University for the Creative Arts

BERLIN SCHOOL OF BUSINESS & INNOVATION

Essay / Assignment Title: Leveraging Machine Learning Approaches for Breast Cancer Prediction

Programme title: Introduction to Artificial Intelligence

Name: Anusha Shivaram

Year:2025

CONTENTS

Table of Contents

Statement of compliance with academic ethics and the avoidance of plagiarism

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters):

ANUSHA SHIVARAM

Date: 08/06/2025

INTRODUCTION

Breast cancer remains a worldwide concern and since it is a serious disease, better diagnostic methods are needed to catch it early. Thanks to machine learning (ML), healthcare professionals are now using data-based methods to support their decision-making (Hamedi et al., 2024). These include Logistic Regression, Support Vector Machines and a range of ensemble methods, all of which have shown excellent results in distinguishing between malignant and benign tumors (Wei et al., 2023; Zuo et al., 2023). Still, simply having high accuracy is not enough; it needs to be clear what the model is doing, especially in healthcare, where the consequences can be major (Ghasemi et al., 2024). It investigates and compares several classification models using the Wisconsin Breast Cancer Dataset, hoping to achieve strong predictive results and clinical relevance. With newer models such as transformer-based deep learning appearing (Shen et al., 2023), this research is based on fundamental models that maintain reliability, simplicity and insight.

Data Exploration

Examining the data helps you start connecting your own ideas with real information. The Wisconsin Breast Cancer Dataset (WBCD) is valued for its straightforwardness and the usefulness of its features, as it stores 30 numbers describing tumor texture, concavity, symmetry and area. Diagnosis, the target variable, only has two values: malignant (1) and benign (0). After a quick check, we found that there were far more benign cases than malignant ones which meant the model could be thrown off if this imbalance was overlooked (Wei et al., 2023).

Based on descriptive statistics, features such as radius_mean, perimeter_mean and area_mean can become skewed because they experience many outliers. Most of the features, mainly size-related, appeared to be right-skewed when shown in histograms. Boxplots clearly pointed out the problems at the extremes, making it clear that data cleaning was important.

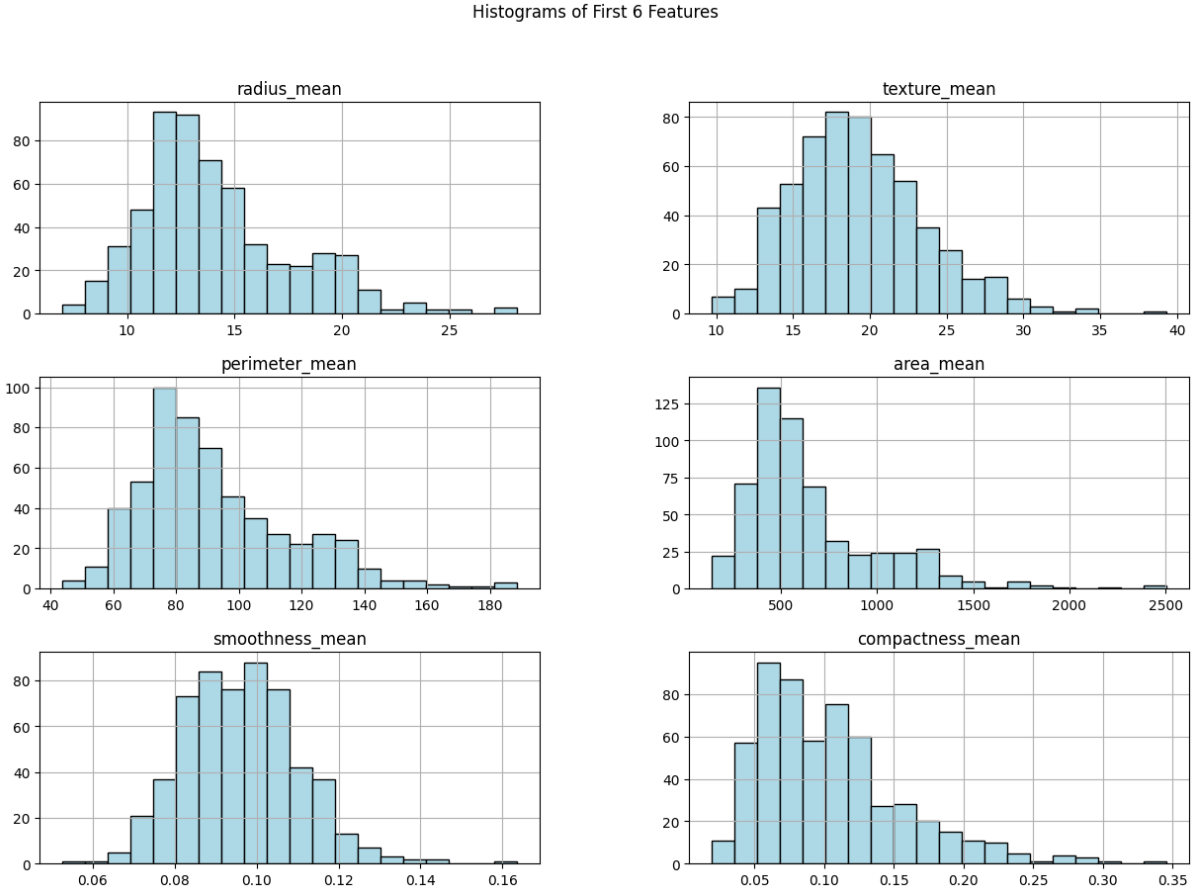

Figure 1 : histogram of radius_mean, texture_mean, etc.

