PREDICTIVE MODELLING OF CRIME DATASET USING DATA MINING

Prajakta Yerpude[1] and Vaishnavi Gudur[2]

Department of Computer Science, Illinois Institute of Technology, Chicago, Illinois

ABSTRACT

With a substantial increase in crime across the globe, there is a need for analysing the crime data to lower the crime rate. This helps the police and citizens to take necessary actions and solve the crimes faster. In this paper, data mining techniques are applied to crime data for predicting features that affect the high crime rate. Supervised learning uses data sets to train, test and get desired results on them whereas Unsupervised learning divides an inconsistent, unstructured data into classes or clusters. Decision trees, Naïve Bayes and Regression are some of the supervised learning methods in data mining and machine learning on previously collected data and thus used for predicting the features responsible for causing crime in a region or locality. Based on the rankings of the features, the Crimes Record Bureau and Police Department can take necessary actions to decrease the probability of occurrence of the crime.

KEYWORDS

Supervised Learning, Unsupervised Learning, Decision Tree, Naïve Bayes, Regression, Data Mining, Machine Learning

1. INTRODUCTION

Criminal activities across the globe have created a menace in the society. Every year large volume of criminal data is generated by the law enforcement organizations and it is a major challenge for them to analyse this data to implement decision for avoiding crimes in future. The crime data has many features including information about immigrants, race, sex, population, demographics and so on. Analysing this data not only helps in recognizing a feature responsible for high crime rate but also helps in taking necessary actions for prevention of crimes. Data mining provides powerful techniques and algorithms to analyse data and extract important information from it.

The knowledge discovery in data mining is gaining useful information or extracting patterns that contribute to an important prediction from large amounts of data. McCue defines [1] knowledge discovery as extraction of operationally actionable output from crime data for solving crimes or explaining criminality. She mentions that crime investigation by police uses case based reasoning techniques and relates it to data mining. Therefore, the data source from police can be used as a source for crime data mining. Predictive analysis according to Nyce [2] is a statistical technique used to develop models that predict future events. These predictive models usually are measured using scores on which the features are extracted are taken as accuracy, precision, recall and F1 score. For this purpose, there is a need to train the data according to a specific data mining algorithm and then test on a new dataset to know how well our model fits the new dataset by measuring in terms of accuracy and F1 scores.

Data mining generally involve two categories with respect to the data to be mined that includes Description [3] mining and Classification with Prediction [4]. Description mining usually is mining of association rules, frequent patterns, clusters, or correlations.

DOI: 10.5121/ijdkp.2017.7404

Classification and Prediction involves classifying a class label for data using probability equations and predicting any feature using numeric measures accordingly.

In this paper, for performing predictive analysis, the Communities and Crime dataset from UCI repository [5] has been used which consists of crime data in Chicago, a city having highest crime rate in the United States of America. It includes features affecting crime rate like population, race, sex, immigrants etc. Many features involving the community like, such as the percent of the population considered urban, and the median family income, and involving law enforcement, such as per capita number of police officers, and percent of officers assigned to drug units are included so that algorithms that select or learn weights for attributes could be tested [5]. The attribute or feature to be predicted is 'Per Capita Violent Crimes' which was pre-calculated in the data using population and the sum of crime variables considered violent crimes in the United States: murder, rape, robbery, and assault.

Here, 80% of data will be trained according to the given algorithms and tested on the rest of 20% data. The aim is to predict top most features with the accurate predictive model that affects the high crime rate which will eventually help police or law enforcement makers take necessary actions. For this purpose, a precise implementation of following data mining algorithmic predictive models has been made:

1. Decision Trees and Random Forest Classification
2. Naïve Bayes Classification
3. Linear Regression

Accuracy [6] measures the performance of each model that gives percentage of features that are predicted correctly among total number of features, Precision [6] which is defined as number of positive features classified by the model that are positive and Recall [6] that gives number of positive features classified correctly by the model. Also, F1 Score [6] is a harmonic mean of precision and recall for balancing out both has been taken as a measure of performance.

