Distinguish abnormal individuals from Neck laser data

Supervised by: Professor Jie Wei

Presented by: Md Ayub Ali Sarker

Problem statement

- Neck laser data is collected from three group of peoples(A:18~30, B:31~50 and C:50+). Each observation contains multiple signal files, and each file is 1D signal values in time domain.

- We also have participant information that contains pulse and health issues of each individual.

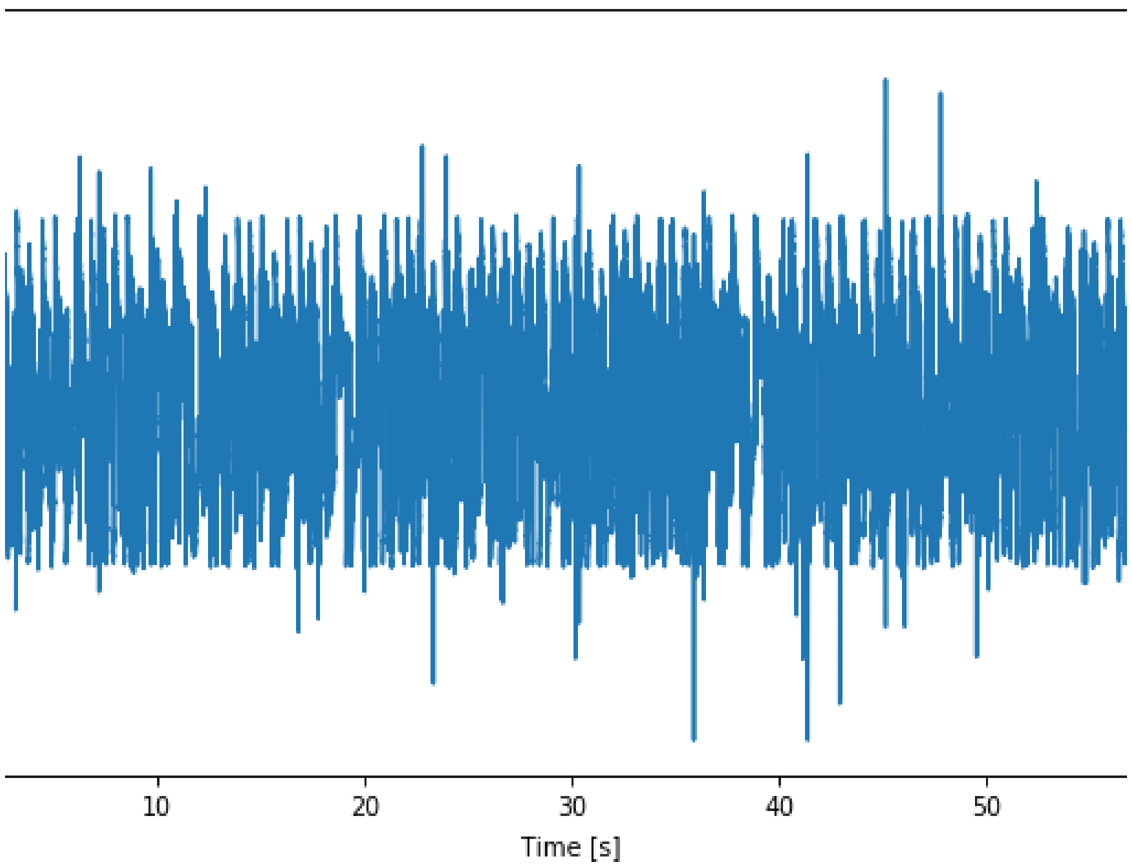
- Motivation: Our laser sensor is remote sensor. It can measure bio sign from 10m. In theory, it could be 400~500m. This can be used to determine remote bio sign. Like covid-19 and other serious illness. That's why I am motivated.

- In this work, We extracted features from signal in time and frequency domain, We determined heartbeat of each individual and health condition (Normal and Not Normal) using signal processing and machine learning technique.

Neck Laser Data

- Neck Laser data contains the human pulse vibrations over the neck artery collected by a laser droppler vibrometer. Each observation is collected for a person from multiple left and right-side scans and saved in .mat format. Each file is the 60 seconds duration with sampling rate 44,100 Hz. We have total of 235 mat files of 39 persons of three groups.

