Prediction of Opiod Overdoses and Misuse of Prescription Drugs

Moenuddeen Ahmad Shaik and Satyen Singh
ms10415 and srs833

Problem Statement

In 2017, the United States Department of Health and Human Services declared Opioid Epidemic as a public health emergency. In just 2016 alone, opioid overdoses claimed the lives of more than 42,000 people, making that year deadlier than any seen before. Even more shockingly, about forty percent of these overdoses were from a prescription opioid. This is highly alarming as it goes to show that one does not necessarily have to acquire opioids from a local drug dealer; these life-threatening and highly addictive substances could be prescribed by your doctor and be found at a well regulated drug store. To put this into perspective, in 2019, roughly 10 million people had abused prescription opioids and more than 1.5 million people were suffering from an opiod addiction. By 2019, the number of lives lost to opiods swelled to more than 70,000. What makes opioids so deeply prevalent in our society is the diversity of methods through which people can acquire it from. As aforementioned, it is available through prescription medication, such as pain relievers(i.e: OxyContin, Vicodin, Codeine, Morphine). It is also available as an illegal drug, commonly known as heroin. In its deadliest form, it is synthetically made, such as fentanyl. Fentanyl is so strong that handlers have to wear protective gear around as just a whiff is enough to cause an overdose. While opioids have a wide variety, our project will focus on the abuse of prescription opioids as they are so deeply pervasive, and more importantly, there is a significant amount of data available whereas it is difficult to acquire data about illicit or scheduled drugs. Our project will take a variety of data points, outlined in the kaggle link, to predict if an individual is likely to overdose on prescription opioids. We will use Neural Networks that will take the data as input and use it to create a predictive model.

Datasets

We are using the following dataset from Kaggle
Accidental death by fatal drug overdose is a rising trend in the United States. What can you do to help?
The dataset that we will use has summaries of prescription records for 250 common opioid as well as non-opioid drugs prescribed by 25,000 unique licensed medical professionals in 2014 for American citizens covered under Class D Medicare. The data also contains metadata about the medical professionals themselves. This portion of the data comes from cms.gov which is the Centers for Medicare and Medicaid Services. All of this information is found in prescriber-info.csv. There is another dataset called opioids.csv which contains the names of all the opioid drugs that are in the dataset. We are also using a third dataset called overdoses.csv that contains information about piod related drug overdose fatalities.
The data contains the following parameters:

- NPI – unique National Provider Identifier number
- Gender - (M/F)
- State - U.S. State by abbreviation
- Credentials - set of initials indicative of medical degree
- A long list of drugs with numeric values indicating the total number of prescriptions written for the year by that individual
- Opioid.Prescriber - a boolean label indicating whether or not that individual prescribed opiate drugs more than 10 times in the year

Model Description

Before we apply deep learning techniques to predict the opioid overdoses, we decided to use simpler machine learning techniques to get an idea of our dataset and see what kind of results we can generate without using neural networks.
For our project, we are planning to implement models such as Deep Neural Network and Recurrent Neural Network. Since it has been outlined extensively in above mentioned literature, it will be a good starting point. The input in our model will be the data that we will be getting from the kaggle dataset that we have stated above. The output of our model will be a score that could help us predict an accidental overdose with proper thresholding. We will set the dimensions of each hidden layer of our neural network to be 256 which we hope will give us a good performance. We will also employ various methods to tackle overfitting such as regularization techniques to make our model robust and have the ability to pick out the most important features. We will also use dropout as another way of combating overfitting for all our hidden layers. We will apply batch normal-

ization on non-recurrent layers to normalize the activations of the previous layer such that the outputs have a mean of 0 and standard deviation of 1 in each mini batch while training.
We especially want to explore the RNN model and improve upon the research articles that we have provided. The reasoning for using the RNN was to handle sequential data of arbitrary length and capture the sequential information from the data.

Data Preparation

- We first scrubbed all the numerical columns and removed all the commas so that we could use them as integers and not as strings. We also removed data which is not part of the 50 states. This included PR, ZZ, AE, VI, GU, and AA.
- One important statistic we wanted to capture was the deaths per capita in each state. This is because we need to be mindful of were any imbalances in the dataset. Upon observation, one can see that California has quite a lot more deaths than other states. It also has a higher population too so its number of deaths were proportional to number of people living there. In order to store this data point, we appended an extra column. We also dropped the NPI column and used the related Speciality column for better representation.
- We encoded the Gender, State and Speciality as Categorical columns.
- In order to track the source of the prescription opioid overdose, we have to follow the trail from the patients to the their prescriptions and finally to their prescriber which is their doctors. To track this, we labelled the opiod prescriptions vs non-opioid prescriptions. This way we check which doctors were prescribing opiates and which were not.