A systematic approach using Software Development Lifecycle [7] was followed that included phases of modules that were planned, designed, coded, tested and integrated effectively. Planning included requirements gathering, technological study, survey of data and deciding upon flow of working. Coding consisted of implementation of predictive models by training and testing of data. The paper introduces the topic followed by the literature search and related work done in predictive analysis and its applications in crime data. It also includes various technologies and methods used for getting results. A detailed description about the libraries used for this prediction, classification and, the performance metrics decided upon, for obtaining the most accurate results has been discussed in modules. Pre-processing of data describes the data in its available format and the techniques used to process it. The data in the given dataset contained many missing, null and erroneous values. Hence, there was a need for the data to be cleaned, transformed, and integrated to reduce noise. To increase the accuracy of prediction, three data mining techniques for developing prediction models were employed, the results of which can then be compared to analyse which model best fits this type of data and to obtain the most accurate prediction. The detailed description of every classifier with its analysis and implementation results shall be seen in this paper. In conclusion, a comparative analysis of these algorithms in terms of their scores with top features affecting crime rate will be discussed.

Decision Trees [8] includes a root node, branches, and leaf nodes with each internal node denoting a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. The Random Forest [21] considers multiple individual trees. Each tree individually classifies the dataset and the algorithm then chooses the classification commonly

chosen by most number of individual trees. Thus, Random Forest Classifier algorithm formed a key factor in getting an accurately predictable set of features with the help of decision trees.
Naïve Bayes Classification [22] is a supervised learning classifier that returns a set of classes, instead of a single output. The classification is thus given by the probability that an object belongs to a class. This approach is mainly used for its ease in implementation and precise results.
Linear Regression [25] allows to study relationships between two continuous (quantitative) variables by fitting a straight line between target feature and remaining features. The best line that fits the data can be found using a straight-line function. Performance of a linear predictive regression model is measured by mean squared error that is average of the squares of the errors or deviations - that is, the difference between the estimator(features) and what is estimated (Target variable).

2. RELATED WORK

Data Mining [8] is the process of analysis to find trends, patterns, and knowledges. For this purpose, two factors are important viz. data used for analysis requiring accuracy and sufficiency and knowledge of expertise. This knowledge in the form of result obtained is used to assist in decision making and solving problems. Growing amount of criminal data gives rise to numerous problems like data storage, warehousing, and analysis [9]. The major areas used in data mining are Association Rule mining, Classification, Prediction, and Clustering. Association mining [9] is used to discover relationship between entities and finding frequent occurring item sets in dataset. The relationship can be one to one, one to many or many to many. This is usually referred in Market Basket analysis that shows customers buying products frequently. Classification and Prediction [9] usually use divide and conquer method of grouping the data and predicting the missing or not defined values in a dataset. Clustering [9] groups similar data into chunks called clusters. In this case, it can be clusters of groups having age greater than 30, less than 30 and so on.

Several techniques have been proposed in the recent years for solving a problem of extracting knowledge from explosive data adopting different algorithms. One of such applications is that of finding knowledge of criminal behaviour from its historical data by studying the frequency of occurring incidents [9]. P.Thongtae [9] studied and gave a comprehensive survey of effective methods on data mining for crime data analysis. One of such proposed information system was that of 'Regional Crime Analysis Program' that is used to turn data into knowledge using data fusion. Data fusion manages, fuses, and interprets information from different sources and overcomes confusion from cluttered backgrounds.

Ozgul [10] studied the sources of crime data mining by pointing out which forms of knowledge discovery if suitable for which methodology. He studied CRISP-DM, Cross-Industry Standard Process for Data Mining like SEMMA (Sample, Explore, Modify, Model, Assess). This method suited best for prediction and clustering tasks. He also studied the Van der Hulst's Methodology that involves criminal network analysis. This included study of network boundaries, defining actors, its attributes, activities, and affiliations that ensure data reliability and change over time. He concluded that selecting the appropriate methodology depends on tasks required or high volume of crime data to be prepared.

Chen Hsinchun [11] proposed a general framework for crime data that shows relationships between data mining techniques applied in criminal and intelligence analysis and the crime types at local, national, and international levels. They used entity extraction, association, prediction, and pattern visualization to categorize each crime type. For example, investigators can use neural networks in crime entity extraction, clustering in association and visualization and so on. Currently they are associated in creating a cybercrime database with the help of this framework.

This research focuses on prediction using:

1. Decision Trees [12] that uses root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. Random Forest takes multiple trees into account and produces the mean result which is useful in balancing the biased data.
2. Naïve Bayes classification [6]that uses Bayes rule to predict class membership probabilities such as the probability that a given tuple belongs to a class. It uses the concept of conditional probability.
3. Linear Regression [6] is a predictive modelling technique where the target variable to be estimated is continuous.

