Internship Report: Parkinson’s Disease Detection

Novelty: Generating and training on synthetic dataset

Adithi Sadananda Upadhya

Summer Intern (Digital), Legato Health Technologies
B.Tech ECE, National Institute of Technology Karnataka, Surathkal
AdithiSadananda.Upadhya@anthem.com
adithi.upadhya710@gmail.com
(Duration: May 16, 2022 - Jul 11, 2022)

Team Members:
Sonu Besra
sonu.besra2@anthem.com
Harsh Tanwar
harsh.tanwar@anthem.com

Supervisor:
Nageshwara Rao Moova
NageshwaraRao.Moova@legato.com

Manager:
Muralidhar Boga
Muralidhar.Boga@legato.com

Acknowledgement

I want to thank my advisers and everyone at Legato for their assistance during my on-site training. Thanks to their guidance, I was able to develop relevant skills and use the necessary tools and frameworks to develop synthetic models for training and testing, and learn more about data analysis, feature selection, data leakage and auto-ml based training. These skills will help me to expand my resume and advance my career.

Table of contents

Project Report

Introduction

Parkinson's disease (PD) is the second most common neurodegenerative disorder after Alzheimer's disease, affecting an estimated 7 to 10 million people worldwide (source: Parkinson's Disease Foundation).

Traditional diagnosis of PD involves a physician taking a neurological history of the patient and performing an examination of a variety of motor skills. Since there is no definitive diagnostic test, the task is often difficult, particularly in the early stages when motor symptoms are not severe. Symptoms can be so subtle in these first stages that they go unnoticed, leaving the disease undiagnosed or misdiagnosed for extended periods of time.

Since the very early stages of PD, there can be subtle abnormalities in speech that might not be perceptible to listeners, but they could be evaluated in an objective way by performing acoustic analyses on recorded speech signals. Vocal impairment can be one of the earliest indicators of PD. The development of accurate systems considering features extracted from voice recordings can be very useful to help diagnose PD in its early stages.

Problem Statement

The aim of the project was to develop a novelty over the traditional PD-detection system by using synthetic data. To overcome the shortcoming of minimal data points, the model has been trained and tested on synthetic data generated from Gaussian Copula, CTGAN, Copula GAN and TVAE models.

Technical Descriptions (Literature Review)

ID	Recording	Status	Gender	Jitter_rel	Jitter_abs	Jitter_RAP	Jitter_PPQ	Shim_loc	Shim_dB	Shim_APQ3	Shim_APQ5	Shi_APQ11	HNR05	H
CONT-01	1	0	1	0.25546	0.000015	0.001467	0.001673	0.030256	0.26313	0.017463	0.019660	0.021882	59.437966	60.7
CONT-01	2	0	1	0.36964	0.000022	0.001932	0.002245	0.023146	0.20217	0.013010	0.014097	0.016828	59.838895	62.6
CONT-01	3	0	1	0.23514	0.000013	0.001353	0.001546	0.019338	0.16710	0.011049	0.012683	0.013038	57.293808	61.8
CONT-02	1	0	0	0.29320	0.000017	0.001105	0.001444	0.024716	0.20892	0.014525	0.015696	0.018330	62.179573	68.6
CONT-02	2	0	0	0.23075	0.000015	0.001073	0.001404	0.013119	0.11607	0.006461	0.008385	0.011037	67.534024	74.9

Methods of data collection:

Original Data Source:

UCI Data Repository
Carlos J. PÃ©rez
Departamento de MatemÃ¡ticas, Universidad de Extremadura, CÃ¡ceres (Spain)
Email: carper '@' unex.es

Participants involved:

A total of 80 subjects older than 50 years were involved in the study.

- 40 of them were healthy: 22 men (55%) and 18 women (45%)
- 40 of them were affected by PD: 27 men (67.5%) and 13 women (32.5%)

The mean (± standard deviation) age was 66.38 ± 8.38 for the control group and 69.58 ± 7.82 for the people with PD.

PD patients presented at least two of the following symptoms: resting tremor, bradykinesia or rigidity.

Speech recordings

The vocal task was the sustained phonation of /a/ vowel at comfortable pitch and loudness, as constant as possible. This phonation had to be kept for at least 5 seconds and on one breath. The task was repeated three times per individual, and all of them were considered as replications.

The speech data were recorded using a portable computer with an **external sound card (TASCAM US322) and a headband microphone (AKG 520)**featuring a cardioid pattern. The digital recording was performed at a **sampling rate of 44.1 KHz and a resolution of 16 bits/sample** by using **Audacity** software (release 2.0.5).

240 rows (80 subjects × 3 replications) and 44 columns (one for every voice feature) are present in the dataset.

