ESPRIT

Machine Learning Final Project

Breast Cancer Detection and Risk Analysis

Team Members:

Mohamed Yassine Janfaoui
Ahmed Tounsi
Ahmed Rayen Aloui
Farah Boubaker
Farah Derbel
Mahdi Saoudi

Academic Year: 2025–2026

Contents

1

Business Understanding

Breast cancer remains one of the most prevalent and life-threatening diseases among women worldwide. Early detection and accurate diagnosis are critical factors that significantly improve patient survival rates and reduce treatment costs. However, traditional diagnostic processes rely heavily on expert interpretation of medical data, which can be time-consuming and subject to human error.

This project follows the CRISP-DM methodology to design and evaluate machine learning models that assist healthcare professionals in breast cancer detection, diagnosis, and prevention. The system is designed as a decision-support tool and does not aim to replace medical expertise.

Business Objectives

- **Confirm the presence of a breast tumor:** Provide reliable support to physicians by analyzing patient data and indicating whether the data suggest the presence of a suspicious breast tumor requiring further medical examination.
- **Characterize the tumor to guide clinical decisions:** Assist clinicians by determining whether a detected tumor is benign or malignant, enabling faster and more informed diagnostic decisions.
- **Detect early risk factors in healthy patients and recommend appropriate preventive actions:** Analyze patient profiles to identify early indicators of increased breast cancer risk and support preventive healthcare strategies.

2

Data Understanding

2.1 DSO1: Classify tumor type

2.1.1 Dataset Description

The Wisconsin Diagnostic Breast Cancer (WDBC) dataset contains features derived from digitized images of breast masses. It is widely used in breast cancer detection research due to its comprehensive and well-structured data.

2.1.2 Variable Description

The dataset includes 30 radiological features calculated for each cell nucleus, grouped into three categories:

- **Mean features** (10 features): Averages of basic measurements
- **Standard error features** (10 features): Standard deviations of measurements
- **Worst features** (10 features): Largest values observed

The features include:

- `radius`: Mean of distances from center to perimeter points
- `texture`: Standard deviation of gray-scale values
- `perimeter`: Cell perimeter
- `area`: Cell area
- `smoothness`: Local variation in radius lengths
- `compactness`: $\frac{perimeter^2}{area} - 1.0$

- `concavity`: Severity of concave portions of the contour
- `concave points`: Number of concave portions of the contour
- `symmetry`: Cell symmetry
- `fractal dimension`: "Coastline approximation" - 1

2.1.3 Dataset Characteristics

- **Total samples:** 569
- **Benign (B):** 357 samples (62.7%)
- **Malignant (M):** 212 samples (37.3%)
- **Features:** 30 numerical features
- **Target:** Diagnosis (Benign/Malignant)

2.1.4 Exploratory Analysis

- **Class distribution:** Moderate imbalance favoring benign cases
- **Missing values:** No missing values in the 30 features
- **Correlations:** Features related to concave points show the strongest correlation with diagnosis
- **Outliers:** Few outliers detected using the IQR method

2.2 DSO2: Histopathology Image Analysis

For DSO2, we worked with a comprehensive histopathology image dataset from Kaggle focused on Invasive Ductal Carcinoma (IDC) detection. This dataset is particularly valuable as it contains actual tissue samples from breast cancer patients.

2.2.1 Dataset Description

The dataset originates from 162 whole mount slide images of Breast Cancer (BCa) specimens scanned at 40x magnification. From these slides, 277,524 patches of size 50×50 pixels were extracted:

- **Total images:** 277,524 patches
- **Class 0 (non-IDC):** 198,738 patches (71.6%)
- **Class 1 (IDC):** 78,786 patches (28.4%)

2.2.2 File Naming Convention

Each patch follows a strict naming convention that encodes important metadata:

`patientID_idx*_x*_y*_class*.png`

Example: `10253_idx5_x1351_y1101_class0.png`

- `u`: Patient ID (e.g., 10253_idx5)
- `X`: x-coordinate where the patch was cropped from the original slide
- `Y`: y-coordinate where the patch was cropped from the original slide
- `C`: Class label where 0 = non-IDC and 1 = IDC

2.2.3 Key Statistics and Observations

- **Class Distribution:** Significant class imbalance with approximately 71.6% non-IDC and 28.4% IDC cases.
- **Data Integrity:** All 277,524 PNG files were validated against the naming convention with zero invalid filenames found.
- **Patient-Wise Splitting:** The data was split at the patient level (not image level) to prevent data leakage, ensuring that images from the same patient do not appear in both training and testing sets.
- **Visual Patterns:** Initial visual inspection revealed meaningful texture and structural patterns that distinguish IDC from non-IDC tissue.