The technology and tools used for Data mining are usually Python, R, Weka, Orange etc. A predictive analytical model has been coded and designed in Python using Scikit [13] Learn Modules. Scikit Learn is an open source and an efficient coding tool for data mining that is built on numpy, Scipy and matplotlib modules of python. The overview and implementation part of these algorithms shall be seen in the next chapter.

3. COMPONENTS USED IN PREDICTIVE MODELLING

3.1 DATA PREPARATION

1. **Pandas:** It is an open source library [14] that provides high-performance, easy-to-use data structures and data analysis tools for the Python programming language. It stores the tabular, matrix data into rows and columns using data frames in Python which helps to process the data dynamically. The CSV file can be loaded and converted into a data frame in Python using a pandas object usually called as 'pd'. The general syntax for this process is:

```
import pandas as pd
df = pd.read_csv('Crime.txt')
```

2. **Numpy:** It is a powerful package [15] for scientific computing in Python. It can be used by creating an N-dimensional object array usually represented by np. The general syntax for creating a numpy object is:

```
import numpy as np
```

3.2 CLASSIFICATION AND REGRESSION

1. **Sklearn – Decision Tree Classifier:** As described in the related work, decision tree model is a supervised learning method of classification. The goal of this classifier [16] is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.A deeper tree indicates complex decision rules and that makes a better fitted model. If y is an attribute to be predicted and X are the attributes used for prediction, then the classifier syntax can be written as:

```
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier(criterion = 'entropy')
#Object of DecisionTreeClassifier with Entropy as the criterion
dt.fit(X, y)
```

   Entropy gives the information gain from a decision rule. Hence for this predictive model, Entropy as a criterion for splitting of branches has been used.

2. **Sklearn – Random Forest Classifier:** This classifier [17] fits n number of decision tree classifiers on various sub-samples of the dataset, controls the over-fitting of data and improves predictive accuracy by averaging. This classifier has been used to gain a good accuracy over singular decision trees obtained in 3. If y is an attribute to be predicted and X are the attributes used for prediction, then the classifier syntax can be written as:

```
from sklearn.tree import RandomForestClassifier
dt = RandomForestClassifier()     #Object of classifier
dt.fit(X, y)
```

3. **Sklearn – GaussianNb:** GaussianN [18] module make use of Naïve Bayes theorem that are a set of supervised learning algorithms with the "naive" assumption of independence between every pair of features. If y is an attribute to be predicted and X are the attributes used for prediction, then the classifier syntax can be written as:

```
from sklearn.naive_bayes import GaussianNB
dt = GaussianNB()      #Object of classifier
dt.fit(X, y)
```

4. **Sklearn – Regression:** Linear Regression [19] in Scikit allows to study relationships between two continuous (quantitative) variables:

   - One variable, denoted x, is regarded as the predictor - Features as variables.
   - The other variable, denoted y, regarded as the response - HighCrime variable.

   A linear regression line has an equation of the form y = a + bX, where X is the explanatory variable and y is the dependent variable. Therefore, then the classifier syntax can be written as:

```
from sklearn.linear_model import LinearRegression
dt = LinearRegression()      #Object of classifier
dt.fit(X, y)
```

3.3 PERFORMANCE METRICS

1. **Cross Validation Score:** In Cross validation [6], each record is used the same number of times for training and exactly once for testing. For example, in a 2-fold cross validation method, choose one of the subsets for training and the other for testing. Then swap the roles of the subsets so that the previous training set becomes the test set and vice versa. For analysis, 10-fold cross validation method has been used thereby ruling out the possibility of over fitting the data. Hence, CV Scores of accuracy, precision, recall and F1 Score have been considered for elevating the performance of our model.

2. **Accuracy, Precision, Recall and F1 Score:** Performance of a model is measured by reflection of well observed actual events. While training any model, a labelled data set that includes the actual values to be predicted is considered. This introduced the concepts of a confusion matrix [6]. It gives a relation between an actual and predicted class.

