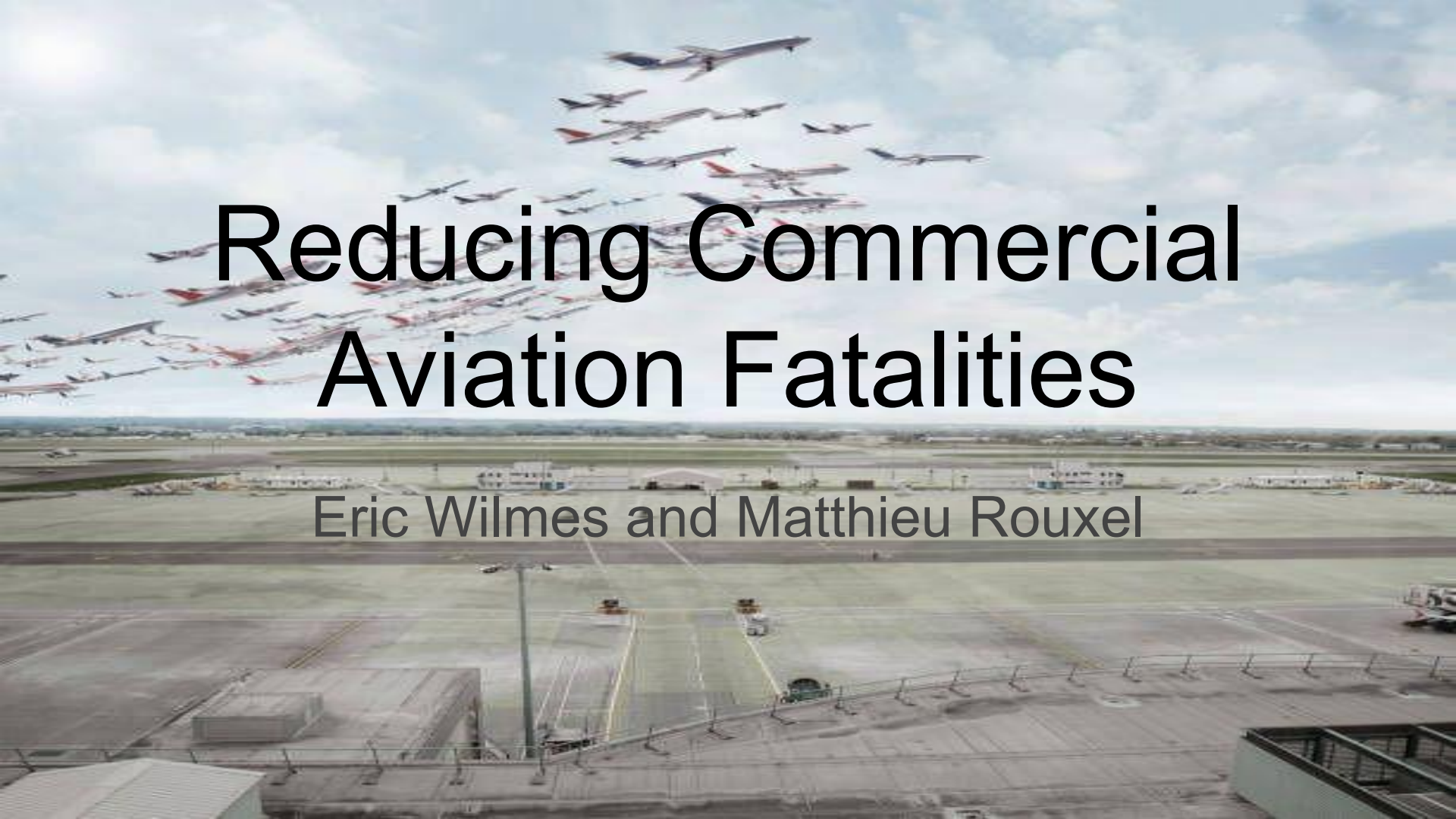


Reducing Commercial Aviation Fatalities

Eric Wilmes and Matthieu Rouxel

Background

Air Travel in the US is probably one of the most frequently public transportation service used.

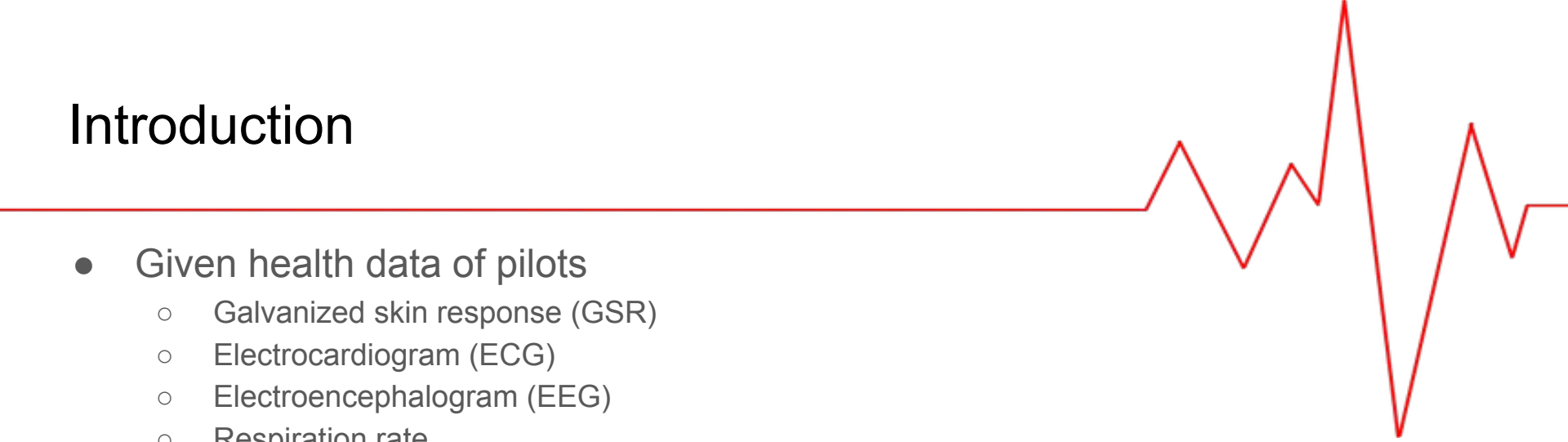
The FAA handles more than 16 million flights a year.

More than 960 million passengers are scheduled to fly on these flights each year.

2.9 M people fly in and out of the US airports yearly.

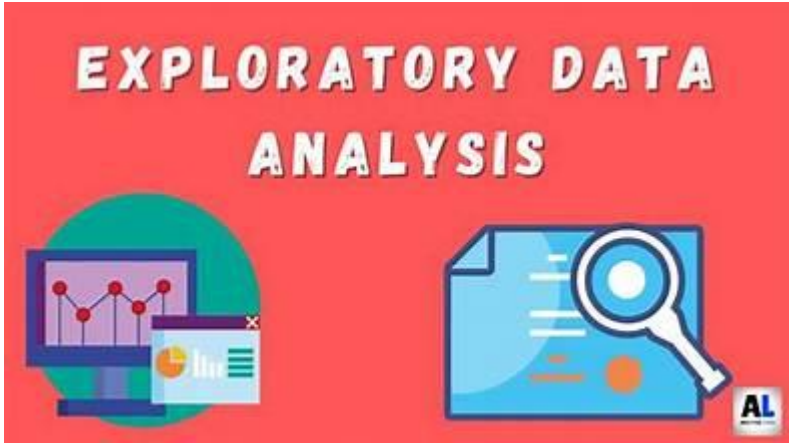
This is why we want to create a safe and efficient environment for all passengers embarking on an airplane.

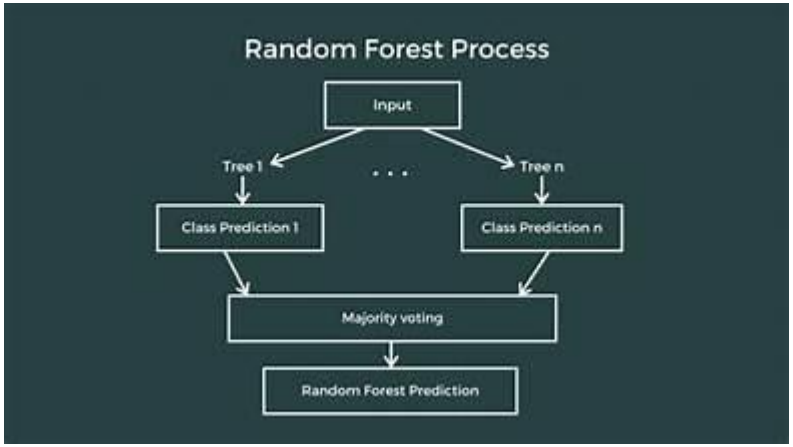
Introduction

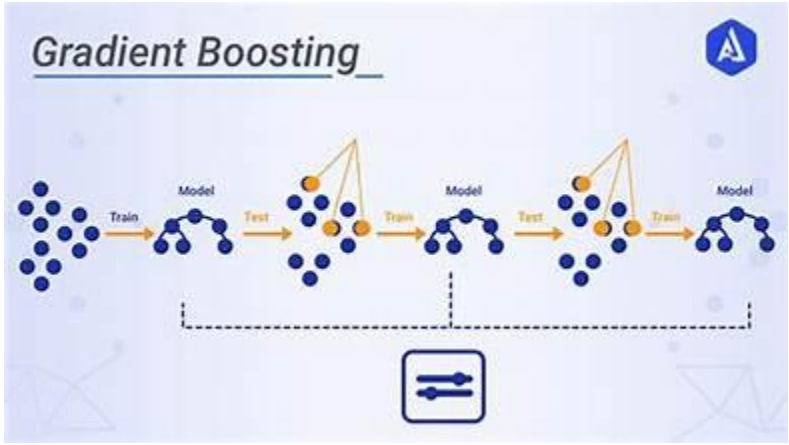


- Given health data of pilots
  - Galvanized skin response (GSR)
  - Electrocardiogram (ECG)
  - Electroencephalogram (EEG)
  - Respiration rate
  - Etc
- Need to predict which state the pilot is in during different times of flight
  - Baseline - Normal
  - Channelized Attention (CA) - Focused
  - Diverted Attention (DA) - Distracted
  - Startle/Surprise (SS) - Jump Scare

Methods


EXPLORATORY DATA
ANALYSIS
AL


Random Forest Process
Input
Tree 1
Tree n
Class Prediction 1
Class Prediction n
Majority voting
Random Forest Prediction


Gradient Boosting_
Train
Model
Test
Train
Model
Test
Train
Model

EDA

EDA

Shape and Dimensions of our dataset: (4867421, 28)

How does our dataset look like?

What features/variables are we working with?

[43]:

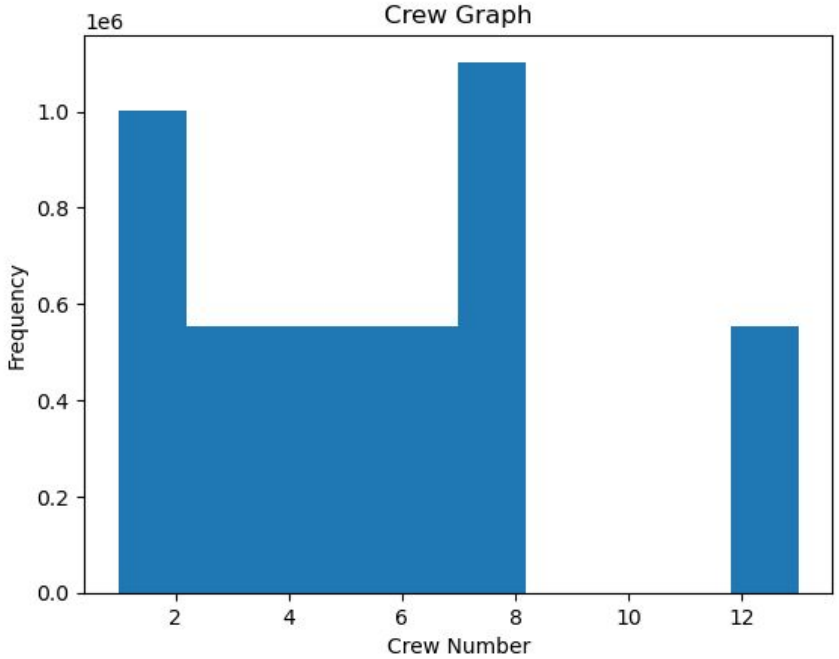
	crew	experiment	time	seat	eeg_fp1	eeg_f7	eeg_f8	eeg_t4	eeg_t6	eeg_t5	...	eeg_c4	eeg_p4	eeg_poz	eeg_c3	eeg_cz	eeg_o2	ecg	r	
0	1	CA	0.011719	1	-5.28545	26.775801	-9.527310	-12.793200	16.717800	33.737499	...	37.368999	17.437599	19.201900	20.596800	-3.951150	14.507600	-4520.0	817.705994	388.8
1	1	CA	0.015625	1	-2.42842	28.430901	-9.323510	-3.757230	15.969300	30.443600	...	31.170799	19.399700	19.689501	21.354700	1.332120	17.750200	-4520.0	817.705994	388.8
2	1	CA	0.019531	1	10.67150	30.420200	15.350700	24.724001	16.143101	32.142799	...	-12.012600	19.396299	23.171700	22.407600	1.537860	22.247000	-4520.0	817.705994	388.8
3	1	CA	0.023438	1	11.45250	25.609800	2.433080	12.412500	20.533300	31.494101	...	18.574100	23.156401	22.641199	19.336700	2.544920	18.998600	-4520.0	817.705994	388.8
4	1	CA	0.027344	1	7.28321	25.942600	0.113564	5.748000	19.833599	28.753599	...	6.555440	22.754700	22.670300	20.293200	1.699620	22.812799	-4520.0	817.705994	388.8
5	1	CA	0.031250	1	6.06746	23.128300	8.645660	14.380800	16.055500	26.925200	...	-9.289120	21.440599	23.253700	19.069599	-0.765018	26.451900	-4520.0	817.705994	388.8
6	1	CA	0.035156	1	-1.37602	20.972000	3.754160	13.766700	18.122000	29.391199	...	-0.604736	20.993401	21.556200	17.327299	1.465000	21.289301	-4520.0	817.705994	388.8
7	1	CA	0.039062	1	1.54787	18.398100	-9.113150	-1.033160	22.627001	32.816601	...	17.483601	22.912600	23.187000	18.462700	0.299232	23.691500	-4520.0	817.705994	388.8
8	1	CA	0.042969	1	-7.78946	12.210700	-8.953760	1.091740	28.526501	35.267200	...	16.681101	28.779600	28.382099	20.697300	-6.620750	31.672701	-4520.0	817.705994	388.8
9	1	CA	0.046875	1	-11.17750	18.235901	-1.035220	4.751110	25.983801	30.499100	...	0.489098	21.937500	21.629299	19.525999	-5.186040	21.618700	-4520.0	817.705994	388.8

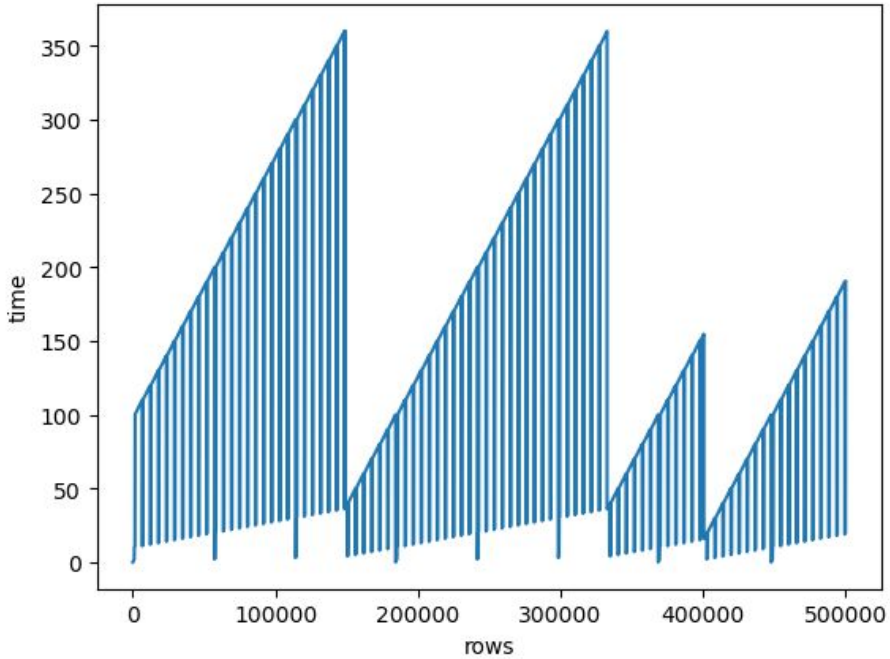
[51]:

```
crew            int64
experiment     object
time          float64
seat             bool
eeg_fp1       float64
eeg_f7        float64
eeg_f8        float64
eeg_t4        float64
eeg_t6        float64
eeg_t5        float64
eeg_t3        float64
eeg_fp2       float64
eeg_o1        float64
eeg_p3        float64
eeg_pz        float64
eeg_f3        float64
eeg_fz        float64
eeg_f4        float64
eeg_c4        float64
eeg_p4        float64
eeg_poz       float64
eeg_c3        float64
eeg_cz        float64
eeg_o2        float64
ecg           float64
r             float64
gsr           float64
event          object
dtype: object
```

EDA

Graphs/Visualizations

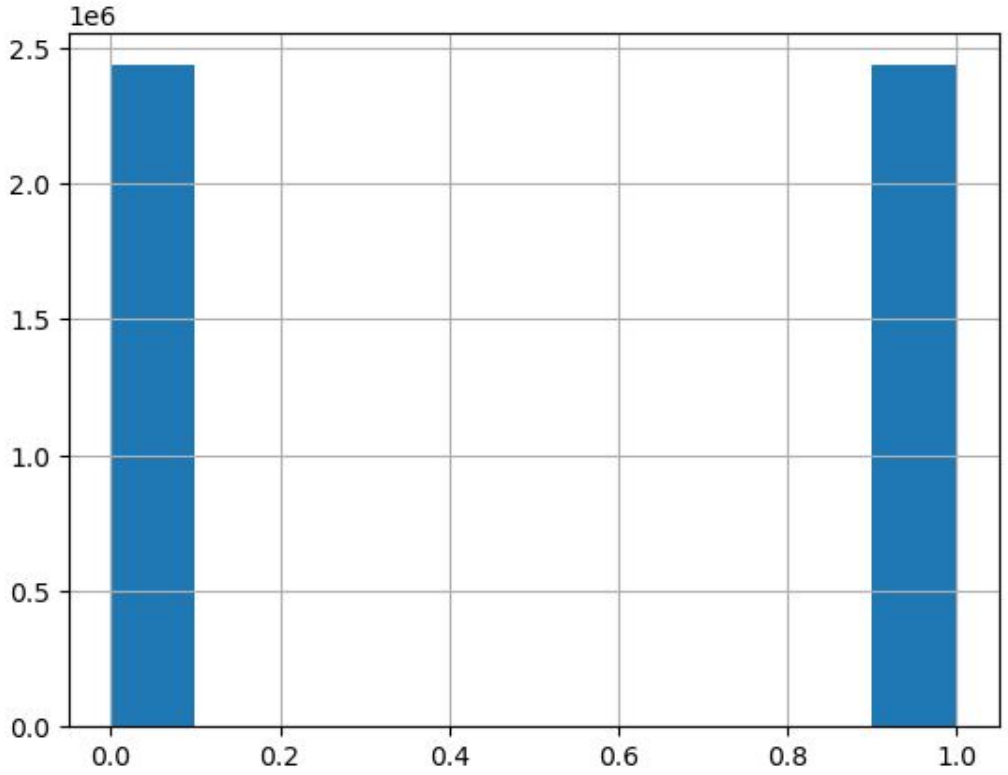




EDA - seat

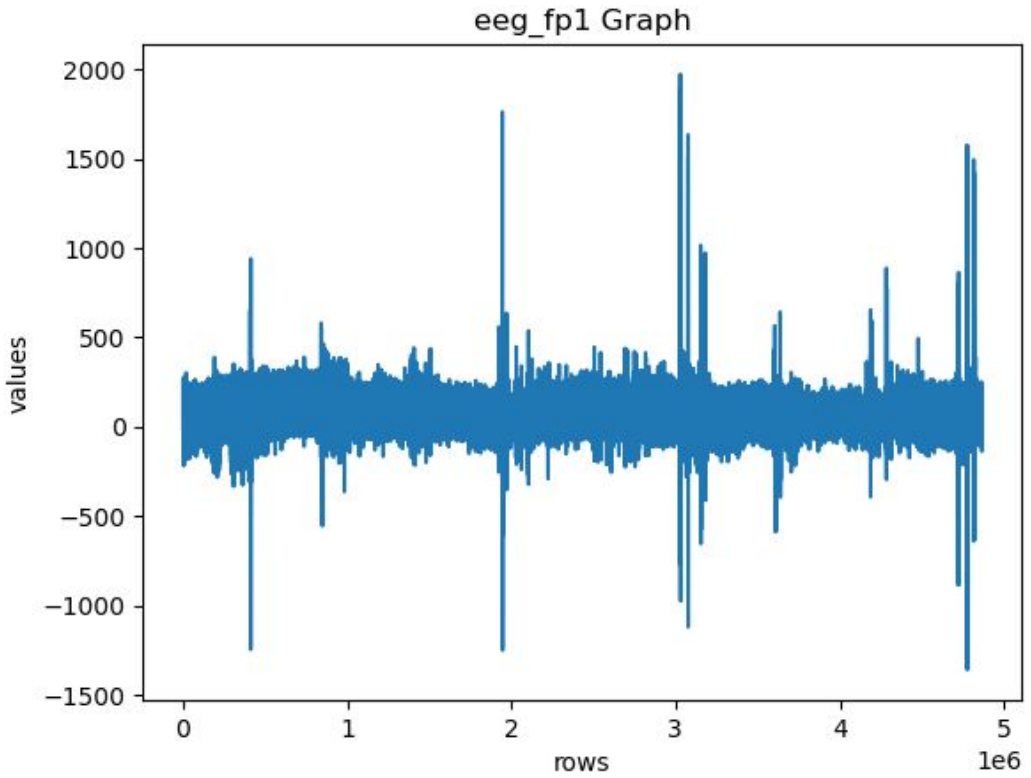
Graphs/Visualizations

- Left (0)
- Right (1)



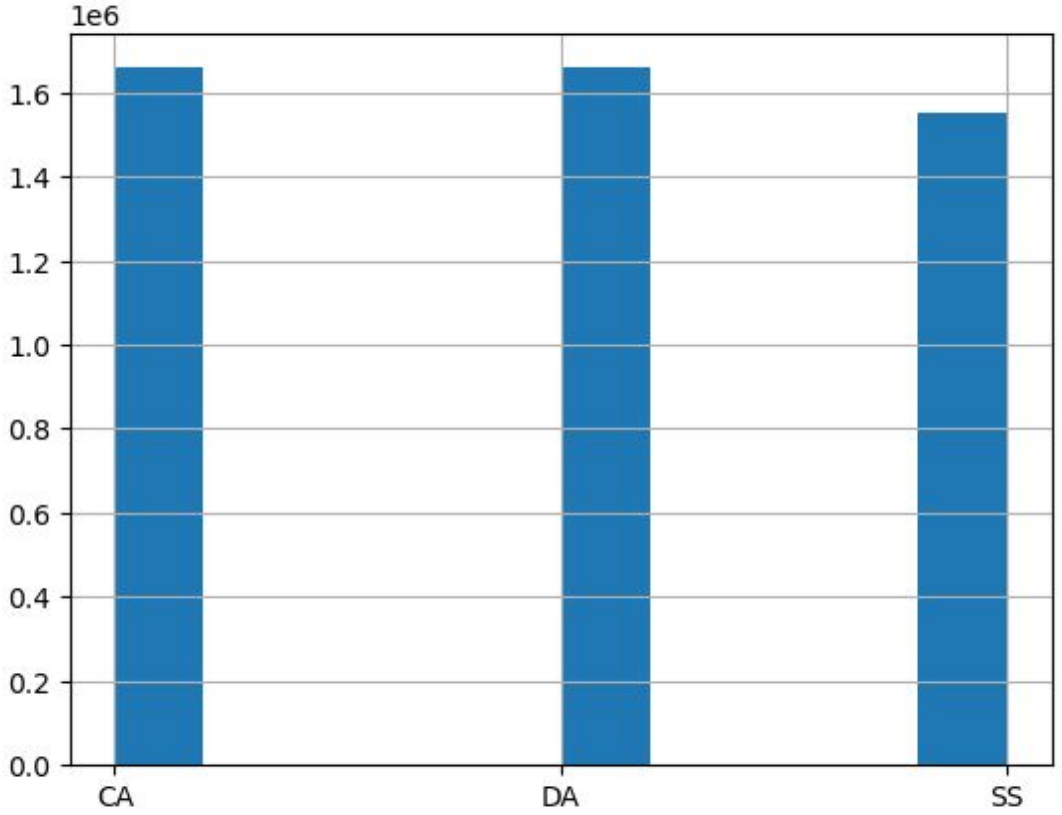
EDA - eeg_fp1

Graphs/Visualizations



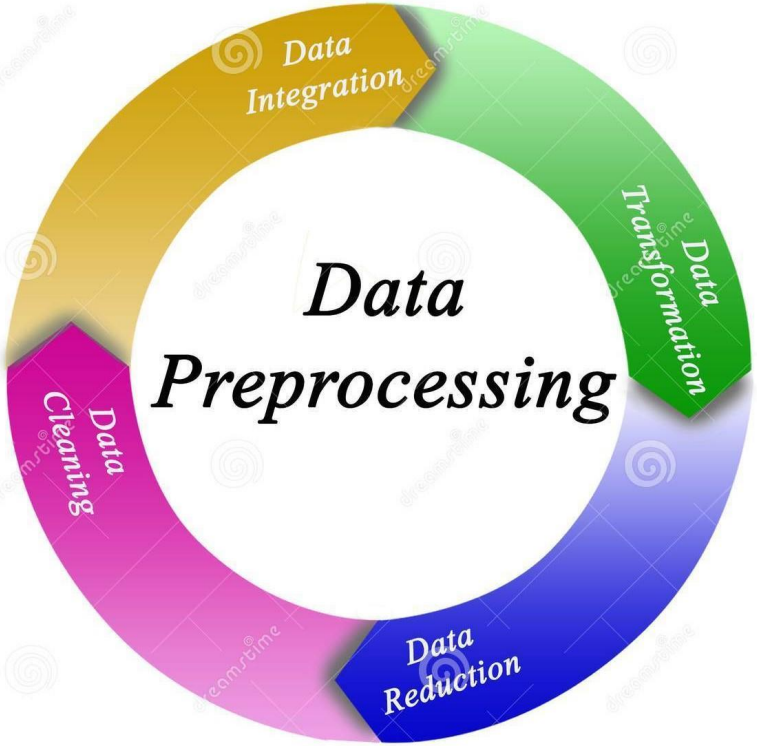
EDA - Target

Graphs/Visualizations



Preprocessing

- Data seemed to be pre cleaned
- Normalization, do we need to normalize? How do you know?
- Were there any NULL values or NA values?
- Were there a lot of zero values? Could we ignore them or not? Are they important or not?



Random Forests - Matthieu

Using the Random Forest Classifier We were able to get around a 95% accuracy.

And using Bootstrapping, we were able to get around the same accuracy.

Process:

- Small samples to test the functionality of the model
- Normalization
- Converted the target to numerical values
- Removed columns (loosely)
- Split the data into: X_train, X_test, y_train, y_test

Creating the model:

- Initializing the model
- Fitting the data
- Predicting the data
- Finding the accuracy of the data

Fine Tuning:

- Bootstrap (multiple samples)

Decision Trees

Accuracy of 98.8%

Without Time: ~94.94%

Process:

- Small samples to test the functionality of the model
- Normalization
- Converted the target to numerical values
- Removed columns (loosely)
- Split the data into: X_train, X_test, y_train, y_test

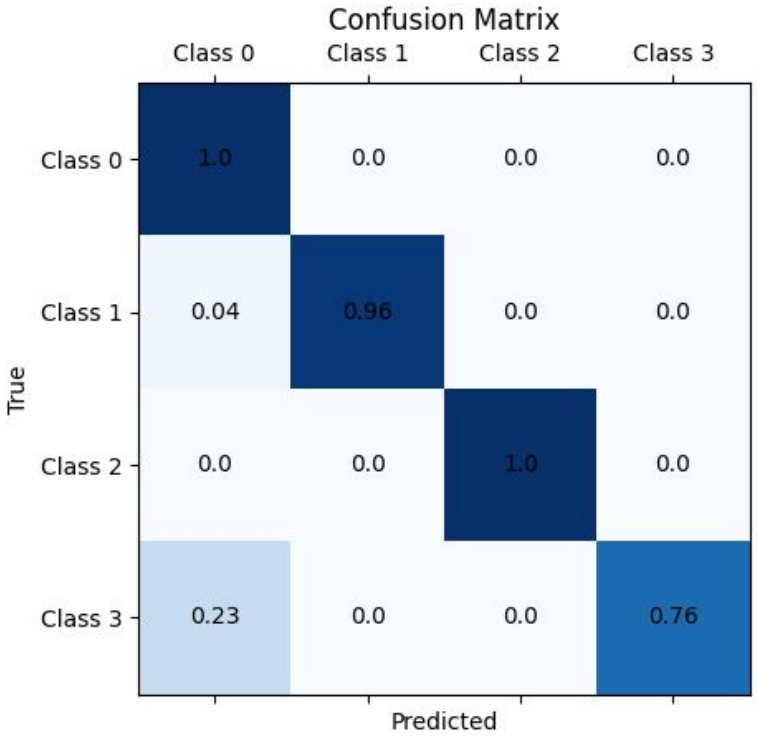
Creating the model:

- Initializing the model
- Fitting the data
- Predicting the data
- Finding the accuracy of the data

Fine Tuning:

- Bootstrap (multiple samples)

Gradient Boosting Machine




```
accuracy:  0.9844229751870855
```

- One of the best models we fit
- 99% accuracy with “time” column
  - Not useful for real life predictions
- 98% without “time” column
- Sometimes would get very low accuracy unexpectedly after training

```
Best parameters from Grid Search: {'learning_rate': 0.2, 'num_leaves': 300}
```