BREAST CANCER PREDICTION

CN240 Data Science

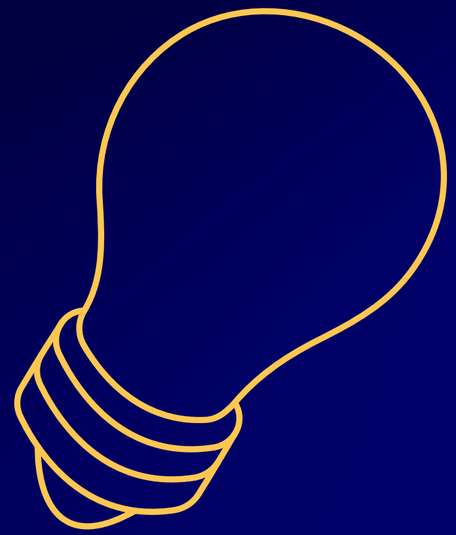
Faculty of Engineering, Thammasat School of Engineering
Thammasat University Rangsit Campus

PRESENTED TO

Prof.Dr.Charturong Tantibundhit

Member Group

- Kun Kerdthaisong 6410685074
- Nattapat Chotisen 6410685116
- Kittisak Suddaen. 6410685082
- Thanakorn Praimanee 6410685157
- Prin Yimrungruang 6410685199
- Natchanon Chanrungrojne 6410615048
- Akapol Mueangham 6410685025



Outline

Problem Statements

Breast cancer is a common and potentially life-threatening disease that requires early detection for optimal treatment and survival. However, current diagnostic and prediction methods can be invasive and costly. To improve patient outcomes, there is a need to develop accurate and reliable breast cancer prediction models based on clinical and demographic factors.

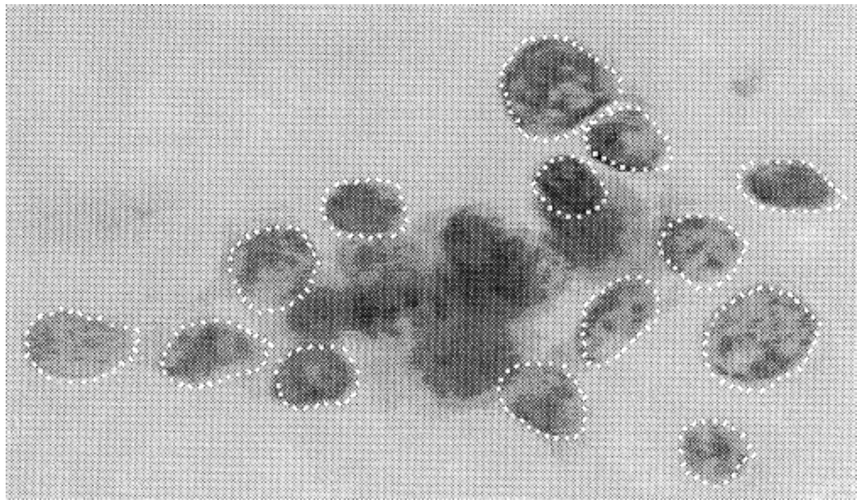
The goal of this study is to develop a predictive model that can identify patients at high risk of developing breast cancer and aid in early detection and treatment planning.

Literature Reviews

Paper

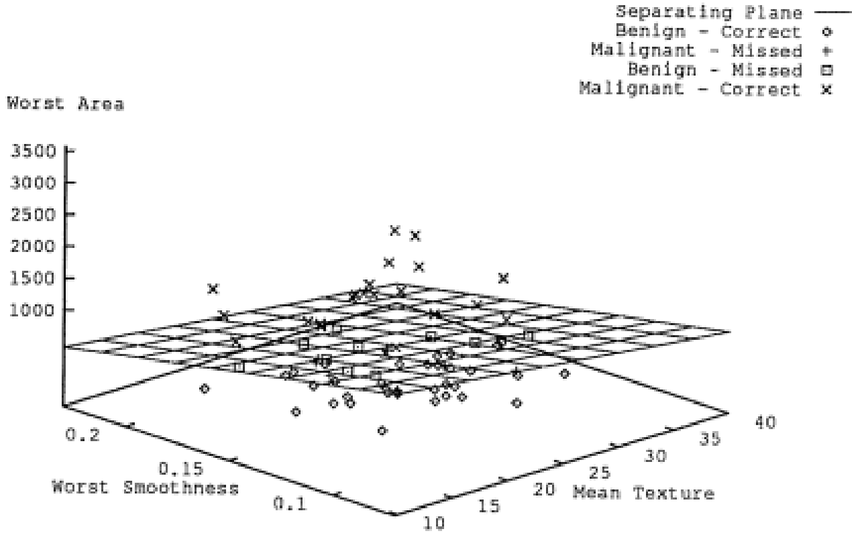
Nuclear Feature Extraction For Breast Tumor Diagnosis (December 1992) **W. Nick Street ,et al**

- **Image preparation**
- **Snakes (active contour)**



Initial Approximate Boundaries of Cell Nuclei.

- **MSM-t(multi surface method-tree)**



Separating Plane in Three Dimensions, Only 10% of correctly classified benign and malignant points are shown here.

10 fold cross-validation =>acc 97.3%

Data Collection

Breast Cancer Wisconsin (Diagostic) Data Set

3 types nuclei's cell that have been collected mean, standard error, worst

1. radius
2. texture
3. perimeter
4. area
5. smoothness
6. compactness
7. concavity
8. concave points
9. symmetry
10. fractal dimension

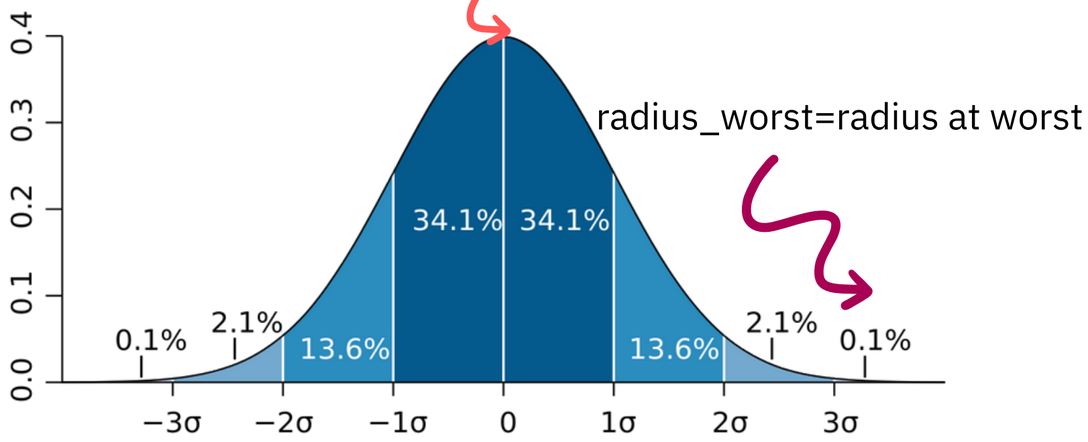
11. ID number
12. Diagnosis
13. Unnamed: 32

Data Collection

collect nuclei's cell's data : mean, standard error, worst.