The **columns of the dataset** represent the following:

1. ID: Subject's identifier.
2. Recording: Number of the recording.
3. Status: 0=Healthy; 1=PD
4. Gender: 0=Man; 1=Woman
5. Pitch local perturbation measures: relative jitter (Jitter_rel), absolute jitter (Jitter_abs), relative average perturbation (Jitter_RAP), and pitch perturbation quotient (Jitter_PPQ).
6. Amplitude perturbation measures: local shimmer (Shim_loc), shimmer in dB (Shim_dB), 3-point amplitude perturbation quotient (Shim_APQ3), 5-point amplitude perturbation quotient (Shim_APQ5), and 11-point amplitude perturbation quotient (Shim_APQ11).
7. Harmonic-to-noise ratio measures: harmonic-to-noise ratio in the frequency band 0-500 Hz (HNR05), in 0-1500 Hz (HNR15), in 0-2500 Hz (HNR25), in 0-3500 Hz (HNR35), and in 0-3800 Hz (HNR38).
8. Mel frequency cepstral coefficient-based spectral measures of order 0 to 12 (MFCC0, MFCC1,..., MFCC12) and their derivatives (Delta0, Delta1,..., Delta12).
9. Recurrence period density entropy (RPDE).
10. Detrended fluctuation analysis (DFA).
11. Pitch period entropy (PPE).
12. Glottal-to-noise excitation ratio (GNE).

The acoustic features represented by the columns can be briefly divided into:

- pitch local perturbation measures
- amplitude
- noise features
- spectral envelope measures
- nonlinear ones

Local perturbation measures: jitter relative (expressed in percentage), jitter absolute, jitter RAP (Relative Average Perturbation) and jitter PPQ (Pitch Perturbation Quotient) were extracted by using a **waveform matching algorithm** consisting in two steps:

1. Calculation of the rough fundamental period length by using the normalized auto-correlation function over 80 ms frames and,
2. Second application of the auto-correlation function on segments with a length equal to twice the mean value of the fundamental periods roughly estimated before, after removing gross pitch errors (halving and doubling) and unvoiced frames.

- **Jitter (rel)**: This is the average absolute difference between consecutive periods, divided by the average period. MDVP calls this parameter Jitt and gives 1.040% as a threshold for pathology.
- **Jitter (local, absolute)**: This is the average absolute difference between consecutive periods, in seconds. MDVP calls this parameter Jita and gives 83.200 μs as a threshold for pathology.
- **Jitter (rap)**: This is the Relative Average Perturbation, the average absolute difference between a period and the average of it and its two neighbours, divided by the average period. MDVP gives 0.680% as a threshold for pathology.
- **Jitter (ppq5)**: This is the five-point Period Perturbation Quotient, the average absolute difference between a period and the average of it and its four closest neighbours, divided by the average period. MDVP calls this parameter PPQ and gives 0.840% as a threshold for pathology.

Amplitude perturbation measures: shimmer local, shimmer dB, APQ3 (3-point Amplitude Perturbation Quotient), APQ5 (5-point Amplitude Perturbation Quotient) and APQ11 (11-point Amplitude Perturbation Quotient)

- **Shimmer (local)** This is the average absolute difference between the amplitudes of consecutive periods, divided by the average amplitude. MDVP calls this parameter Shim and gives 3.810% as a threshold for pathology.
- **Shimmer (local, dB)** This is the average absolute base-10 logarithm of the difference between the amplitudes of consecutive periods, multiplied by 20. MDVP calls this parameter ShdB and gives 0.350 dB as a threshold for pathology.
- **Shimmer (apq3)** This is the three-point Amplitude Perturbation Quotient, the average absolute difference between the amplitude of a period and the average of the amplitudes of its neighbors, divided by the average amplitude.
- **Shimmer (apq5)** This is the five-point Amplitude Perturbation Quotient, the average absolute difference between the amplitude of a period and the average of the amplitudes of it and its four closest neighbors, divided by the average amplitude.
- **Shimmer (apq11)** This is the 11-point Amplitude Perturbation Quotient, the average absolute difference between the amplitude of a period and the average of the amplitudes of it and its ten closest neighbors, divided by the average amplitude. MDVP calls this parameter APQ and gives 3.070% as a threshold for pathology.

Noise features: Harmonic-to-noise ratio (HNR) is a measure of the relative level of noise present in speech. The dataset uses 5 HNR features, corresponding to different frequency bandwidths:

- HNR05 (0-500 Hz)
- HNR15 (0-1500 Hz)
- HNR25 (0-2500 Hz)
- HNR35 (0-3500 Hz)
- HNR38 (0-3800 Hz)

Individual HNR values for each frame are extracted by using the **VoiceSauce toolbox**. These HNR measures are then calculated using a cepstrum-based technique. The final HNR features are calculated as average values of all voiced frames.

Glottal-to-Noise Excitation Ratio (GNE) attempts to quantify the amount of voice excitation by vocal-fold oscillations versus excitation by turbulent noise.