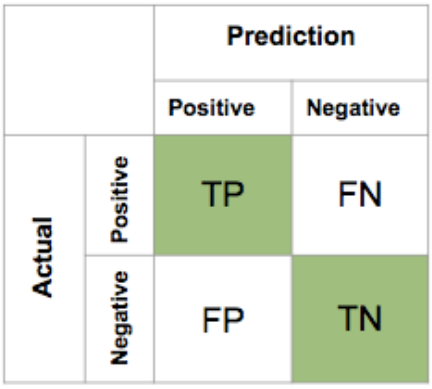

Figure 1 Confusion Matrix

True positives [6] and true negatives [6] are the observations which were correctly predicted, and therefore shown in green. The terms "false positive" and "false negative" can be confusing. False negatives are observations where the actual event was positive. The way to think about it is that the terms refer to the observations and not the actual events. So, if the term starts with "false", the actual value is the opposite of the word that follows it.

Accuracy is simply the ratio of correctly predicted observations and is given by:

$$\text{Accuracy} = \frac{tp+tn}{tp+tn+fp+fn}$$

Precision and Recall is given by:

$$\text{Precision} = \frac{tp}{tp+fp}$$
$$\text{Recall} = \frac{tp}{tp+fn}$$

F1 Score is a harmonic mean of Precision and Recall.

3. **Mean Squared Error:** MSE [25] is the average of the squared errors between actual and estimated readings in a data sample. Negative numbers are taken away by squaring the difference. It also gives bigger differences more weight than smaller differences in the result. Mean square error is widely used in signal processing applications, such as assessing signal quality, comparing competing signal processing methods, and optimizing signal processing algorithms. It is given by:

$$Q = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

where Q is the MSE and the entity $(y_i - \hat{y}_i)^2$ the squared prediction error for data point i.

4. DATA PREPARATION

4.1 DATA PRE-PROCESSING

Data pre-processing [20] is a data mining technique that involves transforming raw data into an understandable format. Often the data is unstructured, inconsistent, has missing values, and lack in certain behaviour or trends that gives many errors. Therefore, it needs to be cleaned, integrated, transformed, and hence reduced. Cleaning fills in the missing values and removes noise.

Integration take the data cubes or chunks together using multiple databases. Transformation uses normalization and aggregates the data and Reduction helps in decreasing the volume of data keeping similar analytical results.

The data set as mentioned above is taken from a UCI Repository: Communities and Crime dataset. It has total 1994 instances and 128 attributes like population, race, and age.The attributes are real and of multivariate characteristics. This data was first converted into CSV file using JSON file from the website using Python. For naming convention, original data was assumed as 'dirty data' and the data with no missing values as 'cleaned data'.

For clean data, removal of missing values was needed to get an appropriate crime data set. Initially, columns that had these missing values or sparse values were deleted as undefined values would have a negative impact on the accuracy of the model. For dirty data, the missing data of a feature were converted into median value of that feature. For predicting feature, 'Per Capita Violent Crimes', a new column called 'High Crime' was created that had a value '1' for Per Capita Violent Crime' greater than 0.1 and '0' otherwise. The threshold of 0.1 was decided upon manual analysis of data by view-through process. All the features had to be predicted using this target feature 'High crime'.

Clean and dirty data sets were converted into different data frames and the target feature 'High crime' was assigned to a variable 'Target' and the remaining features to 'Features'. Following are the steps to implement this conversion using Python:

```
>>>df['HighCrime'] = (df['Per Capita Violent Crimes'] > 0.1)
   #A new column for target feature

>>>df, targets = encode_target(df, " HighCrime", "Target")
    #After encoding from string values to integer values, we name 'High crime' to 'Target'

>>> features = list(df)
    #This consists of all the features in a data frame

>>>features.remove("Per Capita Violent Crimes")
     #As we have a new column 'High crime' based on this feature, we remove this column
   for our model

>>>features.remove("Target")
# Target feature is not included in the remaining features' list
```

The comparative analysis, though is done using both datasets that is, dirty and cleaned data. Because in data mining, sometimes missing values also play an important role in the analysis. Hence implementation and testing of predictive models was performed on both datasets.

5. EXPERIMENTAL RESULTS

5.1 DECISION TREES

The decision tree was built to predict the target column, after splitting the dataset into random training and test sets. The splitting criterion 'Entropy' was decided upon for splitting the dataset. According to the Shannon Information Entropy Theory [23], the entropy of a variable can be defined as, $\sum$ -P * log P, for the probability P of that variable taking values of 0,1,2…. n and, the sum of probabilities of all variables is 1. Dingsheng Wan [24] mentions in his paper that, smaller the value of entropy, better can the system be described.

