STAT 183

Introductory Data Analytics

THE PREDICTIVE ANALYTICS PROJECT

John Robert Torres

Carl Jacob Villar

Ma Andrea Villanueva

AB-1L

ORGANIZATION'S UNDERSTANDING

There is no other city that is filled with so much history as Boston. Apart from its history, it is a city renowned for its sports, universities, culture, and people. In relation, Boston is also one of the major cities in the USA and houses a population of diverse ethnic backgrounds. However, like the majority of cities, it also experiences a high crime rate (Cove, n.d.).

Boston's crime index ranking of 18 indicates that it is safer than just 18% of American cities. The figure is startling in light of the fact that 89.66 percent of Boston's workforce is working in white-collar professions, a figure far higher than the national average (Boston Indicators, n.d.). However, the city's crime data is unusually high and contradicts the city's economic significance within the country. Further, various statistics pertaining to Boston's crime incidents have been compared with the country's average figures in numerous instances. The studies show that the crime rate in Boston is much higher than the national average and that there are more violent crimes per 1000 crimes in Boston than in the rest of the country (Boston, MA Real Estate & Demographic Data, n.d.).

In the United States, crimes are classified into three groups based on their severity: felonies, misdemeanors, and infractions. The most serious offenses are felonies, whereas misdemeanors are the least severe. Infractions are minor; in most cases, they are not even considered crimes, so they will not be of interest in this study (Montalbo, 2019). Similar to the study of Tasnim et al. (2019), a classification approach was utilized based on different forms of time: season, weekdays, and time itself to predict the severity of the crime. This study shows that the origin, structure, intensity, and pattern of crime come from the relationship between these natural things and that this relationship has something to do with how severe the crime is.

By providing a classification-based machine learning technique to predict the severity of occurring crimes at a certain time, this study seeks to increase public awareness and aid law enforcement in identifying criminal patterns that will guarantee the safety of the Boston community.

DATA UNDERSTANDING

BOSTON CRIME DATASET

The Boston Police Department (BPD) provided a crime incident report that was recorded from June 14, 2015, to September 3, 2018. The dataset includes 319,073 instances and 17 attributes in which data about the incident numbers, offense descriptions, districts, locations, and the exact date and time of occurrence of the crime incidents were recorded. The dataset was retrieved at the Kaggle website and can be accessed through the link: Crimes in Boston | Kaggle. Table 1 below shows the data type, level of measurement, and data description of the 17 variables in the dataset.

Table 1. Data dictionary of the Boston crime dataset

Variable Name	Variable Type	Level of Measurement	Variable Description
incident_number	qualitative	nominal	Unique number assigned to an offender
offense_code	qualitative	nominal	Numeric code for each offense classification
offense_code_group	qualitative	nominal	The grouped classification of offenses
offense_description	qualitative	nominal	More specific type of offenses in the grouped classification
district	qualitative	nominal	Code for districts in Boston
reporting_area	qualitative	nominal	Geographic area for which offenses are taken and recorded

shooting	qualitative	nominal	Incidence of shooting (Y - Yes, NA - No)
occured_on_date	qualitative	ordinal	Exact date and time of crime occurrence
year	qualitative	ordinal	Year when the offense was recorded (2015 - 2018)
month	qualitative	ordinal	Month of the year when the offense was recorded (1 - 12)
day_of_week	qualitative	ordinal	Day of the week when the offense was recorded (Sunday - Saturday)
hour	qualitative	ordinal	Hour when the offense was recorded (0 - 23)
ucr_part	qualitative	nominal	Stands for Uniform Crime Reporting that divides the offenses into parts or groups depending on the intensity of crime (Part One, Part Two, Part Three)
street	qualitative	nominal	Name of the streets in Boston
lat	quantitative	interval	Latitude of the location of crime
long	quantitative	interval	Longitude of the location of crime
location	quantitative	interval	Latitude and longitude of the location of crime