① radius 1 people **n = 100 cells**

radius_mean=at mean



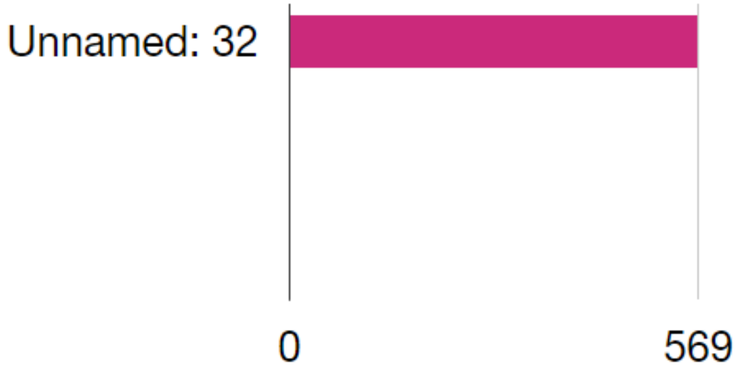
radius_se=radius_sd/sqrt(n)

df[["radius_mean","radius_se","radius_worst"]]

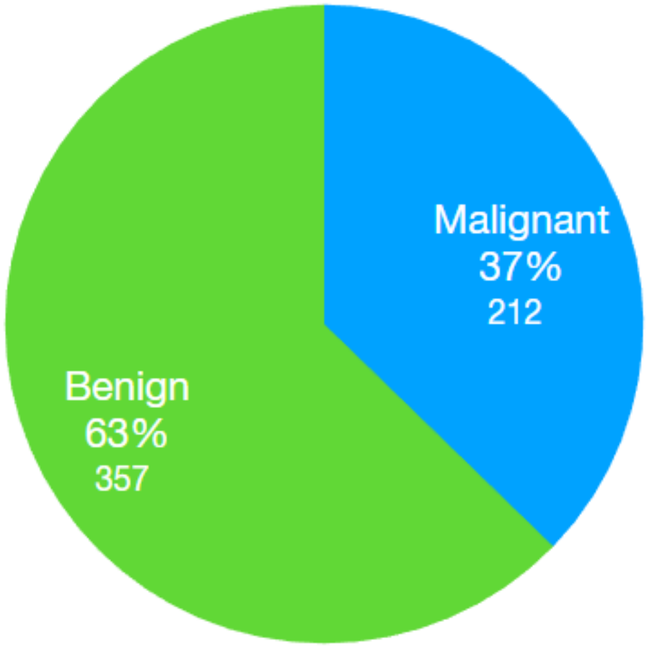
	radius_mean	radius_se	radius_worst
0	17.99	1.0950	25.380
1	20.57	0.5435	24.990
2	19.69	0.7456	23.570
3	11.42	0.4956	14.910
4	20.29	0.7572	22.540

Data Analysis

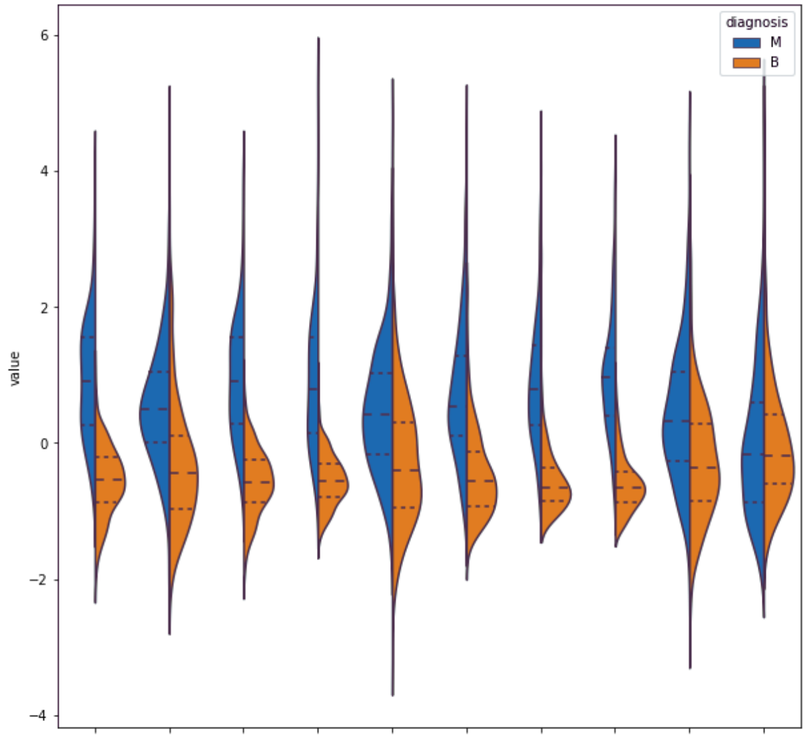
1. Check duplicated id
   : using df['id'].nunique( )
2. Check NAN value in all class
   : using df.isna( ).sum( )



