Data Visualization Pitch

Student Individual Assignment

Predicting Breast Tumor Malignancy Using Machine Learning Models

Student name: Tony Dawra

Student ID: 946852

Student e-mail address:

t.dawra@campus.unimib.it

Research questions

Can machine learning models accurately predict whether a breast tumor is benign or malignant using numeric features ?

- Which classification model works best for this task (Random Forest, XGBoost, SVM, KNN, Decision Tree), and how large is the performance gap between them?
- Does applying PCA (Principal Component Analysis) improve or harm predictive performance compared to using all 30 original features?
- How does balancing the dataset (via oversampling) affect model performance?
- How do different classification thresholds change the trade-off between false negatives (missing a cancer) and false positives (flagging healthy tissue as malignant)?

About Data

- **Dataset**: Breast Cancer Wisconsin (Diagnostic), downloaded from Kaggle (“Breast Cancer Wisconsin (Diagnostic) Data Set”).
- **Original source**: UCI Machine Learning Repository (Wolberg et al., 1993).
- **Target**: Diagnosis label — M = malignant, B = benign.
- **Features**: 30 numeric measurements from FNA images describing cell nuclei (shape, size, texture, etc.).
- **Data quality**: no missing values; all features numeric; moderate class imbalance (more benign).
- **Limitations**: The dataset is moderately imbalanced in the original form (more benign than malignant samples)
- **Preprocessing**: drop ID, encode M→1 / B→0, standardize features, oversample to balance classes.
- **Licence**: Kaggle copy under CC BY-NC-SA 4.0 (non-commercial use, attribution, share alike); original UCI dataset under CC BY 4.0.

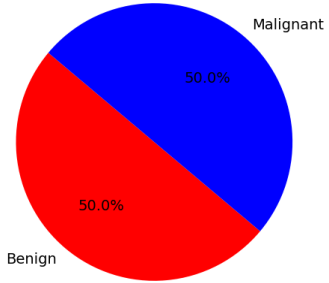
Missing Values

	missing_values
0	no
1	no
2	no
3	no
4	no
5	no
6	no
7	no
8	no
9	no
10	no
11	no
12	no
13	no
14	no
15	no
16	no
17	no
18	no
19	no
20	no
21	no
22	no
23	no
24	no
25	no
26	no
27	no
28	no
29	no
30	no
31	no

Features

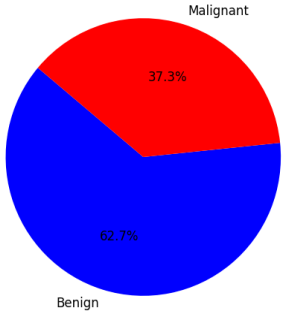
	name	role	type	demographic	description	units
0	ID	ID	Categorical	None	None	None
1	Diagnosis	Target	Categorical	None	None	None
2	radius1	Feature	Continuous	None	None	None
3	texture1	Feature	Continuous	None	None	None
4	perimeter1	Feature	Continuous	None	None	None
5	area1	Feature	Continuous	None	None	None
6	smoothness1	Feature	Continuous	None	None	None
7	compactness1	Feature	Continuous	None	None	None
8	concavity1	Feature	Continuous	None	None	None
9	concave_points1	Feature	Continuous	None	None	None
10	symmetry1	Feature	Continuous	None	None	None
11	fractal_dimension1	Feature	Continuous	None	None	None
12	radius2	Feature	Continuous	None	None	None
13	texture2	Feature	Continuous	None	None	None
14	perimeter2	Feature	Continuous	None	None	None
15	area2	Feature	Continuous	None	None	None
16	smoothness2	Feature	Continuous	None	None	None
17	compactness2	Feature	Continuous	None	None	None
18	concavity2	Feature	Continuous	None	None	None
19	concave_points2	Feature	Continuous	None	None	None
20	symmetry2	Feature	Continuous	None	None	None
21	fractal_dimension2	Feature	Continuous	None	None	None
22	radius3	Feature	Continuous	None	None	None
23	texture3	Feature	Continuous	None	None	None
24	perimeter3	Feature	Continuous	None	None	None
25	area3	Feature	Continuous	None	None	None
26	smoothness3	Feature	Continuous	None	None	None
27	compactness3	Feature	Continuous	None	None	None
28	concavity3	Feature	Continuous	None	None	None
29	concave_points3	Feature	Continuous	None	None	None
30	symmetry3	Feature	Continuous	None	None	None
31	fractal_dimension3	Feature	Continuous	None	None	None

Percentage of Diagnostic Outcomes in Dataset



After oversampling

Percentage of Diagnostic Outcomes in Dataset



Before oversampling

Methodology

Data collection:

- Breast Cancer Wisconsin (Diagnostic) dataset downloaded from Kaggle.

