San Francisco Crime Research

1. Introduction

San Francisco is a city in the USA in the state of California. It is a world tourist destination known for its cold summer fogs, steep hills and a mix of Victorian and modern architecture. For most of its history, it has been the most populated and important city in the San Francisco Bay Area. But if the city is large, it is likely to have a high crime rate. This is what we will investigate in this work.

The idea of this study is to help people planning their journey in the San Francisco. It will help them to choose the right route where the crime is low.

2. Data acquisition and cleaning

We used the dataset from site San Francisco police department. It contains data from 2003 to 2018 (Site: **https://datasf.org/**). There are 2215024 records of crime and 33 different features.

Firstly, we looked the dataset and found the blanks. There were a lot of blanks and we had to solve this problem. Secondly, we selected the main features:

- **IncidntNum**
- **Category**
- **Descript**
- **DayOfWeek**
- **Date**
- **Time**
- **PdDistrict**
- **Address**
- **X**
- **Y**
- **PdId**

Then we made a new features based the old features:

- **Day**
- **Month**
- **Year**
- **Year_month**
- **Hour**

Finally, we dropped the blanks. These features contains few blanks and we just dropped it without lose information.

Below can you see the number of passes after dropping.

```
1  df_import.isna().sum()
```

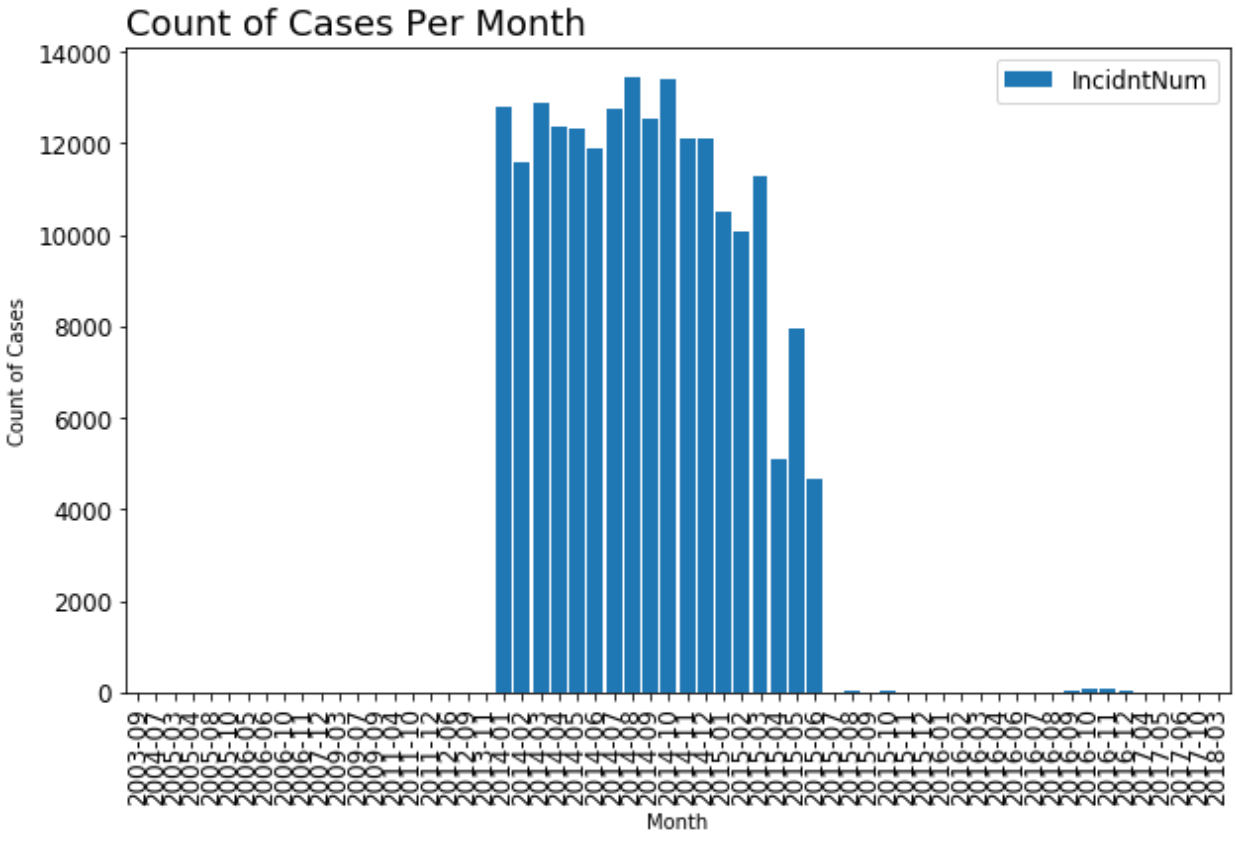
```
IncidntNum     0
Category       0
Descript       0
DayOfWeek      0
Date           0
Time           0
PdDistrict     1
Address        0
X              0
Y              0
PdId           0
Day            0
month          0
year           0
year_month     0
hour           0
dtype: int64
```

3. Exploratory Data Analysis

After preparing the data we decided to analyze the data- we visualized a different charts.

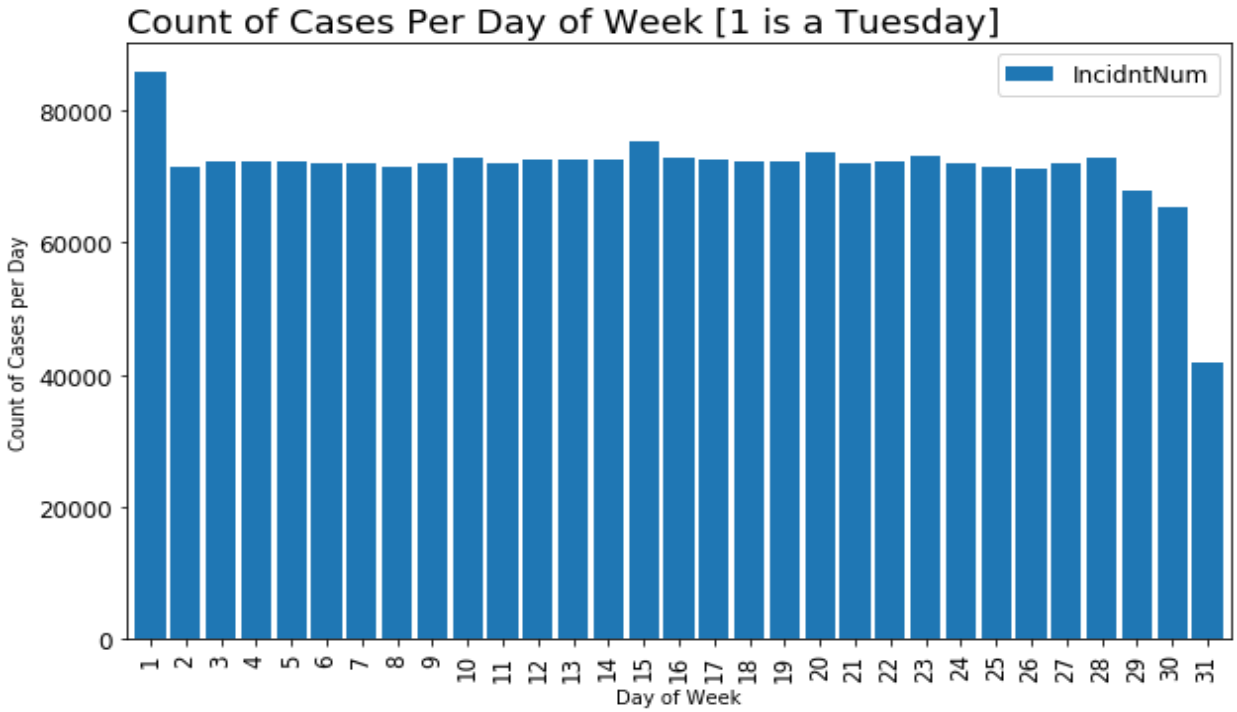
a) Number of crimes per month

We used matplotlib to visualize our charts. Below you can see the dependence of crimes on each month.