2.3 DSO3: Identify risk factors in healthy patients

This DSO focuses on the analysis of clinical and biochemical data to identify early risk factors associated with breast cancer in patients who may not yet show detectable tumors.

2.3.1 Dataset Description

The dataset used for DSO3 contains clinical measurements collected from both healthy individuals and breast cancer patients. All variables are numerical and correspond to metabolic, hormonal, and inflammatory indicators.

2.3.2 Variable Description

The dataset includes the following features:

- **Age**: Age of the patient (years)
- **BMI**: Body Mass Index
- **Glucose**: Blood glucose concentration
- **Insulin**: Insulin level
- **HOMA-IR**: Insulin resistance indicator
- **Leptin**: Hormone related to fat metabolism
- **Adiponectin**: Hormone regulating glucose and lipid metabolism
- **Resistin**: Inflammatory marker
- **MCP-1**: Monocyte Chemoattractant Protein-1
- **Classification**: Target variable (0 = Healthy, 1 = Patient)

2.3.3 Dataset Characteristics

- **Total samples:** 4000
- **Healthy controls:** 1784
- **Patients:** 2216
- **Missing values:** None detected
- **Data distribution:** Slight class imbalance

Initial exploratory analysis revealed that several metabolic indicators (Glucose, BMI, Insulin) show noticeable differences between healthy and patient groups.

3

Data Preparation

3.1 DSO1: Classify tumor type

3.1.1 Data Cleaning

The dataset required minimal cleaning:

- Removal of the `Unnamed: 32` column which was entirely empty
- Conversion of the target variable: M=1 (Malignant), B=0 (Benign)
- Verification of the consistency of the 30 features

3.1.2 Feature-Target Separation

Data separation into features (X) and target (y):

- **Features:** 30 radiological characteristics
- **Target:** Binary variable (0: Benign, 1: Malignant)
- **ID:** The identifier column was removed

3.1.3 Train-Test Split

Data division according to the article methodology:

- **Training set:** 70% (398 samples)
- **Testing set:** 30% (171 samples)
- **Stratification:** Preservation of class distribution
- **Random state:** 42 for reproducibility

3.1.4 Feature Standardization

Feature normalization to improve model performance:

- **Method:** Z-score standardization
- **Formula:** $z = \frac{x-\mu}{\sigma}$
- **Implementation:** StandardScaler from scikit-learn
- **Note:** Parameters are fitted on the training set only

3.2 DSO2: Histopathology Image Analysis

For the image dataset, we implemented a comprehensive preprocessing pipeline tailored for deep learning models:

3.2.1 Data Loading and Preprocessing

- **Image Loading:** Images were loaded using TensorFlow's optimized I/O operations.
- **Resizing:** All images were resized to 50×50 pixels to maintain consistency with the original patch size.
- **Normalization:** Pixel values were scaled to the range [0, 1] by converting to float32 and dividing by 255.
- **Data Augmentation:** Applied random transformations (rotations, flips) to increase dataset variability and prevent overfitting.

3.2.2 Patient-Wise Data Splitting

To ensure realistic evaluation and prevent data leakage:

- **Patient Identification:** Extracted unique patient IDs from filenames.

- **Stratified Splitting:** Split patients (not individual images) into training (70%), validation (10%), and testing (20%) sets.
- **Class Balance Preservation:** Maintained similar class distributions across all splits.

3.2.3 TensorFlow Data Pipeline

Implemented an efficient data pipeline using TensorFlow's `tf.data` API:

- **Parallel Processing:** Used `.map()` for parallel image loading and preprocessing.
- **Shuffling:** Applied shuffling with a buffer size of 5000 to ensure random ordering each epoch.
- **Batching:** Created batches of 64 images for efficient GPU utilization.
- **Prefetching:** Used `.prefetch(1)` to overlap data preprocessing and model execution.

3.3 DSO3: Identify risk factors in healthy patients

3.3.1 Data Cleaning

The dataset required minimal cleaning. Column names were standardized, and all variables were verified to be numeric. No missing or duplicated values were detected.

3.3.2 Feature Engineering

To capture clinically relevant interactions between variables, additional features were created:

- **BMI_Glucose** = BMI × Glucose
- **HOMA_Leptin** = HOMA-IR × Leptin
- **Insulin_Resistin** = Insulin × Resistin
- **Adiponectin_Leptin_ratio** = Adiponectin / Leptin

These features reflect known biological relationships between metabolism, inflammation, and cancer risk.