Class ratio

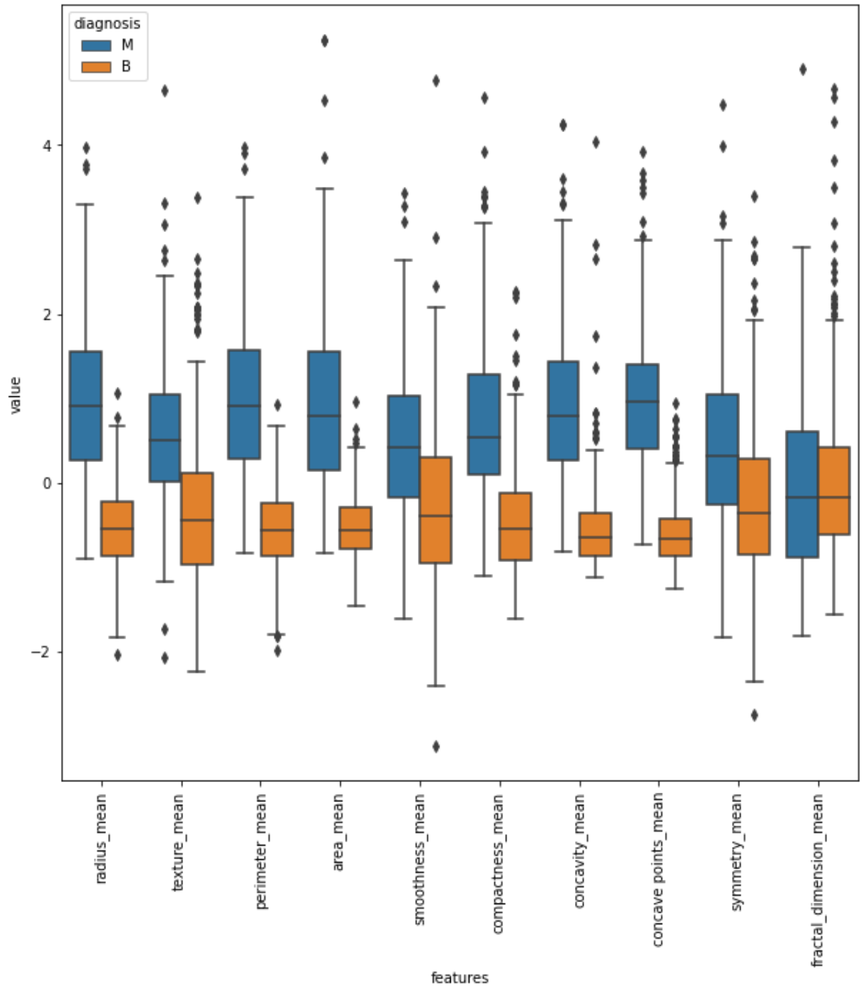


Standardization

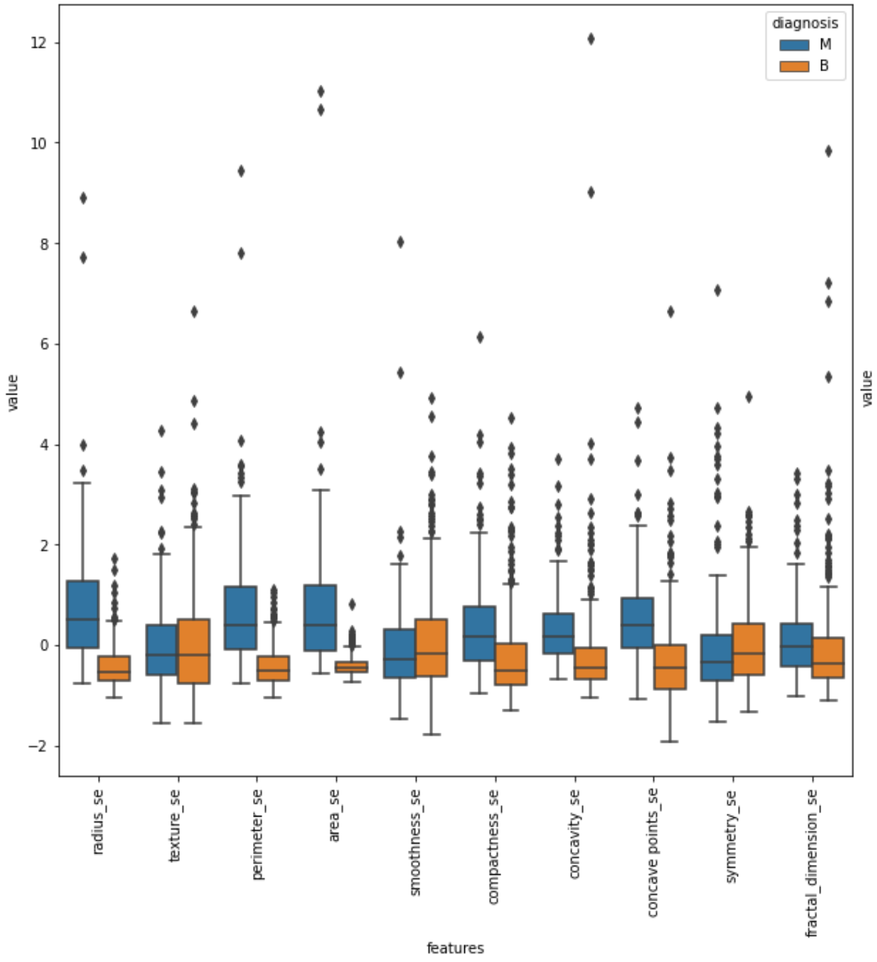


Standardization
Box plot

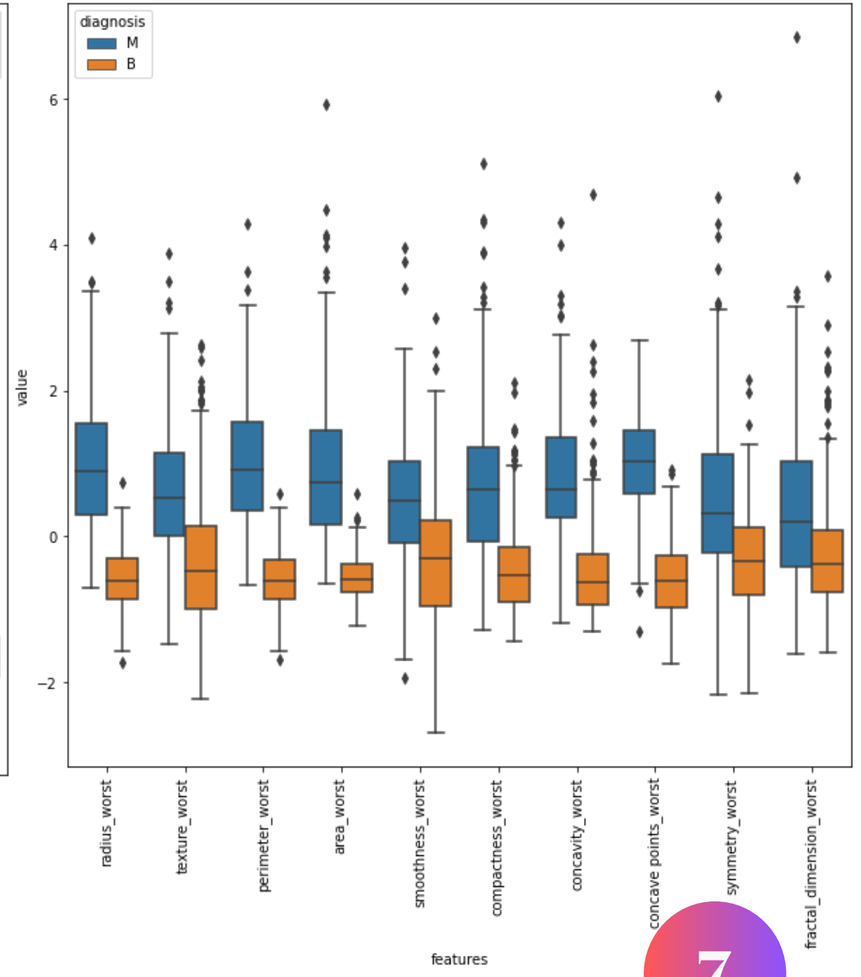
$$Z = \frac{x - \mu}{\sigma}$$



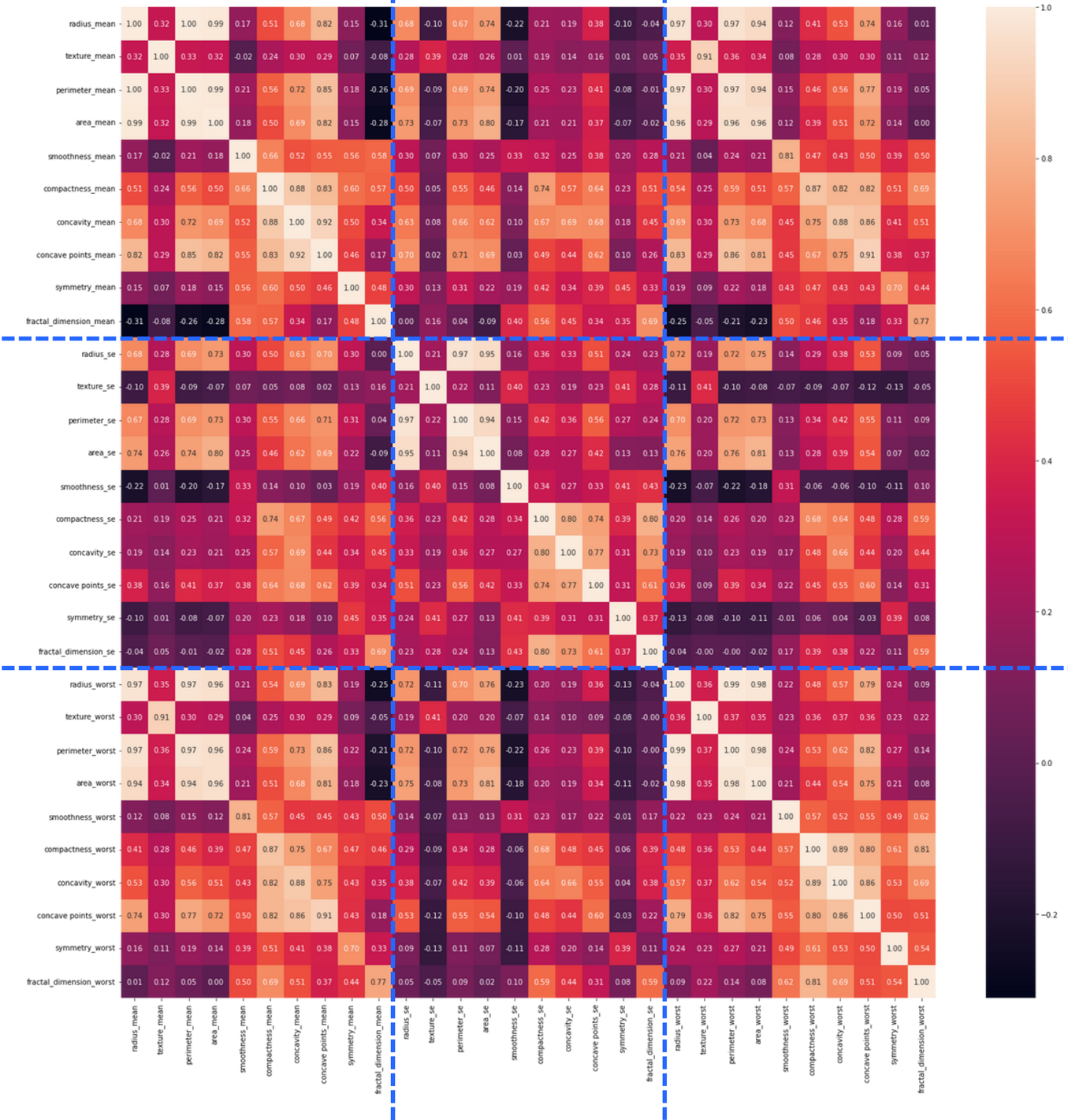
mean



se



worst



Correlation Heatmap

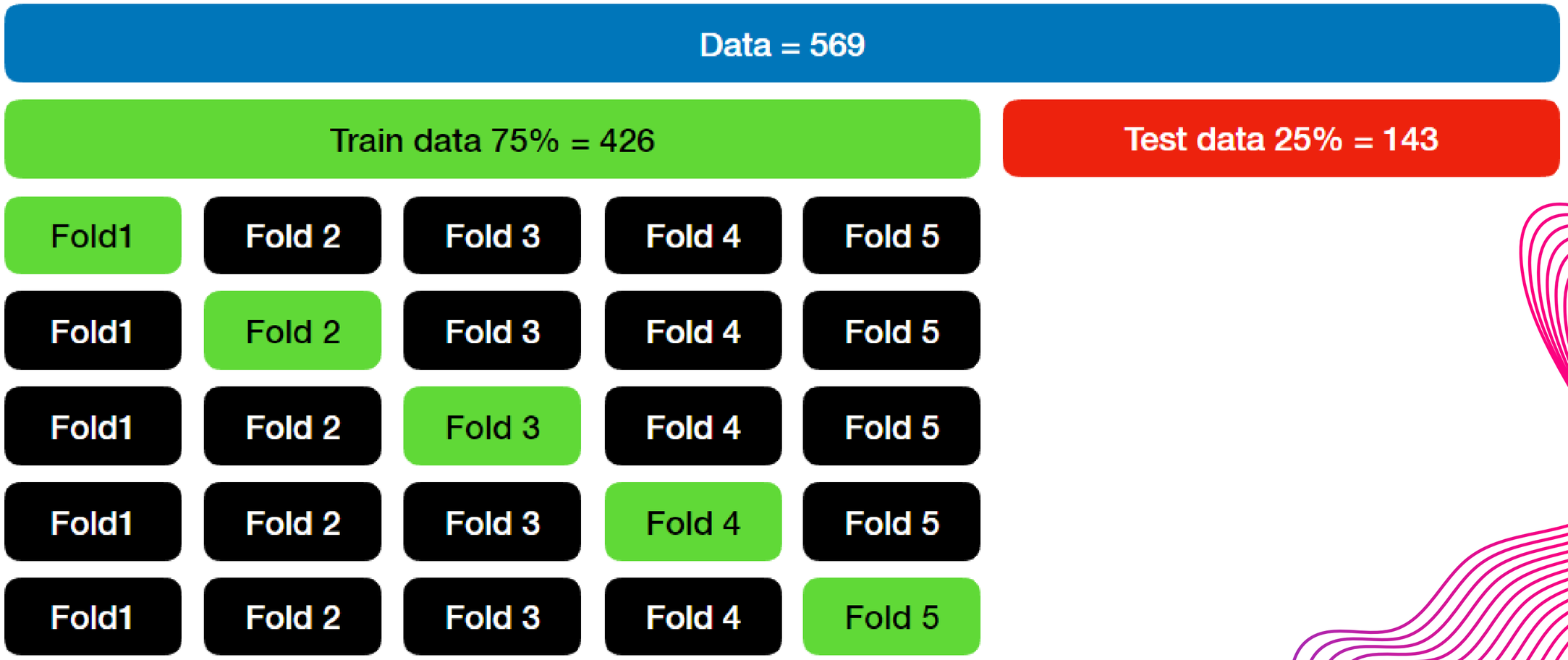
Top10 feature

1. concave points_worst
2. perimeter_worst
3. concave points_mean
4. radius_worst
5. perimeter_mean
6. area_worst
7. radius_mean
8. area_mean
9. concavity_mean
10. concavity_worst