- We also have participant data that contains information like Health issue, Pulse, Age, Sex, Blood pressure, Ethnicity and Weight.



Solution to the problem

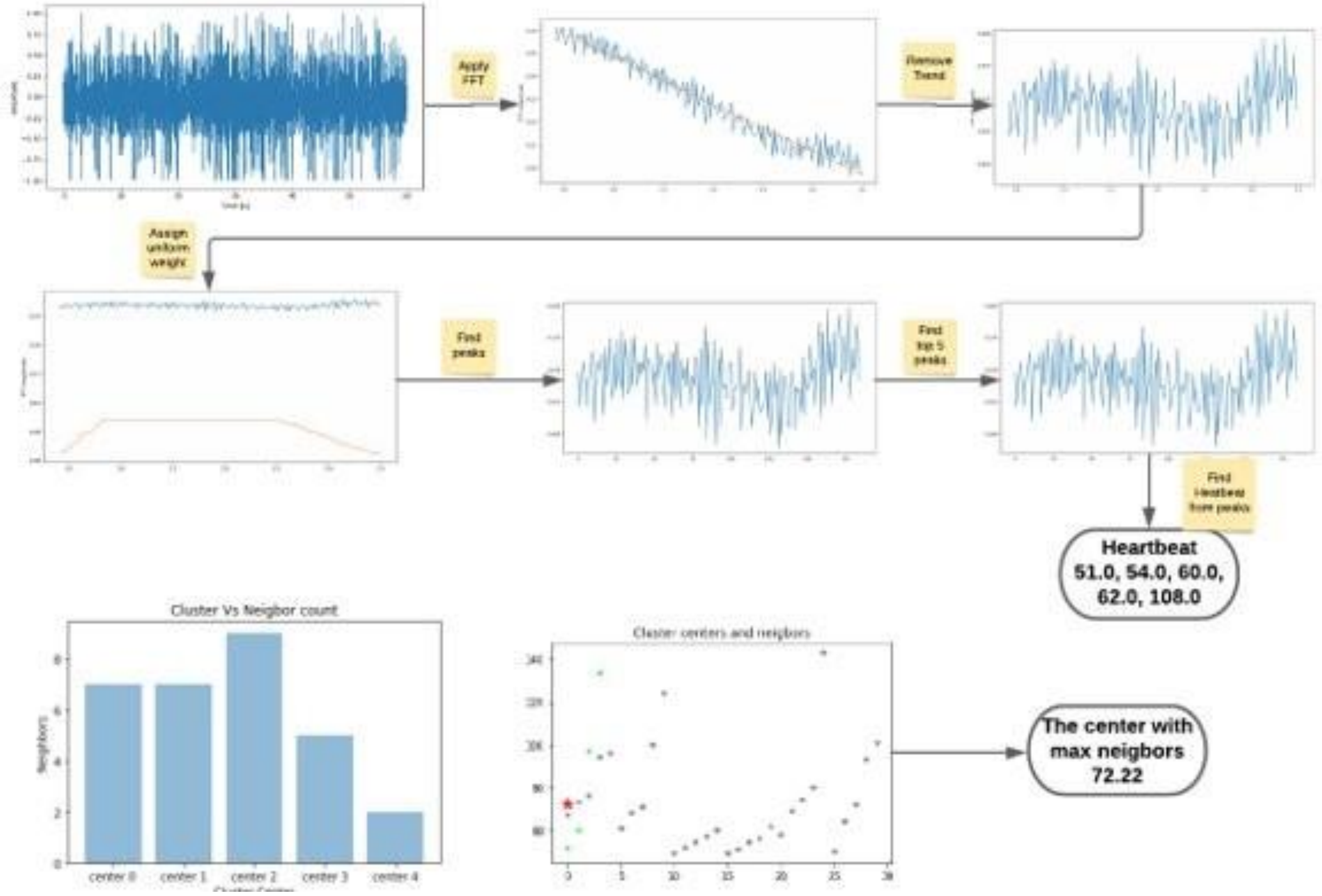
In order to address this problem, we did follow major tasks

