluation of Model Performance through Mean Squared Error (MSE)

This figure depicts the calculated Mean Squared Error (MSE) for our model's predictions. MSE is a pivotal metric in evaluating the performance of our regression model, offering a quantifiable measure of the model's accuracy by assessing the average squared difference between actual and predicted values. The lower the MSE, the more precise the model's predictions, implying a superior model performance. The reported MSE in Figure 1 thus serves as a testament to our model's reliability in predicting crime counts in Calgary communities.