ETHANWALKER-404-FinalProject

April 11, 2020

```
/home/ethan/anaconda3/lib/python3.7/site-packages/dask/config.py:168: YAMLLoadWarning: calling y
  data = yaml.load(f.read()) or {}
/home/ethan/anaconda3/lib/python3.7/site-packages/distributed/config.py:20: YAMLLoadWarning: cal
  defaults = yaml.load(f)
```

1 Abstract

Machine Learning is becomeing a more prevelant tool in the world of criminal justice. Often it is used to predict who will commit a crime or where crimes may occur. Seldom is it used to regulate the criminal justice system, however. In this project I examine prison inmate data and determine what machine learning techniques are effective at detecting the racial bias that has been shown to exist in this data. In this report I find ———

2 Problem Statement and Motivation

Last semester I took a look a data set containing the information of more than 7.5 million individuals that have been processed by the criminal justice system. I found that racial minorities were more likely to receive extreme sentences, agreeing with existing research around bias in the criminal justice system. In this project I will be exploring the data from a machine learning perspective. My goal is to determine if this data can be classified in such a way that is predictive of race. The idea is that perhaps racial bias can be detected in various systems by seeing how effective different machine learning techniques are at classifying an inmate's by race given their data.

This is an unconventional way to approach criminal justice data with machine learning. Often we see machine learning being used to attempt to determine who might be a criminal or where crimal activity may occur using social media data and other public information, which may include data the government owns, but which is not available to te public. These approaches often ignore or discount the ways that these techniques may disproportionately affect people of color and the poor. Many organizations have made official statements reguarding the use of machine learning in this way, often called predicitive policing. The ACLU for example released a statement listing civil rights related concerns about predicitive policing which was signed by several civil rights origanizations includeing the NAACP [1].

My objective is to go against the predictive policing paradigm and use machine learning to benefit these negatively affected classes of people by using machine learning as a diagnostic tool. If it can be shown that certain machine learning techniques are effective at classifying inmates

by race given incarceration related information, then we can inform policies that will attempt to correct for these systemic racial biases.

3 Data

3.1 Source and Credibility

The data that I will be using in this analysis is from one source. It is a database hosted on Data.gov and maintained by the State of Connecticut Department of Corrections. This source is highly credible because it is a primary source for the data. This organization is an official government agency which collects, maintains, and reports on this data.

3.2 Gathering and Cleaning

All the data which I am using in this report are freely available to the public. Collection and cleaning was relatively simple as the source data was well maintained. The file that I obtained from the Connecticut Department of Corrections is a very well maintained database. The largest issue I had with this file was mild inconsistency with the way in which certain data was encoded (ex. race was encoded as both `WHITE` and `WHITE\t`). The file is

`individuals.csv`.

3.3 About the Data

This data set contains individual information for 7.77 million people that have been processed by the justice system and recorded by the Connecticut Department of Corrections. Each individual is recorded along with their age, gender, race, offense, and sentence length, among other things.

Because there is so much to consider in what is found in the data set, I chose not to feature engineer as to avoid unneeded complexity.

The sample sizes among different races that are found in the Connecticut Department of Justice data are not similar. The sample size for American Indians and Asians is much smaller than that of Whites, Hispanics, and Blacks, hence we may see some irregular outcomes in the analysis related to these racial groups.

```
Sample size for Blacks: 3287596
Sample size for Whites: 2393949
Sample size for Hispanic: 2039297
Sample size for American Indian: 21133
Sample size for Asian: 35660
```