The predictor feature 'High crime' was evaluated using a confusion matrix [25] for train and test set, and then using Cross- Validation [26] procedure to reduce this overfitting. In the k-fold cross validation, the training set is split into k- subsets and the model is trained using remaining k-1 folds of training data (which would then become the test set). The average of the values computed is then taken into consideration for calculating the confusion matrix.The accuracy, precision and recall values calculated after performing 10-fold Cross Validation on the Decision Tree gave the values, which are much more realistic and practical as opposed to those obtained with the overfitted data (100%). Following are the steps to be implemented while using a Decision Trees Classifier in Python:

```
>>> from sklearn.tree import DecisionTreeClassifier
# Importing classifier from scikit learn

>>> from sklearn.cross_validation import train_test_split
#Importing train_test_split method from cross validation that splits our data into train (75%)
    and test (25%)

>>> y = df ['Target']
#Target data - HighCrime
>>> x = df2[features]
# Remaining features

>>>X_train, X_test, y_train, y_test = train_test_split(x , y)
#We split the data into train and test using train_test_split()

>>>dt = DecisionTreeClassifier(criterion = 'entropy')
#Object of DecisionTreeClassifier with Entropy as the criterion

>>>dt.fit(X_train, y_train)
Output: DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=None,
max_features=None, max_leaf_nodes=None,
min_impurity_split=1e-07, min_samples_leaf=1,
min_samples_split=2, min_weight_fraction_leaf=0.0,
presort=False, random_state=None, splitter='best')
#This fits the model in our data

>>>y_pred = dt.predict(X_train)
#Pass the training data to predict the feature

>>> from sklearn.metrics import accuracy_score, precision_score, recall_score

>>>accuracy_dt = accuracy_score(y_train,y_pred)*100

>>> print('Accuracy: ', accuracy_dt)

>>>precision_dt = precision_score(y_train,y_pred)*100
#Calculate accuracy, precision, recall and F1 score using metrics in scikit learn
```

Feature Importance in scikit learn measures the importance of feature in relation to target feature. Therefore, the top 10 most important features that helped in prediction of crime, based on this score were fetched. Higher the importance score, more significant is that feature in contribution to crime. The same procedure was applied on both cleaned as well as uncleaned data and the results were as follows:

Table 1: Performance Measures- Decision Trees Clean Data

Evaluating Measure DecisionTreeClassifier- Clean data	10-fold Cross-Validation (%)
Accuracy	75.9%
Precision	80.62%
Recall	81.53%
F1 score	81.22%

Table 2: Performance Measures- Decision Trees Dirty Data

Evaluating Measure DecisionTreeClassifier/Dirty Data	10-fold Cross-Validation (%)
Accuracy	76.77%
Precision	81.56%
Recall	80.58%
F1 score	80.76%

The top 10 features extracted according to the feature importance scores were:

- **PctKids2Par :** Percentage of kids in family housing with two parents
- **RacePctWhite:** Percentage of population that is Caucasian
- **RacePctHisp :** Percentage of race Hispanic
- **PctFam2Par :** Percentage of families (with kids) that are headed by two parents State
- **PctNotSpeakEnglWell:** Percentage of people not speaking English well
- **TotalPctDiv :** Percentage of population who are divorced
- **MalePctDivorce:** Percentage of Male divorced
- **PctWorkMomYoungKids:** Percentage of moms of kids 6 and under in labour force
- **PctIlleg:** Percentage of kids born to never married

Decision Tree Classifier had 'PctKids2Par'as its root implying the topmost predictive feature for 'High crime'. Using entropy as a splitting criterion, the remaining features were extracted.

5.2 RANDOM FOREST CLASSIFICATION

Random Forests Classifiers correct the decision trees' habit of overfitting the training dataset. It constructs multiple trees at the training time and outputs a mean prediction in regression and mode prediction in classification of the data set. Gini Index was used as an impurity measure for constructing trees with the help of Random Forest Classifiers. Gini Index [3] gives the separation measure between the probabilistic measure of the target attribute's values. It is an alternative to Information Gain in Decision trees and Classification. Like Entropy, Gini also reaches its maximum value when all the classes in the data set have equal probability.