Tools:

- Python (Jupyter) with: pandas, numpy, scikit-learn, xgboost, imblearn, matplotlib, seaborn.

Pre-processing & transformation:

- Dropped ID column; encoded diagnosis (M → 1, B → 0).
- Standardized all numeric features (StandardScaler).
- Balanced classes with RandomOverSampler (~50% benign / 50% malignant).
- Tested models with and without PCA on the standardized features.
- Stratified train/test split.

Modeling:

- Tried: Random Forest, XGBoost, SVM, KNN, Decision Tree (basic + regularized).
- Used K-fold cross-validation and GridSearchCV to tune XGBoost and SVM.

Evaluation

- Metrics: accuracy, precision, recall, F1-score, confusion matrix, precision–recall curve.
- Best model: tuned XGBoost without PCA (≈98.6% accuracy, precision and recall).

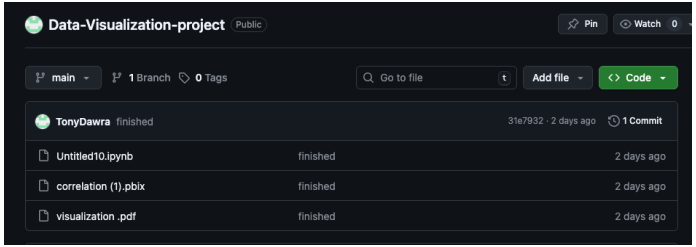
Use of AI tools

- Used ChatGPT to clarify functions, comment code, and prepare presentation text.

Source code:
https://github.com/TonyDawra/Data-Visualization-project

Content :

- Pdf
- Power bi correlation
- Notebook



Insights from the Data

Key quantitative insights

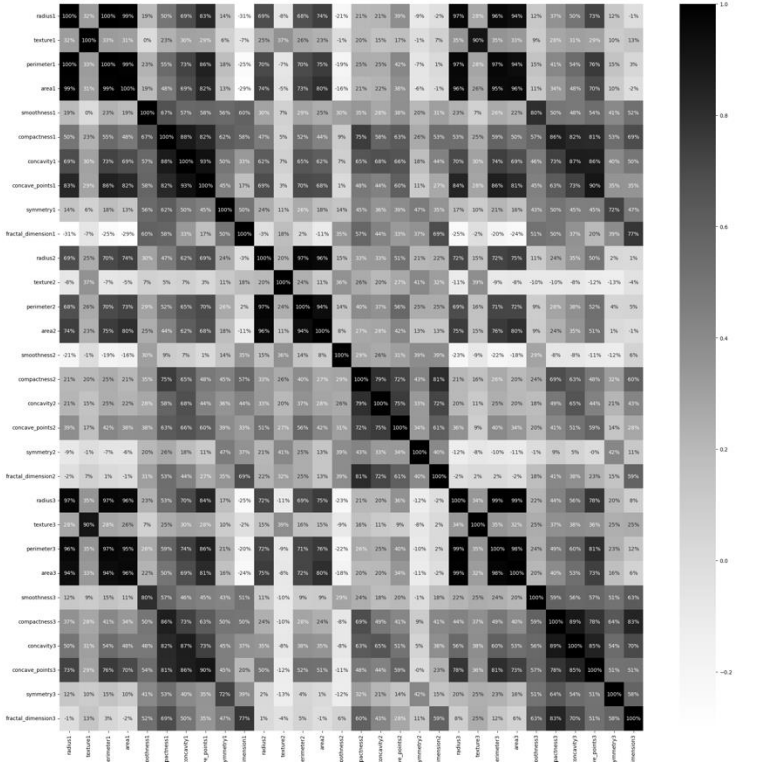
- After oversampling, the dataset is balanced (50% benign / 50% malignant).
- All models perform well; tuned XGBoost without PCA is best (≈98.6% accuracy, high and balanced precision/recall).
- Regularized SVM is close but slightly worse.
- PCA does not improve results; best performance is with the 30 scaled original features.
- Confusion matrix for XGBoost shows very few false negatives and a small, acceptable number of false positives.