DATA PREPARATION

In order to prepare the Boston crime dataset for predictive analysis, the dataset was modified, cleaned, transformed, and enriched to obtain the needed information. First, the attribute names were converted to lowercase for uniformity. Then, the data were modified and enriched by collapsing the variables needed into categories. For instance, the months were converted to four seasons of the year, namely, winter (December to February), spring (March to May), summer (June to August), and fall (September to November). The hour variable was collapsed into two which are day-time (7 AM to 6 PM) and night-time (7 PM to 6 AM). The days of the week were also derived as weekend (Sunday and Saturday) and weekday (Monday to Friday). The last part of the data modification was by identifying the felony and misdemeanor. To eliminate ambiguity in the dataset, only crimes that are definitive enough to be either felonies or misdemeanors are chosen. Among 67 types of crime, 15 crimes were identified as felonies and 20 crimes were identified as misdemeanors. The rest was identified as vague and was dropped later on. The following table shows the types of crimes that were identified as felonies and misdemeanors.

Table 2. Felony and misdemeanor crimes

Felony	**Misdemeanor**
Larceny	Simple Assault
Drug Violation	Verbal Disputes
Vandalism	Towed
Aggravated Assault	Robbery
Property Lost	Harassment
Larceny from Motor Vehicle	Disorderly Conduct
Residential Burglary	Motor Vehicle Accident Response
Auto Theft	Firearm Violations
Homicide	License Violation
Arson	Restraining Order Violations

Manslaughter	Counterfeiting
Human Trafficking	Liquor Violation
Burglary - No Property Taken	Landlord/Tenant Disputes
Human Trafficking - Involuntary Servitude	Assembly or Gathering Violations
	Prostitution
	Criminal Harassment
	Home Invasion
	Bomb Hoax
	Phone Call Complaints

In relation to data modification and enrichment, only the variables needed such as the incident number, offense code group, season, timeday, dayweek, and offense type were kept. These attributes were also converted into binary and numerical attributes as shown in the table below.

Table 3. Modification of types of attributes

Attribute	Description	Binary/Numerical Attribute
offense_type	Felony	1
	Misdemeanor	0
season	Winter	0
	Spring	1
	Summer	2
	Fall	3
timeday	Day-Time	1
	Night-Time	0
dayweek	Weekend	1
	Weekday	0

After data cleaning and modification, the resulting dataset now has 201,486 rows and 6 columns.

MODELING/TOOLS

Following the data preparation, a probability-based classifier called Naive-Bayes and search-based Decision Trees were implemented for the classification of severity (felony or misdemeanor) of each crime based on time-related predictors (season, time of the day, and day of the week). The models were trained on a training dataset (n = 140,714) and their predictive performances were evaluated on a testing dataset (n = 60,772).

NAIVE BAYES CLASSIFICATION

Naive Bayes classification is a probability-based method that involves calculating the posterior probability of a dependent variable by incorporating prior and conditional probabilities. This classification method works best when its predictor variables are categorical, which is the case for this study. It works under the assumption that the predictors in the model are independent of each other. To establish independence among the three predictor variables, Table 4 shows the variance inflation factors (VIF) of each predictor. In practice, VIF values greater than five imply a multicollinearity problem. Since all values are less than five, multicollinearity will not be a problem in building the full models.

Table 4. VIF values of the independent variables

Predictor	VIF
season	1.0004
timeday	1.0085
dayweek	1.0086

The final working model of the naive Bayes classifier can be expressed in terms of prior probabilities and the different conditional probabilities between the dependent variable (severity of the crime) and each predictor. The summary of these probabilities is shown in Table 5.

Table 5. Summary of prior or class probability and conditional probabilities of the naive Bayes model

	Felony (Y = 1)	*Misdemeanor (Y = 0)*
prior or class probability	0.4897	0.5103
season		
winter (season = 0)	0.2111	0.2195
spring (season = 1)	0.2248	0.2388
summer (season = 2)	0.3263	0.3076
fall (season = 3)	0.2377	0.2341
timeday		
night-time (timeday = 0)	0.3590	0.3978
day-time (timeday = 1)	0.6410	0.6022
dayweek		
weekday (dayweek = 0)	0.7357	0.7123
weekend (dayweek = 1)	0.2643	0.2877

As shown above, severe class imbalance is not observed because the number of misdemeanors are only about 2% higher than the number of felonies. Felonies are observed more during summer and fall, but misdemeanors are observed more during spring and summer. In terms of the time of the day when the crimes were committed, most of the crimes for both classes happened during the daytime (6 AM to 6 PM) because it covers crime peak hours of noontime and six in the afternoon. Moreover, most of the crimes were committed during weekdays, which is understandable because weekdays are composed of more days than the weekends.