Table 3: Performance Measures- Random Forest Classifier Clean Data

Evaluating Measure RandomForestClassifier/Clean Data	10-fold Cross-Validation (%)
Accuracy	83.39%
Precision	88.30%
Recall	84.86%
F1 score	86.54 %

Table 4: Performance Measures- Random Forest Classifier Dirty Data

Evaluating Measure RandomForestClassifier/Dirty Data	10-fold Cross-Validation (%)
Accuracy	81.35%
Precision	87.35%
Recall	82.81%
F1 score	84.80%

The top 10 features extracted according to the feature importance scores were:
PctNotSpeakEnglWell: Percent of people who do not speak English well

- **PctFam2Par:** Percentage of families (with kids) that are headed by two parents
- **FemalePctDiv:** Percentage of females who are divorced
- **PctPersDenseHous:**Percent of persons in dense housing (more than 1 person per room)
- **PctKids2Par:** Percentage of kids in family housing with two parents
- **TotalPctDiv:** Percentage of population who are divorced
- **Racepctblack:** Percentage of population that is African American
- **PctWInvInc:** Percentage of households with investment / rent income in 1989
- **racePctWhite:** Percentage of population that is Caucasian
- **PctPopUnderPov:** Percentage of people under the poverty level
- **MedIncome:** Median household income

Since the accuracy, precision and recall values for Random Forest Classifier is almost same on both clean and dirty data, this model perfectly fits for median values for both with and without missing values. Also, the top features like 'PctKids2Par': Percentage of kids in family housing with two parents and 'racePctWhite': Percentage of population that is Caucasian are common to the decision trees all together.

5.3 NAÏVE BAYES CLASSIFICATION

Naïve Bayes Classification uses Bayes theorem [26] that describes the probability of an event, based on prior knowledge of conditions that might be related to the event. It is given as:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

where P(A|B) = Probability of A given B is true
P(B|A) = Probability of B given A is true
P(A) and P(B) = Probability of A and B respectively
A: Target and B: Remaining Features

This classifier is implemented in Python using GaussianNB Classifier that uses Naïve Bayes Rule for calculating feature importance and their score. Again, to reduce over fitting, cross validation was used to measure accuracy, precision, recall and F1 Score. The initial steps of taking target feature in y variable and other features in x variable will be common for all models. Following are the steps to be implemented while using a Naïve Bayes Classifier in Python:

```
>>> from sklearn.naive_bayes import GaussianNB
    #Importing Gaussian Naïve Bayes classifier from scikit learn

>>>clf_gnb = GaussianNB()
    #Creating an object of a model and fitting it over train and test data

>>>clf_gnb.fit(X_train,y_train)
#Fitting the data in the classifier's object
Output: GaussianNB(priors=None)
```

Table 5: Performance Measures- Naïve Bayes Classifier Clean Data

Evaluating Measure NaiveBayesClassifier/Clean Data	10-fold Cross-Validation (%)
Accuracy	77.64 %
Precision	92.53 %
Recall	69.82 %
F1 score	79.58 %

Table 6: Performance Measures- Naïve Bayes Classifier Dirty Data

Evaluating Measure NaiveBayesClassifier /Dirty Data	10-fold Cross-Validation (%)
Accuracy	75.42 %
Precision	82.53 %
Recall	79.45 %
F1 score	79.97 %

The top 10 features extracted according to the feature importance scores were:

NumUnderPov: Number of people under Poverty line, Population
LandArea
NumbUrban: Number of people in Urban Area
HousVacant: Number of vacant houses
RacePctHisp: Percentage of race Hispanic
LemasPctOfficDrugUn: Percentage of officers assigned to drug unit
PctNotSpeakEnglWell: Percentage of people not speaking English well
RacePctAsian: Percentage of race Asian
PctPersDenseHous: Percentage of people in dense housing

The accuracy results for clean data are better than the dirty data implies the clean data set could fit this GaussianNb predictive model well. Sparse and missing data creates inconsistency and that affects the overall performance of the model. Also, the clean data gives maximum number of correctly identified documents according to the precision. However, if recall is taken into consideration with precision, dirty data gives the better performance as the relevant features identified using recall are correct according to a maximum precision.

5.4 Linear Regression

Linear Regression fits a straight line to the data using two continuous variables

- One variable, denoted *x*, is regarded as the **predictor** - Features as variables.
- The other variable, denoted *y*, regarded as the **response** - High crime variable.