Spectral envelope measures: Mel Frequency Cepstral Coefficients (MFCCs) are related to the speech spectral envelope, which depends on articulator position, thus making it possible to detect slight misplacement of the articulators

- PD is known to affect also articulation; therefore, this type of coefficients are promising features to characterize PD

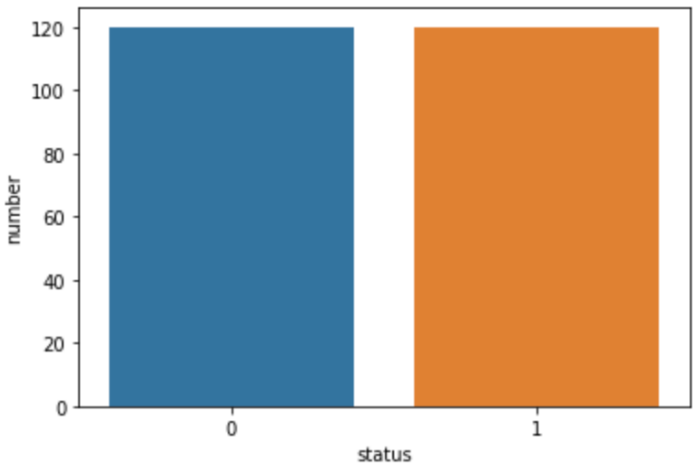
Delta MFCC features are time derivatives of MFCCs, so they can be used to detect subtle changes in the articulator positions (due to tremor) when performing a sustained phonation.

- The length of the feature vector has been chosen to be 26 (13 MFCCs plus 13 Delta coefficients).

Non linear features (of importance because healthy voices are closer to the linear source-filter model): RPDE (Recurrence Period Density Entropy), DFA (Detrended Fluctuation Analysis) and PPE (Pitch Period Entropy)

Exploratory Data analysis

An **absence of data imbalance** was found, as an equal number of Status=0 (Healthy) and Status=1(Diagnosed with Parkinson's) cases were present in the dataset.



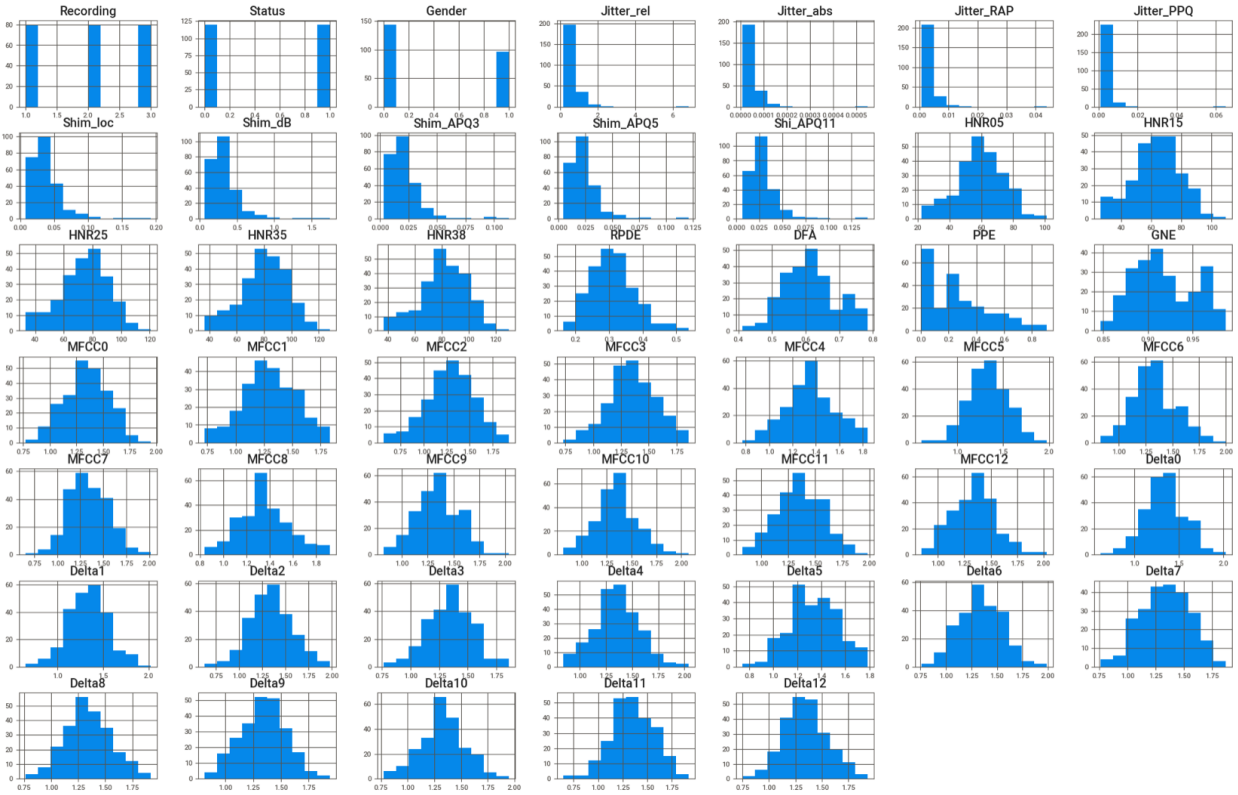
Histogram plots, Bar plots, Gaussian distribution plots, box plots (to identify outliers) and heatmap (to analyze the correlations between the columns) were plotted for better understanding.

This was followed by analyzing all the parameters by comparing the ‘Male’ and ‘Female’ features across all columns using ***sweetviz*** library.

```
report_1 = sweetviz.compare_intra(df, df["Gender"] == 0, ["Male", "Female"], 'Status')
report_1.show_html('Report.html')
```

The comparative-analysis report can be found here.

Sample Histogram plots obtained:



Generation of Synthetic Data

Synthetic data is created algorithmically, and it is used as a stand-in for test datasets of production or operational data, to validate mathematical models and, increasingly, to train machine learning models.

Due to lack of data (240 rows), an additional set of synthetic data of 5000 rows was generated using Gaussian Copula, CTGAN, Copula GAN and TVAE models from the ***sdv.tabular*** library.

sdv.evaluation.evaluate function applies a collection of pre-configured metric functions and returns the average of the scores that the data obtained on each one of them. Both the real and synthetic data are passed as parameters, the output of this function call will be a number between 0 and 1 that will indicate how similar the two tables are, being 0 the worst and 1 the best possible score.

Gaussian Copula

sdv.tabular.GaussianCopula is based on copula functions. In mathematical terms, a copula is a distribution over the unit cube $[0,1]^d$ which is constructed from a multivariate normal distribution over R^d by using the probability integral transform.

	ID	Recording	Status	Gender	Jitter_rel	Jitter_abs	Jitter_RAP	Jitter_PPQ	Shim_loc	Shim_dB	...	Delta3	Delta4	Delta5	Delta6	Delta7	Delta8
0	PARK-25	2	1	0	0.787376	0.000045	0.004457	0.006804	0.060453	0.504122	...	1.510343	1.371496	1.405242	1.422815	1.389876	1.355652
1	PARK-06	2	1	0	1.356952	0.000045	0.005268	0.009093	0.044511	0.406728	...	1.023869	0.954249	1.257329	0.900434	1.305676	1.012784
2	CONT-22	2	0	1	0.712573	0.000044	0.005110	0.005805	0.043585	0.361679	...	1.280589	1.463399	1.332550	1.381185	1.482621	1.204927
3	PARK-29	1	1	1	0.473174	0.000044	0.002792	0.003924	0.019245	0.161346	...	1.331089	1.220882	1.191828	1.381003	1.071129	1.255173
4	PARK-23	2	1	1	0.186585	0.000044	0.000883	0.001414	0.021512	0.181176	...	1.259572	1.241845	1.282063	1.176407	1.244194	1.223163

sdv.evaluation.evaluate score obtained: 0.6605126791826635

Further, the dataset was found to be pretty balanced and **0 duplicates** were present:

Male		Female
2890	ROWS	2110
0	DUPLICATES	0
1.3 MB	RAM	945.3 kb
48	FEATURES	48
4	CATEGORICAL	4
44	NUMERICAL	44
0	TEXT	0

(Fig. source: ***sweetviz*** analysis)

CTGAN

[sdv.tabular.CTGAN] is a collection of Deep Learning based Synthetic Data Generators for single table data, which are able to learn from real data and generate synthetic clones with high fidelity.

sdv.evaluation.evaluate score obtained: 0.38396108754316594

Copula GAN

[sdv.tabular.CopulaGAN] is a variation of the CTGAN model which takes advantage of the CDF based transformation that the GaussianCopulas apply to make the underlying CTGAN model task of learning the data easier.

sdv.evaluation.evaluate score obtained: 0.3761759312832603

TVAE

[sdv.tabular.TVAE] is based on the Variational Auto-encoder based Deep Learning data synthesizer.

sdv.evaluation.evaluate score obtained: 0.6278341755108625

Gaussian Copula and TVAE based synthetic models were found to have around 65% similarity with the original dataset whereas Copula GAN and CTGAN posed around 40% similarity with the original dataset.

Feature Selection

To reduce the number of features (48) for better modeling, the following procedures were followed:

1. Feature selection using algorithms like MRMR and chi square test
2. Manual selection by analyzing the correlations (heatmap) and results obtained from (1)

MRMR (Minimum Redundancy - Maximum Relevance) finds the smallest relevant subset of features for a given Machine Learning task. The top 25 features of the dataset were identified using ***mrmr_selection*** library.

This was followed by **manually analyzing** the top features using the information learnt through the literature review and correlation map. The dimensionality of all the MFCC features (26) were reduced to 10 using Principal Component Analysis (**PCA**). The HNR features were dropped as they had less relevance to the problem statement. The final dataset to be trained on was reduced to 23 columns post feature selection and reduction.

