# Visualizing and Modeling Mental Health

Kelly Farrell, Samy Kebaish, Gretchen Larrick

# Agenda

- Purpose
- Data
  - Origin
  - Tech Stack
- Exploratory Data Analysis
  - Mental Health Data
- Healthcare Quality Measures
  - Follow up visit analysis
  - Logistic Regression
- Trends over Time
  - Machine Learning
  - Website
- Conclusion
- Future Analysis

# How can we see the big picture on mental health and treatment?

- Investigate factors that influence mental health diagnoses and treatment
- Use models to predict some key measures
- Visualize and provide interactive tools to encourage engagement and understanding

# Data Discussion

- Null Values
- Joining Datasets

Medicaid.gov
Keeping America Healthy

https://www.medicaid.gov/medicaid/quality-of-care/performance-measurement/adult-and-child-health-care-quality-measures/adult-health-care-quality-measures/index.html

SAMHSA
Substance Abuse and Mental Health Services Administration

https://www.samhsa.gov/data/data-we-collect/mh-cld-mental-health-client-level-data

KFF

https://www.kff.org/statedata/

# Tech Stack

## Front End

### Web Framework

Flask

### Data Visualization

plotly

bokeh

seaborn

## Back End

### Data Processing

pandas

aws

### Machine Learning

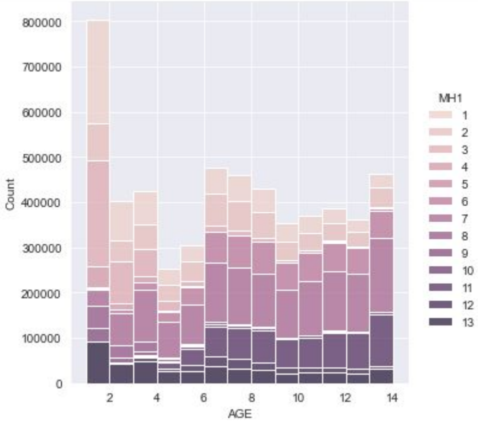
Keras

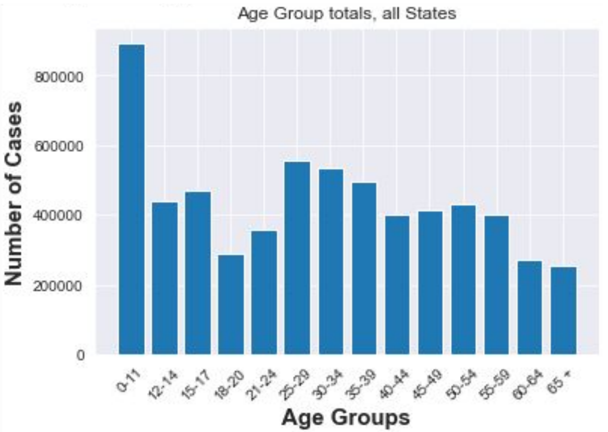
scikit learn

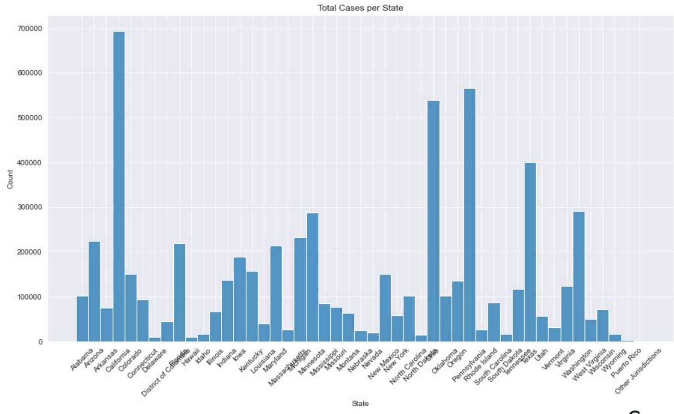
# Exploratory Data Analysis

- Variables
  - Coded
  - Categorical
- Challenges
  - File Size
- EDA Examples
  - Histograms