A linear regression line has an equation of the form $Y = a + bX$, where X is the explanatory variable and Y is the dependent variable.
Target = a + b(Features)
For example: High crime = a + b(Population)
Linear Regression minimizes the sum of squares of the variables predicted by linear approximation. Following are the steps to be implemented while using a Linear Regression predictive model in Python:

```
>>> from sklearn.linear_model import LinearRegression
    # Importing classifier from scikit learn

>>>linreg = LinearRegression()
    #Object of a Linear Regression Model

>>>linreg.fit(X_train,y_train)
#Fitting the data in the classifier's object
     Output: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)

>>> from sklearn.metrics import mean_squared_error
     #Importing mean squared error module from metrics

   >>>mse=cross_val_score(linreg,
X_train,y_train,cv=10,scoring='neg_mean_squared_error')
    #Calculating mean squared error
```

Table 7: Performance Measures- Linear Regression Classifier Clean Data

Evaluating Measure LinearRegression- Clean Data	10-fold Cross-Validation (%)
MSE	0.0179
Accuracy	64.72 %
Precision	88 %
Recall	85.4 %
F1 score	82.3 %

Table 8: Performance Measures- Linear Regression Classifier Dirty Data

Evaluating Measure Linear Regression- Dirty Data	10-fold Cross-Validation (%)
MSE	0.0177
Accuracy	66.93 %
Precision	74 %
Recall	84.98 %
F1 score	86.6 %

The top 10 features extracted according to the feature importance scores were:

FemalePctDiv: Percentage of females who are divorced
PctRecImmig8: Percentage of _population_ who have immigrated within the last 8 years
PctOccupMgmtProf: Percentage of _population_ who have immigrated within the last 8 years
RentHighQ: Rental housing - upper quartile rent
PctForeignBorn: Percentage of people foreign born
PctRecImmig5: Percentage of _population_ who have immigrated within the last 5 years
AgePct16t24: Percentage of population that is 16-24 in age
PctImmigRec10: Percentage of _immigrants_ who immigrated within last 10 years
RacePctHisp: Percentage of race Hispanic
AgePct65up: Percentage of population that is 65 and over in age
Although the values for Mean squared error and Accuracy came out to be close, the dirty data fitted this regression model well. It is a surprising fact that although the accuracy value is low for the clean data, balance in precision and recall values(high) shows that whatever the features identified, are identified correctly.

Table 9: Comparative Results of Data Mining methods and their performance
Clean Data

Classifier	Accuracy	F1 Score
Decision Tree	75.90 %	81.22 %
Random Forest	83.39 %	86.54 %
Naïve Bayes	77.64 %	79.58 %
Linear Regression	64.72 %	82.3 %

Table 10: Comparative Results of Data Mining methods and their performance
Dirty Data

Classifier	Accuracy	F1 Score
Decision Tree	76.77 %	80.76 %
Random Forest	81.35 %	84.80 %
Naïve Bayes	75.42 %	79.97 %
Linear Regression	66.93 %	86.60 %

6. CONCLUSION

The paper concludes withRandom Forest Classifier giving the most balanced results with respect to accuracy, precision, recall and F1 scoreout of three models for prediction of 'Per Capita Violent Crimes' feature. While Linear Regression gave the lowest values in these performance measures, the data could not fit well to the straight line considered using target and remaining features. The high value of accuracy in dirty data for this model as compared to clean data shows that regression needs continuous data including the sparse values. Random Forest Classifier takes multiple trees into account and gives an average of the result which proved to be perfect for this type of data. Naïve Bayes proved to be a balancing quotient for this crime data as it had values close to the Random Forest Classifier. Some common features having high importancescores that proved to be highly predictive of 'High crime' features are 'NumUnderPov': Number of people under poverty level, 'PctNotSpeakEnglWell': Percent of people who do not speak English well, 'LandArea', 'NumbUrban': Number of people in Urban Area using Random Forest Classifier model and Naïve Bayes Classifier model.

Reduction of overfitting using cross validation improves performance by enough training and testing samples that seemed to help in this analysis by giving correct and consistent performance measures. These predicted features will be useful for the Police Department to utilize their resources efficiently and take appropriate actions to reduce criminal activities in the society. This can be used to enhance security and protection of criminal data by a desktop or a mobile application to track the crime rate and take any safety measures based on the relevant features.By maintaining dynamic databases with the criminal records across various countries, this technique can be implemented widely all over the world. The present dataset consists of all types of crimes, this type of analysis can be narrowed down to a single category of crime.