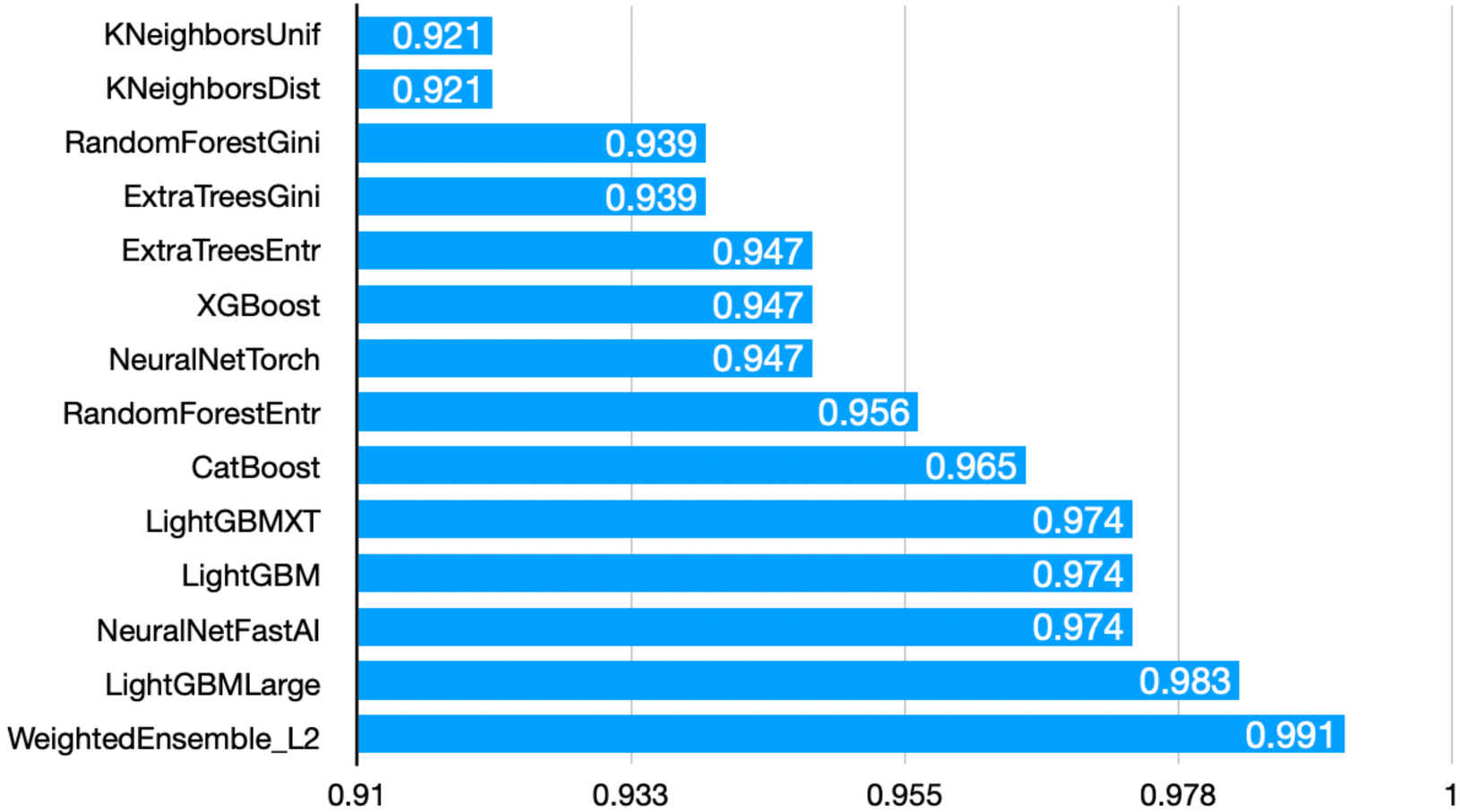
30x30

Split data to Train and Test

- split Train data to 5 folds using staticfied k-folds



AutoGluon



Model	Validation score
KNeighborsUnif	0.9211
KNeighborsDist	0.9211
RandomForestGini	0.9386
ExtraTreesGini	0.9386
ExtraTreesEntr	0.9474
XGBoost	0.9474
NeuralNetTorch	0.9474
RandomForestEntr	0.9561
CatBoost	0.9649
LightGBMXT	0.9737
LightGBM	0.9737
NeuralNetFastAI	0.9737
LightGBMLarge	0.9825
WeightedEnsemble_L2	0.9912

Proposed Methods

1 ramdom forest

-train normally
- train with non feature importance
- trian with feature importance

-train on 5-fold
- non feature importance

2 lgbm

-train normally
- feature importance
- non feature importance

-train on 5-fold
- non feature importance

3 catboost

-train normally
- feature importance
- non feature importance

-train on 5-fold
- non feature importance

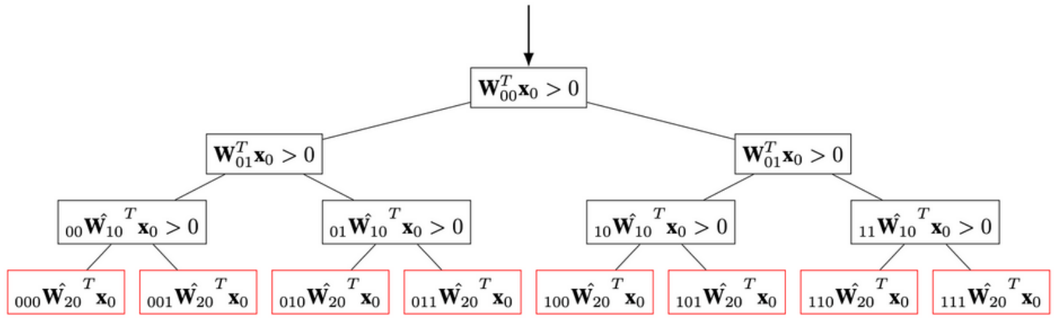
4 ensemble

-majority voting

Neural Networks are Decision Trees

Caglar Aytekin
AI Lead
AAC Technologies
caglaraytekin@aactechnologies.com, cagosmail@gmail.com

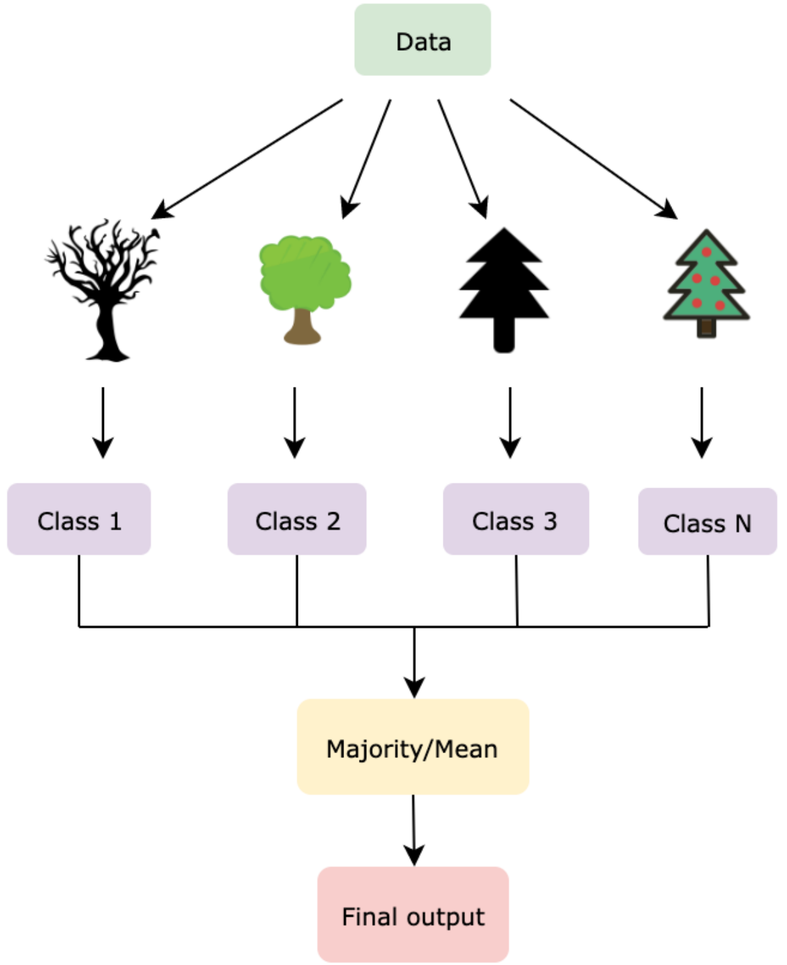
In this manuscript, show that any neural network with any activation function can be represented as a decision tree. The representation is equivalence and not an approximation, thus keeping the accuracy of the neural network exactly as is.



decision tree for a 2-layer ReLu Neural Network

Random Forest

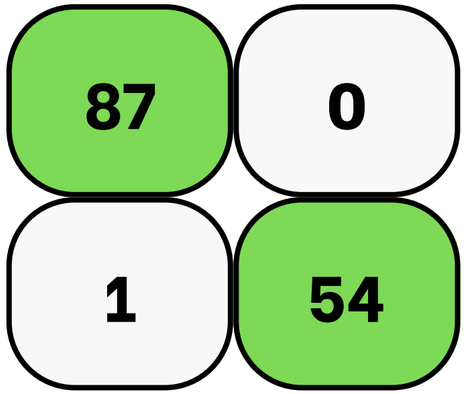
- basic model



Random forest

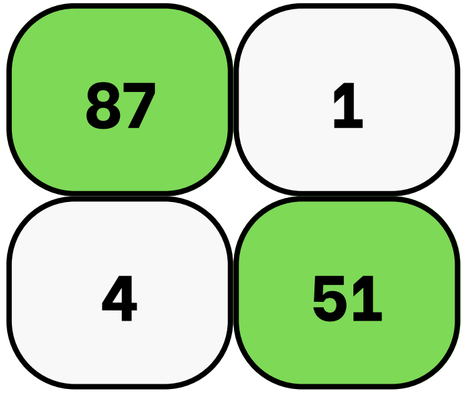
Hyperparameter : criterion='gini'
n_estimator=50

non importance features



accuracy	precision	recall	roc_auc	f1_binary
0.993007	1.0	0.981818	0.990909	0.990826

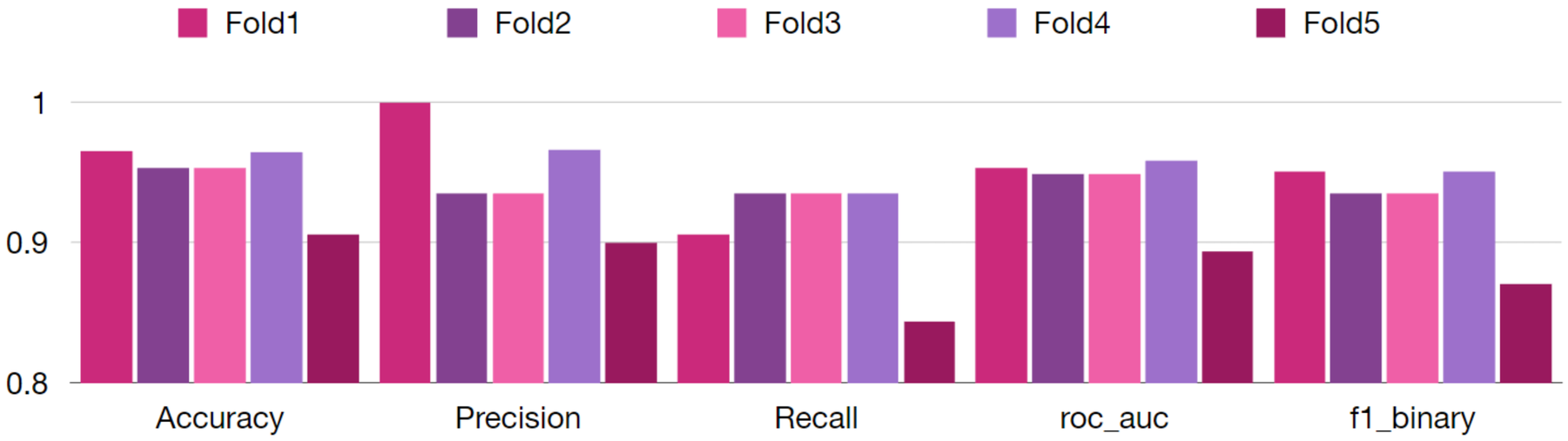
top10 important feature



accuracy	precision	recall	roc_auc	f1_binary
0.965035	0.980769	0.927273	0.957955	0.953271

Random Forest

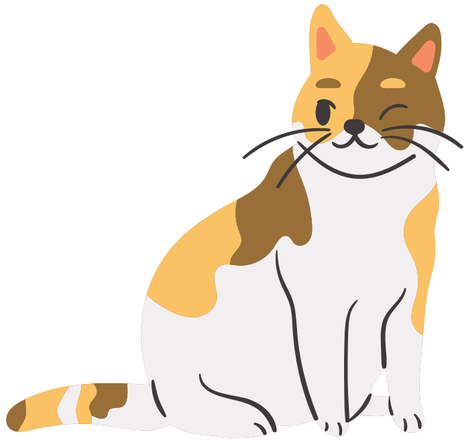
Validation



Test

87	1
2	53

accuracy	precision	recall	roc_auc	f1_binary
0.979021	0.981481	0.963636	0.976136	0.972477



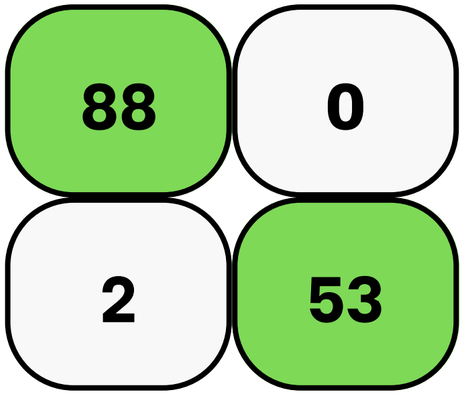
catboost

- good accuracy on validation set
- Categorical features support
- gradient boosting on decision trees

catboost

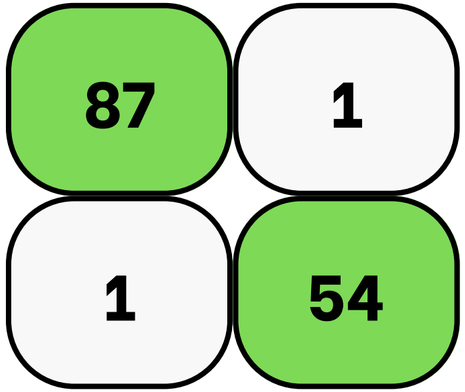
Hyperparameter : iterations=50
learning_rate=0.1
depth=6

non importance features



accuracy	precision	recall	roc_auc	f1_binary
0.986014	1.0	0.963636	0.981818	0.981481

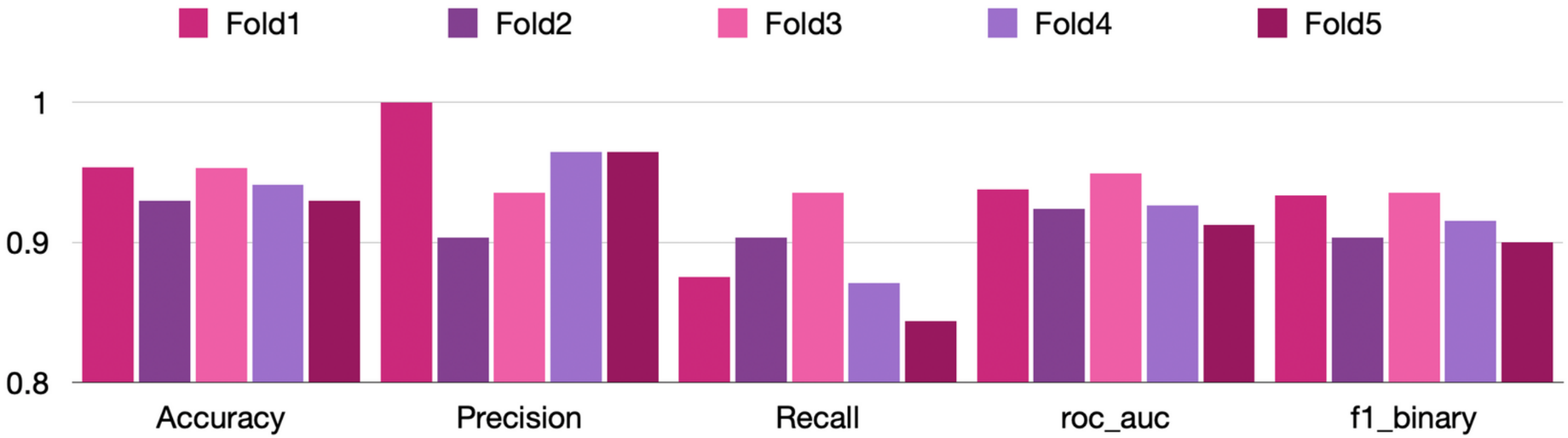
top10 important features



accuracy	precision	recall	roc_auc	f1_binary
0.986014	0.981818	0.981818	0.985227	0.981818

catboost

Validation



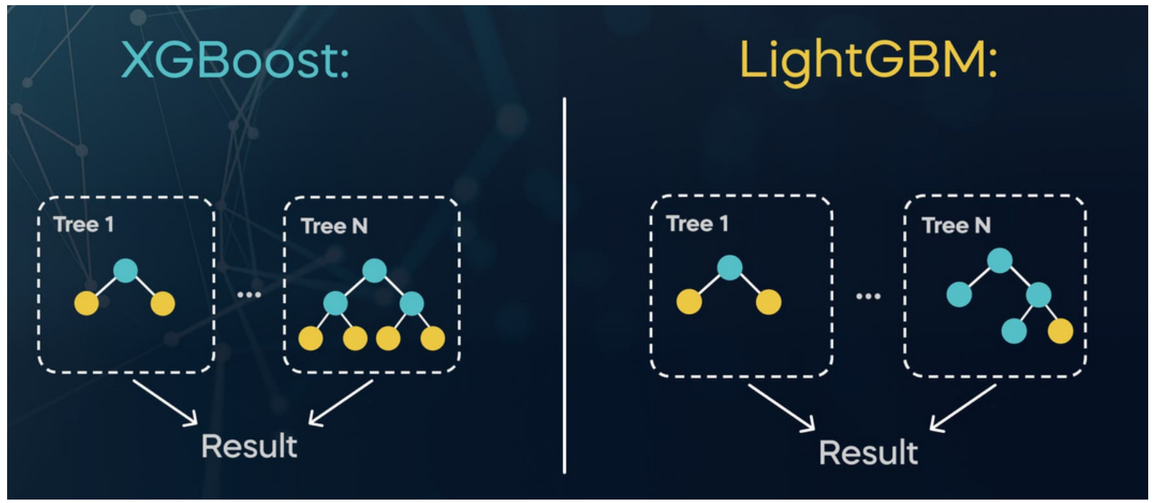
Test

87	0
1	54

accuracy	precision	recall	roc_auc	f1_binary
0.993007	1.0	0.981818	0.990909	0.990826

lgbm

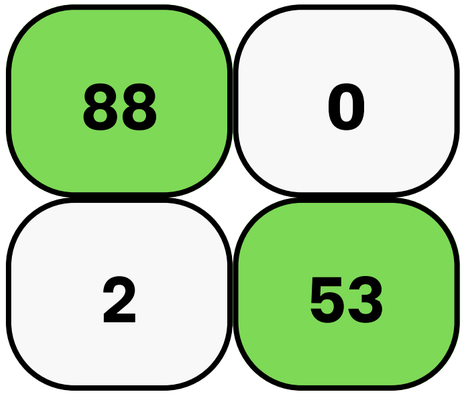
- good accuracy on validation set
- fast
- gradient boosting on decision trees



lgbm

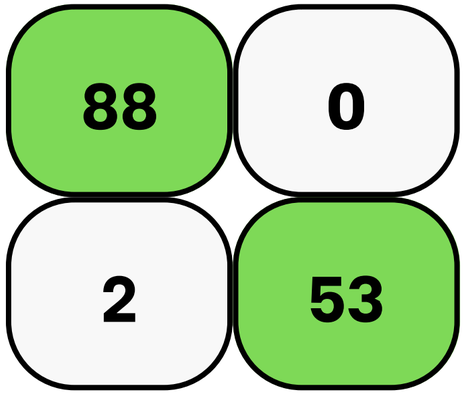
Hyperparameter : num_leaves=31
max_depth=-1
learning_rate=0.1
n_estimator=50

non importance features



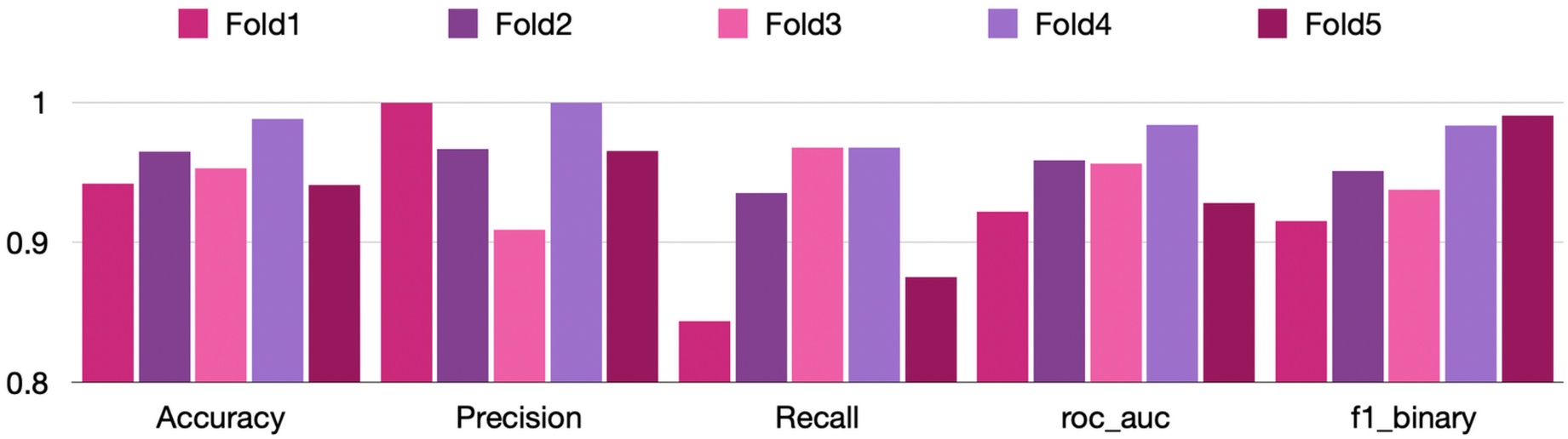
accuracy	precision	recall	roc_auc	f1_binary
0.986014	1.0	0.963636	0.981818	0.981481

top10 important feature

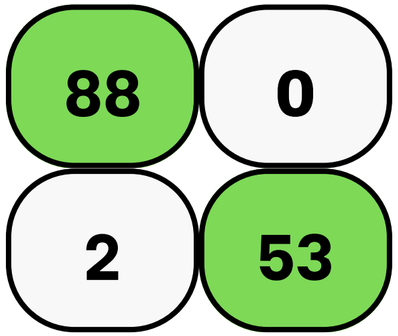


accuracy	precision	recall	roc_auc	f1_binary
0.986014	1.0	0.963636	0.981818	0.981481