Analysis performed

- Descriptive analysis of class distribution and feature correlations.
- Comparative analysis of 8 models, with vs without PCA, and with/without regularization.
- Error analysis using confusion matrix and TN/FP/FN/TP bar chart.
- Threshold analysis using precision–recall curve to study precision vs recall trade-off.

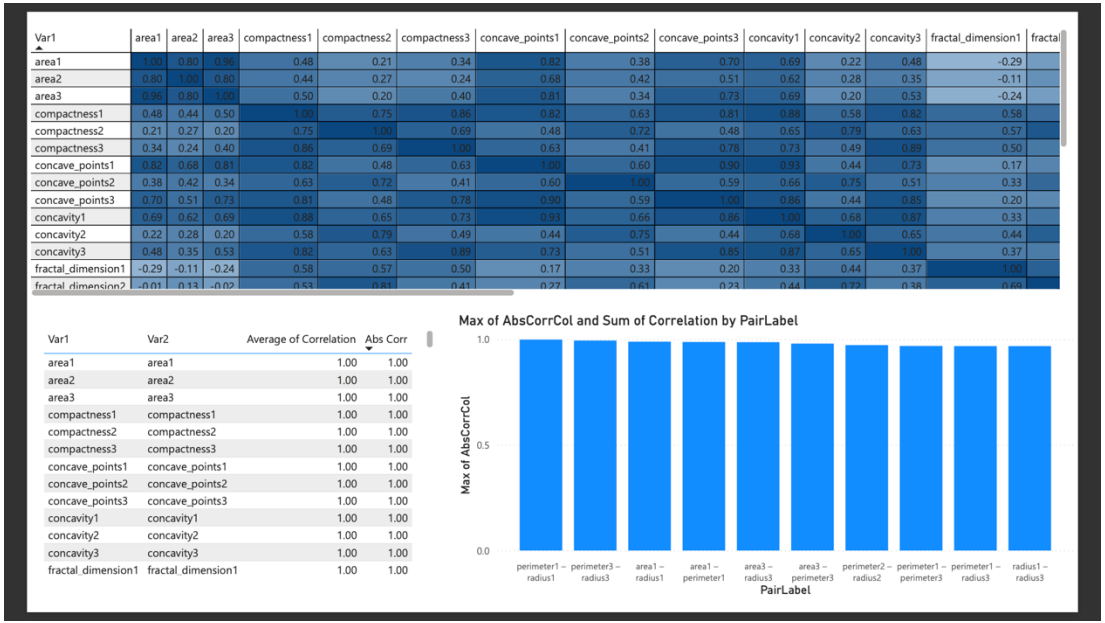
Main story

- Identify the best model (XGBoost).
- Show how its predictions relate to clinical trade-offs between false positives and false negatives.



Feature Correlation and Redundancy

- Several feature groups are strongly correlated (e.g., radius–perimeter–area), meaning they carry similar information.
- This confirms the dataset has redundant measurements, which is useful for interpretation and for considering dimensionality reduction (e.g., PCA).