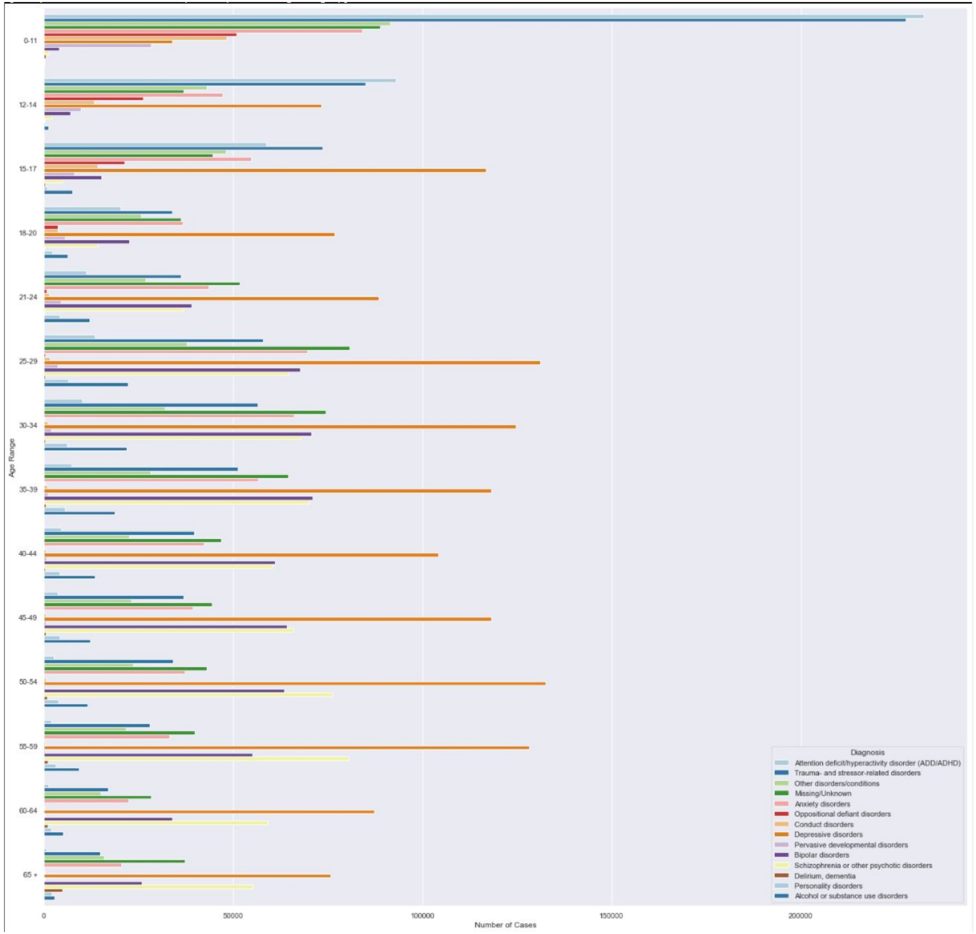
| Data Set | Rows x Columns |
|---|---|
| Mental Health Client-Level Data 2018 | 6213791 x 40 |
| 2018 Child and Adult Health Care Quality Measures | 2856 x 18 |
| Mental Health Client-Level Data 2013 - 2017 | various |