REFERENCES

[1] C.McCue, Data Mining, and Predictive Analysis. Oxford, UK, Elsevier, Butterworth-Heinemann.2007

[2] Nyce, Charles (2007), Predictive Analytics White Paper, American Institute for Chartered Property Casualty Underwriters/Insurance Institute of America, p.

[3] OdedMaimon, LiorRokach, "The Data Mining and Knowledge Discovery Handbook", Springer 2005, Page 6

[4] Han, Jiawei et.al "Data Mining", Second Edition, Page 285

[5] UCI Repository Communities and Crime dataset, Retrieved from http://archive.ics.uci.edu/ml/datasets/communities+and+crime

[6] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, "Introduction to Data Mining", Pearson Addison Wesley, 2006, Page 187,227, 296, 297, 729

[7] Pressman Rodger S 6th edition," Software Engineering – A Practitioner's Approach "

[8] ThongsatapornwatanaUbon, "A survey of data mining techniques for analysing crime patterns", Second Asian Conference on Defence Technology(ACDT), 2016

[9] P.Thongtae and S.Srisuk, "An analysis of data mining applications in crime domain", IEEE 8th International Conference on Computer and IT Workshops, 2008

[10] OzgulFatih et al., "Incorporating data sources and methodologies for crime data mining", IEEE International Conference on Intelligence and Security Informatics, Proceedings of 2011

[11] Chen Hsinchun et.al, "Crime Data Mining: A General Framework and Some Examples", IEEE Journals and Magazines, 2004, Vol 37, Issue 4

[12] Decision trees: LiorRokach, OdedMaimon, "Data Mining with Decision Trees: Theory and Applications", Second Edition, Pages 15,27

[13] Scikit Learn, Retrieved from http://scikit-learn.org/stable/

[14] Pandas Module in Python, Retrieved from http://pandas.pydata.org/

[15] Numpy Module in Python, Retrieved from http://www.numpy.org/

[16] Sklearn Decision Tree in Python, Retrieved from http://scikit-learn.org/stable/modules/tree.html

[17] Sklearn Random Forest in Python, Retrieved from http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[18] Sklearn Naïve Bayes Module in Python, Retrieved from http://scikit-learn.org/stable/modules/naive_bayes.html

[19] Sklearn Linear Regression in Python, Retrieved from http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

[20] Data preprocessing: S. Kotsiantis, D. Kanellopoulos, P. Pintelas, "Data Preprocessing for Supervised Learning", International Journal of Computer Science, 2006, Vol 1 N. 2, Pages 111–117

[21] Gaikwad Bhagyashri and Halkarnikar P.P, "Random Forest Technique for E-mail Classification", International Journal of Scientific & Engineering Research, Issue 3, March-2014, Volume 5.

[22] SathyadevanShiju et.al, "Crime Analysis and Prediction Using Data Mining",2014 First International Conference on Networks & Soft Computing

[23] Xing Li-Ning (College of Information System and Management,National University of Defense Technology), Tang Hua, "Data mining algorithm based on genetic algorithm and entropy," Journal of Computational Information Systems, Volume 3, May 2007

[24] Wan Dingsheng et.al, "Data Mining Algorithmic Research and Application Based on Information Entropy", 2008 International Conference on Computer Science and Software Engineering

[25] Simple Linear Regression, Retrieved from https://onlinecourses.science.psu.edu/stat501/node/250

[26] Jeffreys, Harold (1973). Scientific Inference (3rd ed.). Cambridge University Press. Page 31.

AUTHORS

Prajakta R. Yerpudeis pursuing her Masters in Computer Science 2016-18, from Illinois Institute of Technology, Chicago, USA. She has been working in the domain of Data Science since the past 4 years. Her interests include Advanced Data Mining, Machine Learning, Data analytics, Databases, and Artificial Intelligence. She has published a research paper titled "Algorithms for text to graph conversion and text summarization using NLP: A business solution" at International Journal of Natural Language Computing

Vaishnavi V. Gudur is pursuing her Masters in Computer Science 2016-18, from Illinois Institute of Technology, Chicago, USA. She has been working in the domain of Database Management and Web Enterprise Applications since the past 3 years. Herinterests include Big Data Analytics,Advanced Data Mining,Web Development, Databases.