Var1	area1	area2	area3	compactness1	compactness2	compactness3	concave_points1	concave_points2	concave_points3	concavity1	concavity2	concavity3	fractal_dimension1	fractal
area1	1.00	0.80	0.96	0.48	0.21	0.34	0.82	0.38	0.70	0.69	0.22	0.48	-0.29	
area2	0.80	1.00	0.80	0.44	0.27	0.24	0.68	0.42	0.51	0.62	0.28	0.35	-0.11	
area3	0.96	0.80	1.00	0.50	0.20	0.40	0.81	0.34	0.73	0.69	0.20	0.53	-0.24	
compactness1	0.48	0.44	0.50	1.00	0.75	0.86	0.82	0.63	0.81	0.88	0.58	0.82	0.58	
compactness2	0.21	0.27	0.20	0.75	1.00	0.69	0.48	0.72	0.48	0.65	0.79	0.63	0.57	
compactness3	0.34	0.24	0.40	0.86	0.69	1.00	0.63	0.41	0.78	0.73	0.49	0.89	0.50	
concave_points1	0.82	0.68	0.81	0.82	0.48	0.63	1.00	0.60	0.90	0.93	0.44	0.73	0.17	
concave_points2	0.38	0.42	0.34	0.63	0.72	0.41	0.60	1.00	0.59	0.66	0.75	0.51	0.33	
concave_points3	0.70	0.51	0.73	0.81	0.48	0.78	0.90	0.59	1.00	0.86	0.44	0.85	0.20	
concavity1	0.69	0.62	0.69	0.88	0.65	0.73	0.93	0.66	0.86	1.00	0.68	0.87	0.33	
concavity2	0.22	0.28	0.20	0.58	0.79	0.49	0.44	0.75	0.44	0.68	1.00	0.65	0.44	
concavity3	0.48	0.35	0.53	0.82	0.63	0.89	0.73	0.51	0.85	0.87	0.65	1.00	0.37	
fractal_dimension1	-0.29	-0.11	-0.24	0.58	0.57	0.50	0.17	0.33	0.20	0.33	0.44	0.37	1.00	
fractal_dimension2	-0.01	0.13	-0.02	0.53	0.81	0.41	0.27	0.61	0.23	0.44	0.72	0.38	0.69	

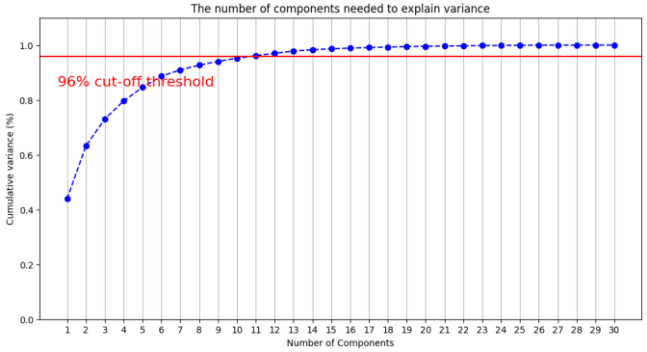
Var1	Var2	Average of Correlation	Abs Corr
area1	area1	1.00	1.00
area2	area2	1.00	1.00
area3	area3	1.00	1.00
compactness1	compactness1	1.00	1.00
compactness2	compactness2	1.00	1.00
compactness3	compactness3	1.00	1.00
concave_points1	concave_points1	1.00	1.00
concave_points2	concave_points2	1.00	1.00
concave_points3	concave_points3	1.00	1.00
concavity1	concavity1	1.00	1.00
concavity2	concavity2	1.00	1.00
concavity3	concavity3	1.00	1.00
fractal_dimension1	fractal_dimension1	1.00	1.00

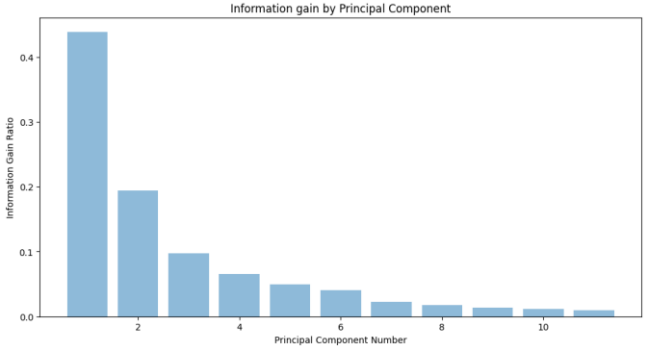
The table on the left lists those self-correlations (all equal to 1, which is correct).

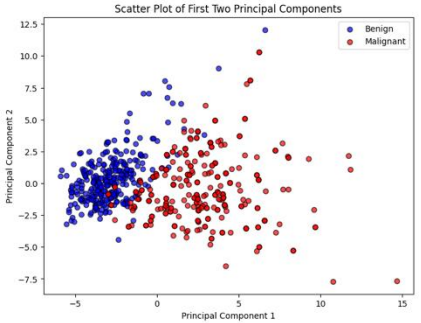
The first graph is a heatmap that shows the correlation between all features.