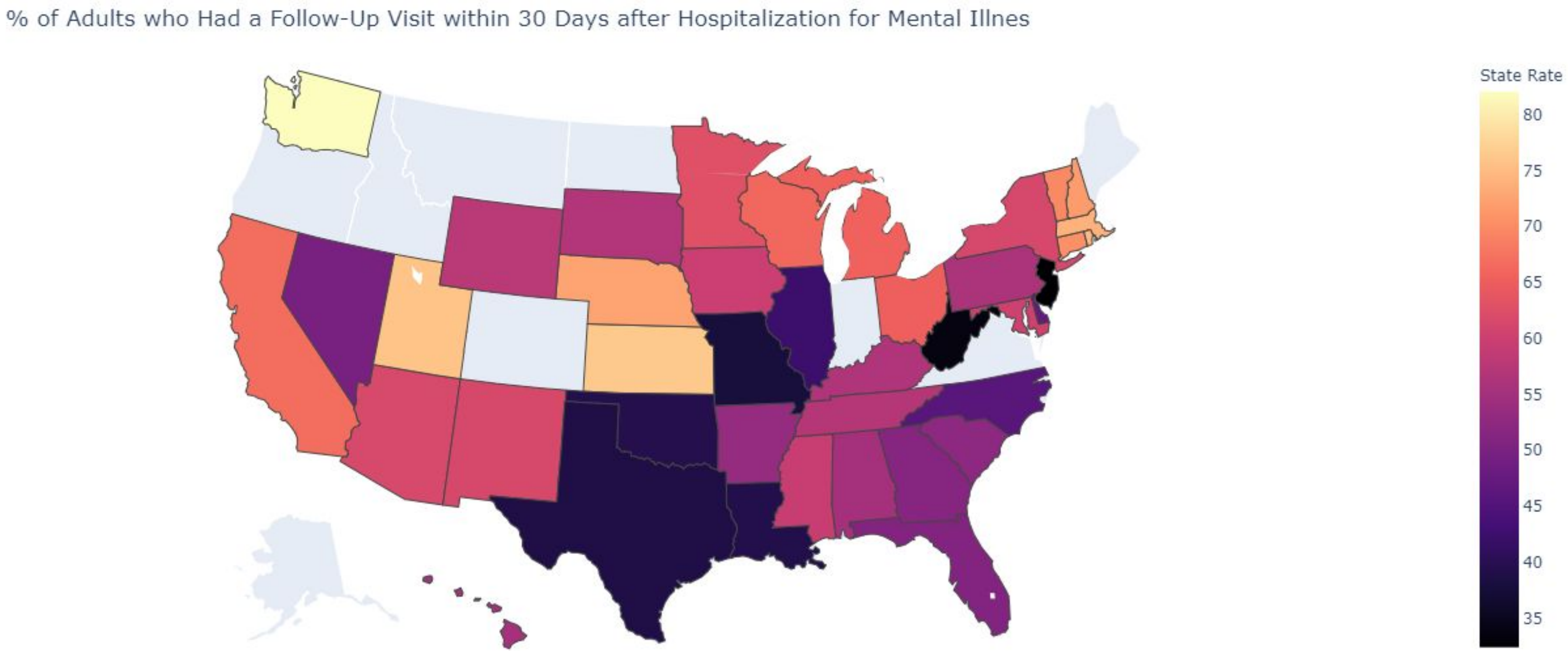



# Age Range and Diagnosis Analysis



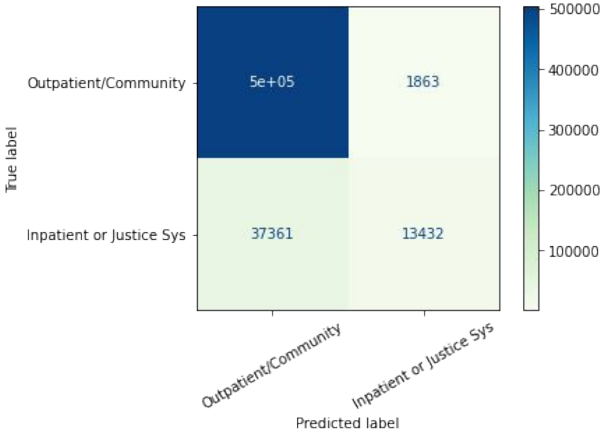

- Age Range 0-11 has the highest total cases.
- Diagnosis shift after the age of 17.
  - ADD/ADHD Decline
  - Trauma/Stressors Decline
  - Oppositional defiant disorders Decline
  - Depressive Disorders increase

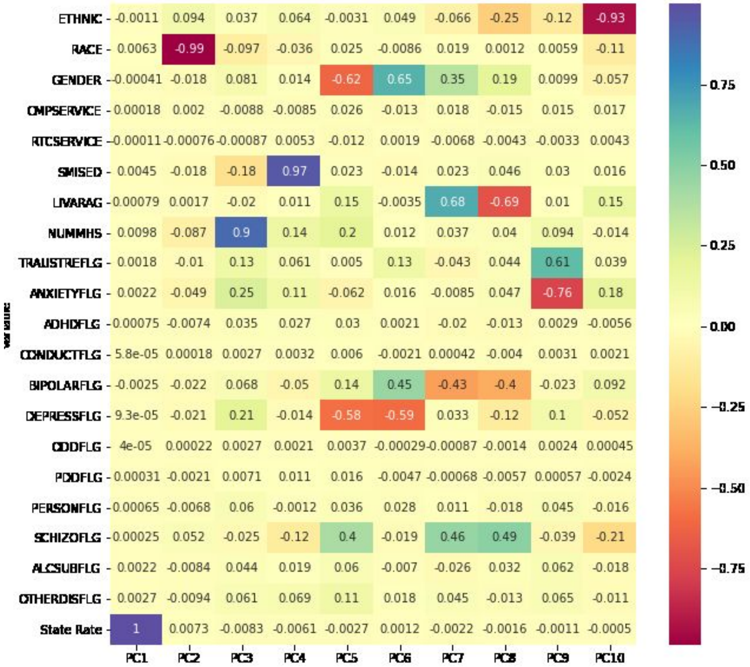
# Healthcare Quality Measures: Geography



# Healthcare Quality Measures/Outcomes:  Logistic Regression

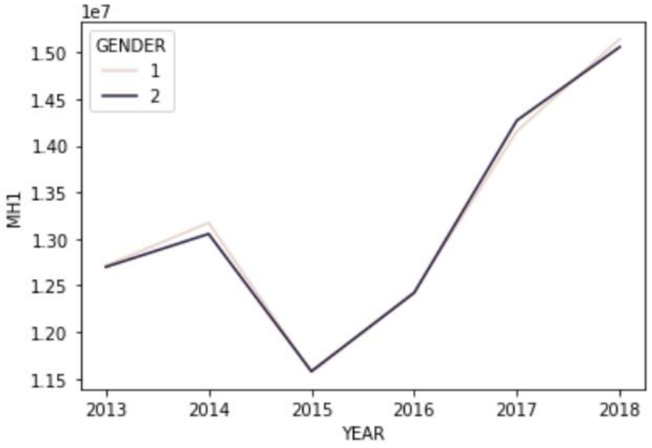
| | Variable | Coefficient |
|---|---|---|
| 0 | ETHNIC | -0.119943 |
| 1 | RACE | -0.048001 |
| 2 | GENDER | -0.554407 |
| 3 | CMPSERVICE | 4.750801 |
| 4 | RTCSERVICE | 0.904576 |
| 5 | SMISED | -0.179109 |
| 6 | LIVARAG | 0.753326 |
| 7 | NUMMHS | 0.320968 |
| 8 | TRAUSTREFLG | -0.060183 |
| 9 | ANXIETYFLG | -0.544588 |
| 10 | ADHDFLG | -0.731544 |
| 11 | CONDUCTFLG | -0.204433 |
| 12 | BIPOLARFLG | 0.151147 |
| 13 | DEPRESSFLG | -0.124426 |
| 14 | ODDFLG | -0.756464 |
| 15 | PDDFLG | -1.477699 |
| 16 | PERSONFLG | 0.547705 |
| 17 | SCHIZOFLG | 0.508583 |
| 18 | ALCSUBFLG | -0.287129 |
| 19 | OTHERDISFLG | 0.024449 |
| 20 | State Rate | 0.021482 |



# Historical Data (2013-2018)

- To evaluate possible trends, data was loaded from similar datasets for the years 2013-2018.
- For the most part, the incidence of diseases was fairly stable and there were no major differences of note amongst groups. This could be due to oversampling of certain populations, capturing an unrepresentative depiction of mental health.
    - Or perhaps that mental health affects more people, more broadly, than originally thought.

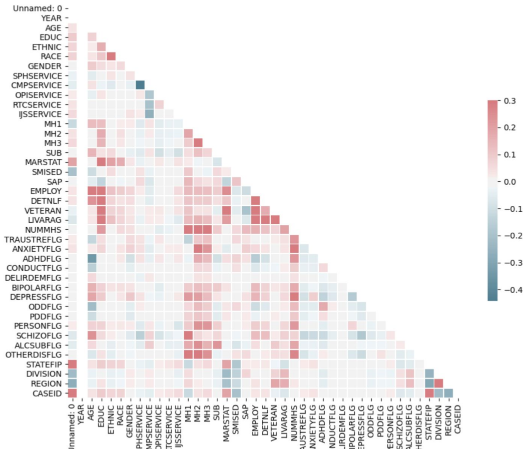


# Interactive Website (Brief Demo)

Data Analysis Project CS5010 Home Data Analysis Data Dictionary Chart

## Correlation Matrix

As can be visualized by the correlation matrix, none of the variables show a profound correlation with each other. Potential features, such as a age, sex, and ethnicity, have a multifactor relationship with labels such as type of mental health classification and/or disorder.




Data Analysis Project CS5010 Home Data Analysis Data Dictionary Chart

**YEAR: Reporting Period**

| Value | Label | Frequency | % |
|---|---|---|---|
| 2018 | 2018 | 6,213,791 | 100 |

**AGE: Age (recorded)**

| Value | Label | Frequency | % |
|---|---|---|---|
| 1 | 0-11 years | 892,484 | 14.4% |
| 2 | 12-14 years | 438,967 | 7.1% |
| 3 | 0-11 years | 892,484 | 14.4% |
| 4 | 12-14 years | 438,967 | 7.1% |
| 5 | 15-17 years | 469,162 | 7.6% |
| 6 | 18–20 years | 288,195 | 4.6% |
| 7 | 21–24 years | 357,257 | 5.7% |
| 8 | 25–29 years | 557,653 | 9.0% |
| 9 | 30–34 years | 534,107 | 8.6% |
| 10 | 35–39 years | 494,186 | 8.0% |
| 11 | 40–44 years | 400,992 | 6.5% |
| 12 | 45–49 years | 414,109 | 6.7% |
| 13 | 50–54 years | 414,109 | 6.7% |
| 14 | 55–59 years | 402,710 | 6.5% |
| 15 | 55–59 years | 402,710 | 6.5% |
| 16 | 60–64 years | 273,170 | 4.4% |
| 17 | 65 years and older | 256,693 | 4.1% |

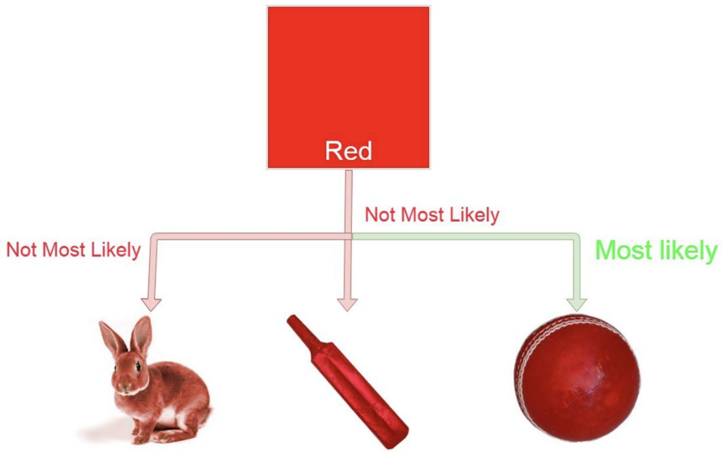
**EDUC: Education**

# Machine Learning

- Due to the large sample size and high number of labels and features, machine learning algorithms were applied towards predicting the presence or lack thereof mental health disease/disorders.
- These included:
  - Naive Bayes
  - KNN
  - Multilabel Classification (ADAM Optimization

# Naive Bayes

Naive Bayes Classifiers can be simplified as a classifier which counts how many times each attributes co-occurs with each class.




Likelihood

Class Prior Probability

$$P(c \mid x) = \frac{P(x \mid c)\,P(c)}{P(x)}$$

Posterior Probability

Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

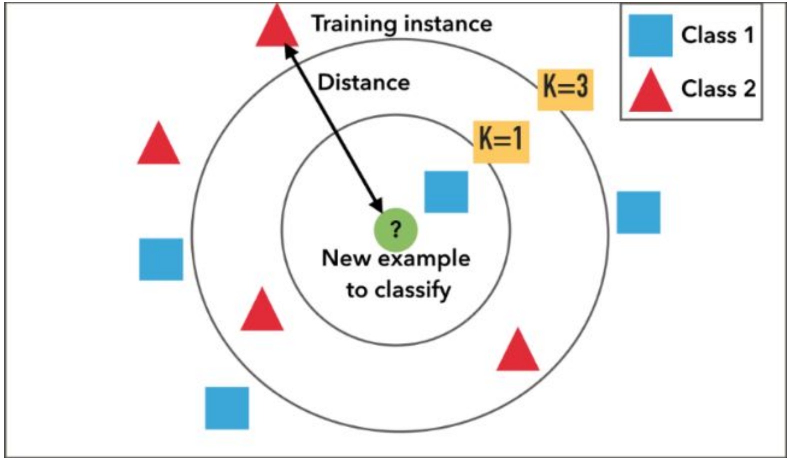
- $P(c|x)$ is the posterior probability of *class* c given *predictor* (*features*).
- $P(c)$ is the probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

# Naive Bayes

```
import pandas as pd
from google.colab import  drive
import numpy as np
from sklearn.preprocessing import LabelEncoder
import random
from sklearn.naive_bayes import GaussianNB
import scipy
learn.model_selection import train_test_split




#drive.mount('/drive', force_remount = False)
#check clean data; if dataset is small can use python or scikit


df = pd.read_csv("https://csprojectdatavisualizationsample50k.s3.us-east-2.amazonaws.com/sample_df.csv")
df_columns = df.columns
df_feature_names = (df_columns[1:6]).to_list()
df_features = df.iloc[:,2:6].values
df_label_names = (df_columns[26:26]).to_list()
df_labels = df.iloc[:, 26:26].values
#Input
print(df_label_names)
print(df_labels.shape)
print(df_features.shape)

# Split our data
train, test, train_labels, test_labels = train_test_split(df_features,
                                                          df_labels,
                                                          test_size=0.50,
                                                          random_state=42)

print(train.shape)
print(test.shape)

# Initialize our classifier
gnb = GaussianNB()

# Train our classifier
model = gnb.fit(train, train_labels)

# Make predictions
preds = gnb.predict(test)
print(preds)
```

# K-Nearest Neighbors

- K Nearest Neighbors is an algorithm which assumes that similar data points are close to one another.
- Can be used to solve classification and regression problems.



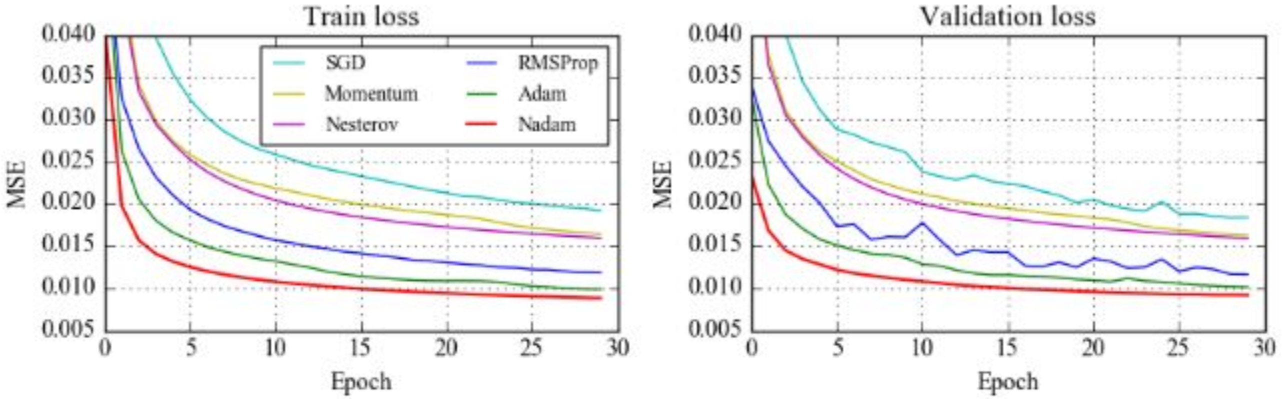
# K-Nearest Neighbors

```
import pandas as pd
from google.colab import  drive
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
import scipy
from sklearn.model_selection import train_test_split


# drive.mount('/drive', force_remount = False)
#check clean data; if dataset is small can use python or scikit


df = pd.read_csv("https://csprojectdatavisualizationsample50k.s3.us-east-2.amazonaws.com/sample_df.csv")
df_columns = df.columns
df_feature_names = (df_columns[2:15]).to_list()
print("Features to be analyzed ", df_feature_names)
df_features = df.iloc[:,2:15].values
df_label_names = (df_columns[26:36]).to_list()
df_labels = df.iloc[:, 26:36].values
print("Labels to be analyzed", df_label_names)

#Input


# Split our data
train, test, train_labels, test_labels = train_test_split(df_features,
                                                          df_labels,
                                                          test_size=0.25,
                                                          random_state=42)

row = [[3, 2, 3, 5, 2, 1, 1, 1, 2, 1, 1,1,6]]

from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 7).fit(train, train_labels)
result = knn.predict(row)
print("Prediction is ", result)
# accuracy on X_test
accuracy = knn.score(train, train_labels)
print("The accuracy is", accuracy)
```

```
Features to be analyzed  ['AGE', 'EDUC', 'ETHNIC', 'RACE', 'GENDER', 'SPHSERVICE', 'CMPSERVICE', 'OPISERVICE', 'RTCSERVICE', 'IJSSERVICE', 'MH1', 'MH2', 'MH3']
Labels to be analyzed ['ADHDFLG', 'CONDUCTFLG', 'DELIRDEMFLG', 'BIPOLARFLG', 'DEPRESSFLG', 'ODDFLG', 'PDDFLG', 'PERSONFLG', 'SCHIZOFLG', 'ALCSUBFLG']
(37500, 13)
(12500, 13)
Prediction is  [[0 0 0 0 1 0 0 0 0 0]]
```

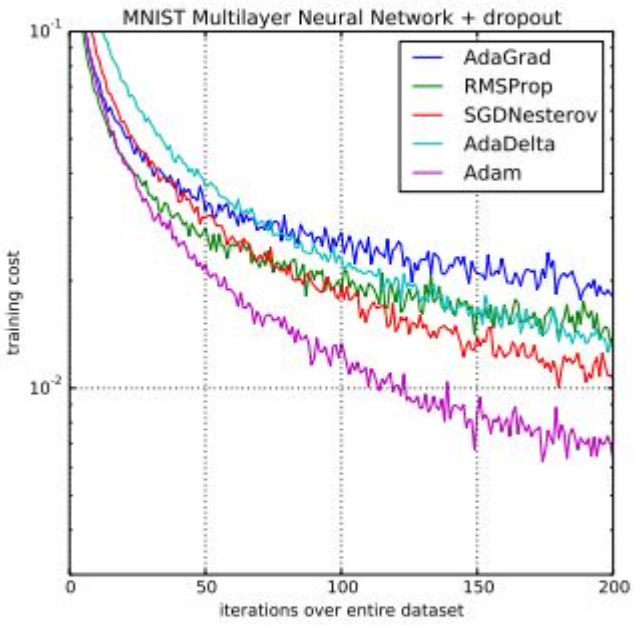
# Multilabel Classification using Keras (ADAM optimization algorithm)

- Keras is an intuitive deep learning API which acts as an interface for the Tensorflow Library.
- We implemented a Multilabel Classification using the ADAM optimization algorithm.



# Adam Optimization Algorithm

The Adam optimization algorithm is used, as opposed to stochastic gradient descent, which stands for Adaptive Moment Estimation. Unlike stochastic gradients, which use single learning rates, here the Adam algorithm implements both Adaptive Gradient Algorithm and Root Mean Square Propagation. These two mechanisms calculate an exponential moving average of the gradient and the squared gradient.



# Keras Code

```python
# mlp for multi-label classification
from numpy import mean
from numpy import std
from sklearn.datasets import make_multilabel_classification
from sklearn.model_selection import RepeatedKFold
from keras.models import Sequential
from keras.layers import Dense
from sklearn.metrics import accuracy_score
import pandas as pd

# get the model
def get_model(n_inputs, n_outputs):
  model = Sequential()
  model.add(Dense(20, input_dim=n_inputs, kernel_initializer='he_uniform', activation='relu'))
  model.add(Dense(n_outputs, activation='sigmoid'))
  model.compile(loss='binary_crossentropy', optimizer='adam')
  return model

# evaluate a model using repeated k-fold cross-validation
def evaluate_model(X, y):
  results = list()
  n_inputs, n_outputs = X.shape[1], y.shape[1]
  # define evaluation procedure
  cv = RepeatedKFold(n_splits=10, n_repeats=3, random_state=1)
  # enumerate folds
  for train_ix, test_ix in cv.split(X):
    # prepare data
    X_train, X_test = X[train_ix], X[test_ix]
    y_train, y_test = y[train_ix], y[test_ix]
    # define model
    model = get_model(n_inputs, n_outputs)
    # fit model
    model.fit(X_train, y_train, verbose=0, epochs=10)
    # make a prediction on the test set
    yhat = model.predict(X_test)
    # round probabilities to class labels
    yhat = yhat.round()
    # calculate accuracy
    acc = accuracy_score(y_test, yhat)
    # store result
    print('>%.3f' % acc)
    results.append(acc)
  return results

df = pd.read_csv("https://csprojectdatavisualizationsample50k.s3.us-east-2.amazonaws.com/sample_df.csv")
df_columns = df.columns
df_feature_names = (df_columns[2:7]).to_list() + (df_columns[37:38]).to_list() + (df_columns[16:17]).to_list() +  (df_columns[19:20]).to_list()
print("Features to be analyzed ", df_feature_names)
df_features = pd.concat([df.iloc[:,2:7], df.iloc[:,37:38], df.iloc[:,16:17], df.iloc[:,19:20]], axis = 1)
#Features to be analyzed  ['YEAR', 'AGE', 'EDUC', 'ETHNIC', 'RACE']
df_label_names = (df_columns[26:36]).to_list()
print("Labels to be analyzed", df_label_names)
df_labels = df.iloc[:, 26:36].values
#Labels to be analyzed ['ADHDFLG', 'CONDUCTFLG', 'DELIRDEMFLG', 'BIPOLARFLG', 'DEPRESSFLG', 'ODDFLG', 'PDDFLG', 'PERSONFLG', 'SCHIZOFLG', 'ALCSUBFLG']
print(df_features)
# results = evaluate_model(df_features, df_labels)
print('Accuracy: %.3f (%.3f)' % (mean(results), std(results)))

n_inputs, n_outputs = df_features.shape[1], df_labels.shape[1]
# get model
model = get_model(n_inputs, n_outputs)
# fit the model on all data
model.fit(df_features, df_labels, verbose=0, epochs=10)
# make a prediction for new data
row = [9, 4, 3, 5, 2, 6, 4, 1]
#Prediction for a 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced
newX = list([row])
yhat = model.predict(newX)
print('Accuracy: %.3f (%.3f)' % (mean(results), std(results)))
print('Predicted: %s' % yhat[0])
```

# Example using Keras

Imagine the following scenario: A 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced. How well would we be able to diagnose her from the following mental health disorders?

| |
|---|
| Trauma- and stressor-related disorders |
| Anxiety disorders |
| Attention deficit/hyperactivity disorder (ADD/ADHD) |
| Conduct disorders |
| Delirium, dementia |
| Bipolar disorders |
| Depressive disorders |
| Oppositional defiant disorders |
| Pervasive developmental disorders |
| Personality disorders |
| Schizophrenia or other psychotic disorders |
| Alcohol or substance use disorders |

# Example using Keras

Imagine the following scenario: A 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced. How well would we be able to diagnose her from the following mental health disorders?

**Not very well. A Multi Label Classification Algorithm using ADAM optimization and 10 epochs was only able to achieve a maximum 36% accuracy.**

```
Features to be analyzed  ['AGE', 'EDUC', 'ETHNIC', 'RACE', 'GENDER', 'STATEFIP', 'MARSTAT', 'EMPLOY']
Labels to be analyzed ['ADHDFLG', 'CONDUCTFLG', 'DELIRDEMFLG', 'BIPOLARFLG', 'DEPRESSFLG', 'ODDFLG', 'PDDFLG', 'PERSONFLG', 'SCHIZOFLG', 'ALCSUBFLG']
>0.344
>0.359
>0.357
>0.347
>0.344
>0.350
>0.357
>0.349
>0.341
>0.345
>0.348
>0.345
>0.339
>0.346
>0.337
>0.350
>0.350
>0.340
>0.345
>0.344
>0.334
>0.341
>0.355
>0.352
>0.335
>0.350
>0.348
>0.359
>0.342
>0.347
Accuracy: 0.347 (0.007)
```

```
                Predicted:
['ADHDFLG',     [2.9082805e-02
'CONDUCTFLG',   1.8648803e-03
'DELIRDEMFLG',  2.6618242e-03
'BIPOLARFLG',   1.8475160e-01
'DEPRESSFLG',   4.4119683e-01
'ODDFLG',        1.9347668e-04
'PDDFLG',       2.0621717e-03
'PERSONFLG',    6.7612886e-02
'SCHIZOFLG',    1.4146024e-01
'ALCSUBFLG']    8.1029296e-02]
```

# Example using Keras

Imagine the following scenario: A 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced. How well would we be able to diagnose her from the following mental health disorders?

What if you also knew she **was not**

1. **A veteran**
2. **Engaging in substance abuse**
3. **Homeless**

| |
|---|
| Trauma- and stressor-related disorders |
| Anxiety disorders |
| Attention deficit/hyperactivity disorder (ADD/ADHD) |
| Conduct disorders |
| Delirium, dementia |
| Bipolar disorders |
| Depressive disorders |
| Oppositional defiant disorders |
| Pervasive developmental disorders |
| Personality disorders |
| Schizophrenia or other psychotic disorders |
| Alcohol or substance use disorders |

# Example using Keras

Imagine the following scenario: A 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced. How well would we be able to diagnose her from the following mental health disorders?

What if you also knew **she was not**

1. **A veteran**
2. **Engaging in substance abuse**
3. **Homeless**

**Accuracy Increases to 38%; Diagnosis changes minimally**

```
['ADHDFLG',       [1.73549056e-02
'CONDUCTFLG',     8.18073750e-04
'DELIRDEMFLG',    5.06639481e-03
'BIPOLARFLG',     1.70569807e-01
'DEPRESSFLG',      4.50651914e-01
'ODDFLG',         2.84641981e-04
'PDDFLG',         2.84901261e-03
'PERSONFLG',      3.72844636e-02
'SCHIZOFLG',       1.16393566e-01
'ALCSUBFLG']      5.05395234e-02]
```

# Example using Keras

Imagine the following scenario: A 40-44 year old, High School Educated, Hispanic (other than Mexico or Puerto Rico), White in Ethnicity, Female), From California, Divorced. How well would we be able to diagnose her from the following mental health disorders?

What if you knew **she was**

1. **A veteran**
2. **Engaging in substance abuse**
3. **Homeless**

**Accuracy Increases to 38%; Diagnosis undergoes notable changes**

```
['ADHDFLG',        [1.5554756e-02
'CONDUCTFLG',      6.6286325e-04
'DELIRDEMFLG',     3.0833483e-04
'BIPOLARFLG',      4.1382189e-01
'DEPRESSFLG',      4.5767653e-01
'ODDFLG',           2.5886297e-04
'PDDFLG',          4.2591393e-03
'PERSONFLG',       6.6187352e-02
'SCHIZOFLG',       6.6005898e-01
'ALCSUBFLG']       4.9500734e-02]
```

# Example using Keras

Finally, i**ncluding additional features leads to greater accuracy**. In this case, the addition of hospitalization information and prior history of mental illness (6 additional features), substantially improves the accuracy from the original 12, demographic-heavy features (e.g. 18 total features evaluated).

```
>0.613
>0.728
>0.667
>0.714
>0.757
>0.603
>0.607
>0.735
>0.650
>0.601
>0.618
>0.698
>0.694
>0.685
>0.686
>0.718
>0.687
>0.622
>0.648
>0.686
>0.758
>0.644
>0.638
>0.711
>0.614
>0.707
>0.704
>0.593
>0.735
>0.669
Accuracy: 0.673 (0.049)
```

**Features to be analyzed** ['AGE', 'EDUC', 'ETHNIC', 'RACE', 'GENDER', 'SPHSERVICE', 'CMPSERVICE', 'OPISERVICE', 'RTCSERVICE', 'IJSSERVICE', 'MH1', 'MH2', 'MH3', 'STATEFIP', 'MARSTAT', 'SAP', 'EMPLOY', 'VETERAN', 'LIVARAG']
**Labels to be analyzed** ['ADHDFLG', 'CONDUCTFLG', 'DELIRDEMFLG', 'BIPOLARFLG', 'DEPRESSFLG', 'ODDFLG', 'PDDFLG', 'PERSONFLG', 'SCHIZOFLG', 'ALCSUBFLG']

```
['ADHDFLG',       [0.08255884
'CONDUCTFLG',     0.00623584
'DELIRDEMFLG',    0.01521513
'BIPOLARFLG',     0.34578314
'DEPRESSFLG',     0.70010173
'ODDFLG',         0.00119707
'PDDFLG',         0.00349078
'PERSONFLG',      0.06941462
'SCHIZOFLG',      0.4814202
'ALCSUBFLG']      0.02444795]
```

# Findings

- In the case of this dataset, training the Keras Multilabel Classification model with larger dataset (6 million variables) vs. a sample size of 200,000 variables led to only marginal improvement.
- The addition of demographic variables as features led to only minimal improvements in accuracy, whereas adding information about hospitalizations, treatment and substance abuse were more meaningful feature additions.

# Conclusions

- Traditional and ML models can characterize complex questions about mental health, although ML models require a careful selection of features.
- Need to consider dataset characteristics, null/missing data, and imbalanced sample sizes in order to get a full picture
- Trends in diagnoses are relatively steady over time

# Future Analysis

The expansive nature of the data allows for continuing investigations to clarify and broaden understanding.

- ❏ Mental Health Diagnosis Impact Investigation
  - ❏ How do factors, such as race, socioeconomic conditions, and geographical location, impact the type of diagnosis received?
- ❏ Healthcare quality measures
  - ❏ Logistic regression sampling
- ❏ Future Trends
  - ❏ Covid 19 impact on mental health