Initial Results

Models compared

We used a plethora of different models to derive predictions. These include the following:

- Ada Boost Classifier
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Decision Tree
- Random Forest
- K-Nearest Neighbors (KNN)
- Gradient Boosting
- Bagging Classifier
- Ensemble

Mean Accuracy scores

We ran our models once with "Gender" included and once without.

With Gender:

- Ada Boost Classifier: 0.791
- LDA: 0.715
- QDA: 0.637
- Decision Tree: 0.770
- Random Forest: 0.830
- KNN: 0.779
- Gradient Boosting: 0.823
- Bagging Classifier: 0.810
- Ensemble: 0.828

Without Gender:

- Ada Boost Classifier: 0.782
- LDA: 0.710
- QDA: 0.637
- Decision Tree: 0.772
- Random Forest: 0.834
- KNN: 0.779
- Gradient Boosting: 0.823
- Bagging Classifier: 0.810
- Ensemble: 0.826

As one can see from the results, ignoring "Gender" improved our accuracy results.

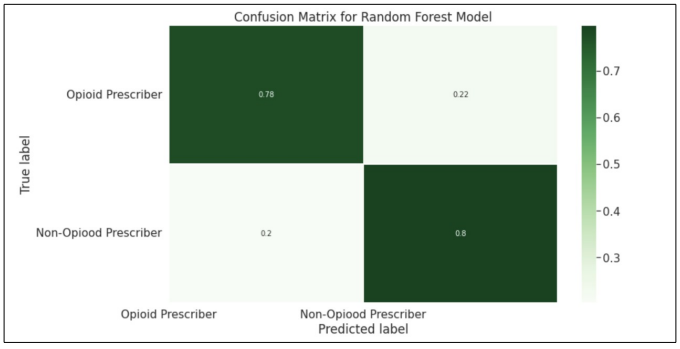
Best Model metrics

From the above results we see Random Forest Model has the best accuracy score.

Confusion matrix

```
array([[1983,  565],
       [ 740, 2902]])
```

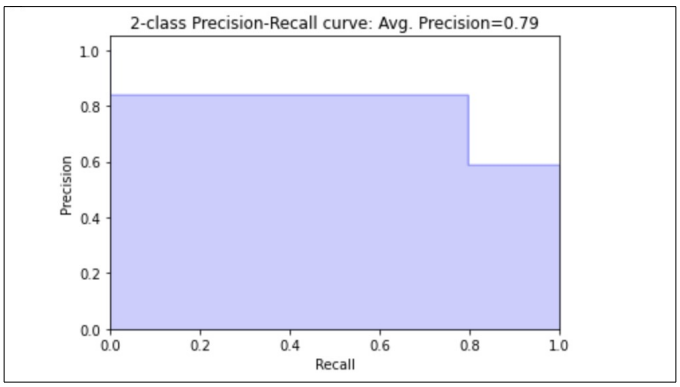
Confusion matrix (As a plot)



Average Precision Recall Score

```
Mean Precision-Recall score: 0.79
```

Precision Recall Curve



Classification Report

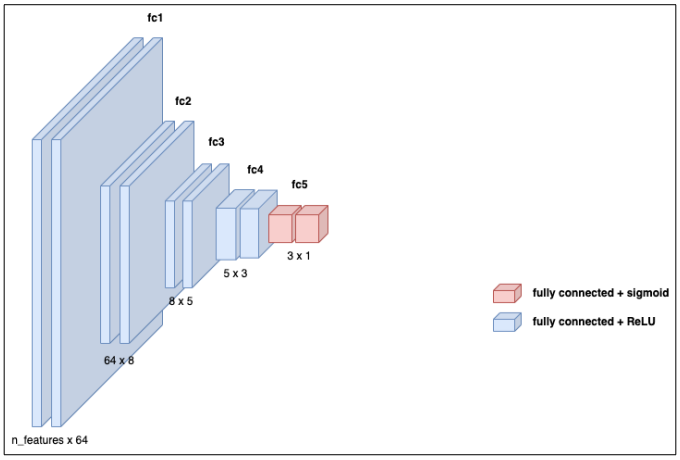
	precision	recall	f1-score	support
0	0.73	0.78	0.75	2548
1	0.84	0.80	0.82	3642
accuracy			0.79	6190
macro avg	0.78	0.79	0.78	6190
weighted avg	0.79	0.79	0.79	6190

Training with Deep Neural Network

Following are some variations in DNNs we tried.

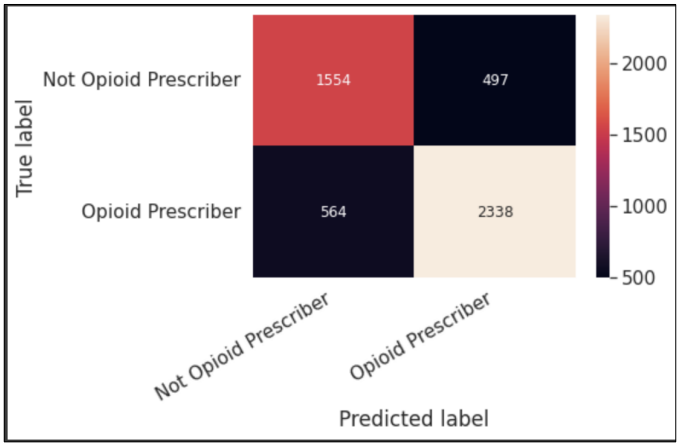
DNN Model 1

This is a simple model containing 5 fully connected layers and RELU activation for each layer. The architecture of this simple DNN is illustrated in the following image:



Categorical Variables We converted Gender, State and Specialty as Categorical columns.We used Binary Cross Entropy Loss and the Adam Optimizer with the learning rate set to 0.01. After training our model for 100 epochs, we were able to achieve an accuracy of 0.965 for the training set and 0.80 for the test set. The rest of our model statistics are illustrated in the following images.

Confusion Matrix

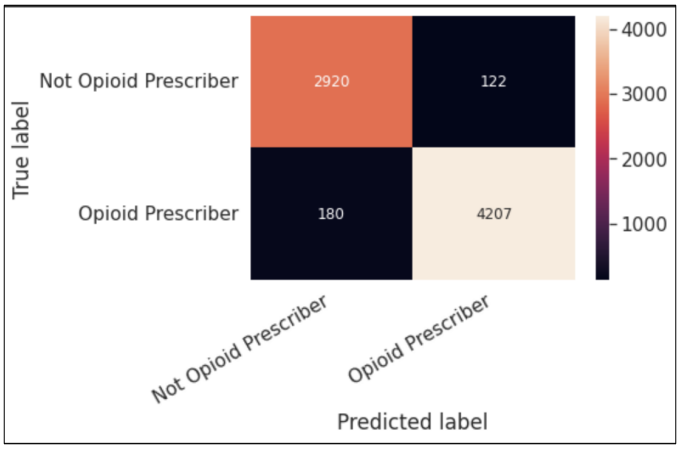


Classification Report

	precision	recall	f1-score	support
Not Opioid Prescriber	0.73	0.76	0.75	2051
Opioid Prescriber	0.82	0.81	0.82	2902
accuracy			0.79	4953
macro avg	0.78	0.78	0.78	4953
weighted avg	0.79	0.79	0.79	4953

One Hot Encoding Keeping the model same we encoded Gender and Specialty with One Hot Encoding to observe the difference in performance. After training our model for 100 epochs, we were able to achieve an accuracy of 0.965 for the training set and 0.80 for the test set. The rest of our model statistics are illustrated in the following images.

Confusion Matrix

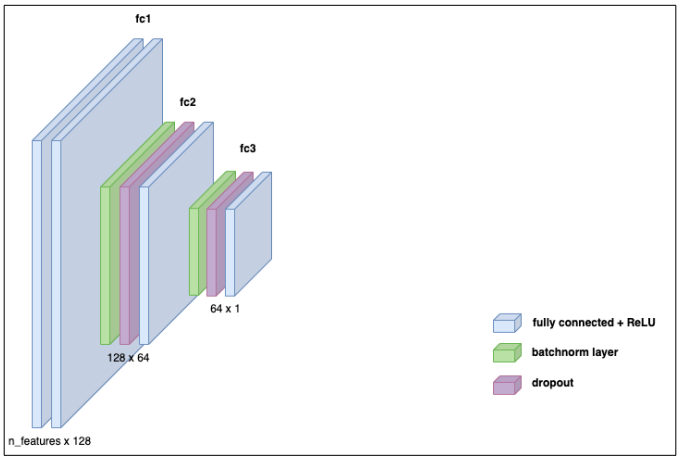


Classification Report

	precision	recall	f1-score	support
Not Opioid Prescriber	0.94	0.96	0.95	3042
Opioid Prescriber	0.97	0.96	0.97	4387
accuracy			0.96	7429
macro avg	0.96	0.96	0.96	7429
weighted avg	0.96	0.96	0.96	7429

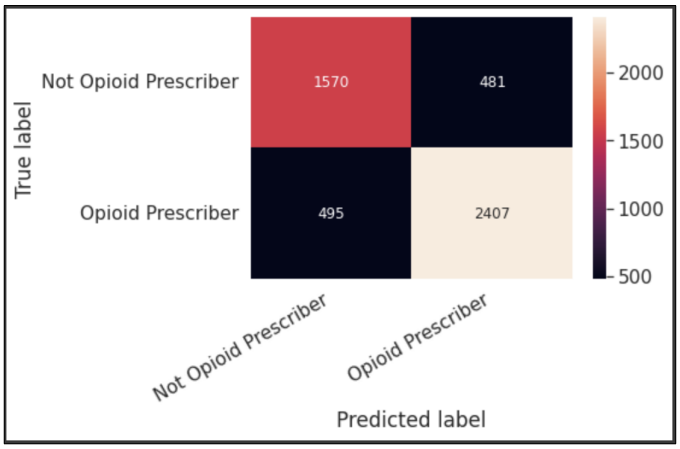
DNN Model 2

We modified the DNN model to contain 2 fully connected layers with ReLU activation and dropout layers to observe the improvement in performance.
The architecture of our updated DNN is illustrated in the following diagram:



Categorical Variables We converted Gender, State and Specialty as Categorical columns.We used Binary Cross Entropy with Logits Loss and the Adam Optimizer with the learning rate set to 0.01.
The batch size was set to 64 and the learning rate was 0.001. After training for 100 epochs, the best train accuracy was 0.92 and the best test accuracy was 0.81. The rest of our model statistics are illustrated in the following images.

Confusion Matrix

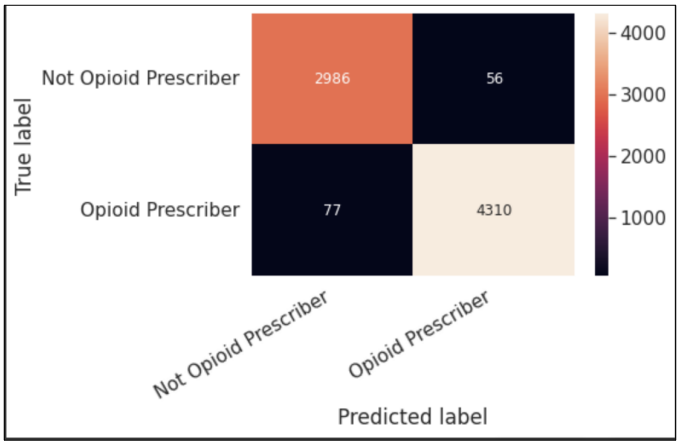


Classification Report

	precision	recall	f1-score	support
Not Opioid Prescriber	0.76	0.77	0.76	2051
Opioid Prescriber	0.83	0.83	0.83	2902
accuracy			0.80	4953
macro avg	0.80	0.80	0.80	4953
weighted avg	0.80	0.80	0.80	4953

One Hot Encoding Keeping the model same we encoded Gender and Specialty with One Hot Encoding to observe the difference in performance.
The batch size was set to 64 and the learning rate was 0.001. After training for 100 epochs, the best train accuracy was 0.99 and the best test accuracy was 0.98. The rest of our model statistics are illustrated in the following images.

Confusion Matrix



Classification Report

	precision	recall	f1-score	support
Not Opioid Prescriber	0.97	0.98	0.98	3042
Opioid Prescriber	0.99	0.98	0.98	4387
accuracy			0.98	7429
macro avg	0.98	0.98	0.98	7429
weighted avg	0.98	0.98	0.98	7429

Observation

We see that we have better performance for our DNN models when we consider the Gender and Specialty columns with One Hot Encoding improving our performances close to 99%. Although we have observed the test performance also has the same accuracy there's still a possibility of overfitting. Unfortunately, we don't have enough data to confirm this.

Conclusion

Our objective was to analyze the dataset and build a model which can efficiently predict the possibility of Opioid overdose. For this we were able to run Machine Learning models like Gradient Boosting, Random Forest, Ada Boost Classifier and an Ensemble model created by combining the above models and found Random Forest had the best performance with a precision score of 83%. Next we created a deep neural network and tuned the parameters and layers to achieve a better performance that the ML models previously used and this was achieved as you can see from the results presented above.

Github Repository link

https://github.com/ssingh1997/Deep_Learning_Final_Project

References

- Prevalence of opioid abuse and addiction.Link
- Analytic Models to Identify Patients at Risk for Prescription Opioid Abuse Link
- https://jamanetwork.com/journals/jamanetworkopen/..
- https://journals.plos.org/plosone/article?...
- https://www.ncbi.nlm.nih.gov/pmc/articles/..
- https://www.kaggle.com/apryor6/us-opiate-prescriptions
- https://github.com/IBM/predict-opioid-prescribers/blob/master/README.md
- https://www.kaggle.com/yevgenpukhta/quick-and-dirty-attempt-on-voting-classifier
- https://curiousily.com/posts/build-your-first-neural-network-with-pytorch/
- https://towardsdatascience.com/pytorch-tabular-binary-classification-a0368da5bb89
- https://www.kaggle.com/greenmaverick/predictopioidprescribers-nn-binaryclassification