DECISION TREES

Decision trees as a classification method involve subsetting a dataset according to a certain parameter while building and developing a classification tree at the same time. Its final result is a tree-like structure composed of several nodes i.e., root nodes, decision nodes, or terminal nodes. Figure 1 shows the final structure of the decision tree classifier used in this study.

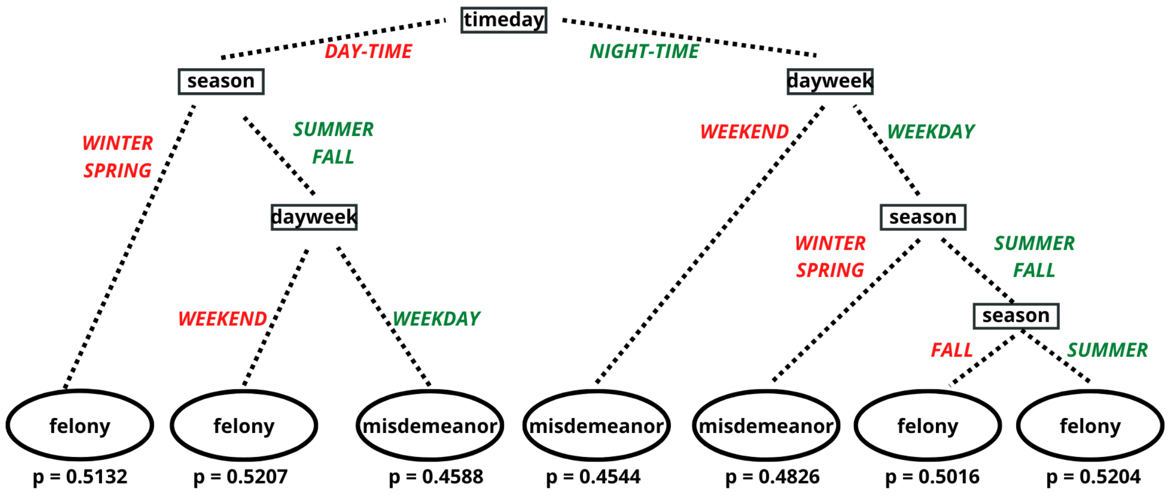

Figure 1 . Decision tree model for crime severity classification

The root node contains the *timeday* variable, the first parameter by which the training dataset was split. If the crime was committed during daytime, the next decision parameter will be *season* followed by *dayweek* if the crime was during summer and fall. On the other hand, If the crime was committed at nighttime, the next decision parameter will be *dayweek* followed by *season* when the crime happened on a weekday. The leaf nodes at the bottom of the structure show the probability and classification of each branching node.

SUMMARY AND COMPARISON OF PREDICTIVE PERFORMANCE

To show how each model performs, a summary of their classification performance is shown in a confusion matrix. The confusion matrix is a table that shows the frequency of actual and predicted classes coming from each model. It shows the true positives (felonies classified as felonies), false positives (misdemeanors classified as felonies), true negatives (misdemeanors classified as misdemeanors), and false negatives (felonies classified as misdemeanors) that resulted from the classification of the observations in the testing dataset. Tables 6 and 7 show the summary for the naive Bayes classifier and decision trees, respectively.

Table 6. Confusion matrix for the results of the naive Bayes classification

n = 60,772	Actual Felony	Actual Misdemeanor
Predicted Felony	11,354	10,580
Predicted Misdemeanor	18,356	20,482

Table 7. Confusion matrix for the results of the decision tree classification

n = 60,772	Actual Felony	Actual Misdemeanor
Predicted Felony	7,587	6,725
Predicted Misdemeanor	22,123	24,337

From the results above, it can be observed that the naive Bayes classifier performs better in classifying actual felonies, but it produces more false positives compared to the decision tree. On the other hand, the decision tree performs better in identifying actual misdemeanors but is producing more false negatives than the naive Bayes model. To further assess their predictive performances, a summary of different accuracy metrics for both models is shown in the table below.