b) Number of crimes occurring on each day

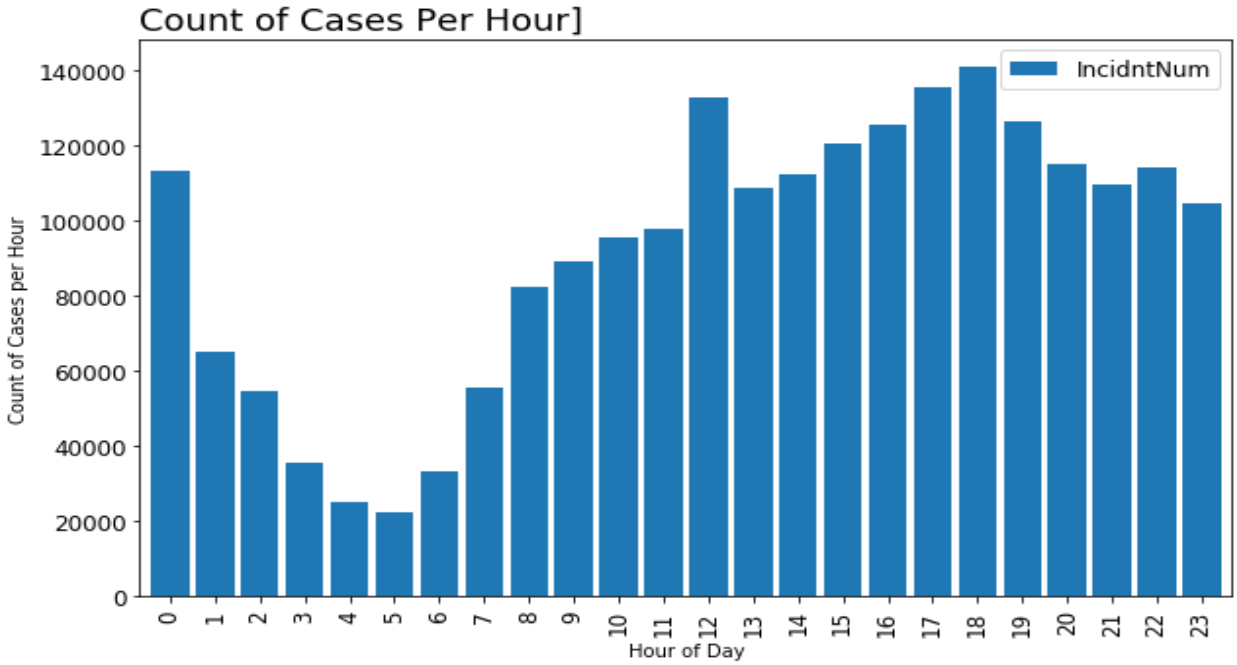
Below you can see the dependence of crimes on each day.



We can see that the most crime day is the first day of month and it is Tuesday.

c) Number of crimes occurring on each hour

Below you can see the dependence of crimes on each hour.



Here you can see that the crimes are committed most often at lunchtime or in the evening.

4. Predictive Modeling

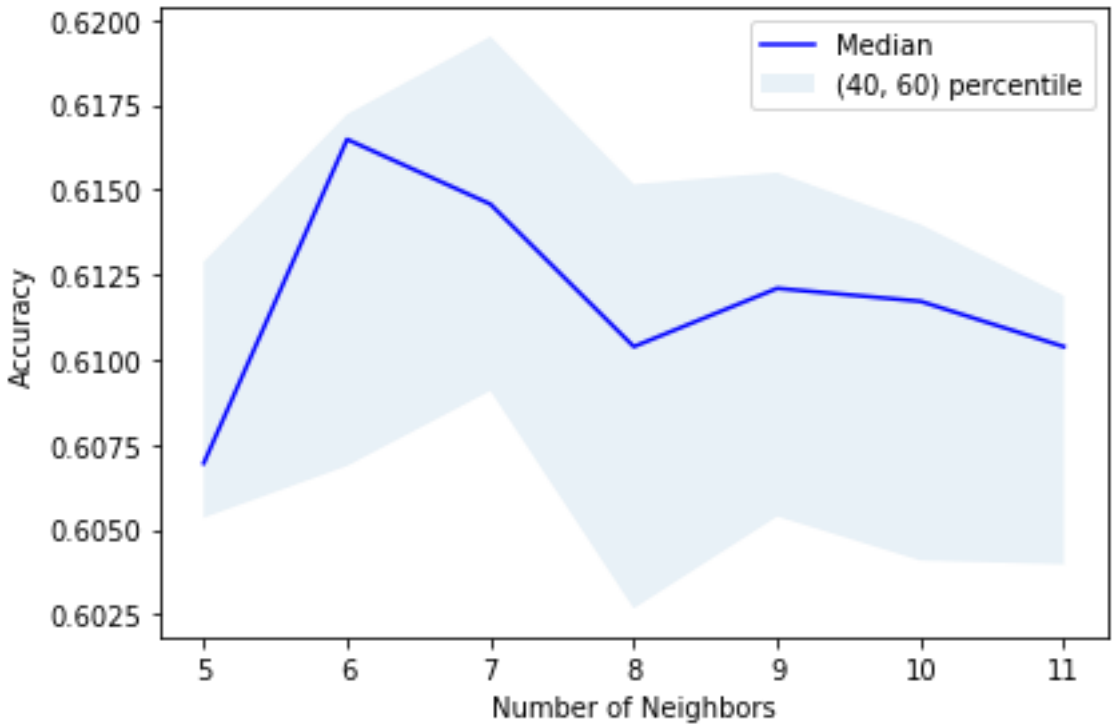
There are of 4 different models to predict new crimes:

- **Logistic Regression**
- **Random forest**
- **kNN (k Nearest Neighbor)**
- **Decision Tree**

Firstly I selected the best parameters for models (e.x. the number of neighbors for kNN), and then used these parameters. I used the accuracy metric to control the models.

a) k Nearest Neighbor

Below you can see the selecting of the best number of neighbors.



The best number of neighbors is 6.

Here we use this number to build the model.

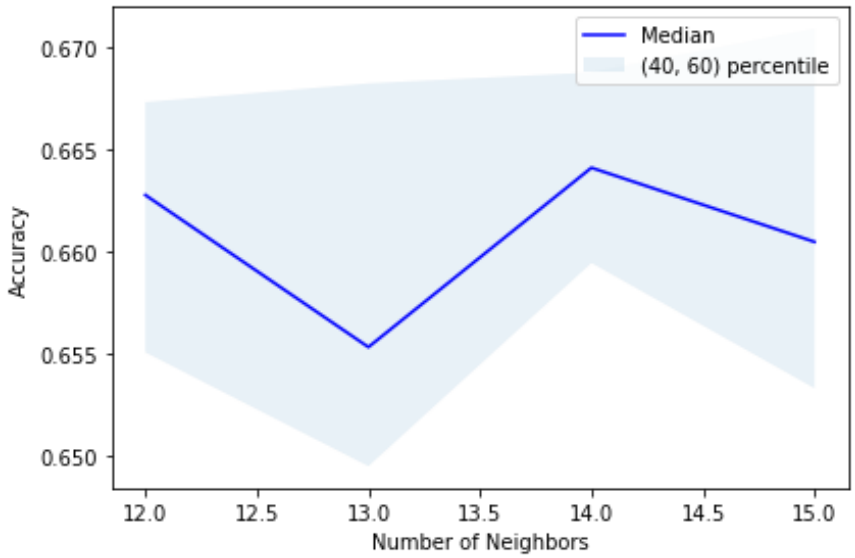
```
from sklearn.metrics import f1_score
from sklearn.metrics import accuracy_score
#K Nearest Neighbor(KNN)
model = KNeighborsClassifier(n_neighbors = 6)
model.fit(X_train, y_train)
y_pred=model.predict(X_test)
accuracy_score(y_test, y_pred)
```

```
0.51
```

The accuracy score is equal 0.51. It is low, we will not use this model.

b) Random forest

Below you can see the selecting of the best estimator.



The best number of neighbors is 14.0.

Here we use this number to build the model

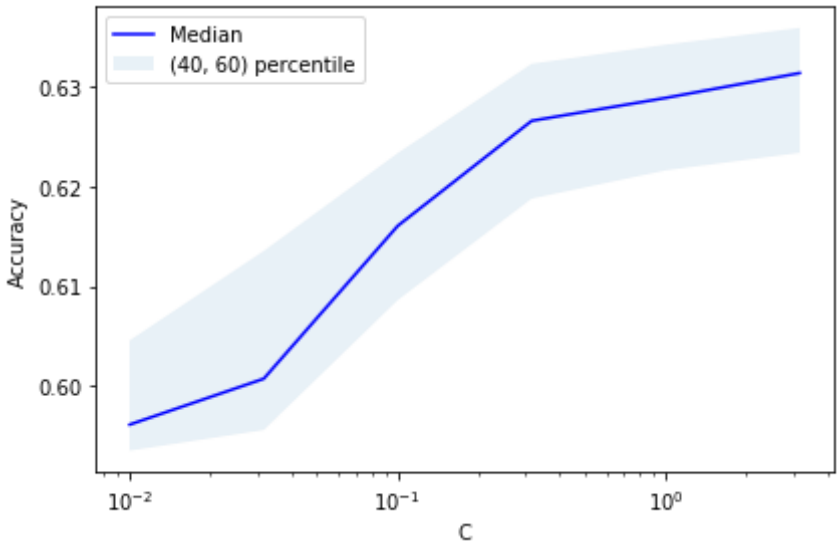
```
#Random Forest
model = RandomForestClassifier(n_estimators = 14, max_features = 'sqrt')
model.fit(X_train, y_train)
y_pred=model.predict(X_test)
accuracy_score(y_test, y_pred)
```

```
0.89
```

The accuracy score is equal 0.89. It is the highest score. This model is convenient to us.

c) Logistic Regression

Below you can see the selecting of the best estimator.



The best C estimator is 3.1622776601683795.

Here we use this number to build the model.

```
#Logistic Regression
model = LogisticRegression(C = 3.1622776601683795  , solver = 'liblinear')
model.fit(X_train, y_train)
y_pred=model.predict(X_test)
accuracy_score(y_test, y_pred)
```

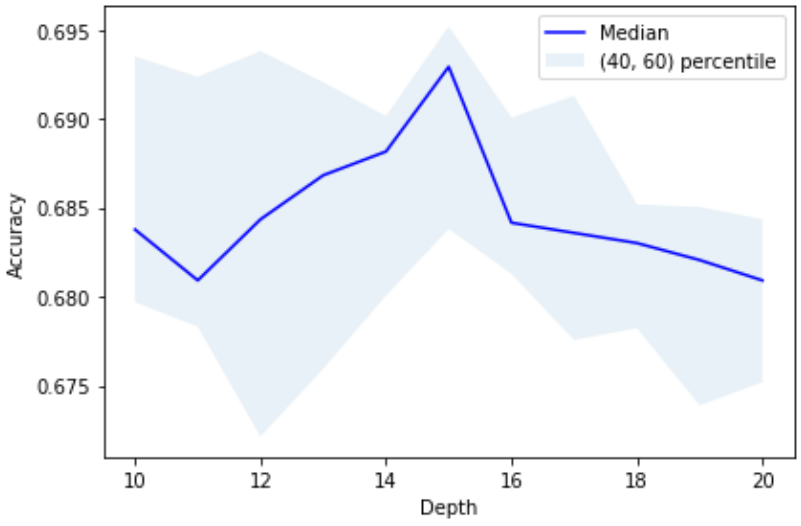
```
C:\Users\slava\Anaconda3\lib\site-packages\sklearn\linear_model\logistic.py:469:
nged to 'auto' in 0.22. Specify the multi_class option to silence this warning.
  "this warning.", FutureWarning)
```

```
0.3389090909090909
```

The accuracy score is equal 0.34. It is low, we will not use this model.

d) Decision Tree

Below you can see the selecting of the best depth.



The best depth is 15.

Here we use this number to build the model.

```
#Decision Tree
model = DecisionTreeClassifier(criterion = "entropy", max_depth = 100)
model.fit(X_train, y_train)
y_pred=model.predict(X_test)
accuracy_score(y_test, y_pred)
```

```
0.6321
```

The accuracy score is equal 0.6321. It is low, we will not use this model.

5. Conclusions

In this study, I analyzed the history of crimes in Chicago. I found dependencies between crimes and hours or days. After that I build a few models to predict a new crimes.

I hope my this research will help people planning their journey in the San Francisco. If I visit San Francisco, firstly, I will use this my research**.** And some people can use this program to analyze and predict the crimes in their city (as for me I can use it to analyze cities in my country).