AutoML based Training

In order to select the best possible training model for the synthetic datasets TPOT library was used. **Tree-Based Pipeline Optimization Tool (TPOT)** - automates the building of ML pipelines by combining a flexible expression tree representation of pipelines with stochastic search algorithms such as genetic programming. TPOT makes use of the Python-based scikit-learn library as its ML menu.

The optimum models and cross validation scores for the synthetic datasets are listed below:

Gaussian Copula

```
Average CV score on the training set was:  0.7464
Best pipeline: SGDClassifier(RobustScaler(input_matrix), alpha=0.001, eta0=0.01,
fit_intercept=True, l1_ratio=0.25, learning_rate=constant, loss=hinge,
penalty=elasticnet, power_t=0.0)
TPOTClassifier(cv=StratifiedKFold(n_splits=10, random_state=None,
shuffle=False),generations=5, n_jobs=-1, population_size=50, random_state=1,
scoring='accuracy', verbosity=2)
```

CTGAN

```
Average CV score on the training set was: 0.6476
Best pipeline: LinearSVC(ZeroCount(input_matrix), C=10.0, dual=False,
loss=squared_hinge, penalty=l1, tol=1e-05)
Pipeline(steps=[('zerocount', ZeroCount()),('linearsvc', LinearSVC(C=10.0,
dual=False, penalty='l1', random_state=1, tol=1e-05))])
```

Copula GAN

```
Average CV score on the training set was: 0.5346
Best pipeline: GaussianNB(RFE(input_matrix, criterion=gini,
max_features=0.7000000000000001, n_estimators=100, step=0.5))
Pipeline(steps=[('rfe',RFE(estimator=ExtraTreesClassifier(max_features=0.700000000000
0001,random_state=1),step=0.5)),('gaussiannb', GaussianNB())])
```

TVAE

```
Average CV score on the training set was: 0.9092
Best pipeline: RandomForestClassifier(input_matrix, bootstrap=True,
criterion=entropy, max_features=0.9500000000000001, min_samples_leaf=8,
min_samples_split=9, n_estimators=100)
```

```
Pipeline(steps=[('randomforestclassifier',RandomForestClassifier(criterion='entropy',
max_features=0.9500000000000001, min_samples_leaf=8, min_samples_split=9,
random_state=1))])
```

Model Testing

Trained on: Synthetic Data, Tested on: Original Data (240 rows)

Gaussian Copula

- Confusion matrix and classification report:

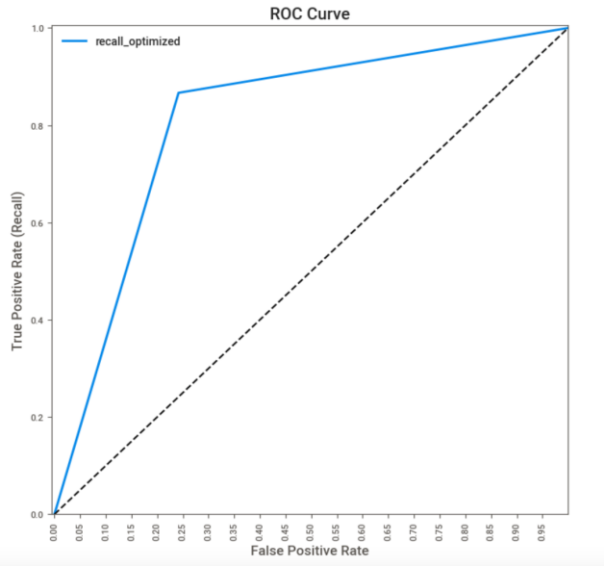
```
[[ 91  29]
 [ 16 104]]
              precision    recall  f1-score   support

           0       0.85      0.76      0.80       120
           1       0.78      0.87      0.82       120

    accuracy                           0.81       240
   macro avg       0.82      0.81      0.81       240
weighted avg       0.82      0.81      0.81       240
```

- AUC score and ROC curve:

```
0.8125
```



CTGAN

- Confusion matrix and classification report:

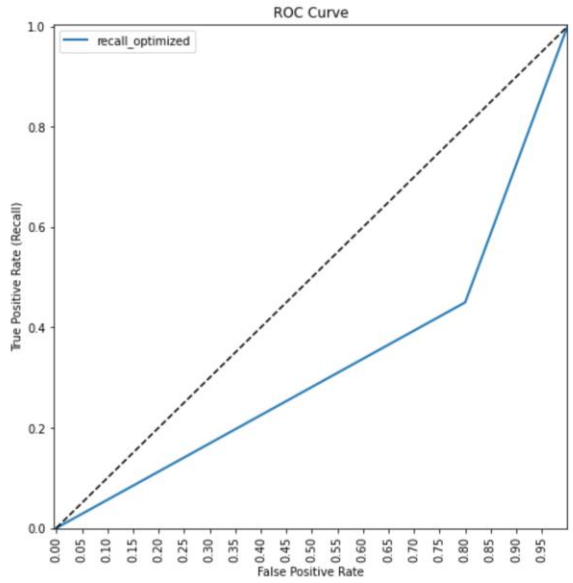
```
[[24 96]
 [66 54]]
              precision    recall  f1-score   support

           0       0.27      0.20      0.23       120
           1       0.36      0.45      0.40       120

    accuracy                           0.33       240
   macro avg       0.31      0.33      0.31       240
weighted avg       0.31      0.33      0.31       240
```

- AUC score and ROC curve:

```
0.32499999999999996
```



Copula GAN

- Confusion matrix and classification report:

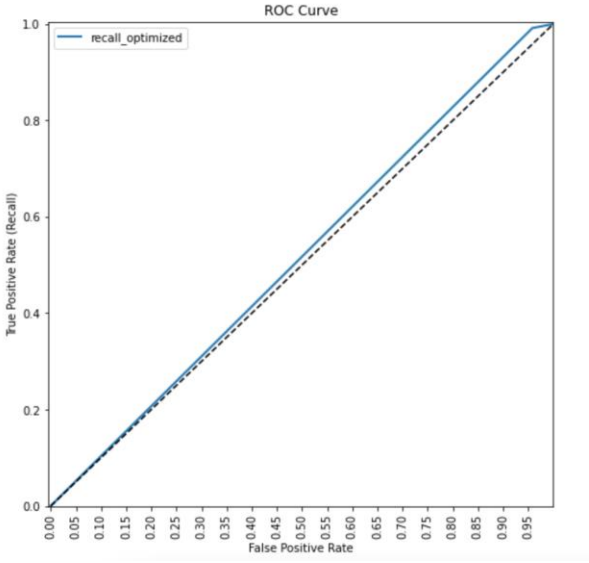
```
[[  5 115]
 [  1 119]]
              precision    recall  f1-score   support

           0       0.83      0.04      0.08       120
           1       0.51      0.99      0.67       120

    accuracy                           0.52       240
   macro avg       0.67      0.52      0.38       240
weighted avg       0.67      0.52      0.38       240
```

- AUC score and ROC curve:

```
0.5166666666666666
```



TVAE

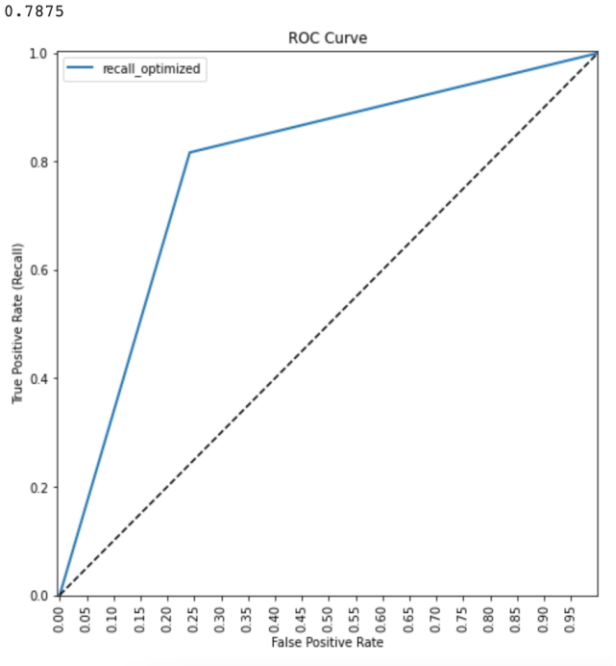
- Confusion matrix and classification report:

```
[[91 29]
 [22 98]]
              precision    recall  f1-score   support

           0       0.81      0.76      0.78       120
           1       0.77      0.82      0.79       120

    accuracy                           0.79       240
   macro avg       0.79      0.79      0.79       240
weighted avg       0.79      0.79      0.79       240
```

- AUC score and ROC curve:



More tests on Gaussian Copula based model

Trained on: Original Data (240 rows), Tested on: Synthetic Data

- **Model used:** ExtraTreesClassifier
- Confusion matrix and classification report:

```
[[791 457]
 [271 981]]
              precision    recall  f1-score   support

           0       0.74      0.63      0.68      1248
           1       0.68      0.78      0.73      1252

    accuracy                           0.71      2500
   macro avg       0.71      0.71      0.71      2500
weighted avg       0.71      0.71      0.71      2500
```

Trained and tested on combined dataset of real and synthetic data

- **Model used:** ExtraTreesClassifier
- Confusion matrix and classification report:

```
[[256  74]
 [ 98 257]]
              precision    recall  f1-score   support

           0       0.72      0.78      0.75       330
           1       0.78      0.72      0.75       355

    accuracy                           0.75       685
   macro avg       0.75      0.75      0.75       685
weighted avg       0.75      0.75      0.75       685
```

Hyperparameter tuning using GridSearchCV(SGD and Logistic)

- **SGD Classifier**

- The following parameters were used for identifying the optimum parameters for the SGD Classifier:

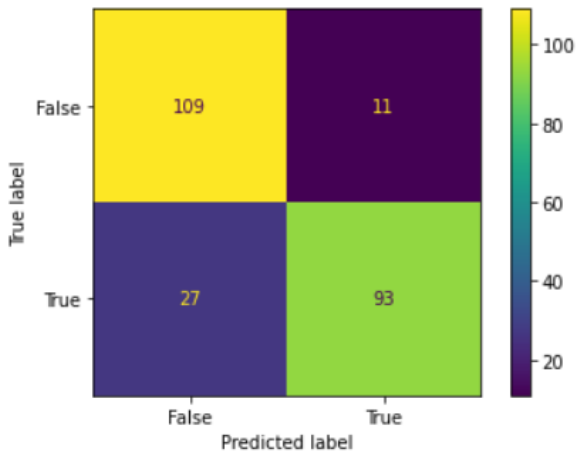
```
from sklearn.model_selection import GridSearchCV

params = {
    "loss" : ["hinge", "log", "squared_hinge", "modified_huber"],
    "alpha" : [0.0001,0.0001, 0.001, 0.01, 0.1, 1, 10],
    "penalty" : ["l2", "l1", "none"],
    'n_jobs': [-1]
}

model = SGDClassifier(max_iter=1000)
clf = GridSearchCV(model, param_grid=params)
```

- Best classifier: `SGDClassifier(alpha=0.01, loss='modified_huber', n_jobs=-1)`

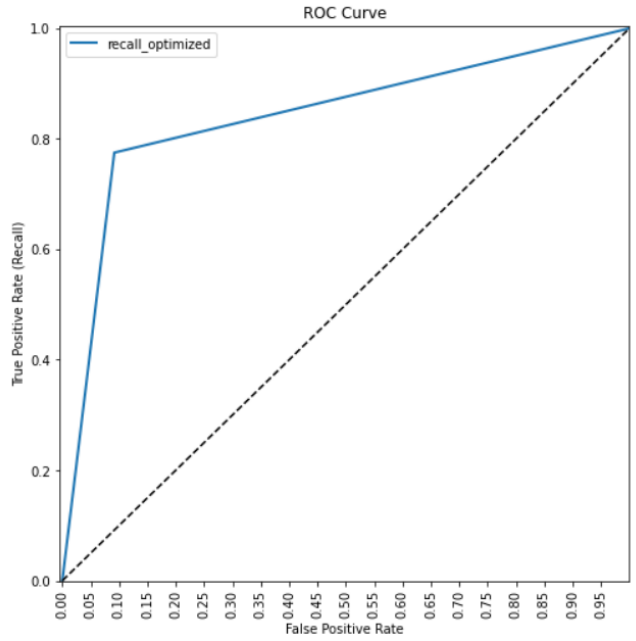
- Confusion matrix and classification report obtained for the best classifier:



	precision	recall	f1-score	support
0	0.80	0.91	0.85	120
1	0.89	0.78	0.83	120
accuracy			0.84	240
macro avg	0.85	0.84	0.84	240
weighted avg	0.85	0.84	0.84	240

- AUC Score and ROC curve:

```
AUC: 0.8416666666666667
```



- **<u>Logistic Regression</u>**

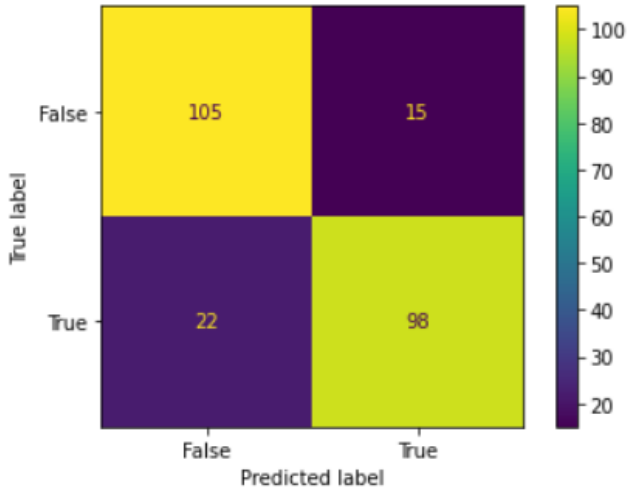
- The following parameters were used for identifying the optimum parameters for training:

```
from sklearn.linear_model import LogisticRegression
clf_log = LogisticRegression()
grid_values = {'penalty': ['l1', 'l2'],'C':np.logspace(-4, 4, 50), 'solver': ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga']}
grid_clf_acc = GridSearchCV(clf_log, param_grid = grid_values)
grid_clf_acc.fit(X_train, y_train)
```

- Best Classifier: `LogisticRegression(C=0.08685113737513521, penalty='l1', solver='liblinear')`