Table 8. Accuracy metrics for naive Bayes and decision tree classification

n = 60,772	Naive Bayes	Decision Tree
Accuracy	52.39%	52.53%
Error Rate	47.61%	47.47%
Sensitivity	38.22%	25.54%
Specificity	65.94%	78.35%

The accuracy rates for the naive Bayes model and the decision tree are almost equal at around 52%. However, their specificity and sensitivity rates differ by more than 10%. Sensitivity describes the ability of a classifier to predict true positives, while specificity refers to the ability of a classifier to predict true negatives.

In the context of crime severity, the consequences of a false negative, or the instance of predicting a misdemeanor when in fact the crime was a felony, will be lighter than the consequence of a false positive, or the instance of predicting a felony when in fact the crime was a misdemeanor. Based on these, a classifier that is more sensitive is preferred because false negatives should be minimized. The naive Bayes model has higher sensitivity, while the decision tree has higher specificity. However, a sensitivity of about 38% can be considered low, and adjustments should be made in order to increase it. One of the adjustments that can be made is to adjust the classification threshold. Unfortunately, in this problem, only a 2% increase can be accommodated so that the accuracy and specificity of the models are not greatly affected negatively. This is an insignificant increase, but the impact on the sensitivity of each model is large.

Table 9. Accuracy metrics for naive Bayes and decision tree classification at increased threshold

n = 60,772	Naive Bayes	Decision Tree
Accuracy	51.18%	51.41%
Error Rate	48.82%	48.59%
Sensitivity	65.30%	58.81%
Specificity	37.68%	44.34%

In terms of accuracy, the models still perform alike at around 51%. However, it can be observed that the sensitivity of each model significantly increased compared to the values in the previous table. A trade-off between sensitivity and specificity inevitably took place, but in this specific problem of classifying the severity of a crime based on time-related predictors, a model that minimizes false negatives is more preferred.

EVALUATION

Performing an in-depth predictive analysis of the Boston Crime dataset enabled us to find the relationship between crime and time factors. Exploring the internal relationship among the time factors of nature helped in finding an interesting crime pattern. By utilizing classification algorithms and providing them with time factors such as season, days, and time of the week, we were able to classify the severity of crime.

Based on the produced Naive Bayes model, felonies are observed more during summer and fall, but misdemeanors are observed more during spring and summer. In terms of the time of the day when the crimes were committed, most of the crimes for both classes happened during the daytime (6 AM to 6 PM) because it covers the crime peak hours of noontime and six in the afternoon. Moreover, most of the crimes were committed on weekdays. The results are in line with the study of Almanie et al. (2015), wherein weekends have the lowest crime rate. At the same time, in the recent study of Favre (2019), he found out that most crimes committed in big cities occur during the day. But at the same time, it is also in contrast since both in the studies of Almanite et al. (2015), and Favre (2019), more violent offenses happen more frequently at night. Meanwhile, the decision tree model shows that if the crime was committed during the daytime, the next decision parameter will be *season* followed by *dayweek* if the crime was committed during summer and fall. On the other hand, if the crime was committed at night, the next decision parameter will be *dayweek* followed by *season* when the crime happened on a weekday.

In relation, by comparing various accuracy metrics acquired from the two classification approaches, it was found that the Naive Bayes algorithm works best for our model. Even though the accuracy rates for the naive Bayes model and the decision tree are almost equal at around 52%, the naive Bayes model has higher sensitivity, which is more preferred In the context of crime severity.

Overall, these findings have encouraged us to relate crime with time factors. This information may be utilized for a variety of reasons, depending on the unique needs of the concerned stakeholders. For instance, the Boston Police Department would be interested in gaining insights in order to implement more effective crime control measures. Similarly, the insights may also support the business community in enhancing its security based on time. Additionally, the information may alert the healthcare sector depending on the time and criticality of the crime.