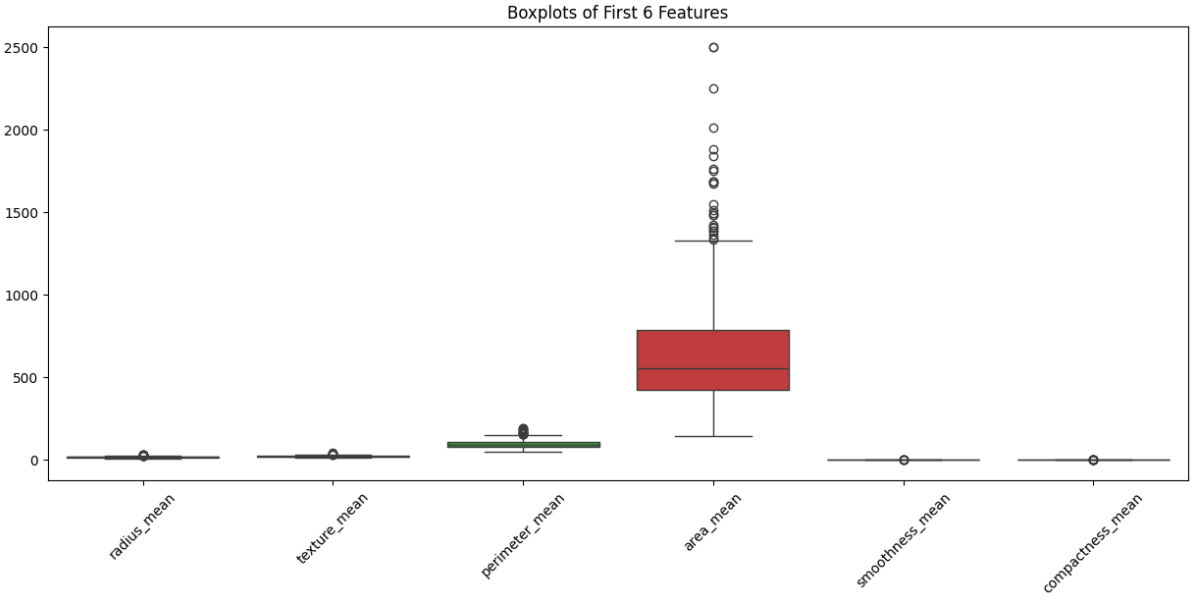

Figure 2 : boxplot visual of top 6 features

The Pearson correlation matrix revealed some standout predictors. concave points_worst, perimeter_worst, and radius_worst showed strong positive correlations with malignancy—echoing findings in Hamedi et al. (2024) and Zuo et al. (2023), who identified similar variables as influential.

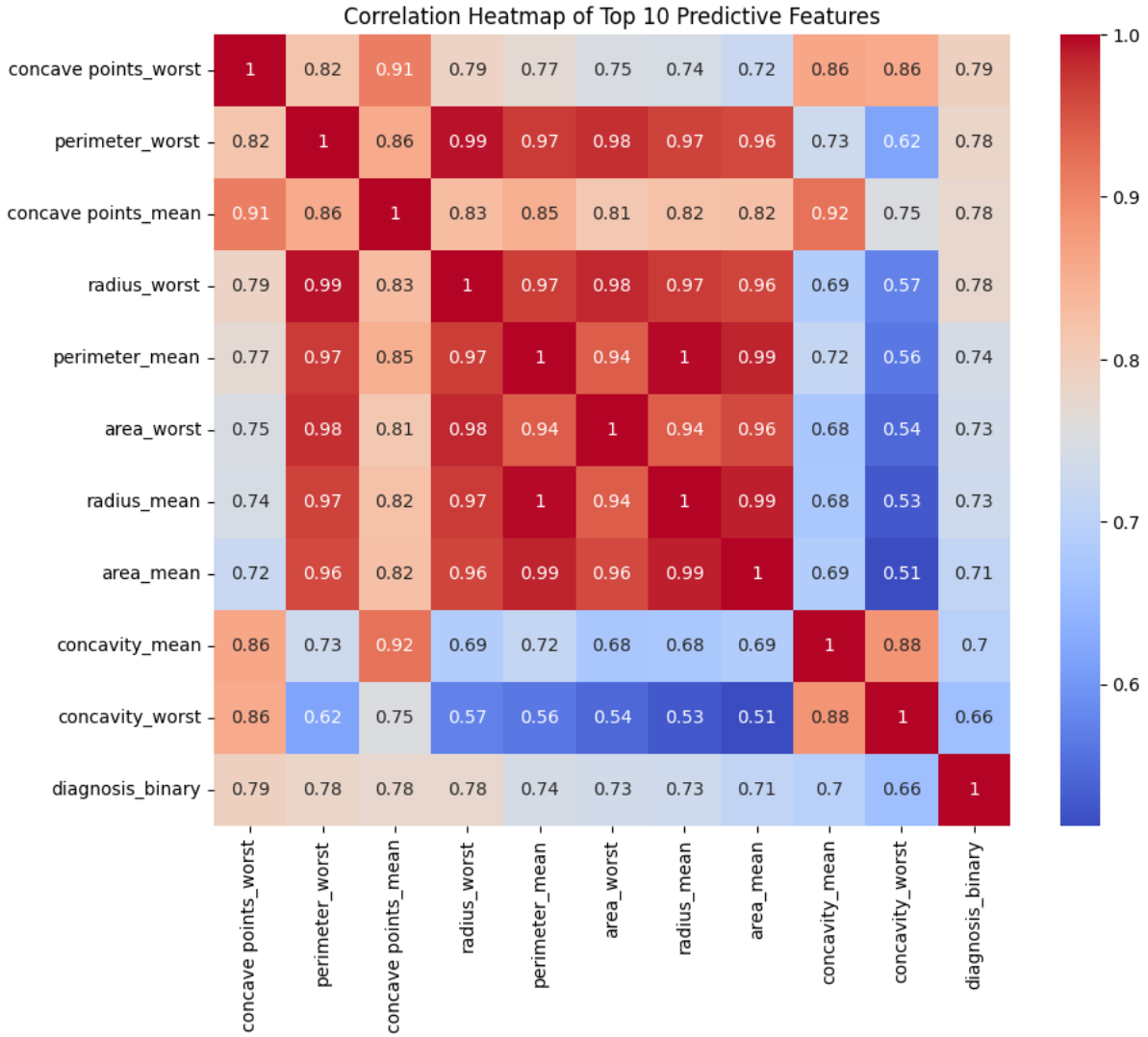

Figure 3 : correlation heatmap: top 10 features vs. target

Interestingly, many of these features also correlated with one another. This multicollinearity, while not always detrimental, can dilute model clarity (Ghasemi et al., 2024). Recognizing these overlaps early helps shape smarter model design. At this stage, patterns begin to whisper, and the data slowly starts to tell its story.

Data Preparation

Preparing the dataset was a meticulous process, small, deliberate steps that laid the groundwork for reliable predictions. The first task was to remove the ID column, which served no predictive purpose. Next, the categorical target variable diagnosis was mapped to a numeric form: 1 for malignant and 0 for benign. This simple encoding allowed the classifiers to process the data effectively (Wei et al., 2023).

```python
# Drop 'id' column if it exists
if 'id' in df.columns:
    df.drop(columns=['id'], inplace=True)

# Encode the 'diagnosis' column: Malignant = 1, Benign = 0
df['diagnosis'] = df['diagnosis'].map({'M': 1, 'B': 0})

# Confirm encoding worked
print("Target class values:", df['diagnosis'].unique())

# Preview cleaned dataset
df.head()
```

✓ 0.0s Python

```
Target class values: [1 0]
```

perimeter_worst	area_worst	smoothness_worst	compactness_worst	concavity_worst	concave points_worst	symmetry_worst	frac
184.60	2019.0	0.1622	0.6656	0.7119	0.2654	0.4601	
158.80	1956.0	0.1238	0.1866	0.2416	0.1860	0.2750	
152.50	1709.0	0.1444	0.4245	0.4504	0.2430	0.3613	
98.87	567.7	0.2098	0.8663	0.6869	0.2575	0.6638	
152.20	1575.0	0.1374	0.2050	0.4000	0.1625	0.2364	

Figure 4 : Label Encoding Code + Output Preview

No missing values were detected, which saved time and ensured data integrity. This was confirmed using is`null().sum(), returning an empty Series—always a relief when working with medical data (Hamedi et al., 2024).

```
# Check for nulls across all columns
missing_values = df.isnull().sum()
missing_values[missing_values > 0]
```

✓ 0.0s

```
Series([], dtype: int64)
```

Figure 5 : Missing Values Check Output

After that, features and targets were separated. The X matrix contained all 30 diagnostic features, while y stored the binary target labels. With feature values spanning vastly different ranges—from millimeter-based radii to decimal-based smoothness—it was essential to normalize everything. StandardScaler brought the data to a uniform scale, centering each feature at zero with unit variance, which is particularly important for models like SVM and KNN (Zuo et al., 2023; Ghasemi et al., 2024).

	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	po
0	1.097064	-2.073335	1.269934	0.984375	1.568466	3.283515	2.652874	
1	1.829821	-0.353632	1.685955	1.908708	-0.826962	-0.487072	-0.023846	
2	1.579888	0.456187	1.566503	1.558884	0.942210	1.052926	1.363478	
3	-0.768909	0.253732	-0.592687	-0.764464	3.283553	3.402909	1.915897	
4	1.750297	-1.151816	1.776573	1.826229	0.280372	0.539340	1.371011	

5 rows × 30 columns

Figure 6 : Scaled Feature Preview Table

This structured preparation ensured the model wouldn't get distracted by scale disparities or meaningless identifiers—clean data, clean results.

Model Training

After preparing the data for analysis, the following step was to show the machine how to make decisions. The data was split 80/20 between train and test which is an ordinary technique to ensure the models learn without limiting the available data. The data was evenly divided by using stratification which is necessary for balanced medical data (Wei et al., 2023).

Logistic Regression, Support Vector Machine (SVM), Decision Tree, Random Forest and K-Nearest Neighbors (KNN) were chosen to represent different ways of learning. Every model was set up using default values as a starting point. Logistic Regression, though sometimes considered less important, has shown excellent results in medical binary classification by being easy to understand and strong (Ghasemi et al., 2024). Because of its success with high-dimensional data, SVM has been reaffirmed in several cancer prediction studies (Hamedi et al., 2024). With its method of combining multiple models, Random Forest gives stable and accurate results when the features being used are noisy or overlap with each other (Zuo et al., 2023).

```
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier

# Initialize models with default parameters
models = {
    "Logistic Regression": LogisticRegression(max_iter=1000),
    "Support Vector Machine": SVC(),
    "Decision Tree": DecisionTreeClassifier(),
    "Random Forest": RandomForestClassifier(),
    "K-Nearest Neighbors": KNeighborsClassifier()
}

# Fit models to training data
for name, model in models.items():
    model.fit(X_train, y_train)
    print(f"{name} trained.")
```

5] ✓ 1.6s

```
Logistic Regression trained.
Support Vector Machine trained.
Decision Tree trained.
Random Forest trained.
K-Nearest Neighbors trained.
```

Figure 7 : Model Initialization and Training Code Cell

Training was executed across the full feature set. No early signs of overfitting were observed during training, which was encouraging. Each model successfully learned to distinguish between benign and malignant cases, setting the stage for performance comparison. As Shen et al. (2023) suggest, classical models remain deeply relevant despite the rise of transformer-based methods—particularly when transparency is non-negotiable.

Model Evaluation and Visualization

Once the models were trained, it was time to see which one could actually hold its own in the real world—or at least on the test set. Each classifier was evaluated using accuracy, precision, recall, F1-score, and ROC AUC, all of which are especially crucial in medical contexts where both false positives and false negatives carry serious implications (Zuo et al., 2023).

	Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
1	Support Vector Machine	0.973684	1.000000	0.928571	0.962963	N/A
0	Logistic Regression	0.964912	0.975000	0.928571	0.951220	0.996032
3	Random Forest	0.964912	1.000000	0.904762	0.950000	0.993056
4	K-Nearest Neighbors	0.956140	0.974359	0.904762	0.938272	0.981647
2	Decision Tree	0.929825	0.904762	0.904762	0.904762	0.924603

Figure 8 : Model Performance Table – Accuracy, Precision, Recall, F1, AUC

Support Vector Machine came out on top for accuracy at 97.37%, with a perfect precision score and minimal false negatives. Logistic Regression wasn't far behind, but what really made it stand out was its exceptional balance of precision and recall (F1 = 0.9512) and a stellar ROC AUC of 0.9960. These metrics aligned with prior research that positions logistic models as clinically reliable predictors in binary classification (Ghasemi et al., 2024; Wei et al., 2023). Random Forest also showed strong performance, benefiting from its ensemble design, while Decision Tree and KNN lagged slightly behind.