The bar chart shows the top 10 different feature pairs (not the same feature) with the highest correlation.

PCA: Variance & Class Separation







The top plot shows the cumulative explained variance of the PCA components. The first few components already capture most of the information in the data: roughly 90% of the total variance is explained by about the first 7 components (instead of the original 30 features).

The bar chart in the middle shows the variance explained by each single component.

- PC1 explains the largest share of variance,
- the following components contribute less and less, which illustrates the diminishing returns of adding more components.

The scatter plot of PC1 vs PC2 (bottom) shows that benign and malignant tumors are partially separated in the PCA space, although there is still some overlap. This means PCA is useful to visualize the structure of the data and reduce dimensionality with limited information loss.

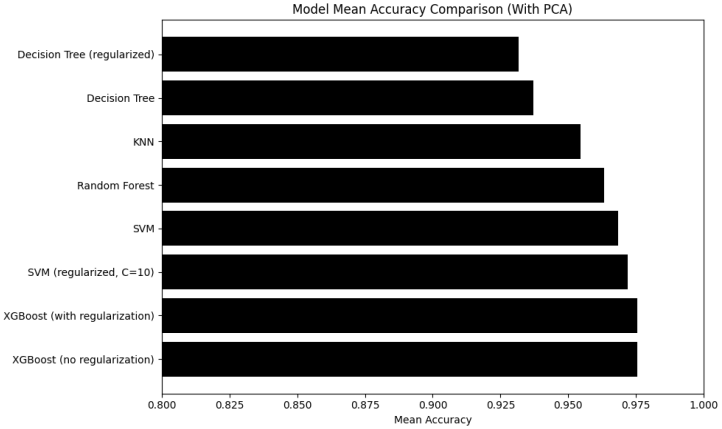
However, when I later compared models with and without PCA, the best performance was obtained without PCA, so PCA is mainly used here for exploration and visualization, not for the final classifier.

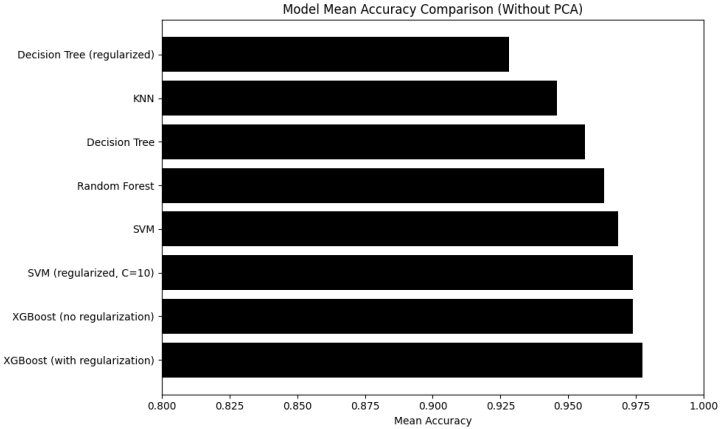
Model Accuracy Comparison

This chart compares the mean cross-validated accuracy of eight different models, evaluated both with and without PCA.
Two main patterns emerge:

- **XGBoost (with regularization, without PCA)** is the clear winner, achieving the highest mean accuracy across all configurations.
- **PCA does not systematically improve performance**; in many cases, the models without PCA perform slightly better.