4 Possible questions

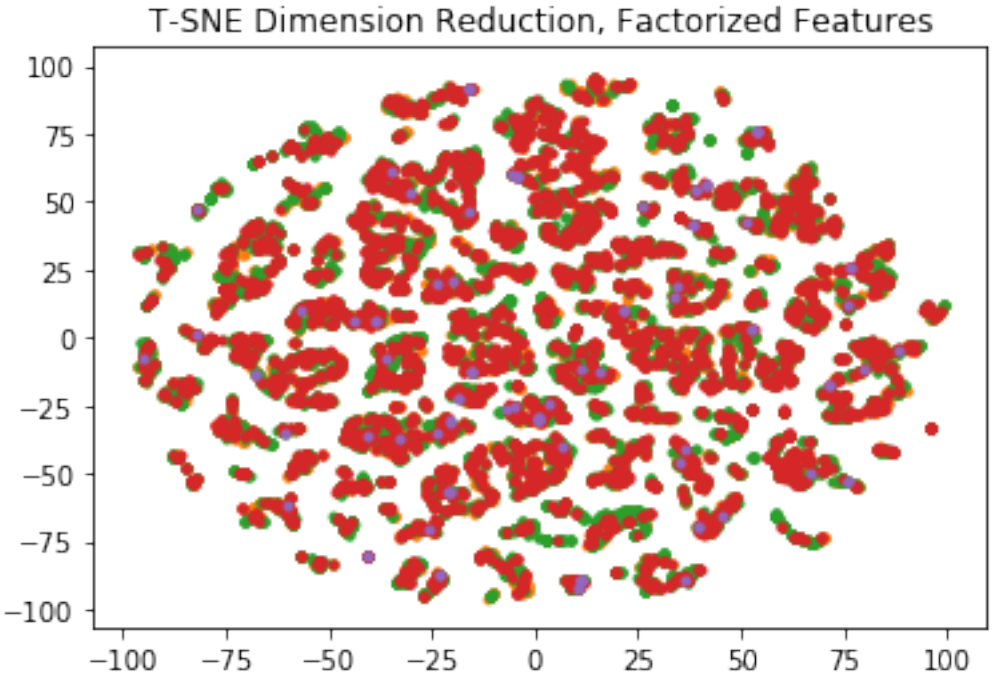
can we use ml techniques to correctly classify this data? which ones fail and why? can we create a predictive model for sentence lengths? should sentencing be offloaded to a ml algorithm? what does it mean to have an effective classfifier for this data set.

5 Methods

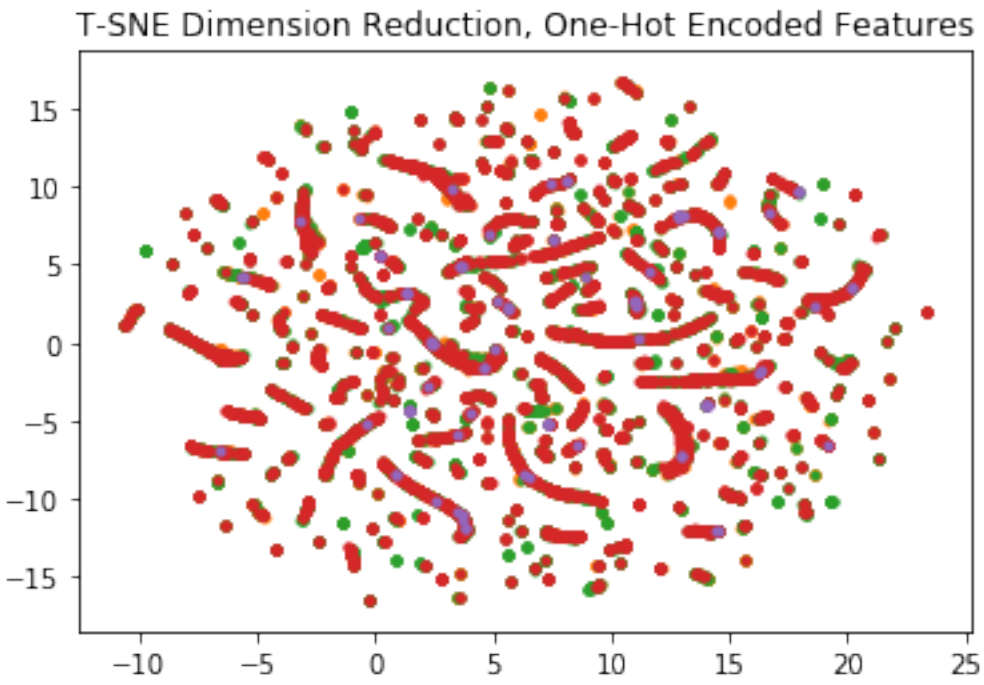
Before I begin discussing the methods that I did use, I will talk about some of the methods that I did not use. There are manay techniques that are not applicable to this data. For example, since this data set is not a time series models like ARMA and HMM are not applicable here.

A method that I attempted to use, but found little success with were dimension reduction tools like PCA, T-SNE, and UMAP. This dimension reduction would have been helpful, especially because once one-hot encoded this data becomes extremely high dimensional. However PCA, T-SNE, and UMAP all failed to create meaningful clusters, so I abandoned the the pursuit of clustering early on. Perhaps some kernel methods would have been helpful in this endeavor, however I could not find a kernel that could create a metric on crimes and I dp not feel qualified to write one myself.

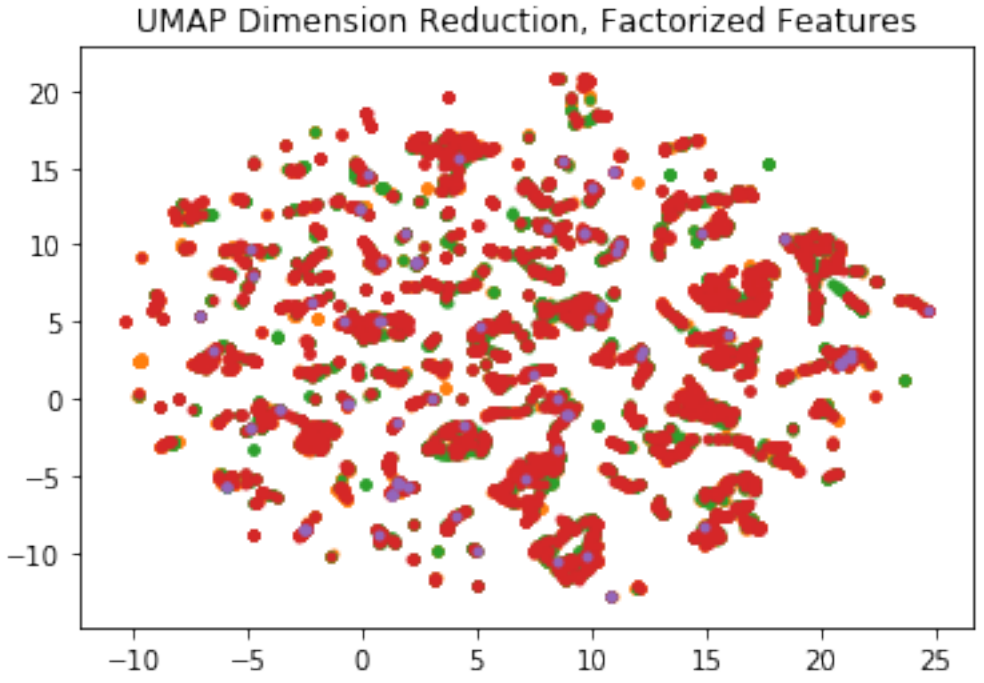
Below I have images of my attempt to use PCA, T-SNE, and UMAP to cluster the data. It is apperent that it simply is not effective.



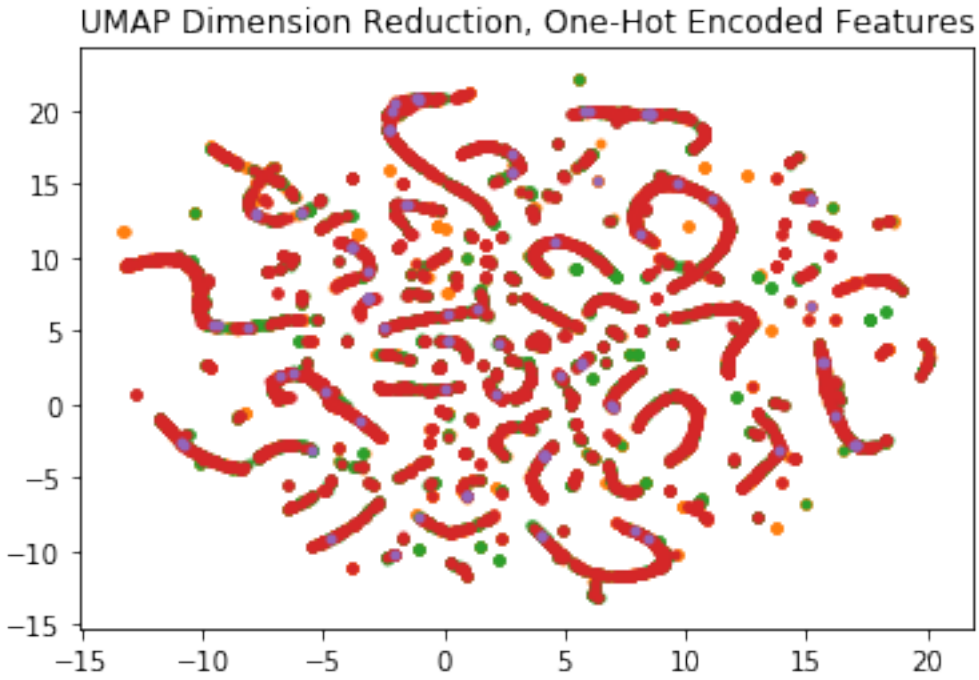

```
/home/ethan/anaconda3/lib/python3.7/site-packages/sklearn/manifold/spectral_embedding_.py:237: U
  warnings.warn("Graph is not fully connected, spectral embedding"
```

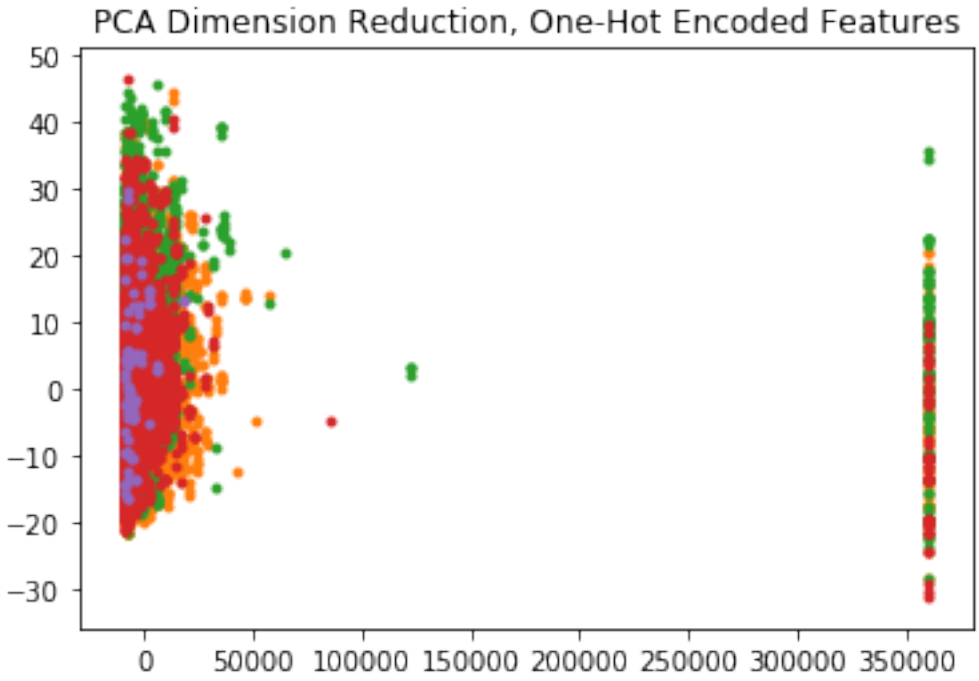



```
/home/ethan/anaconda3/lib/python3.7/site-packages/sklearn/manifold/spectral_embedding_.py:237: U
  warnings.warn("Graph is not fully connected, spectral embedding"
```



```
/home/ethan/anaconda3/lib/python3.7/site-packages/sklearn/manifold/spectral_embedding_.py:237: U
  warnings.warn("Graph is not fully connected, spectral embedding"
```





This is only a small sample of things that I tried to do to get useful clusterings given the dimension reduction, though it is representitive of the results I found

5.1 Ensamble Methods

Ensamble methods seemed imediately like the methods that I should be using in this project. I did not intially expect to successfully classify this data, however as I learned about how ensamble classifiers worked I became more confident that I would get decent results.

The primary reason that I though I would not get good results has to do with how I identified racial bias in the data. In my previous project I determined that there was racial bias in the criminal justice system based on

5.1.1 Random Forest Classifier

In [44]:

In [46]:
```
samp = df.sample(20000)
samp.RACE = pd.factorize(samp['RACE'])[0] + 1
samp.GENDER = pd.factorize(samp['GENDER'])[0] + 1
samp.OFFENSE = pd.factorize(samp['OFFENSE'])[0] + 1
samp.DETAINER = pd.factorize(samp['DETAINER'])[0] + 1
samp.FACILITY = pd.factorize(samp['FACILITY'])[0] + 1
samp_y = samp.RACE
samp_X = samp[['GENDER','AGE','OFFENSE','FACILITY','DETAINER','SENTENCE DAYS']]

param_grid = {
    'n_estimators': np.arange(100,400,20),
    'max_depth': np.arange(10,100,10),
    'max_features': np.arange(1,6)
}



clf = RandomForestClassifier(oob_score=True)

s = time.time()
clf = GridSearchCV(clf, param_grid, scoring=None, cv=5)
clf = clf.fit(samp_X, samp_y)
e = time.time()
print(f'time to train is {(e-s)/60} minutes')

clf = clf.best_estimator_
clf = clf.fit(samp_X, samp_y)
print(f'oob score is {clf.oob_score_}')

with open('RandomForestClf.pickle', "wb+") as f:
    pickle.dump(clf, f)
```

```
time to train is 223.73655876318614 minutes
oob score is 0.70645
```

```
In [4]: with open('RandomForestClf.pickle',"rb") as f:
            clf = pickle.load(f)
        print(clf.oob_score_)
        print(clf.feature_importances_)
```

```
/home/ethan/.local/lib/python3.7/site-packages/sklearn/base.py:306: UserWarning: Trying to unpic
  UserWarning)
```

```
0.70645
[0.01022266 0.26942225 0.2124008  0.17300606 0.0338177  0.30113052]
```

```
/home/ethan/.local/lib/python3.7/site-packages/sklearn/base.py:306: UserWarning: Trying to unpic
  UserWarning)
```

here we can see that the most important features are sentence length, age, and offense. And the high OoB score is promising.

```
Out[5]: 0.420315
```

However the score is not great. Random chance would be .2, so it does do better than chance, though not much better.

5.2 Gradient Descent Boosted Classification

```
In [35]:
```

```
Out[6]: 0.296365
```

```
[0.02090027 0.24716962 0.25775903 0.10433119 0.03748648 0.33235341]
```

```
/home/ethan/.local/lib/python3.7/site-packages/sklearn/base.py:306: UserWarning: Trying to unpic
  UserWarning)
/home/ethan/.local/lib/python3.7/site-packages/sklearn/base.py:306: UserWarning: Trying to unpic
  UserWarning)
```

Here we see the same feature importances as we did with the random forest

```
Out[39]: {'criterion': 'friedman_mse',
          'init': None,
          'learning_rate': 0.34,
          'loss': 'deviance',
```

```
 'max_depth': 4,
 'max_features': None,
 'max_leaf_nodes': None,
 'min_impurity_decrease': 0.0,
 'min_impurity_split': None,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 600,
 'n_iter_no_change': None,
 'presort': 'auto',
 'random_state': None,
 'subsample': 1.0,
 'tol': 0.0001,
 'validation_fraction': 0.1,
 'verbose': 0,
 'warm_start': False}
```

5.3 XG Boost

```
time was 1.7242353409528732 hours


0.41374
[0.2596316  0.132617   0.1624777  0.12035192 0.18768501 0.13723671]
```

This score is very promiseing since the score is about double chance. There actually is a lot of correct classification going on here.

Interestingly however, it seems that the feature importances are very different for this model. Gender is the most important feature nad every other feature is about equally important.

5.4 K-Nearest Neighbors

In [ ]:

5.5 KD Trees

In [ ]:

6 GDA

In [ ]:

7 Results

8 Analysis

9 Conclusion

10 References

[1] Statement Of Concern About Predictive Policing By Aclu and 16 Civil Rights Privacy, Racial Justice, and Technology Organizations https://www.aclu.org/other/statement-concern-about-predictive-policing-aclu-and-16-civil-rights-privacy-racial-justice