SUMMARY OF R CODES AND DATASET

```
# for cleaning
# libraries
library(tidyr)
library(dplyr)
library(writexl)
library(janitor)

# loading raw dataset
crime =  read.csv('crime.csv', sep = ",",
        na.strings =
        c('','NA','na','N/A','n/a','NaN','nan
        '), strip.white = TRUE,
        stringsAsFactors = FALSE)
crime = clean_names(crime)
nrow(crime)

# summary of missing data per column
sort(sapply(crime, function(x)
        sum(is.na(x))), decreasing =
        TRUE)
summary(crime)

# derive season from months
crime = crime %>% mutate(season =
                    ifelse(month %in%
        c(6,7,8), "Summer",
                        ifelse(month
        %in% c(9,10,11), "Fall",
                             ifelse(month
        %in% c(12,1,2), "Winter",
        "Spring"))))

# derive time of the day from hour
crime = crime %>% mutate(timeday =
                    ifelse(hour %in% c(7,
        8, 9, 10, 11, 12, 13, 14, 15, 16,
        17, 18), "Day-Time",
        "Night-Time"))

#derive weekday/weekend
crime = crime %>%
  mutate(dayweek =
        ifelse(day_of_week %in%
        c("Sunday", "Saturday"),
        "Weekend", "Weekday"))

# drop unnecessary columns
colnames(crime)
```

```
raw_crime = subset(crime,
            select = c(incident_number,
        offense_code_group, season,
        timeday, dayweek))
summary(raw_crime)
View(raw_crime)

#identify felony/misdemeanour
felony = c(
  "Larceny",
  "Drug Violation",
  "Vandalism",
  "Aggravated Assault",
  "Property Lost",
  "Larceny From Motor Vehicle",
  "Fraud",
  "Residential Burglary",
  "Auto Theft",
  "Homicide",
  "Arson",
  "Manslaughter",
  "HUMAN TRAFFICKING",
  "Burglary - No Property Taken",
  "HUMAN TRAFFICKING -
        INVOLUNTARY SERVITUDE"
)
misdemeanour = c(
  "Simple Assault",
  "Verbal Disputes",
  "Towed",
  "Robbery",
  "Harassment",
  "Disorderly Conduct",
  "Motor Vehicle Accident Response",
  "Firearm Violations",
  "License Violation",
  "Restraining Order Violations",
  "Counterfeiting",
  "Liquor Violation",
  "Landlord/Tenant Disputes",
  "Assembly or Gathering Violations",
  "Prostitution",
  "Criminal Harassment",
  "HOME INVASION",
  "Bomb Hoax",
  "Phone Call Complaints",
  "Biological Threat"
)
```

```
#derive dependent variable
raw_crime = raw_crime %>%
  mutate(offense_type =
        ifelse(offense_code_group
        %in% felony, "Felony",

        ifelse(offense_code_group
        %in% misdemeanour,
        "Misdemeanour", "Vague")))

#drop vague
raw_crime = subset(raw_crime,
        offense_type != "Vague")

#convert to codes
y_name = c(Felony = 1,
        Misdemeanour = 0)
raw_crime$offense_type =
        as.factor(y_name[(raw_crime$o
        ffense_type)])

season_name = c(Winter = 0, Spring =
        1, Summer = 2, Fall = 3)
raw_crime$season =
        as.factor(season_name[(raw_cri
        me$season)])

timeday_name = c('Day-Time' = 1,
        'Night-Time' = 0)
raw_crime$timeday =
        as.factor(timeday_name[(raw_c
        rime$timeday)])

dayweek_name = c(Weekend = 1,
        Weekday = 0)
raw_crime$dayweek =
        as.factor(dayweek_name[(raw_
        crime$dayweek)])

# save as excel file
write_xlsx(raw_crime,
        "task3_final_crime.xlsx")

#for modelling
library(readxl)
library(caret)
library(party)
library(rpart)
library(rpart.plot)
library(RColorBrewer)
library(rattle)
library(ipred)
library(adabag)

data =
        read_excel("task3_final_crime.x
        lsx")

set.seed(1779)
sample = sample(c(TRUE, FALSE),
        nrow(data), replace = T, prob =
        c(0.7,0.3))
train = data[sample, ]
test = data[!sample, ]
nrow(train) + nrow(test) == nrow(data)
train$season = as.factor(train$season)
train$timeday =
        as.factor(train$timeday)
train$dayweek =
        as.factor(train$dayweek)

train$offense_type =
        as.factor(train$offense_type)
test$season = as.factor(test$season)
test$timeday = as.factor(test$timeday)
test$dayweek =
        as.factor(test$dayweek)
test$offense_type =
        as.factor(test$offense_type)

train_x = subset(train, select =
        c(season, timeday, dayweek))
train_y = train$offense_type
nb_4 = train(train_x,
          train_y,
          'nb',
          trControl =
        trainControl(method = 'cv',
        number = 10))

tree = ctree(offense_type ~ season +
        timeday + dayweek,
          data = train)
traverse <- function(node) {
  if (node$terminal) {
    return(node$prediction[2])
  }
  return(c(node$prediction[2],
        traverse(node$left),
        traverse(node$right)))
}
traverse(tree@tree)[c(3,5,6,8,10,12,13
        )]

#function for accuracy metrics
cmd = function(cm){
  TP = cm$table[1,1]
  TN = cm$table[2,2]
  FP = cm$table[1,2]
  FN = cm$table[2,1]
  n = TP + TN + FP + FN
  p = TP / (TP + FP)
  r = TP / (TP + FN)
  fm = (2 * p * r) / (p + r)
  cat("TP = ", TP, "\n")
  cat("TN = ", TN, "\n")
  cat("FP = ", FP, "\n")
  cat("FN = ", FN, "\n")
  cat("accuracy = ", (TP + TN) / n, "\n")
  cat("error = ", (FP + FN) / n, "\n")
  cat("sensitivity = ", TP / (TP + FN),
        "\n")
  cat("specificity = ", TN / (TN + FP),
        "\n")
  cat("precision = ", p, "\n")
  cat("recall = ", r, "\n")
  cat("fm = ", fm, "\n")
}

pred_nb = ifelse(prob_nb[,2] > 0.52,
        "1", "0")
nbcm =
        caret::confusionMatrix(as.factor(
        pred_nb),
        as.factor(test$offense_type))
cmd(nbcm)

prob_tree = predict(tree, test, type =
        "prob")
preds = c()
for(i in 1:length(prob_tree)){
  preds = c(preds, prob_tree[[i]][2])
}
pred_tree = ifelse(preds > 0.52, "1",
        "0")
dtcm =
        confusionMatrix(as.factor(pred_
        tree),
        as.factor(test$offense_type))
cmd(dtcm)
```

REFERENCES

Almanie, T., Mirza, R. &Lor, E. (2015).Crime Prediction Based on Crime Types and Using Spatial and Temporal Criminal Hotspots, International Journal of Data Mining & Knowledge Management Process, Vol. 5, No. 4, pp. 1-19.

Barresi, S. (2021, August 22). *Boston MA crime rate: is Boston safe? [data, stats, reports, map]*. Mass Bay Movers. Retrieved April 25, 2022, from https://massbaymovers.com/blog/boston-crime-rate/

Boston Indicators. (n.d.). *Boston's more diverse than you may realize*. Boston Indicators. Retrieved April 25, 2022, from https://www.bostonindicators.org/article-pages/2018/september/boston-diversity

Boston Police Department. (n.d.). *The Boston Police Department's virtual community*. bpdnews.com. Retrieved April 25, 2022, from https://bpdnews.com/?category=Crime%2BStats

Cove. (2022, April 11). *Boston crime rate: Is Boston a safe city?* Self-Installed DIY Home Security Systems. Retrieved April 25, 2022, from https://www.covesmart.com/blog/boston-crime-rate-is-boston-a-safe-city/

Crime Grade. (n.d.). *The safest and most dangerous places in Boston Metro, MA* . Retrieved April 25, 2022, from https://crimegrade.org/safest-places-in-boston-ma-metro/

Favre. (2019). *Study finds crime in big cities is more likely during the day | cities* U.S. News. Retrieved May 16, 2022, from https://www.usnews.com/news/cities/articles/2019-06-12/study-finds-crime-in-big-cities-is-more-likely-during-the-day

Neighborhood Scout. (n.d.). *Boston, MA crime rates*. NeighborhoodScout. Retrieved April 25, 2022, from https://www.neighborhoodscout.com/ma/boston/crime