Classification Model Comparison: NYC Shooting Incidents

Executive Summary

Classification Model Comparison

- NYC Shooting Incident Report (Historic)
  - Classify Incidents as Murders Based on Incident Attributes
  - Accuracy / Computational Resources
- EDA
  - NA / Content
  - Feature Mapping (Multi-value to Binary)
- Models
  - K Nearest Neighbors
  - Random Forest
  - Gradient Boosting
- Evaluation
  - Tables / Plots
- Discussion
  - Expected vs Surprise
- Conclusion
  - Top Performer: Gradient Boosting

Conclusion: Gradient Boosting Best Performer

Project Description

Analyze the NYPD Shooting Incident Data data set in an attempt to classify shooting incidents as murders.

- Data Set
  - NYPD Shooting Incident Data (Historic)
  - Every shooting incident that occurred in New York City from 1/1/2006 through 12/31/2021
  - Importance – Situational Avoidance, Murder Incident Reduction
  - Location: https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic
- Models (sklearn)
  - K Nearest Neighbors
  - Random Forest
  - Gradient Boosting
  - Varying Conditions
- Metrics
  - Classification Accuracy
  - Run Time
- Evaluation
  - Histograms
  - Plots
  - Tables

Exploratory Data Analysis and Cleansing

- High Level
  - 25,596 Observations
  - 19 Features
  - Type → OCCUR_DATE, OCCUR_TIME From Object to Date Time

```
INCIDENT_KEY                int64
OCCUR_DATE                 object
OCCUR_TIME                 object
BORO                       object
PRECINCT                    int64
JURISDICTION_CODE         float64
LOCATION_DESC              object
STATISTICAL_MURDER_FLAG      bool
PERP_AGE_GROUP             object
PERP_SEX                   object
PERP_RACE                  object
VIC_AGE_GROUP              object
VIC_SEX                    object
VIC_RACE                   object
X_COORD_CD                float64
Y_COORD_CD                float64
Latitude                  float64
Longitude                 float64
Lon_Lat                    object
dtype: object
```

- NA Analysis
  - 4 Features (LOCATION_DESC, PERP_AGE_GROUP, PERP_SEX, PERP_RACE)
  - 9300+ Observations
  - >5% - Throw

```
INCIDENT_KEY 0 0
OCCUR_DATE 0 0
OCCUR_TIME 0 0
BORO 0 0
PRECINCT 0 0
JURISDICTION_CODE 2 0
LOCATION_DESC 14977 0
STATISTICAL_MURDER_FLAG 0 0
PERP_AGE_GROUP 9344 0
PERP_SEX 9310 0
PERP_RACE 9310 0
VIC_AGE_GROUP 0 0
VIC_SEX 0 0
VIC_RACE 0 0
X_COORD_CD 0 0
Y_COORD_CD 0 0
Latitude 0 0
Longitude 0 0
Lon_Lat 0 0
```

- Irrelevant
  - Throw → INCIDENT_KEY PRECINCT, JURISDICTION_CODE, Coordinates

- Redundant
  - Throw → X_COORD_CD, Y_COORD_CD, Latitude, Longitude, Lon, Lat

Exploratory Data Analysis (EDA)

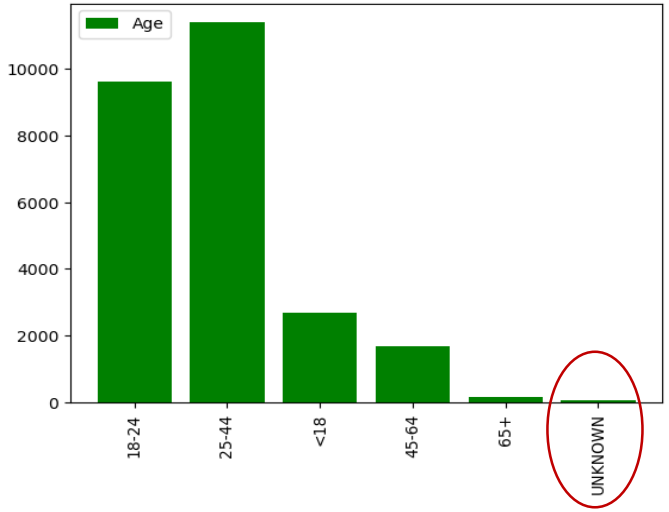
- Correctness
  - VIC_AGE_GROUP, VIC_RACE, VIC_SEX → U and UNKNOWN Values
  - <.5% of Observations → Removed

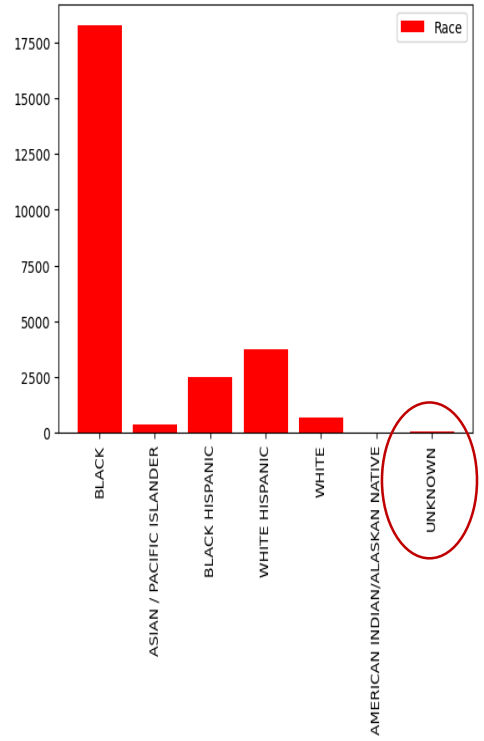
```
Unknown Age Rows: 60
Percent of Rows: 0.0023441162681669013
Unknown Sex Rows: 11
Percent of Rows: 0.0004297546491639319
Unknown Age Rows: 65
Percent of Rows: 0.0025394592905141427
```

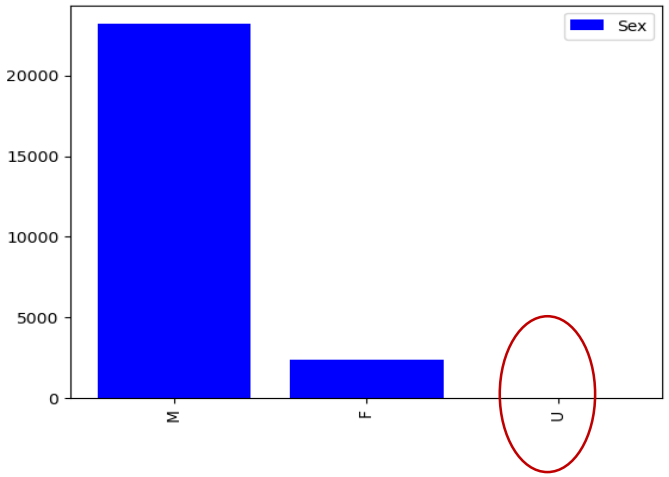
- Multi-Value to Binary Conversion
  - 19 – 61 Features

```
INCIDENT_KEY                int64
OCCUR_DATE                 object
OCCUR_TIME                 object
BORO                       object
PRECINCT                    int64
JURISDICTION_CODE         float64
LOCATION_DESC              object
STATISTICAL_MURDER_FLAG      bool
PERP_AGE_GROUP             object
PERP_SEX                   object
PERP_RACE                  object
VIC_AGE_GROUP              object
VIC_SEX                    object
VIC_RACE                   object
X_COORD_CD                float64
Y_COORD_CD                float64
Latitude                  float64
Longitude                 float64
Lon_Lat                    object
dtype: object
```

```
0   murder                          25482 non-null  bool
1   sex                             25482 non-null  bool
2   d0                              25482 non-null  bool
3   d1                              25482 non-null  bool
4   d2                              25482 non-null  bool
5   d3                              25482 non-null  bool
6   d4                              25482 non-null  bool
7   d5                              25482 non-null  bool
8   d6                              25482 non-null  bool
9   h0                              25482 non-null  bool
10  h1                              25482 non-null  bool
11  h2                              25482 non-null  bool
12  h3                              25482 non-null  bool
13  h4                              25482 non-null  bool
14  h5                              25482 non-null  bool
15  h6                              25482 non-null  bool
16  h7                              25482 non-null  bool
17  h8                              25482 non-null  bool
18  h9                              25482 non-null  bool
19  h10                             25482 non-null  bool
20  h11                             25482 non-null  bool
21  h12                             25482 non-null  bool
22  h13                             25482 non-null  bool
23  h14                             25482 non-null  bool
24  h15                             25482 non-null  bool
25  h16                             25482 non-null  bool
26  h17                             25482 non-null  bool
27  h18                             25482 non-null  bool
28  h19                             25482 non-null  bool
29  h20                             25482 non-null  bool
30  h21                             25482 non-null  bool
31  h22                             25482 non-null  bool
32  h23                             25482 non-null  bool
33  m1                              25482 non-null  bool
34  m2                              25482 non-null  bool
35  m3                              25482 non-null  bool
36  m4                              25482 non-null  bool
37  m5                              25482 non-null  bool
38  m6                              25482 non-null  bool
39  m7                              25482 non-null  bool
40  m8                              25482 non-null  bool
41  m9                              25482 non-null  bool
42  m10                             25482 non-null  bool
43  m11                             25482 non-null  bool
44  m12                             25482 non-null  bool
45  BROOKLYN                        25482 non-null  bool
46  QUEENS                          25482 non-null  bool
47  BRONX                           25482 non-null  bool
48  MANHATTAN                       25482 non-null  bool
49  STATEN ISLAND                   25482 non-null  bool
50  BLACK                           25482 non-null  bool
51  ASIAN / PACIFIC ISLANDER        25482 non-null  bool
52  BLACK HISPANIC                  25482 non-null  bool
53  WHITE HISPANIC                  25482 non-null  bool
54  WHITE                           25482 non-null  bool
55  AMERICAN INDIAN/ALASKAN NATIVE  25482 non-null  bool
56  18-24                           25482 non-null  bool
57  25-44                           25482 non-null  bool
58  <18                             25482 non-null  bool
59  45-64                           25482 non-null  bool
60  65+                             25482 non-null  bool
```

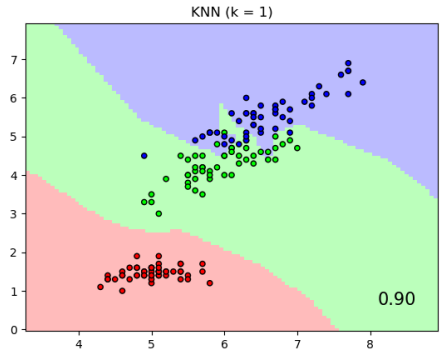






Models

- K Nearest Neighbor
  - sklearn.neighbors
  - KNeighborsClassifier(n_neighbors = k)
  - k = 2 to 22




```
y = classificationData['murder']
X = classificationData.drop(['murder'], axis = 1)
X.info()
y.info()
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.25, random_sta

dfAccuracy = pd.DataFrame()
dfTimes = pd.DataFrame()

#KNN Evaluation
knnStartTime = time.time()

knnScores = [None]
knnTimes =[None]
for k in range(2, 22):
    knn = KNeighborsClassifier(n_neighbors=k)
    knn.fit(X_train, y_train)
    knnPredictions = knn.predict(X_test)
    knnScores.append(metrics.accuracy_score(y_test, knnPredictions))
    knnTimes.append(time.time() - knnStartTime)

dfAccuracy['KNN'] = knnScores
dfTimes['KNN'] = knnTimes
```

Sample Model

- Gradient Boost
  - sklearn.ensemble
  - GradiebntBoostClassifier(n_estimators = e)
  - estimators = [25, 50, 75, 100, 125, 150, 175, 200, 225, 250, 275, 300, 325, 350, 375, 400, 425, 450, 475, 500]

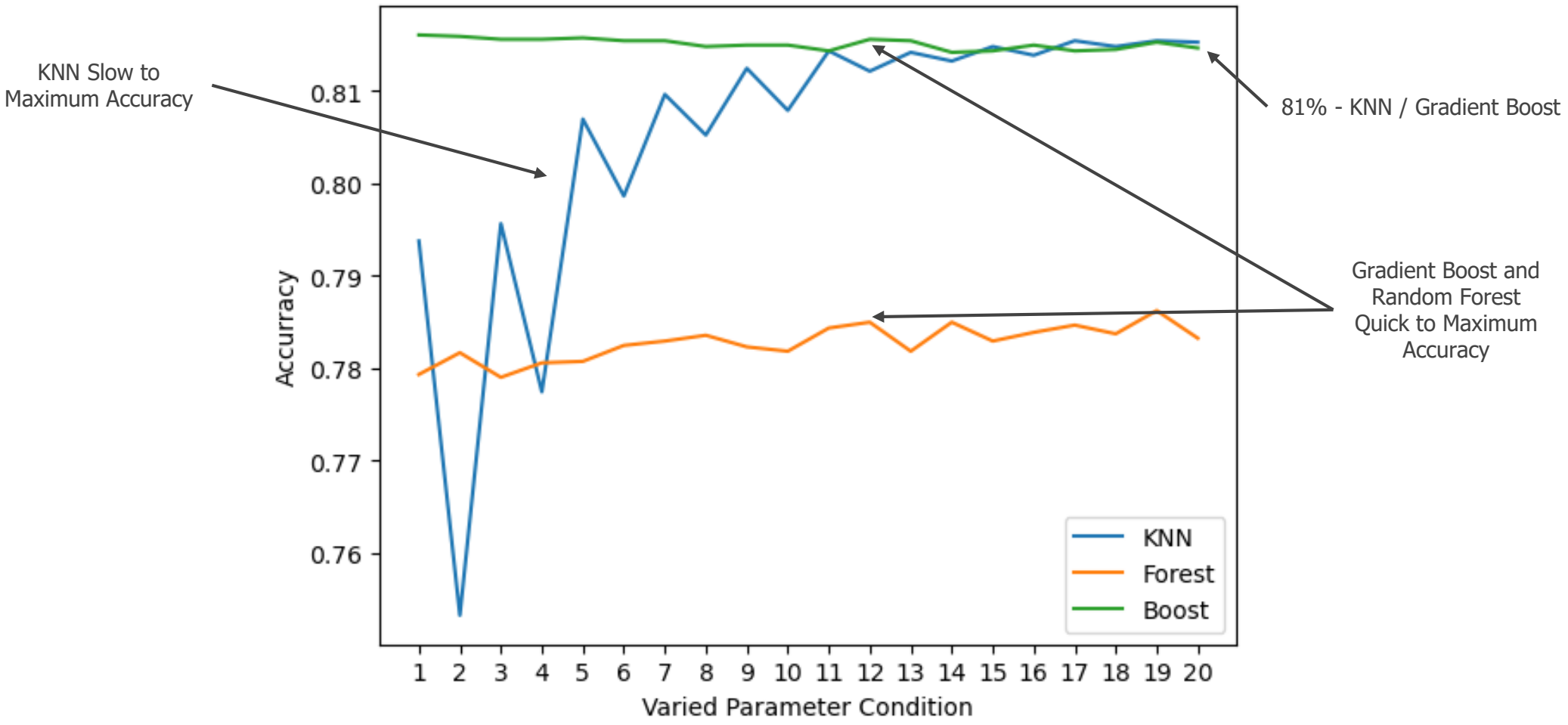
- Random Forest
  - sklearn.ensemble
  - RandomForestClassifier(n_estimators = e)
  - estimators = [25, 50, 75, 100, 125, 150, 175, 200, 225, 250, 275, 300, 325, 350, 375, 400, 425, 450, 475, 500]

Decision tree trained on all the iris features

Images from sklearn Website

Evaluation

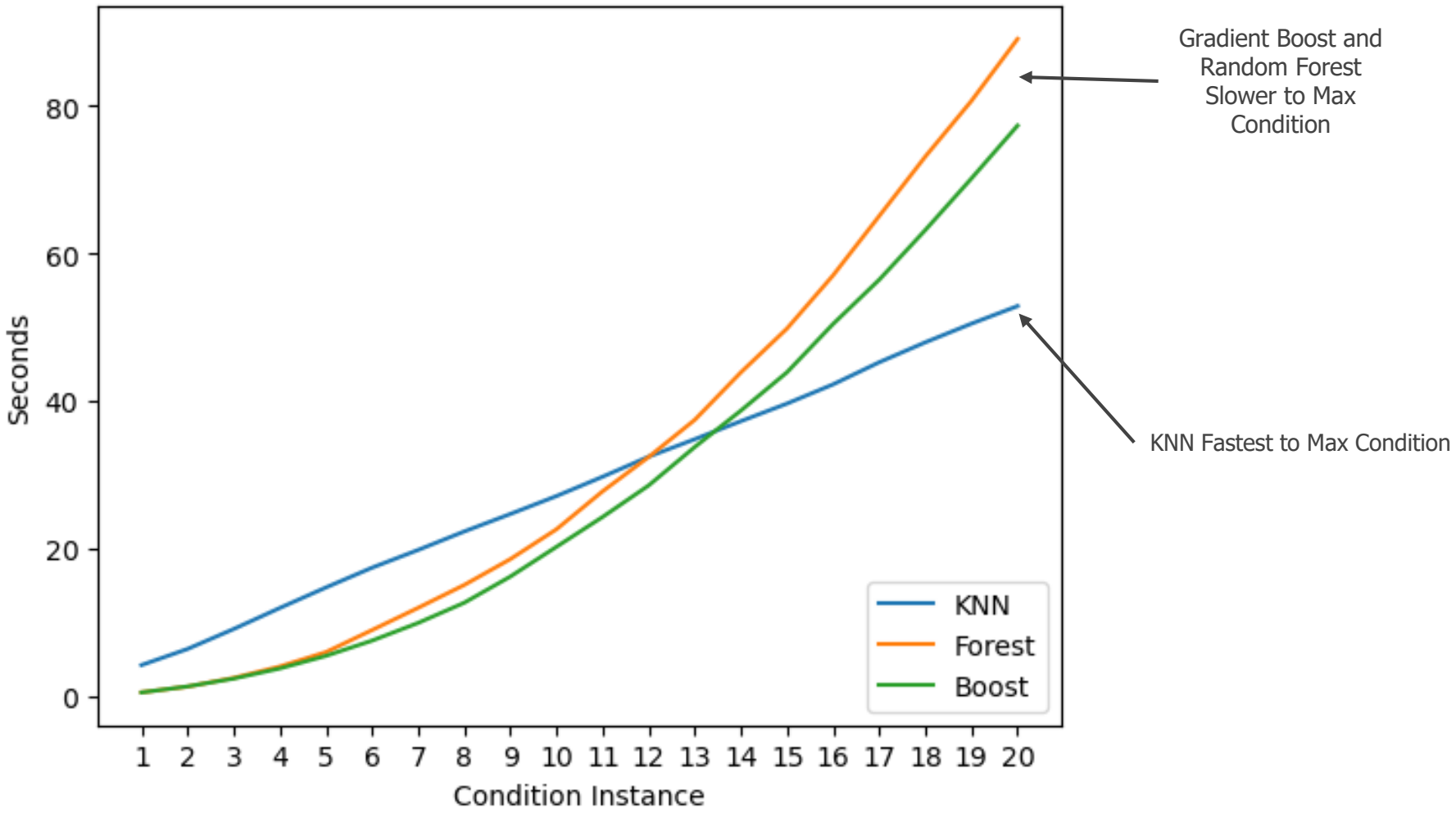
Accuracy



Best: Gradient Boost

Evaluation

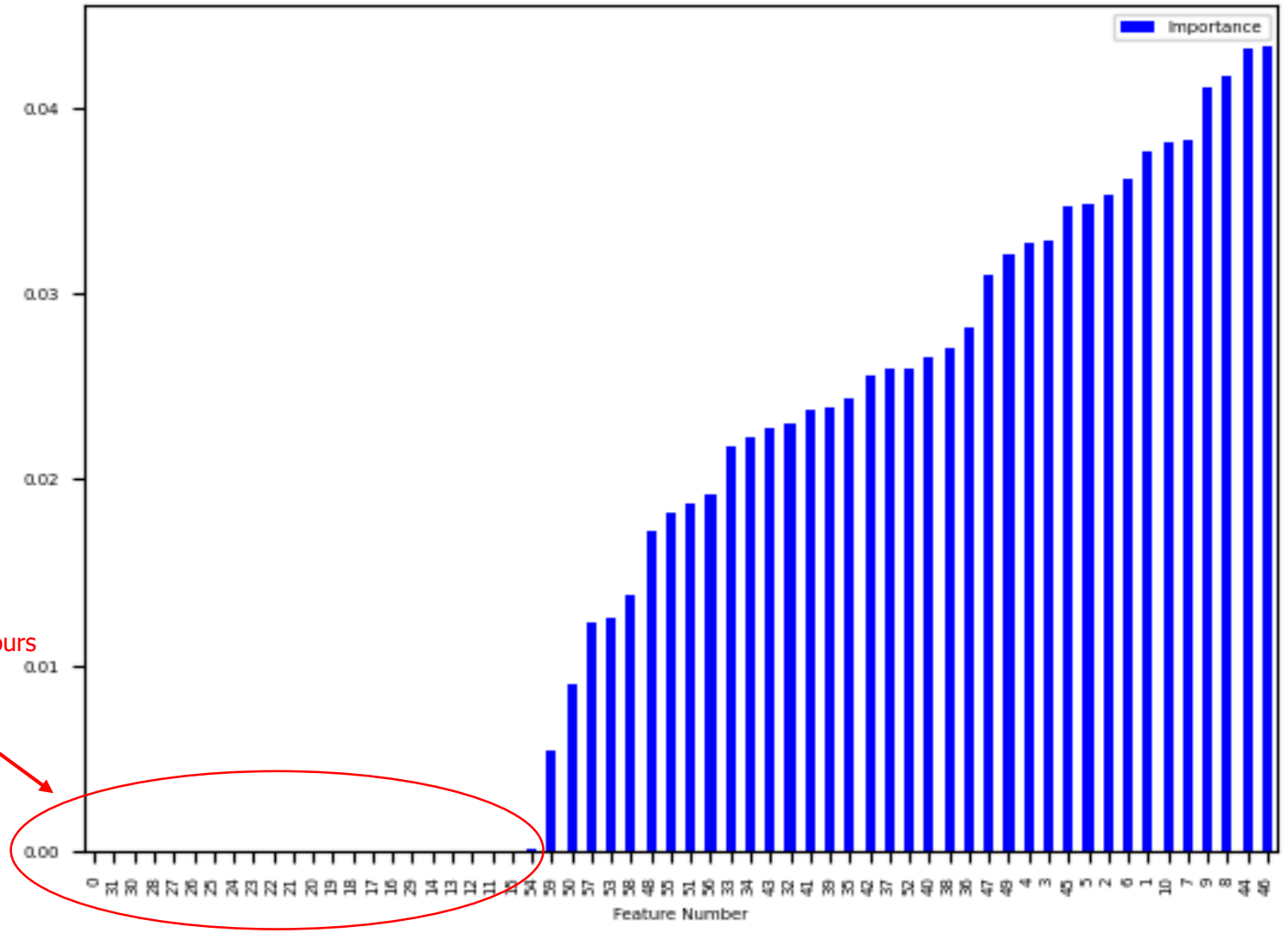
Run Time



Best: KNN

Evaluation

Feature Importance

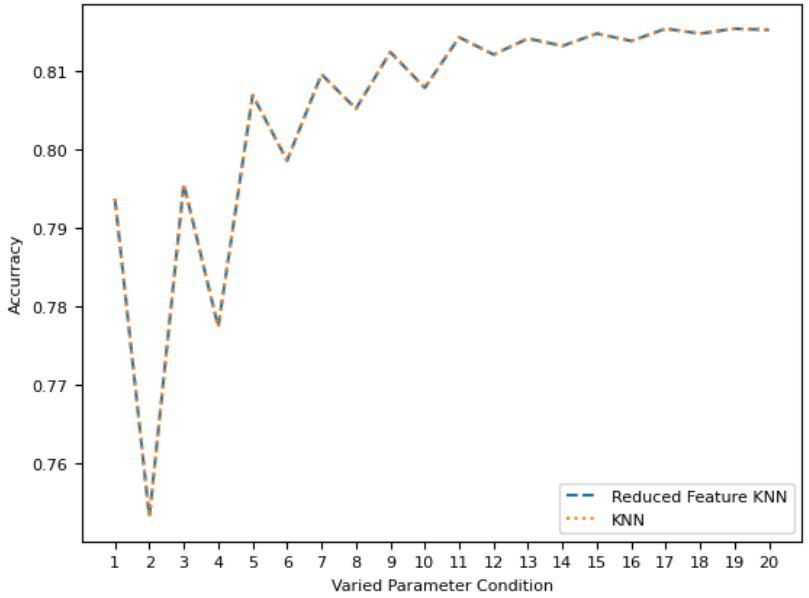


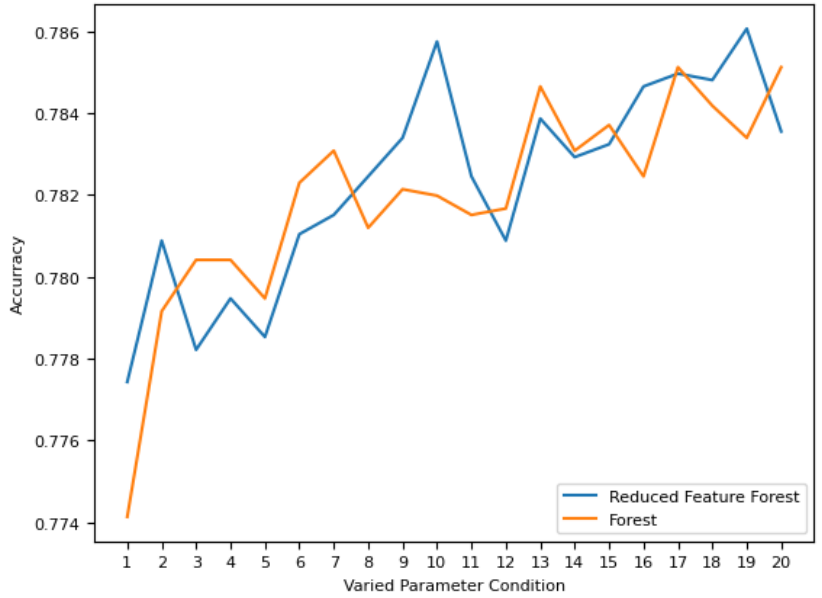
Unimportant Features
- Victim Sex
- Daytime and Evening Hours

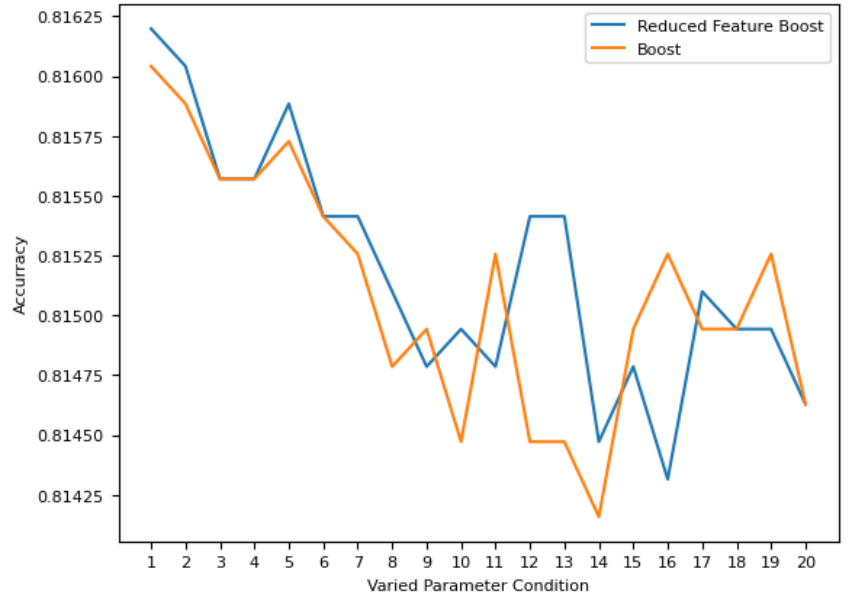
23 Unimportant Features

Evaluation

Accuracy: Reduced Feature Set





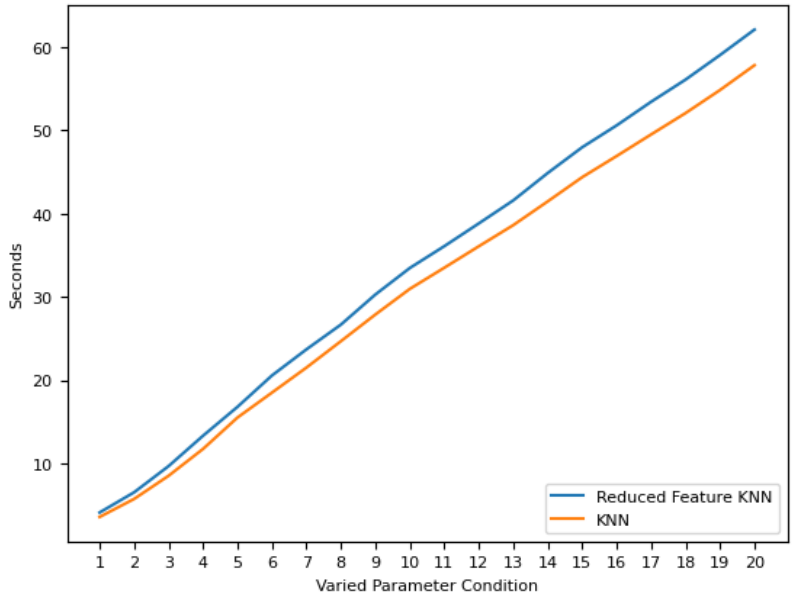


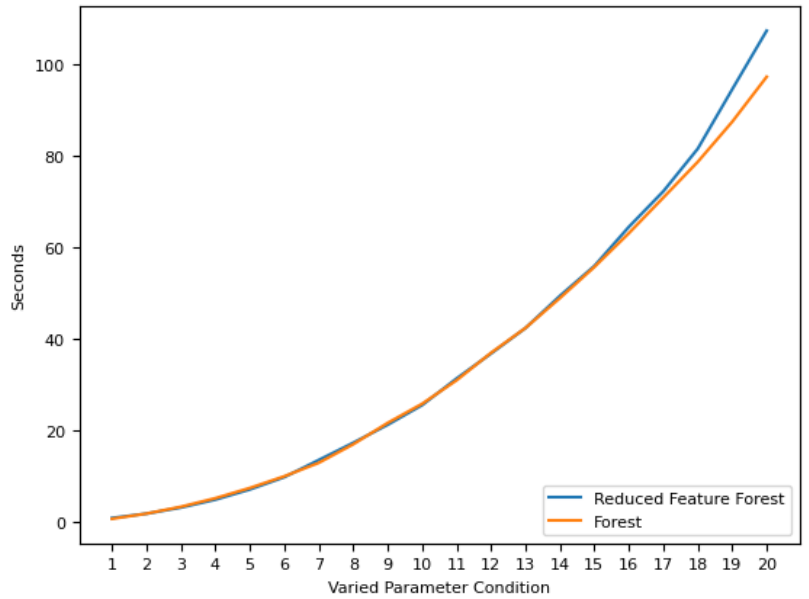
KNN	Random Forest	Gradient Boost
Exact Overlay	Similar Shape Variance Due to Randomness of Individual Tree Features	

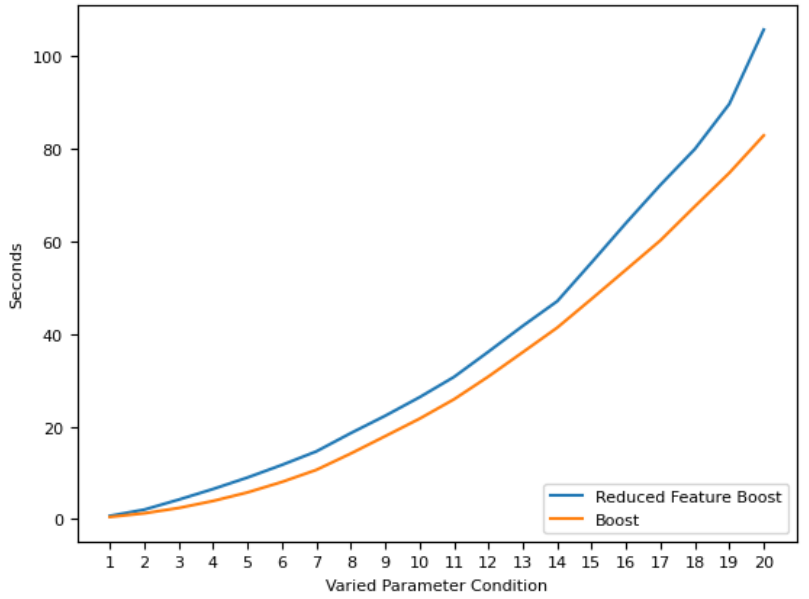
No Impact on Accuracy

Evaluation

Run Time: Reduced Feature Set







KNN	Random Forest	Gradient Boost
↓9 %	↓10%	↓28%

Reduced Run Time on All Models

Discussion

- Accuracy
  - KNN and Gradient Boost Realized 81%
  - Random Forest 78%
  - Observations
    - Tree Models Much Better Accuracy Acceleration
    - KNN Noticeably Slower Accuracy Acceleration
- Feature Reduction
    - Daytime and Evening Hours
    - Sex
- Reduced Set Accuracy
  - No Change in Accuracy
  - Feature Reduction Impact Not As expected
  - Unexpected But Explainable
    - KNN
      - Removed Features Never Part on Nearest Neighbor Set
      - But Included In Calculations
    - Trees
      - Removed Features Eliminated Early
      - Resultant Sets Did Not Change

Discussion

- Run Time
  - KNN Outperformed Others But At Highest K Condition
  - Total Varied Condition Runs Took Longer With Little Gain in Accuracy

- Over All Model Compare
  - KNN Will Get You There If you Have the Time and Resources
  - Random Forest Won't Get You There
  - Gradient Boost Best Overall
    - Quick To Accuracy
    - Minimal Amount of Learners

- Future Work
  - Repeat With Multi-Variable Set
  - Compare

Conclusion

Rating			
Accuracy	Best	Better	Worst
Speed	Best	Worst	Good
Rate	Best	Worst	Good

Image Provided By: https://stock.adobe.com/search?k=open+nutshell&asset_id=35826460

Resources

- Data Source
  - https://catalog.data.gov/dataset/nypd-shooting-incident-data-historic

- GitHub Repository
  - https://github.com/jmskeet/DTSA-5509

Image Provided By: https://stock.adobe.com/search?k=open+nutshell&asset_id=35826460