- Confusion matrix and classification report obtained for the best classifier:

	precision	recall	f1-score	support
0	0.83	0.88	0.85	120
1	0.87	0.82	0.84	120
accuracy			0.85	240
macro avg	0.85	0.85	0.85	240
weighted avg	0.85	0.85	0.85	240



Results

Table 1 - Optimum training models

Synthetic Data	Training Model
Gaussian Copula	SGDClassifier, Logistic Regression
CTGAN	LinearSVC
Copula GAN	GaussianNB
TVAE	RandomForestClassifier

Table 2 - Training model: RobustScaler & SGDClassifier for all the synthetic datasets, Tested on: Original Data (240 rows)

Synthetic Data	Recall score		Accuracy	AUC score
	1 (PD)	0 (Healthy)		
Gaussian Copula	0.76	0.87	0.81	0.812
CTGAN	0.20	0.45	0.33	0.325
Copula GAN	0.04	0.99	0.52	0.516
TVAE	0.76	0.82	0.79	0.787

It can be concluded that the best results were obtained for the synthetic model based on Gaussian Copula, closely followed by TVAE.

Table3: Hyperparmeter Tuning Results on Gaussian Copula, Test data: Original data

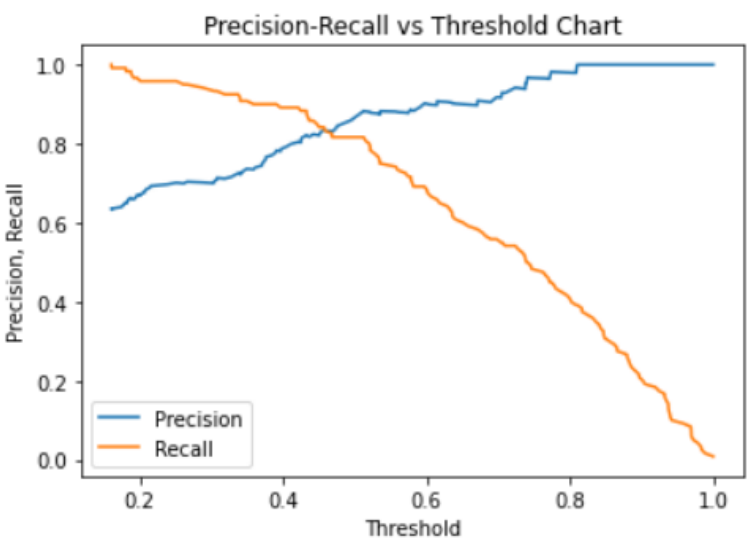
Model	Recall score		Accuracy	AUC score
	1 (PD)	0 (Healthy)		
`SGDClassifier(alpha=0.01, loss='modified_huber', n_jobs=-1)`	0.78	0.91	0.84	0.841
`LogisticRegression(C=0.08685113737513521, penalty='l1', solver='liblinear')`	0.82	0.88	0.85	0.85

Best model-

```
LogisticRegression(C=0.08685113737513521, penalty='l1', solver='liblinear')

Training Dataset- Synthetic Data (Gaussian Copula)
Testing Dataset- Original Data(240 rows)
Recall score- PD: 0.82 and Healthy: 0.88
Accuracy-0.85
AUC score- 0.85
```

Precision-recall vs Threshold to further optimise the recall for Logistic Regression model



From the graph, the **optimum threshold** was found to be `0.4576891645085667`.

Conclusion

A system that distinguishes people with PD from healthy controls based on acoustic features extracted from voice recordings has been developed. The optimum model was derived from the **Gaussian Copula synthetic model trained using SGD Classifier (~84% accuracy) and Logistic Regression (~85% accuracy)**. The identified optimum threshold for Logistic Regression (**0.457**) can be used to **further optimize the recall score.**

The proposed approach has been implemented and tested for two-classes classification problems, but it **can be easily extended to the multi-class problem with stage-wise PD detection**. Although the model has been developed w.r.t PD detection, it can be used in similar experiments for different purposes. Yet another future prospect could include the development of an intelligent telediagnosis system prototype based on mobile devices for PD detection.

Relevant Papers

Sajal, M., Rahman, S., Ehsan, M., Vaidyanathan, R., Wang, S., Aziz, T. and Mamun, K.A.A., 2020. Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis. Brain Informatics, 7(1), pp.1-11.

Naranjo, L., PÃ©rez, C.J., Campos-Roca, Y., MartÃ-n, J.: Addressing voice recording replications for Parkinsonâ€™s disease detection. Expert Systems With Applications 46, 286-292 (2016)

Winursito, A., Hidayat, R., Bejo, A. and Utomo, M.N.Y., 2018, July. Feature data reduction of MFCC using PCA and SVD in speech recognition system. In 2018 international conference on smart computing and electronic Enterprise (ICSCEE) (pp. 1-6). IEEE.