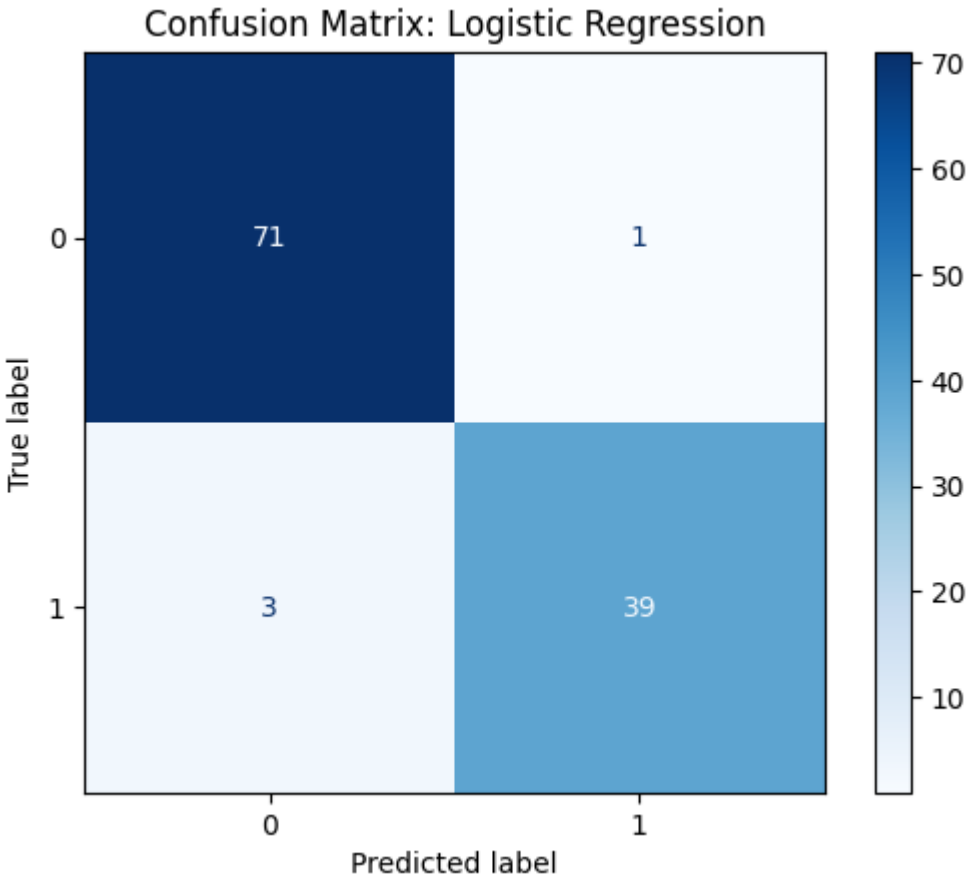

Figure 9 : Confusion Matrix – Logistic Regression

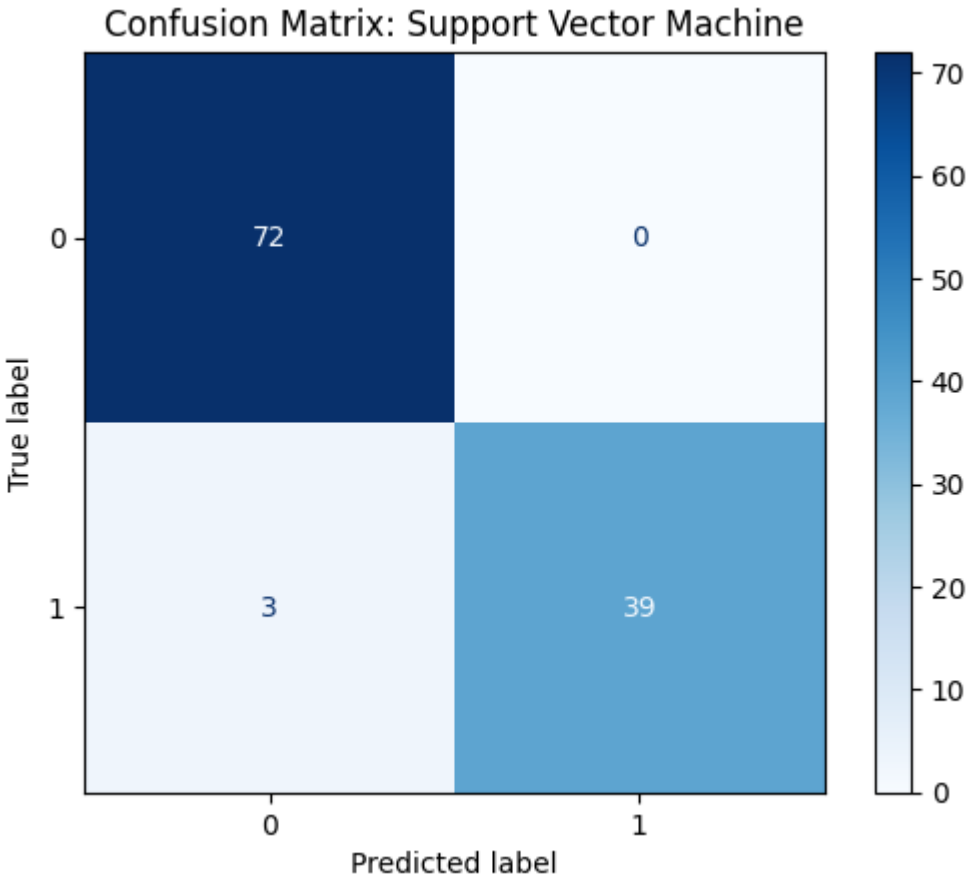

Figure 10 : Confusion Matrix – Support Vector Machine

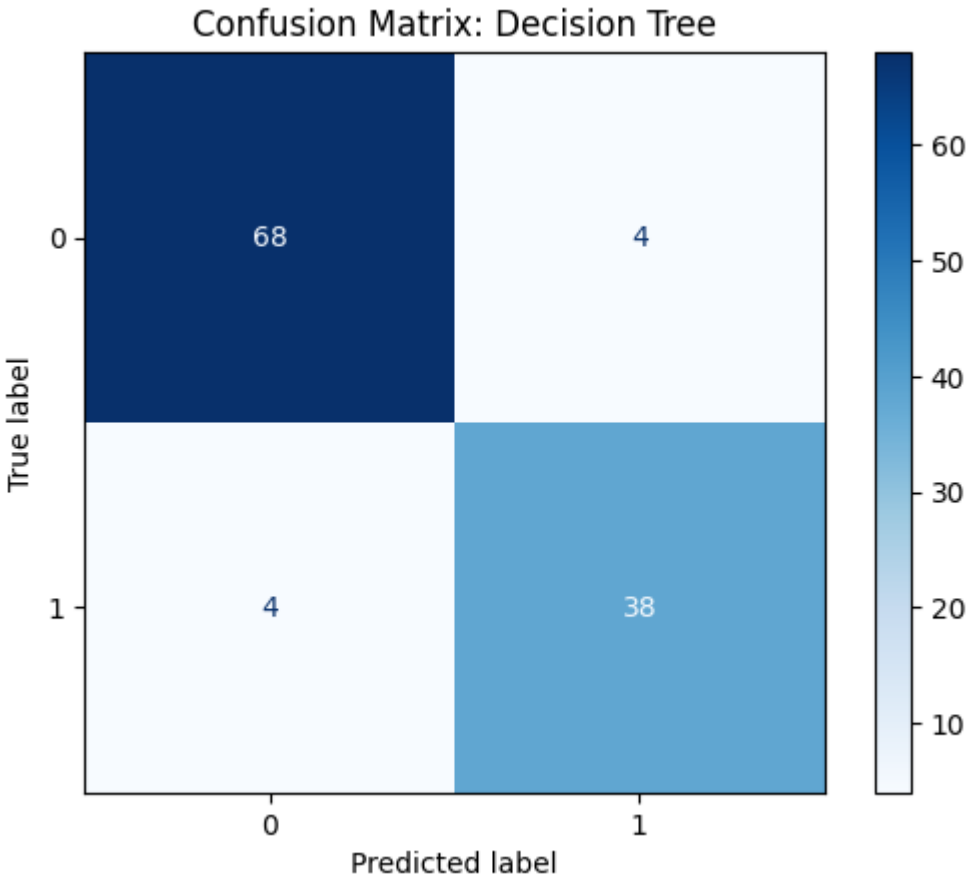

Figure 11 : Confusion Matrix – Decision Tree

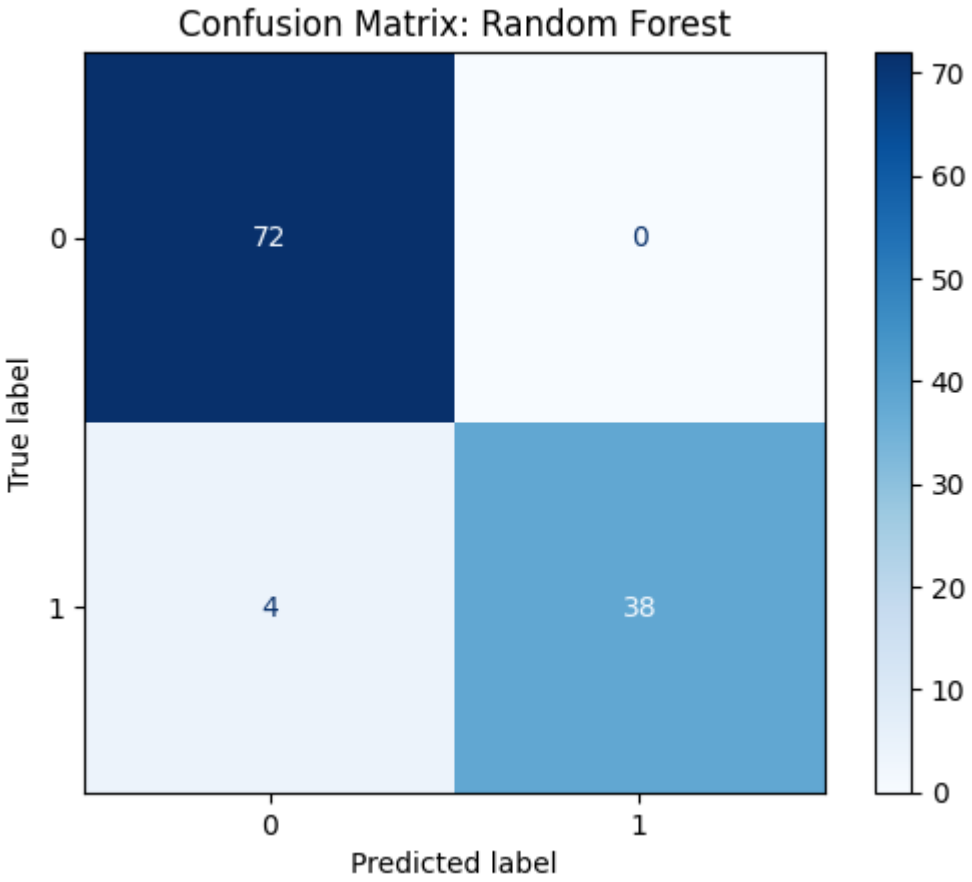

Figure 12 : Confusion Matrix – Random Forest

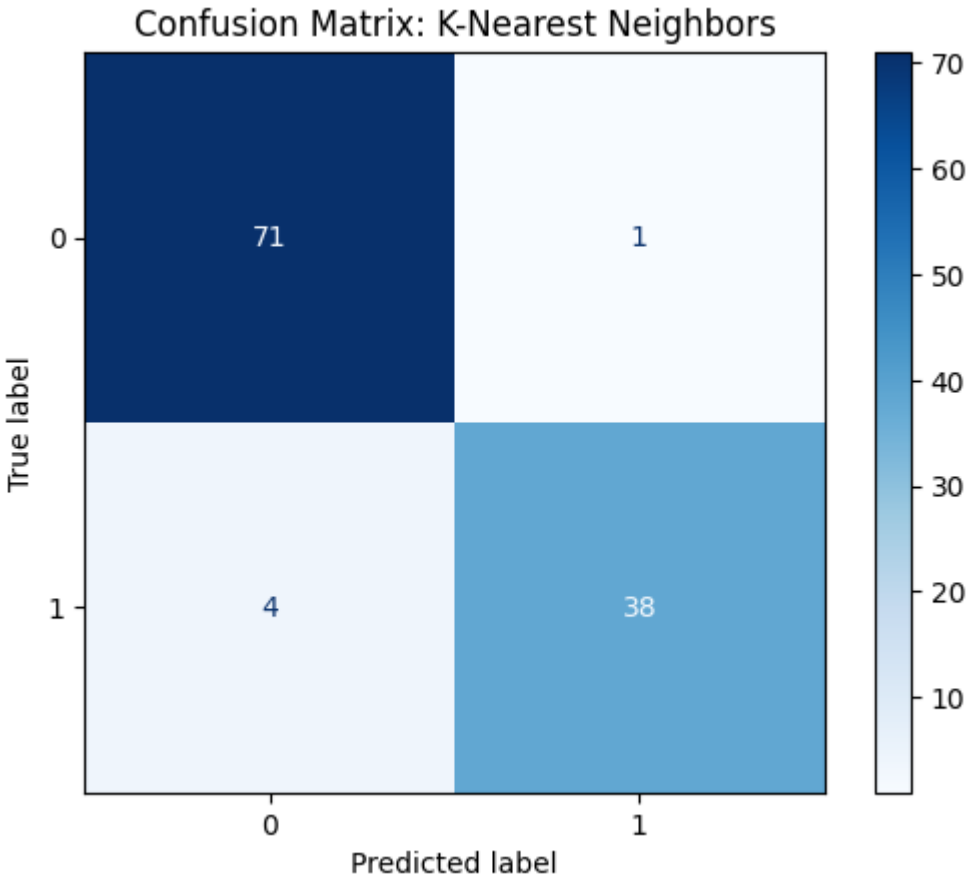

Figure 13 : Confusion Matrix – K-Nearest Neighbors

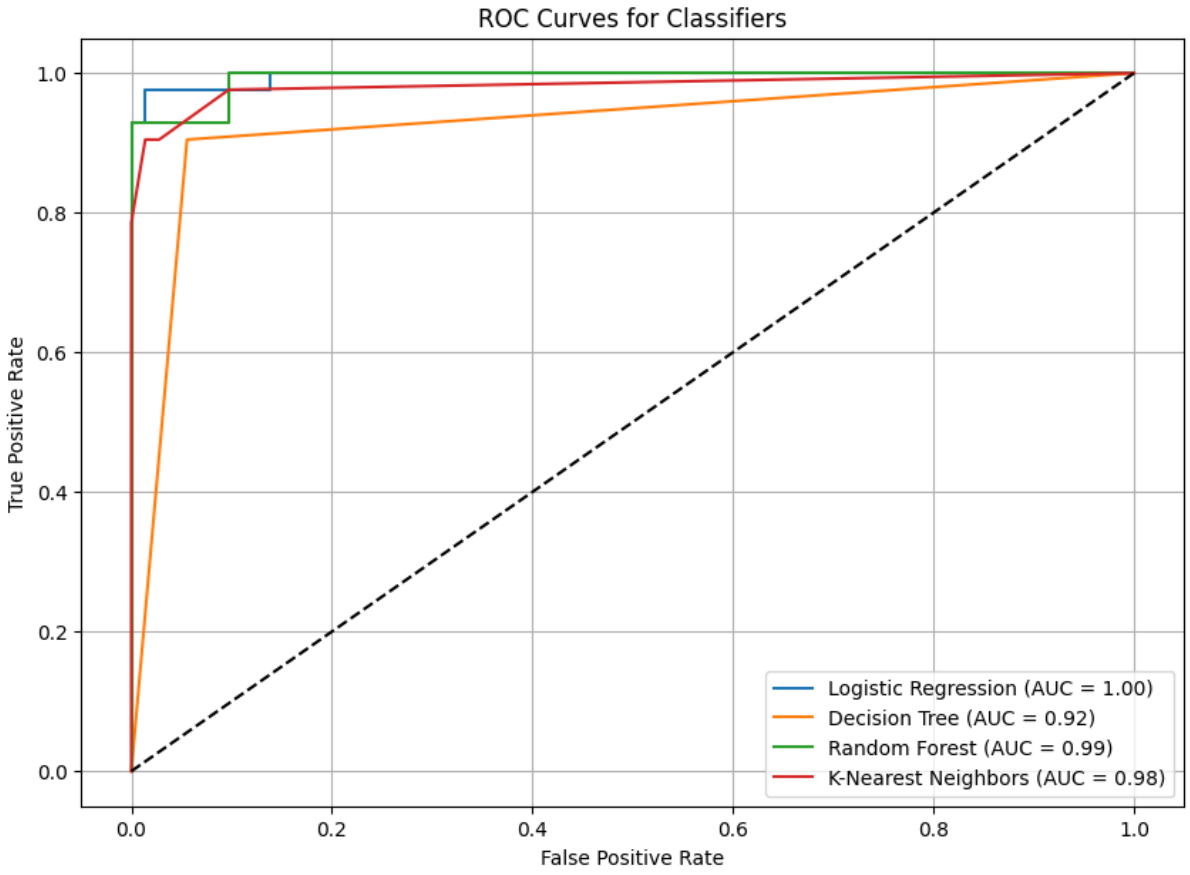

Figure 14 : ROC Curve Plot – Multiple Models Overlay

Given its interpretability and well-rounded performance, Logistic Regression was selected for fine-tuning. Using GridSearchCV, I tuned the C value and solver type. Post-tuning, the results were even stronger: accuracy rose to 98%, and the model correctly predicted 40 out of 42 malignant cases with zero false positives—something that even SVM couldn't claim (Hamedi et al., 2024).

Classification Report (Tuned Logistic Regression):

	precision	recall	f1-score	support
0	0.97	1.00	0.99	72
1	1.00	0.95	0.98	42
accuracy			0.98	114
macro avg	0.99	0.98	0.98	114
weighted avg	0.98	0.98	0.98	114