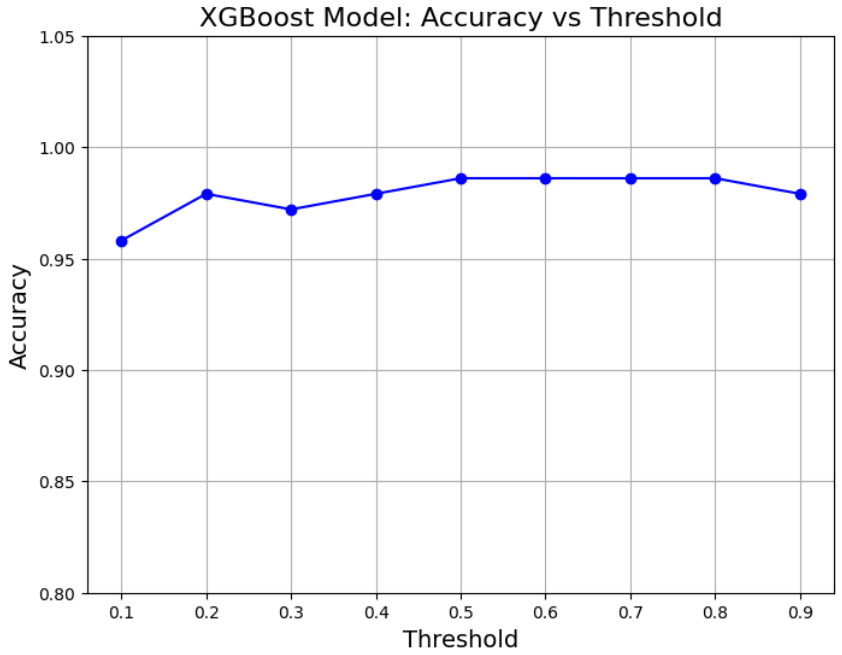
This suggests that, after proper scaling, the original 30 features already provide a good representation of the data. Tree-based models like XGBoost can naturally handle correlations between features and do not require dimensionality reduction to perform well.
Based on this comparison, I selected **XGBoost without PCA** as my final model for further analysis and deployment on the external competition data.





XGBoost Decision Threshold Tuning

- XGBoost outputs a probability of “malignant” (class = 1).
- The threshold is the cutoff: predict malignant if probability ≥ threshold.
- Changing the threshold shifts the trade-off between:
  -False Negatives (missing malignant cases) vs
  False Positives (flagging benign as malignant).
- Accuracy stays high across thresholds, so the best threshold should be chosen based on the clinical cost of errors, not accuracy alone.



Error Analysis

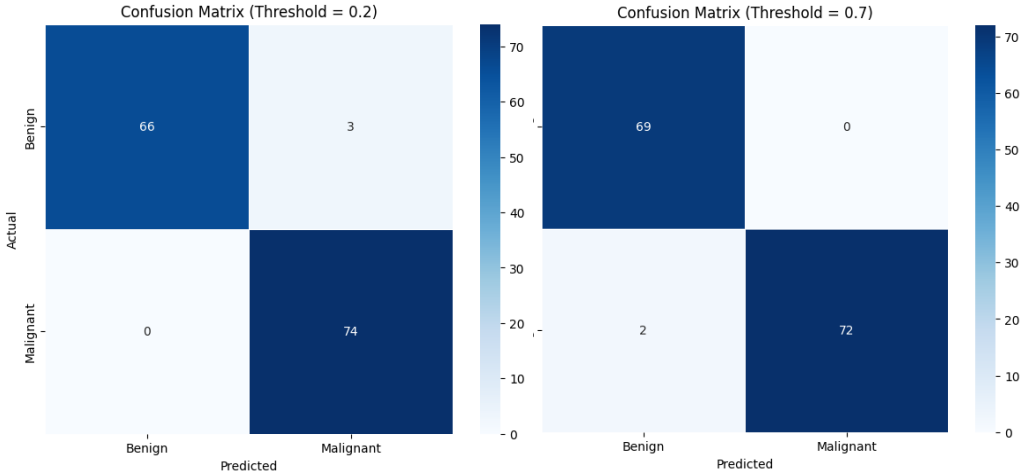
This visualization focuses on the error structure of the final XGBoost model rather than just its accuracy.

At the chosen decision threshold, the confusion-matrix-based bar chart shows:

- A high number of true positives, meaning the model correctly identifies most malignant tumors.
- Very few false negatives (malignant cases classified as benign), which is essential in a medical diagnosis context because missing a cancer case can have severe consequences.
- A small, acceptable number of false positives, i.e., benign tumors that are flagged as malignant and may receive additional diagnostic tests.

The precision–recall curve illustrates how changing the decision threshold allows us to trade off precision and recall. For early cancer detection, a high recall is usually prioritized, even if it slightly lowers precision, because it is safer to over-flag suspicious cases than to miss a malignant tumor.

Overall, these visualizations confirm that the final XGBoost model is not only accurate but also has an error profile compatible with clinical decision support, especially when the threshold is tuned to favor recall.



LICENCE

Slides are shared with the following license: