CYO

CRIME IN BOSTON

I. Problem

This project analyses the crime statistic of the Boston Police Department as found on Kaggle.com (https://www.kaggle.com/sourinroy/boston-crime-dataset-updated-july-2020) and tries to find a pattern that can be used in order to predict shootings based on other information involved in an offense. This is done by applying a KNN-model on a set of 17 base variables. Knowing more about the structure of crime and the occurrence of shootings could be of practical use for crime prevention.

II. Data

The dataset consists of 501070 observations of the following 17 variables:

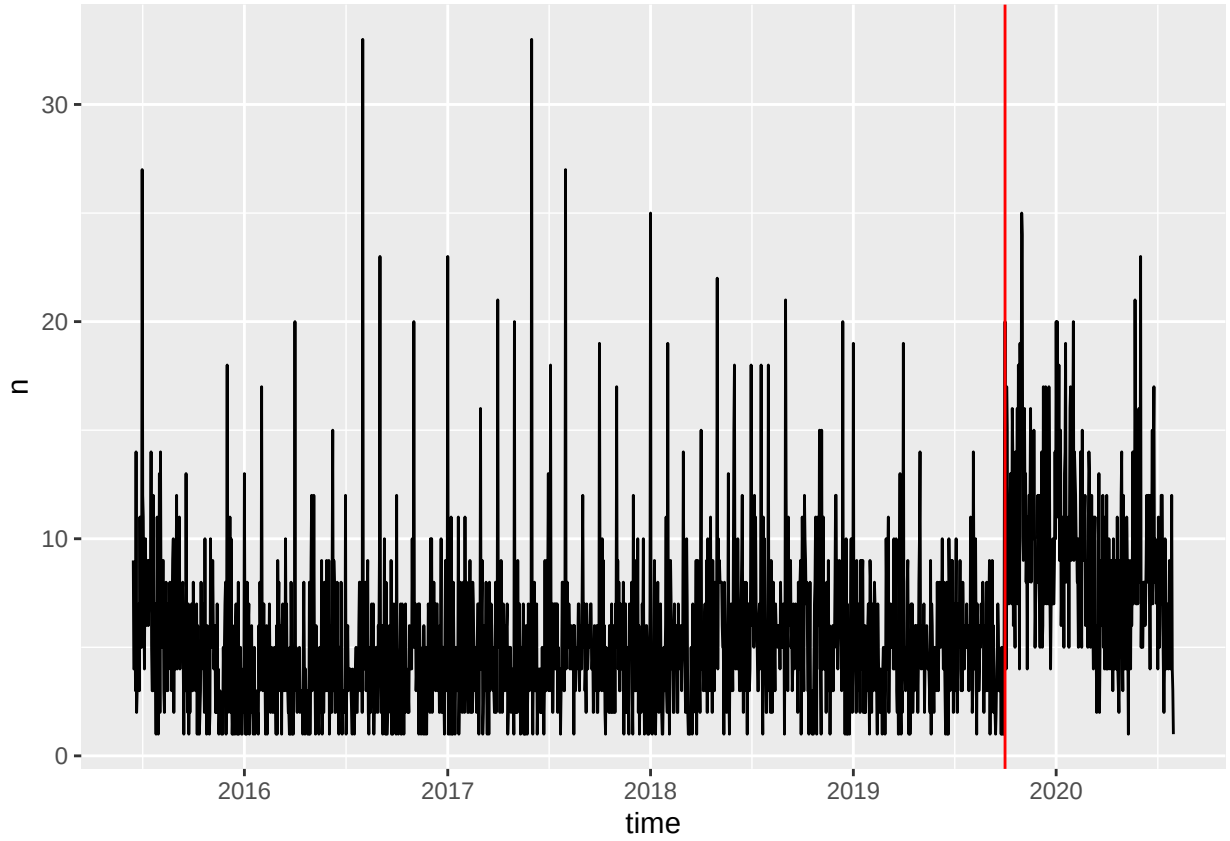
INCIDENT_NUMBER, OFFENSE_CODE, OFFENSE_CODE_GROUP, OFFENSE_DESCRIPTION, DISTRICT, REPORTING_AREA, SHOOTING, OCCURRED_ON_DATE, YEAR, MONTH, DAY_OF_WEEK, HOUR, UCR_PART, STREET, Lat, Long, Location

It is obvious that some of the variables are redundant, such as "OCCURED_ON_DATE" and "YEAR", nevertheless this will make extracting information easier. The data covers the entire period between the 15th of June 2015 until the 30th of July 2020. With the aim of finding a periodic pattern in the data, the total occurrences of daily offenses is plotted and analysed.

2.1. Time

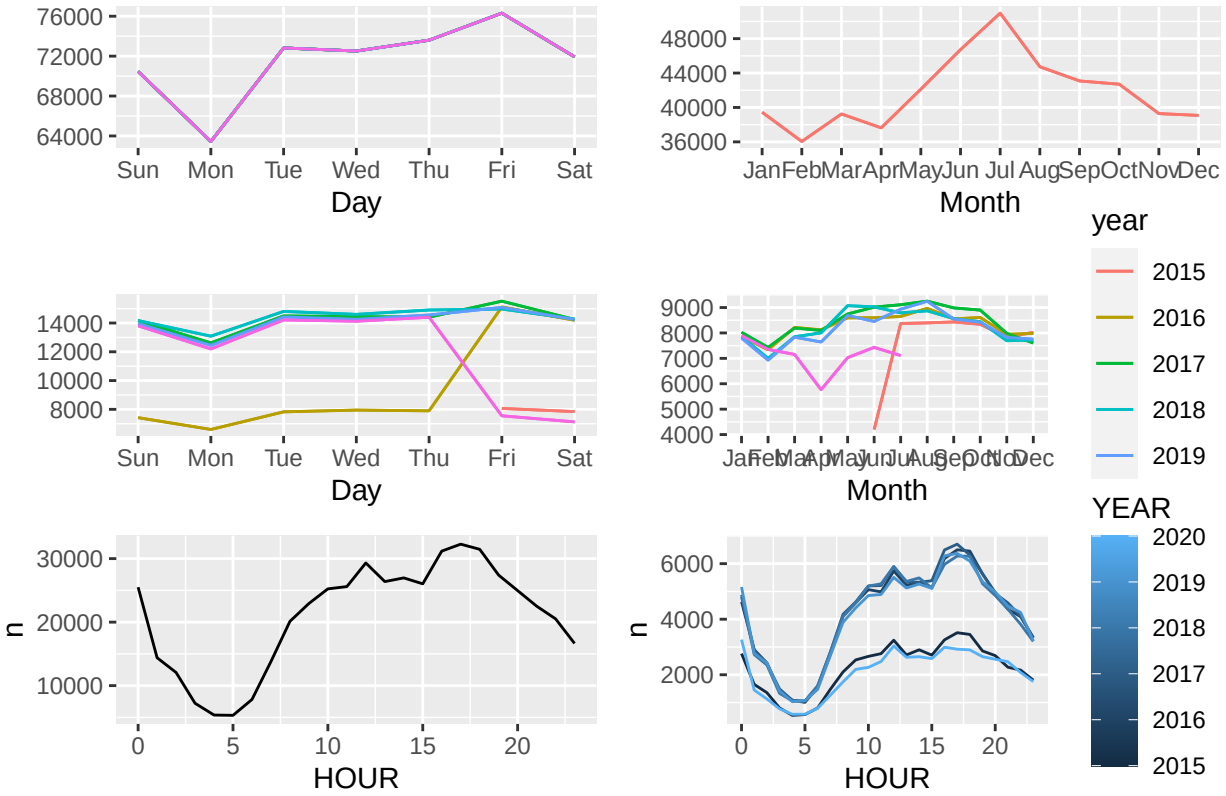
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
crimedata2 %>% ggplot(aes(time,n))+
  geom_line()+
  geom_vline(xintercept = as.numeric(crimedata2$time[1570]),color="red")
```



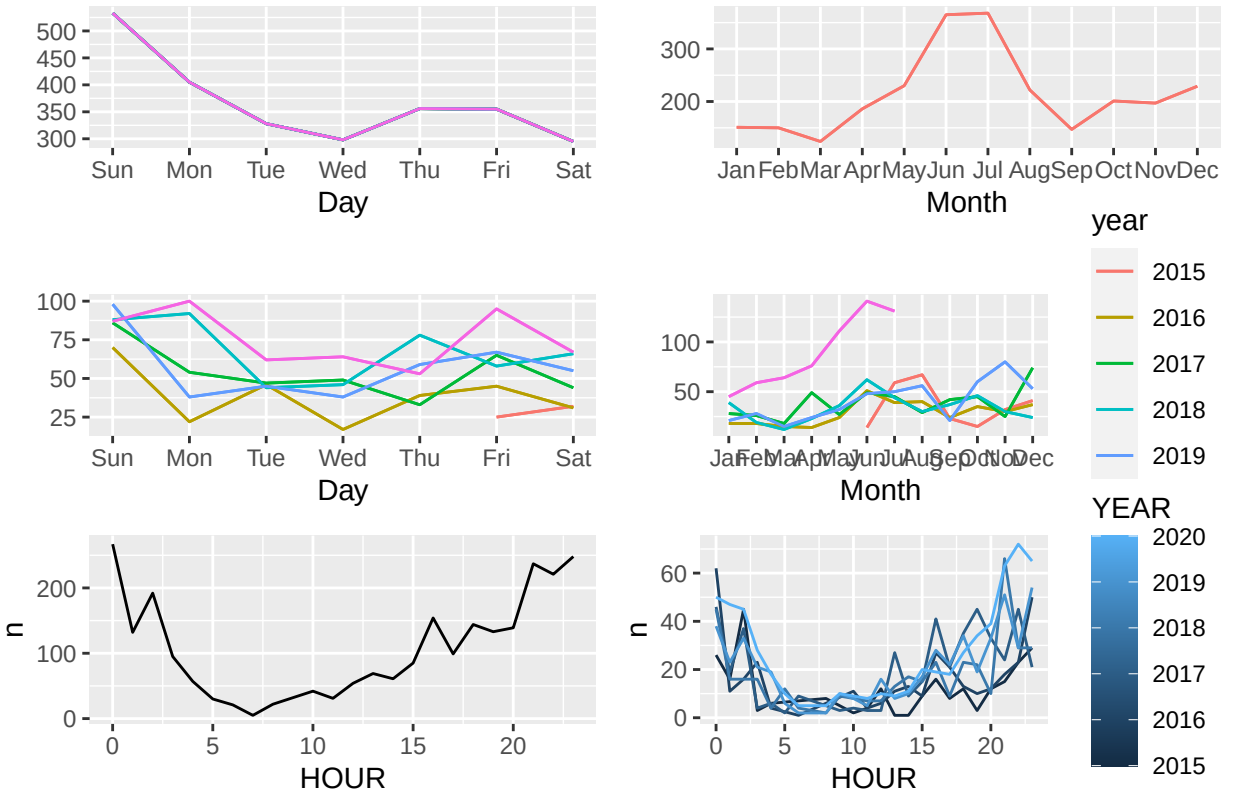
In the plot displayed above, a structural break in form of an increase in the level of daily crime can be noted just before the beginning of the year 2020 (marked in red).

```
## `summarise()` ungrouping output (override with `.groups` argument)
## `summarise()` ungrouping output (override with `.groups` argument)

## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)

## `summarise()` ungrouping output (override with `.groups` argument)

## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
```



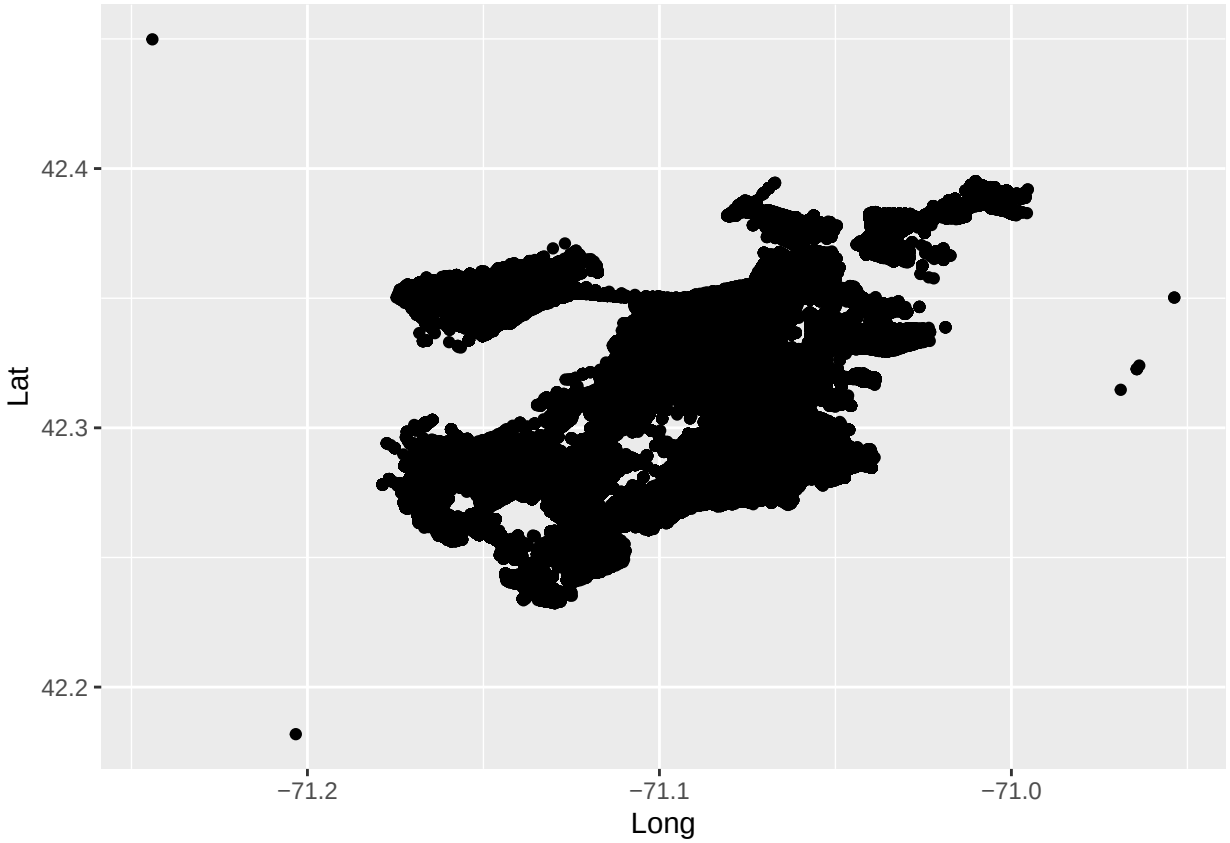
When taking a look at the weekly and monthly data it can be noted that crime spikes in the summer months and at the weekends. It is particularly low on mondays. While the level of crime remains relativly constant over the years (contrary to the first indication), the level is particularly low for the year 2020. A possible explanation could be the COVID-pandemic. A look at the hourly data reveals that most offenses take place at 5 p.m., while the minimum lies at 5 a.m.. The hourly pattern stays comparatively constant over time.

```
## `summarise()` ungrouping output (override with `.groups` argument)
## `summarise()` ungrouping output (override with `.groups` argument)

## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)

## `summarise()` ungrouping output (override with `.groups` argument)

## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
```



When compared to the overall occurrence of crime, it can be noted that offenses that included a shooting were a lot higher in 2020 than in the rest of the years. Also it has to be noted that the distribution of the shooting-related incidents is less seasonal, when analyzed on a yearly basis, although the total number of shootings spikes in summer as well. The hourly data shows a spike at 0 o'clock, different to the maximum of the sum of all the offenses.

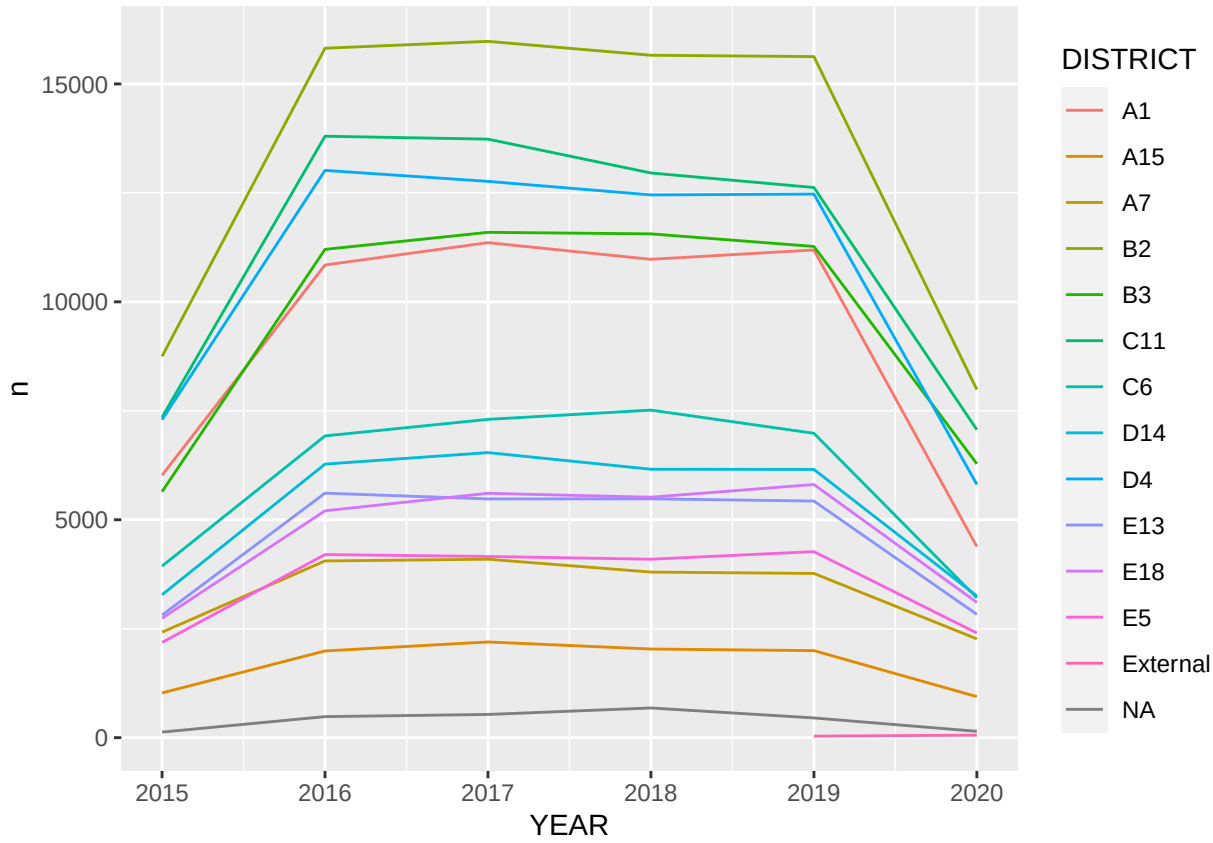
2.2. Location



The location of all accumulated offenses occurred during the observation period of the data matches the shape of the municipal boundaries of Boston quite well. That means that before stratifying the data more, no prior conclusion can be drawn from the plot.

Next, the distribution of offenses per district is plotted and analysed:

```
## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
```



As can be seen in the graph, the distribution does not change over the years. Districts with a high number of offenses in 2015 also tend to have a relatively high number of crimes in 2020.

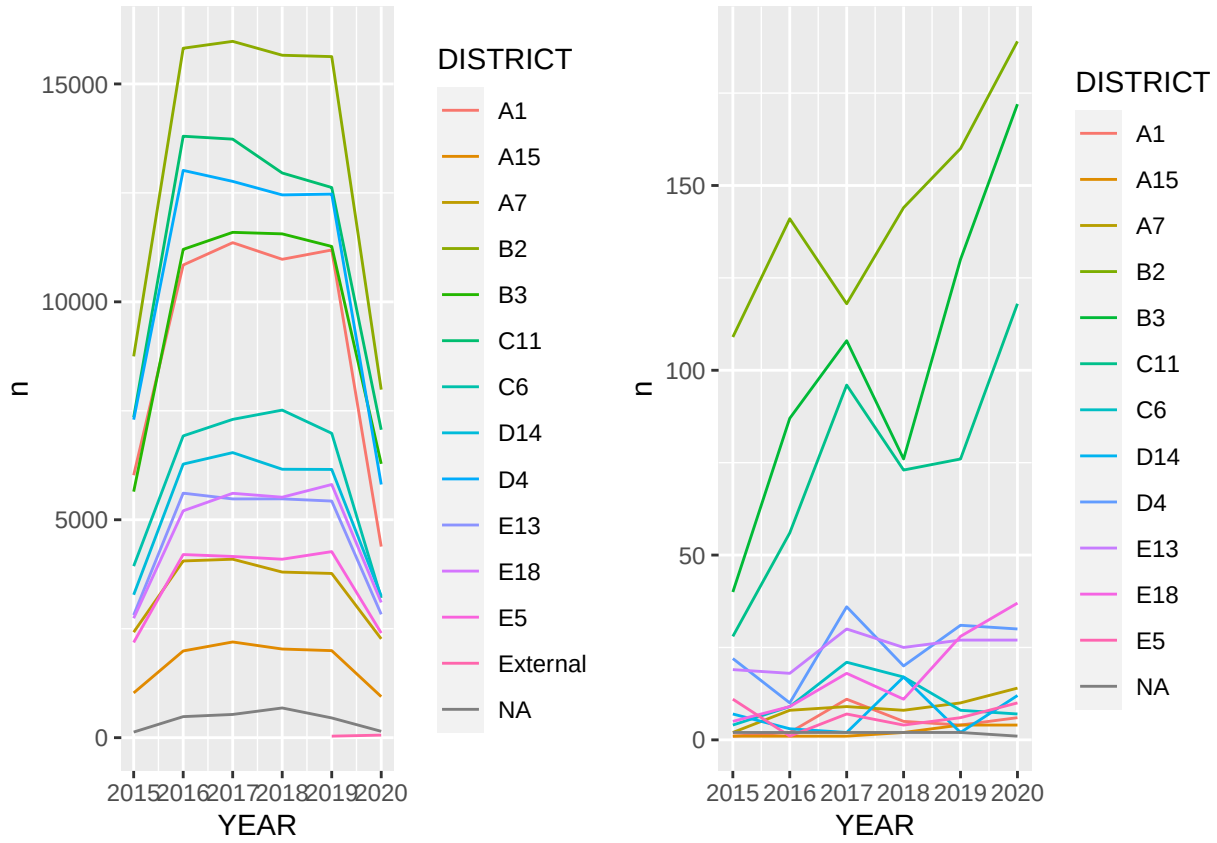
```
#type by district

#per year
crime_pYD <- crimedata %>% group_by(YEAR,DISTRICT) %>% summarise(n=n())
```

```
## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
```

```
cpy <- crime_pYD %>% ggplot(aes(YEAR,n,color=DISTRICT))+
  geom_line()
scpy <- crimedata%>%
  filter(SHOOTING==1)%>%
  group_by(YEAR,DISTRICT) %>%
  summarise(n=n())%>%
  ggplot(aes(YEAR,n,color=DISTRICT))+
  geom_line()
```

```
## `summarise()` regrouping output by 'YEAR' (override with `.groups` argument)
```

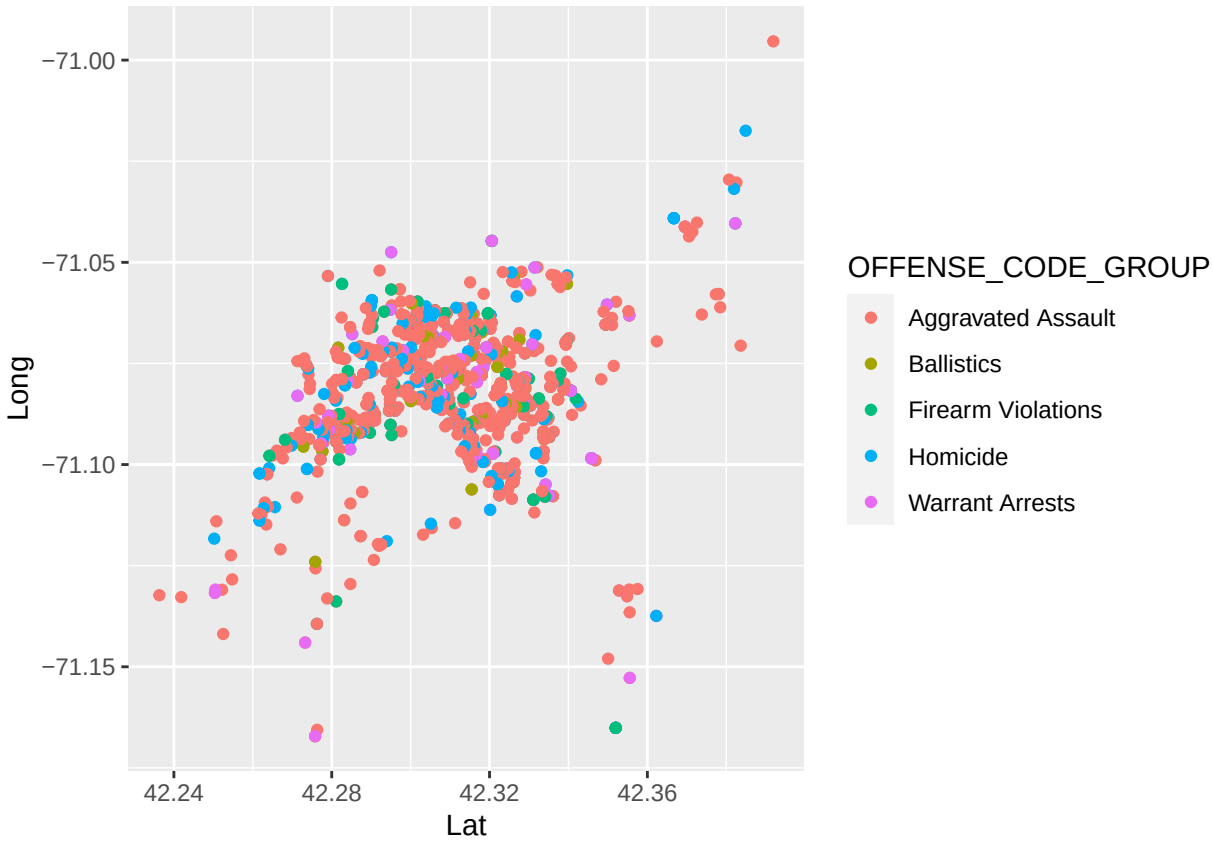
```
grid.arrange(cpy,scpy,nrow=1)
```



As can be seen in the graph, the shooting crimes occure mostly in the districts B2,B3 and C11 and they are increasing over the years, while crimes in general are more broadly distributed and stay more or less constant over time.

2.3. Types of crime

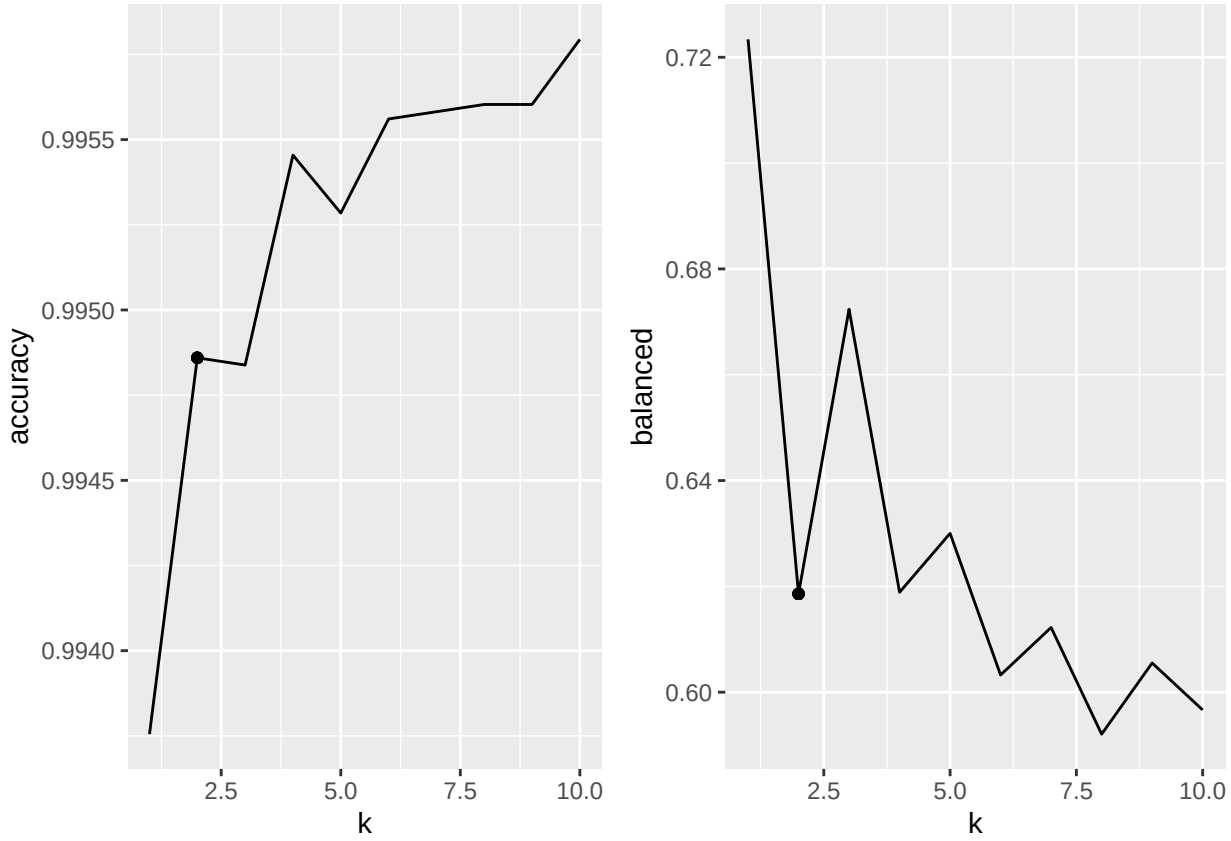
```
## # A tibble: 5 x 3
##   OFFENSE_CODE_GROUP     n      p
##   <chr>              <int>  <dbl>
## 1 Aggravated Assault   840 0.327
## 2 Homicide             223 0.0868
## 3 Warrant Arrests      140 0.0545
## 4 Firearm Violations   130 0.0506
## 5 Ballistics            82 0.0319
```



The main crimes related to shootings that account for over 50% of the shootings in boston are listed above. When plotted, it becomes obvious that these crimes are more common in the city-center than in the sub-urbs.The main crime connected to shootings is the aggravated assault.

III. KNN-Model

In continuation, the dataset will be divided into test and training set in order to be able to assess the accuracy of the dataset properly. Then, in a next step, a knn model is estimated for a set of values for k in order to find the one that optimizes the accuracy. In theory, the best model that is found by this method could be applied in order to predict the involvement of shootings in crimes in the future. Nevertheless, this will not be done in this context, i.e. no hold-out set will be predicted. The optimal number of k is found by comparing the predictions of each model to the real data of the test set. For each row of the test set, the KNN algorithm finds the k-nearest neighbors according to the criteria fixed in the creation of the training-based model and decides by majority vote whether or not this group is likely to have a shooting involved or not. In R, the knn3 function is used in order to train the algorithm and the predict function with the type "prob" is used in order to estimate the values for the test set. That means that in some cases, the output will be a value between 0 and 1 and a cutoff value has to be specified. Since in the context of shootings it might be more important to predict a shooting that actually happened as such, than predicting a shooting that did not happen, sensitivity is valued over specificity. Therefore, the cutoff value is fixed at >0.5, the lowest reasonable percentage.



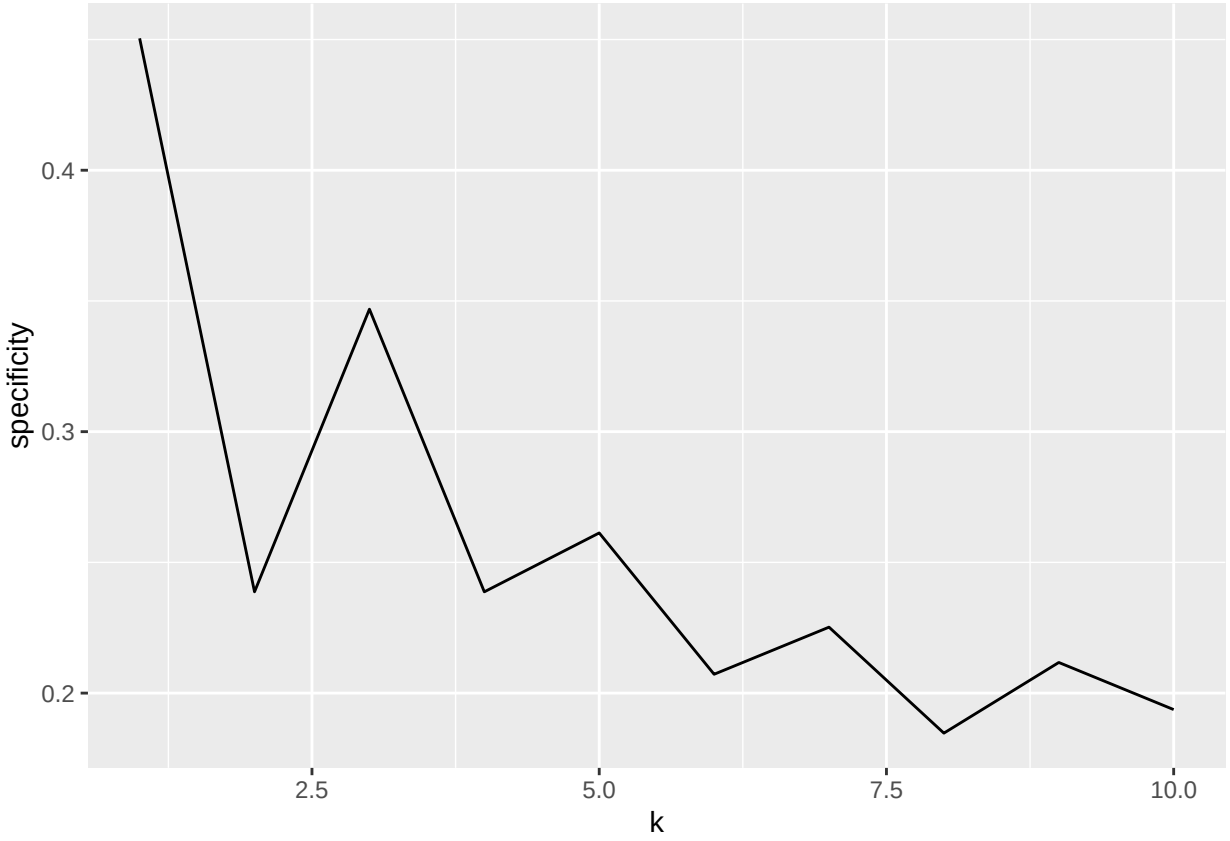
As can be seen in the plot, the optimal value for k for the balanced accuracy would be 1 or 3. Nevertheless, the optimal value of k is decided by the accuracy in this case. This is done because the balanced accuracy is the average of sensitivity and specificity, meaning that in this case (desired high sensitivity and low specificity) it is not a good measure. The accuracy in turn is the percentage of true predictions of both cases and more suited to assess the model. Since the accuracy always increases for any increase in k, the k chosen should be the one for which the increase is maximal. According to the plot this is the case for k=2.

```
## Confusion Matrix and Statistics
##
##
##         0     1
##   0 46783   169
##   1    73    53
##
##                Accuracy : 0.9949
##                  95% CI : (0.9942, 0.9955)
##     No Information Rate : 0.9953
##     P-Value [Acc > NIR] : 0.9146
##
##                   Kappa : 0.3022
##
##  Mcnemar's Test P-Value : 1.016e-09
##
##             Sensitivity : 0.9984
##             Specificity : 0.2387
##          Pos Pred Value : 0.9964
```

```
##          Neg Pred Value : 0.4206
##              Prevalence : 0.9953
##          Detection Rate : 0.9937
##    Detection Prevalence : 0.9973
##       Balanced Accuracy : 0.6186
##
##        'Positive' Class : 0
##
```

The test set results of the final model with k=2 show a very high sensitivity (0.9984420) and a very low specificity (0.2387387). Unfortunately, a look at the confusion matrix reveals that the function automatically calculated the sensitivity using value=0 as the positive value. Thus, in the context of shootings this means that an incident without shooting almost always gets classified as such, although offenses that do involve a shooting in many cases are not classified as such. That means that in order to get the best model, one should have selected according to the specificity and not to the overall accuracy. Hence, the main underlying problem is that the data contains a lot of negative values and only a few positive ones.

```
accresult %>% ggplot(aes(k,specificity))+
  geom_line()
```



According to the data, k=1 would maximize the specificity, i.e. the rate of shootings that are identified as such. Althoug k would equal 1, an overfitting is most likely not given, since the true values are binary. Nevertheless, the final specificity rate would only be ~.45, meaning that shootings would be identified only in ~45% of the cases. That is why the model will not be analyzed further.

IV. Conclusion

This project built a KNN model in order to predict shootings. The optimal model found was a KNN model with k=1. Unfortunately the model is most likely of no use since it fails in almost 55% of the test cases with shootings to predict the involvement of such. Further analysis could try do model the data with a different method in order to get a better result.