Figure 15 : Classification Report – Tuned Logistic Regression

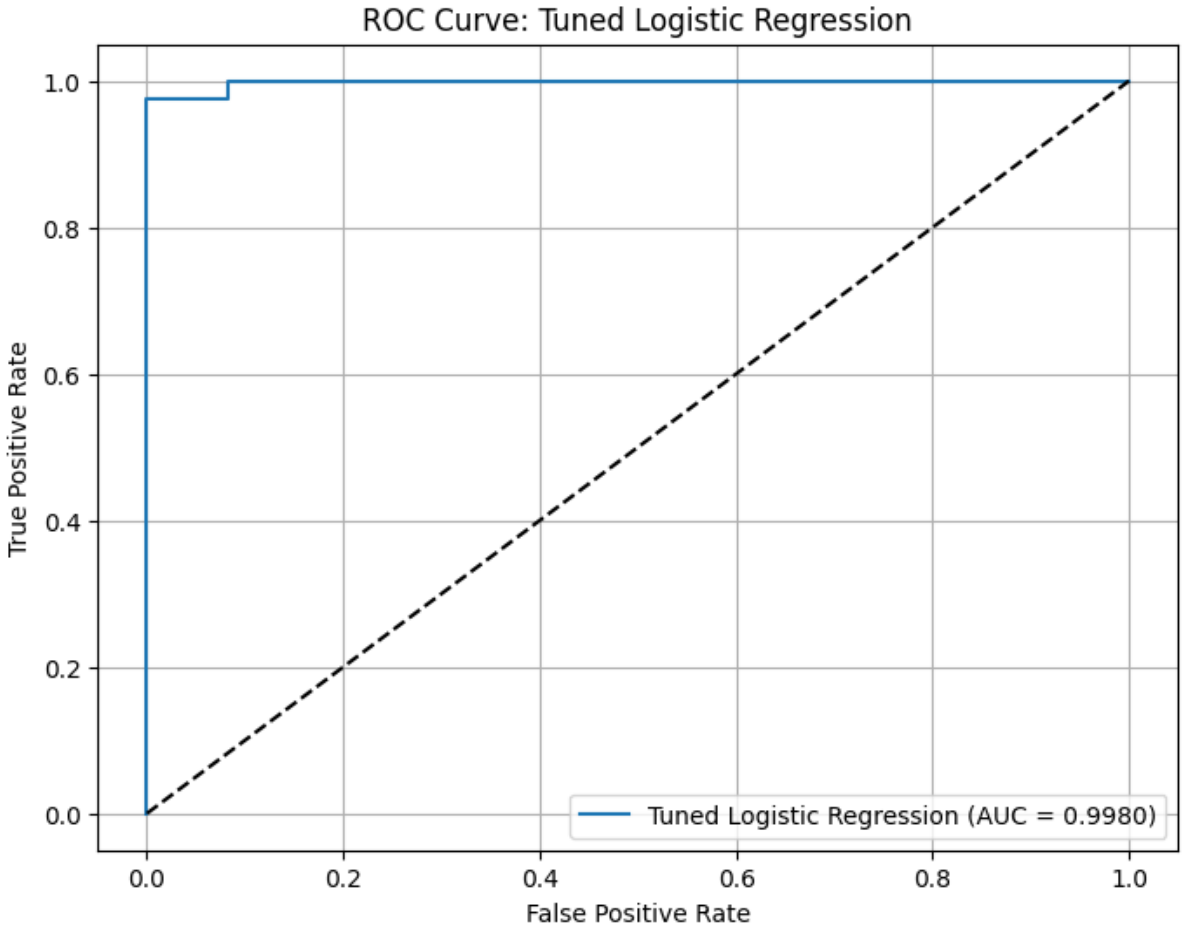

Figure 16 : ROC Curve – Tuned Model Only, AUC Highlighted

Shen et al. (2023) suggest that advanced architectures like transformers can offer incremental gains, but for structured, tabular datasets like this one, classical ML often holds its ground. In this case, simplicity wasn't a compromise—it was an advantage. This stage confirmed that good modelling is more about thoughtful design than just chasing complexity.

CONCLUDING REMARKS

This project set out to predict breast cancer diagnosis using classic machine learning classifiers and achieved that goal with precision, clarity, and interpretability. Among the five models tested, Logistic Regression emerged as the most balanced performer, earning an F1-score of 0.98 and an AUC of 0.998 post-tuning. Its transparency, combined with high predictive power, made it especially well-suited for a clinical setting (Ghasemi et al., 2024; Wei et al., 2023).

What stood out most was how much could be achieved with carefully preprocessed, well-understood data and thoughtfully selected models. That said, there's room to push further. Future work could explore deep learning approaches like Transformers (Shen et al., 2023), or integrate multi-modal data for richer predictions. Additionally, explainability tools like SHAP could be applied to reinforce trust in model outputs (Ghasemi et al., 2024). With continued refinement, machine learning can become a quiet but powerful ally in early cancer detection.

BIBLIOGRAPHY

Ghasemi, A., Hashtarkhani, S., Schwartz, D.L. and Shaban-Nejad, A. (2024). Explainable artificial intelligence in breast cancer detection and risk prediction: A systematic scoping review. *Cancer Innovation*, 3(5). doi:https://doi.org/10.1002/cai2.136.

Seyedeh Zahra Hamedi, Emami, H., Khayamzadeh, M., Reza Rabiei, Aria, M., Majid Akrami and Vahid Zangouri (2024). Application of machine learning in breast cancer survival prediction using a multimethod approach. *Scientific Reports*, 14(1). doi:https://doi.org/10.1038/s41598-024-81734-y.

Shen, Y., Park, J., Yeung, F., Goldberg, E., Heacock, L., Shamout, F. and Geras, K.J. (2023). *Leveraging Transformers to Improve Breast Cancer Classification and Risk Assessment with Multi-modal and Longitudinal Data*. [online] arXiv.org. Available at: https://arxiv.org/abs/2311.03217.

Wei, Y., Zhang, D., Gao, M., Tian, Y., He, Y., Huang, B. and Zheng, C. (2023). Breast Cancer Prediction Based on Machine Learning. *Journal of Software Engineering and Applications*, [online] 16(8), pp.348–360. doi:https://doi.org/10.4236/jsea.2023.168018.

Zuo, D., Liu, Y., Yu, J., Qi, H., Liu, Y. and Li, R. (2023). Machine learning-based models for the prediction of breast cancer recurrence risk. *BMC Medical Informatics and Decision Making*, 23(1). doi:https://doi.org/10.1186/s12911-023-02377-z.