- Heartbeat extraction
- Extract features in time and frequency domain
- Extract Level from participant data
- Feature Selection
- Feed the model with original data
- Generate 200 synthetic data using TCGAN
- Feed the model with original plus synthetic data

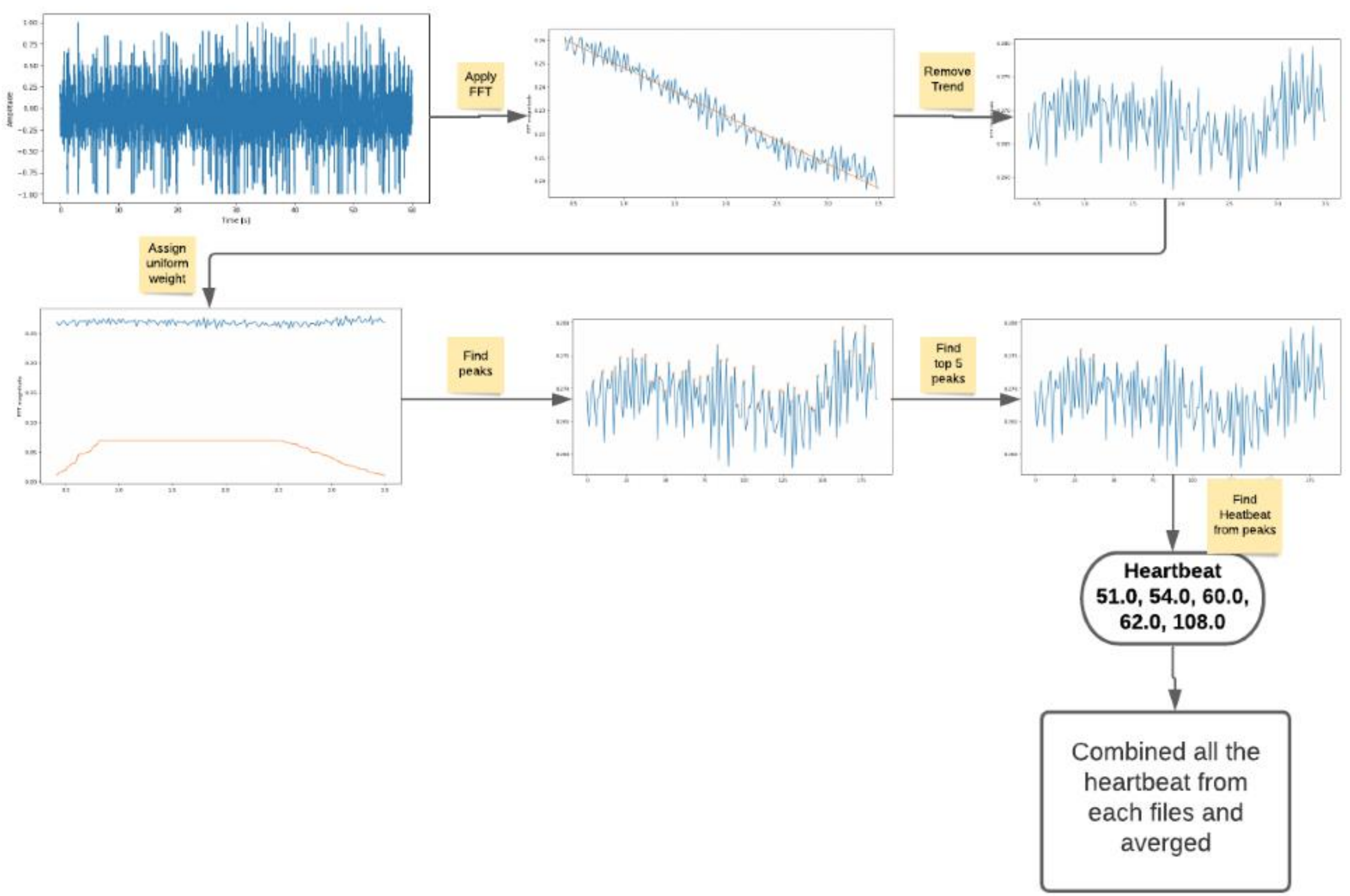
Heartbeat Extraction

- We developed two procedures to extract heartbeat from signal
  - Clustering Approach
  - Average Approach

Heartbeat Extraction-Clustering Approach

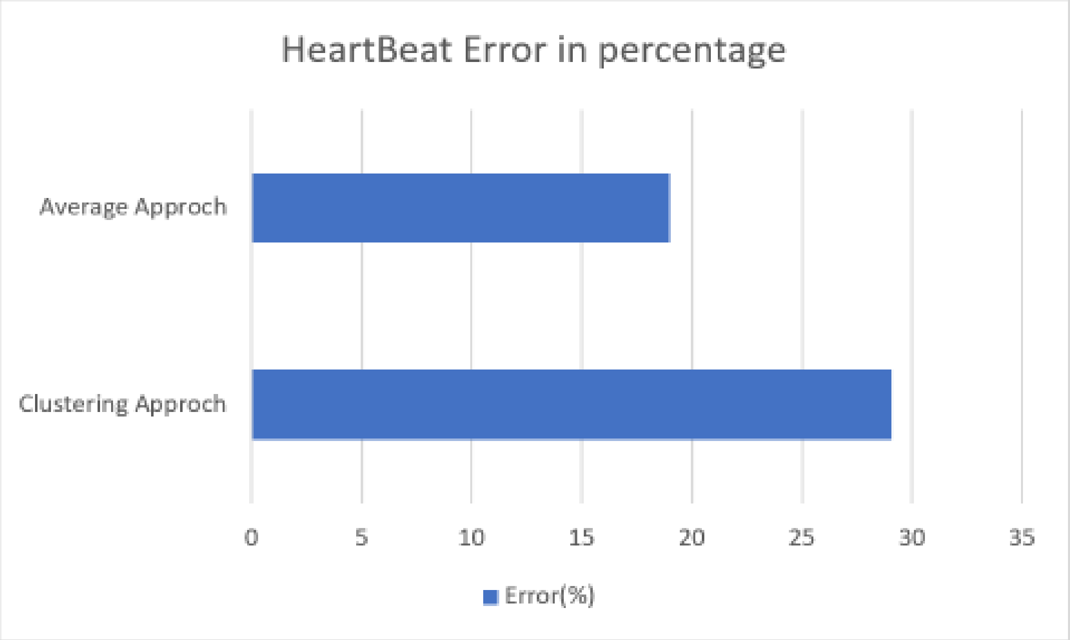


Heartbeat Extraction-Average Approach



Heartbeat Extraction

- We compared the derived heartbeat with actual heartbeat from the participation data. We calculated error of each approach.



- We took heartbeat calculated from average approach as the final heartbeat to use in later machine learning approach

Extract features in time and frequency domain

- **Feature extraction in frequency spectrum**: We divided the signal in frequency domain into six bands and found out peak frequency as a feature in that band using cluster approach. Those bands are
  - 0~0.7hz
  - 2.6~10hz
  - 11~20hz
  - 21~30hz
  - 31~40hz
  - 41~50hz

- **Feature extraction in time domain**: We extracted following features from the signal in time domain
  - Zero crossing rate average
  - Spectral rolloff average
  - Spectral centroid average
  - Spectral bandwidth average
  - Poly features average
  - RMS average
  - Spectral flatness average

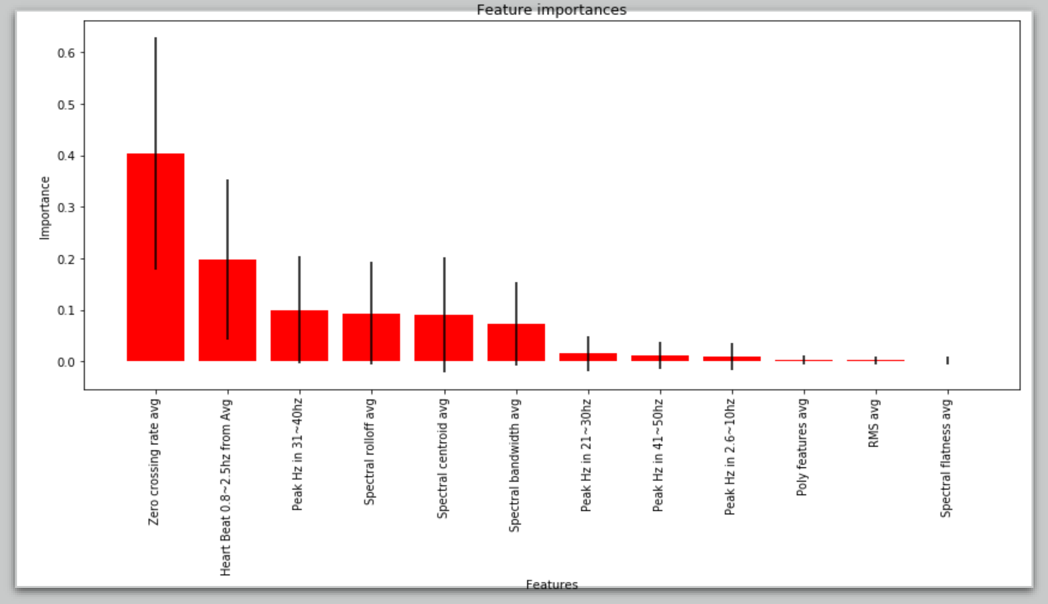
Extract Level from participant data

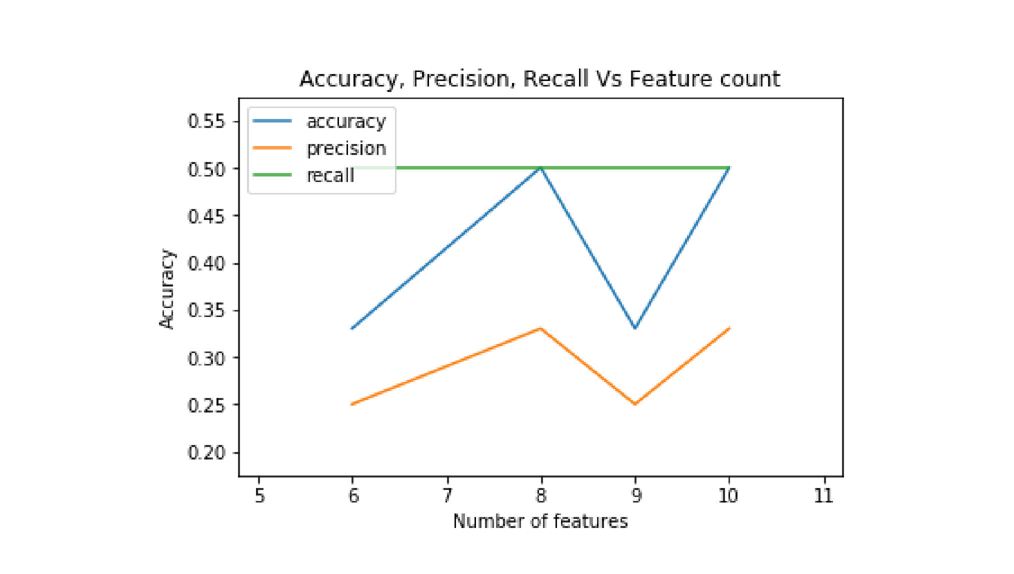
- Participant data contains health issues of each individual. We used this health issues information as an indication of health condition (Normal/ Not Normal)

Hz in hz	Peak Hz in 2.6~10hz	Peak Hz in 21~30hz	Peak Hz in 31~40hz	Peak Hz in 41~50hz	RMS avg	Zero crossing rate avg	Spectral flatness avg	Spectral rolloff avg	Spectral centroid avg	Poly features avg	Spectral bandwidth avg	Level
.0	5.50	23.33	31.00	43.67	0.2528	0.0280	0.0202	3931.6415	3655.5029	0.8441	5149.4551	abnormal
.0	3.12	30.00	40.00	43.25	0.1949	0.0372	0.0211	5369.9529	4887.3982	0.6950	6042.2007	normal
.0	3.00	24.71	39.50	41.50	0.2060	0.0275	0.0148	5252.5443	4598.1605	0.6630	6013.8062	abnormal
.0	4.00	25.00	40.00	50.00	0.2054	0.0720	0.0470	7429.2183	7041.1570	1.1817	6818.1680	normal
.0	5.78	22.40	38.78	45.00	0.2440	0.0008	0.0000	71.4086	128.8602	0.4138	605.7551	abnormal
.0	6.00	26.00	37.40	46.00	0.2539	0.0013	0.0000	72.5249	130.6257	0.4349	600.0443	normal
.0	6.12	23.88	40.00	49.75	0.2299	0.0423	0.0258	6426.0823	5796.0671	0.9369	6385.6767	normal
.0	10.00	21.09	40.00	43.00	0.2813	0.0764	0.0676	5556.0980	5706.2273	1.9134	5553.6838	abnormal
.0	3.00	26.00	37.00	50.00	0.3111	0.0170	0.0138	2613.0217	2380.4616	0.9283	4423.1727	normal
.0	9.09	29.67	40.00	43.67	0.3109	0.0597	0.0499	5209.9116	5016.2431	1.6762	5553.5699	normal
.0	9.10	26.50	36.00	50.00	0.2565	0.0009	0.0000	96.7716	147.9980	0.4341	623.3123	normal
.0	6.00	30.00	34.38	45.75	0.2195	0.0425	0.0248	6508.9343	5974.1852	0.8806	6485.1432	normal
.0	3.00	25.12	40.00	47.44	0.1906	0.0244	0.0104	5342.5278	4564.8693	0.5465	6166.3718	abnormal
.0	7.00	30.00	35.00	41.00	0.1900	0.0424	0.0211	6538.1594	5847.1872	0.6881	6555.1387	normal
.0	6.00	30.00	36.00	50.00	0.2009	0.0289	0.0161	5303.3936	4649.6131	0.6449	6116.2743	abnormal
.0	3.00	26.12	35.00	48.00	0.2271	0.0221	0.0125	4491.5204	3952.0604	0.6559	5555.7867	abnormal
.0	3.00	24.50	33.57	42.43	0.2035	0.0556	0.0397	6579.2238	5980.8607	1.0353	6496.4923	abnormal
.0	4.00	23.33	39.71	41.50	0.2212	0.0353	0.0225	4739.5686	4232.8570	0.8236	4682.7387	normal
.0	3.70	30.00	36.00	41.67	0.1796	0.0345	0.0163	6633.9151	5703.6169	0.5981	6663.8258	normal

Feature Selection

- We used RandomForestClassifier to see feature's importance.
- We can see the zero-crossing rate average has the highest contribution, then Heartbeat and peak frequency between 31~40Hz and then spectral Rolloff frequency and so on.
- In order to found out top important sets of features we feed RandomForestClassifier to five different sets of top important features [6, 8, 9, 10]. and plot the accuracy, precision and recall
- We can see that recall are same for all, but accuracy and procession are higher in 8 sets of features. So, we used first 8 importance features to feed our model





Feed the model with original data

Model	Accuracy
AdaBoostClassifier	67%
DecisionTreeClassifier	67%
KNeighborsClassifier	50%
RandomForestClassifier	50%

Generate 200 synthetic data using TCGAN

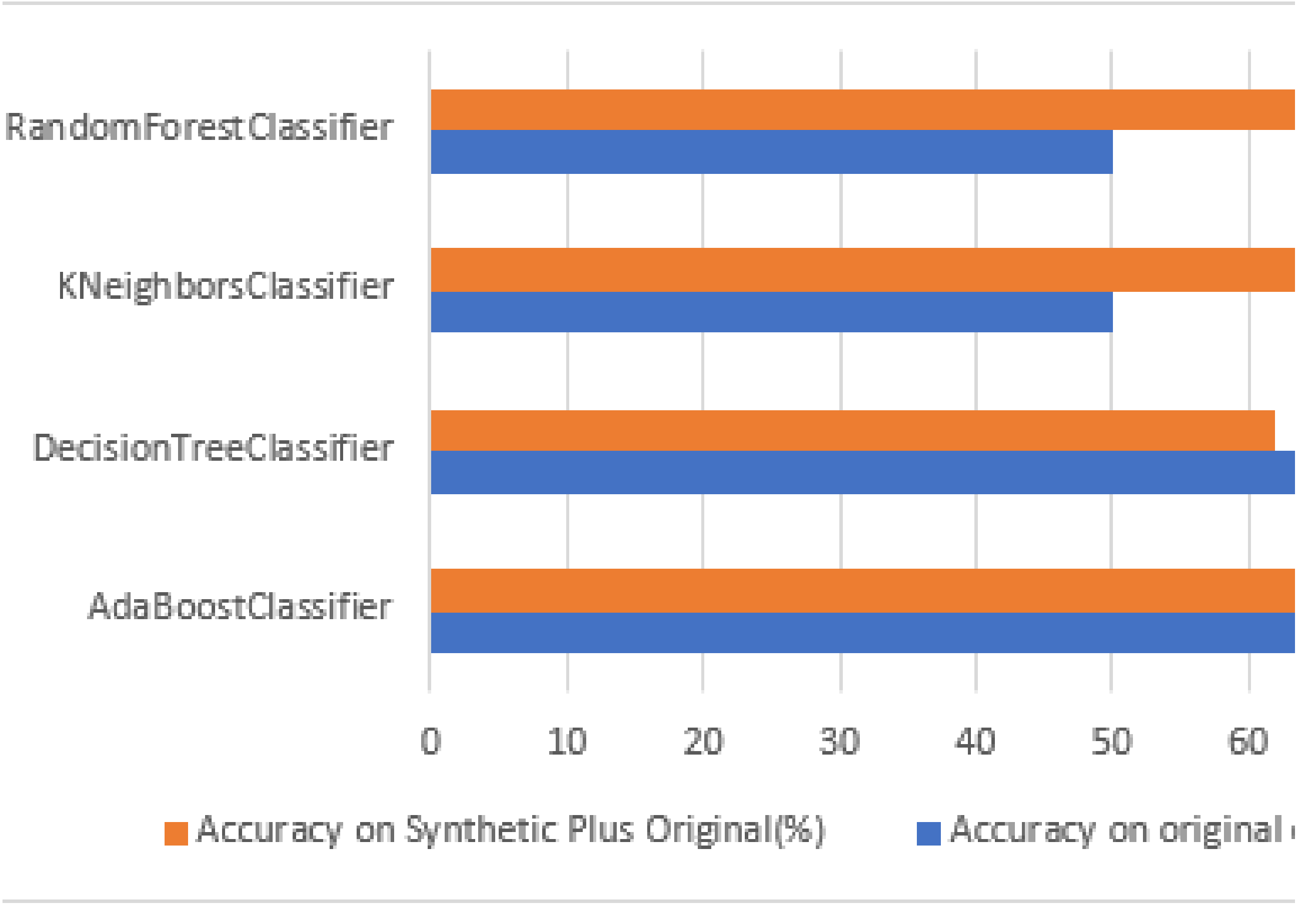
- We have small set of observations. A small data set suffers from high bias and overfitting. Getting more data in our case it very difficult and costly.

- We used TCGAN to generate 200 synthetic data. We used that 200 synthetic plus original data to feed our model.

Feed the model with original plus synthetic data

MODEL	ACCURACY
AdaBoostClassifier	71%
DecisionTreeClassifier	62%
KNeighborsClassifier	74%
RandomForestClassifier	73%

Summary of result



- Looking at the result we can see that Adaboost has moderate accuracy as 67% original and 71% in synthetic plus original data.
- Although Random forest(73) and KNN(74) has high accuracy in synthetic plus original data but they have low accuracy in original data
- So Adaboost is the best choice for our data set

Conclusion and Future plan

- We have small set of data. Most experimental involving primary research with real people have small data due to cost of conduction in person. In our case, this collection process is costly, too difficult and time-consuming. That's why we used to TCGAN to generate some synthetic data.
- We tried four different classifiers like AdaBoost, Decision Tree, KNN and Random Forest. We saw that Adaboost is the best choice.
- In future we planned to extend this work to identify sex and age group of individual
- https://github.com/msarker000/dse-capstone

Question?