lgbm
Validation

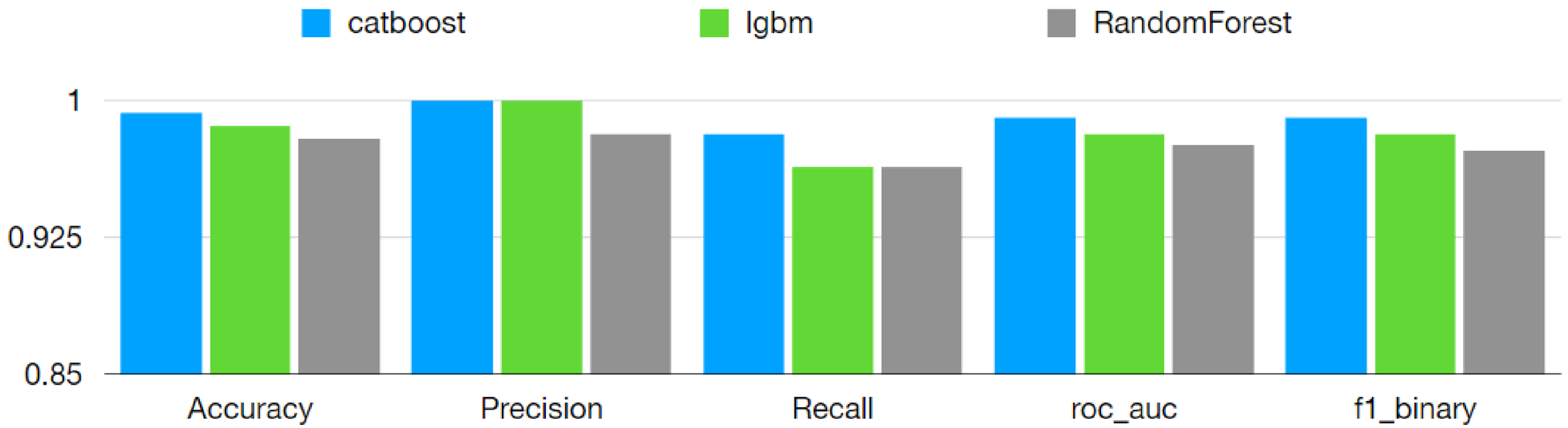


Test



accuracy	precision	recall	roc_auc	f1_binary
0.986014	1.0	0.963636	0.981818	0.981481

Result from test set



88	0
2	53

catboost

88	0
2	53

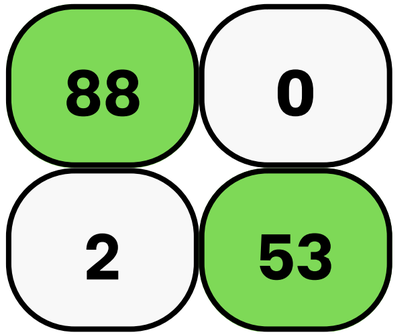
lgbm

87	1
2	53

random forest

Ensemble majority voting

- catboost(k-fold)
- lgbm(k-fold)
- randomforest(k-fold)



accuracy	precision	recall	roc_auc	f1_binary
0.986014	1.0	0.963636	0.981818	0.981481

catboost

incorrect index : 261

lgbm

incorrect index : 193, 261

Random Forest

incorrect index : 193, 261, 421

Analyze from Feature

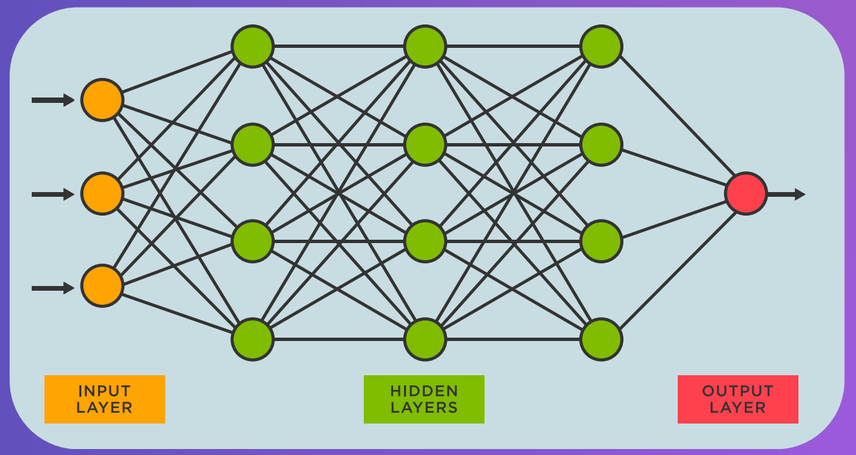
We analyze data by using mean, standard error, worst . Then we compare with Mean, Min, Max values of people with disease and without disease

		radius			texture			perimeter			area			smoothness		
index	FALSE	mean	se	worst	mean	se	worst	mean	se	worst	mean	se	worst	mean	se	worst
193	FN	12.34	0.4053	15.65	26.86	1.809	39.34	81.15	2.642	101.7	477.4	34.44	768.9	0.10340	0.0091	0.1785
261	FN	17.35	0.4007	19.85	23.06	1.317	31.47	111.00	2.577	128.2	933.1	44.41	1218.0	0.08662	0.00573	0.1240
421	FP	14.69	0.5462	16.46	13.98	1.511	18.34	98.22	4.795	114.1	656.1	49.45	809.2	0.10310	0.00998	0.1312

		compactness			concavity			concave points			symmetry			fractal dimension		
index	FALSE	mean	se	worst	mean	se	worst	mean	se	worst	mean	se	worst	mean	se	worst
193	FN	0.1353	0.03845	0.4706	0.10850	0.03763	0.4425	0.04562	0.01321	0.14590	0.1943	0.01878	0.3215	0.06937	0.00567	0.12050
261	FN	0.0629	0.01106	0.1486	0.02891	0.01246	0.1211	0.02837	0.00767	0.08235	0.1564	0.01411	0.2452	0.05307	0.00158	0.06515
421	FP	0.1836	0.05244	0.3635	0.14500	0.05278	0.3219	0.06300	0.0158	0.11080	0.2086	0.02653	0.2827	0.07406	0.00544	0.09208

1 index 2 index 3 index

Discussion & Conclusion



DATA

- UNDERSAMPLING
- OVERSAMPLING,with noise

MODEL

- decision tree
- randomforest
- catboost
- lightGBM
- ensemble

Future Works

- try using another models
- try training 5-Folds cross validation on torchNN
- preprocess - feature engineering

References

- Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, Andrey Gulin. CatBoost: unbiased boosting with categorical features. [Online] 2019. Available from: https://arxiv.org/abs/1706.09516 [Accessed 5th april 2023].
- Caglar Aytekin. Neural Networks are Decision Trees. [Online] 2022. Available from: https://arxiv.org/abs/2210.05189 [Accessed 6th april 2023].
- Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, Alexander Smola. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data. [Online] 2022. Available from: https://arxiv.org/abs/2210.05189 [Accessed 7th april 2023].
- Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. [Online] 2017. Available from: https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf [Accessed 8th april 2023].
- W. Nick Street, William H. Wolberg, O.L. MAngasarian. Nuclear Feature Extraction For Breast Tumor Diagnosis [Online] 1992. Available from: https://minds.wisconsin.edu/bitstream/handle/1793/59692/TR1131.pdf?sequence=1 [Accessed 15th March 2023].
- W. Nick Street, William H. Wolberg, O.L. MAngasarian. Breast Cancer Diagnosis and Prognosis via Linear Programming [Online] 1992. Available from: https://www.researchgate.net/publication/2302195_Breast_Cancer_Diagnosis_and_Prognosis_Via_Linear_Programming [Accessed 16th March 2023].