

## distances

September 14, 2022

```
[7]: import sys
sys.path.append("../")
import pandas as pd
from sklearn.neighbors import BallTree
import numpy as np

def distance():
    dfa = pd.read_csv("calls_for_service_2022_9_14_2022.csv")

    # filter for violent crimes (list of all crimes below)
    dfa = dfa[
        (
            dfa.TypeText.isin(
                [
                    "AGGRAVATED ASSAULT",
                    "CARJACKING",
                    "AGGRAVATED ASSAULT DOMESTIC",
                    "ARMED ROBBERY WITH GUN",
                    "AGGRAVATED BATTERY BY SHOOTING",
                    "AGGRAVATED BATTERY DOMESTIC",
                    "AGGRAVATED BURGLARY",
                    "AGGRAVATED BATTERY BY CUTTING",
                    "AGGRAVATED RAPE",
                    "ARMED ROBBERY",
                    "SIMPLE RAPE",
                    "HOMICIDE BY SHOOTING",
                    "ARMED ROBBERY WITH KNIFE",
                    "AGGRAVATED KIDNAPPING",
                    "SIMPLE ASSAULT DOMESTIC",
                    "AGGRAVATED RAPE UNFOUNDED BY SPECIAL VICTIMS OR CHILD_
↪ABUSE",
                    "AGGRAVATED BURGLARY DOMESTIC",
                    "AGGRAVATED RAPE MALE VICTIM",
                    "HOMICIDE",
                    "HOMICIDE BY CUTTING",
                    "ILLEGAL CARRYING OF WEAPON- KNIFE",
```

```

        "SIMPLE RAPE MALE VICTIM",
        "AGGRAVATED ARSON",
        "SIMPLE RAPE UNFOUNDED BY SPECIAL VICTIMS OR CHILD ABUSE",
    ]
    )
)
]

dfa.loc[:, "TypeText"] = dfa.TypeText.fillna("")
dfa = dfa[~(dfa.TypeText == "")]

locations = (
    dfa.Location.str.lower()
    .str.strip()
    .str.extract(r"point \((-.\+\.+\) (.\+\.+\)\)")
)

dfa.loc[:, "latitude"] = locations[1].fillna("")
dfa = dfa[~((dfa.latitude == ""))]
dfa.loc[:, "latitude"] = dfa.latitude.astype(float)

dfa.loc[:, "longitude"] = locations[0].fillna("")
dfa = dfa[~((dfa.longitude == ""))]
dfa.loc[:, "longitude"] = dfa.longitude.astype(float)

dfb = pd.read_csv("new_orleans_cameras_3_11_2022_french_quarter_filtered.
↪csv")

bt = BallTree(np.deg2rad(dfa[["latitude", "longitude"]].values), ↪
↪metric="haversine")
distances, indices = bt.query(np.deg2rad(np.c_[dfb["latitude"], ↪
↪dfb["longitude"]]))

l = []
for d in distances:
    miles = d * 3958.8
    yards = miles * 1760
    l.append(yards)
    df = pd.DataFrame(l, columns=["distances"])
return df

```

```
[8]: data = distance()
```

```
[9]: # of cameras
data.nunique()
```

```
[9]: distances    34  
     dtype: int64
```

```
[10]: # avg distance of camera (34) to a call for service for a violent crime (1/1/  
      ↪ 2022 - 9/13/2022)  
      data.sum()/len(data)
```

```
[10]: distances    27.840212  
     dtype: float64
```