3.3.3 Feature Scaling

All numerical features were standardized using the **StandardScaler** to ensure:

- Equal contribution of variables
- Improved convergence for SVM and logistic models
- Better performance for distance-based classifiers

3.3.4 Handling Class Imbalance

To address class imbalance, resampling techniques were applied:

- SMOTE
- SMOTETomek

The SMOTETomek method was selected for final training as it produces a balanced dataset while reducing noise near class boundaries.

4

Modeling

4.1 DSO1: Classify tumor type

Six machine learning algorithms were implemented and compared for binary classification of breast tumors. All models were trained with the exact hyperparameters specified in the reference article.

4.1.1 Model 1: Linear Regression

Rationale: Linear regression for binary classification uses a threshold function (0.5) to predict classes. While simple, it can provide a baseline for more complex models.

Parameters used:

- Batch size: 128
- Learning rate: 0.001
- Number of epochs: 3000
- Loss function: MSE (Equation 9 from the article)
- Optimizer: SGD

Architecture:

- Single linear layer: $y = WX + b$
- Threshold function: $\hat{y} = 1_{(y \geq 0.5)}$
- 31 trainable parameters (30 weights + 1 bias)

4.1.2 Model 2: Nearest Neighbor Classification (k=1)

Rationale: The k-nearest neighbors algorithm (with k=1) is a non-parametric geometric method that classifies points based on similarity to training samples.

Distance metrics evaluated:

- **Manhattan (L1):** $d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$ (Equation 12)
- **Euclidean (L2):** $d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$ (Equation 13)

Characteristics:

- No training required
- Direct geometric computation
- Sensitive to feature scaling (hence the importance of standardization)

4.1.3 Model 3: Softmax Regression

Rationale: Softmax regression extends logistic regression to multi-class problems. For binary classification, it is equivalent to logistic regression but formulated in a multi-class framework.

Parameters used:

- Batch size: 128
- Learning rate: 0.001
- Number of epochs: 3000
- Loss function: Cross-entropy (Equation 15)
- Optimizer: SGD

Architecture:

- Linear layer followed by softmax: $P(y = j|x) = \frac{e^{w_j^T x}}{\sum_{k=1}^{K} e^{w_k^T x}}$
- 62 trainable parameters (30×2 weights + 2 biases)
- One-hot encoding for labels

4.1.4 Model 4: Support Vector Machine (SVM)

Rationale: SVMs seek the optimal hyperplane that maximizes the margin between classes. The L2-SVM version used implements a quadratic penalty for errors.

Parameters used:

- Batch size: 128
- Learning rate: 0.001
- Number of epochs: 3000
- Parameter C: 5 (regularization strength)
- Norm: L2 (Equation 20)
- Optimizer: Adam

Loss function:

$$L = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \max(0, 1 - y_i(w^T x_i + b))^2$$

4.1.5 Model 5: Multilayer Perceptron (MLP)

Rationale: The MLP is a feed-forward neural network capable of learning complex non-linear relationships between features and target.

Architecture:

- **Input layer:** 30 neurons
- **Hidden layers:** 3 layers of 500 neurons each
- **Activation function:** ReLU
- **Output layer:** 2 neurons with softmax
- **Dropout:** Not used (according to article specifications)

Parameters used:

- Batch size: 128
- Learning rate: 0.01
- Number of epochs: 3000
- Loss function: Cross-entropy
- Optimizer: SGD

4.1.6 Model 6: GRU-SVM

Rationale: The GRU-SVM hybrid combines the sequential learning capabilities of GRUs with the classification power of SVMs. Although the dataset is not sequential, GRUs can learn hierarchical feature representations.

Architecture:

- **GRU layer:** 128 cells
- **Dropout:** 0.5 (training only)
- **SVM layer:** Linear dense layer
- **Parameter C:** 5

Parameters used:

- Batch size: 128
- Learning rate: 0.001
- Number of epochs: 3000
- Optimizer: Adam

4.2 DSO2: Histopathology Image Analysis

For DSO2, we developed and compared two deep learning architectures for IDC detection from histopathology images:

4.2.1 Model 1: Custom Convolutional Neural Network (CNN)

We designed a custom CNN architecture specifically tailored for the 50×50 histopathology images:

- **Architecture Rationale:** CNNs are ideal for image data as they automatically learn hierarchical spatial patterns (edges → textures → tissue structures) through convolutional layers with weight sharing.
- **Layer Structure:**
  1. Conv2D (32 filters, 3×3) + ReLU + MaxPooling2D
  2. Conv2D (64 filters, 3×3) + ReLU + MaxPooling2D
  3. Conv2D (128 filters, 3×3) + ReLU + MaxPooling2D

4. Flatten layer
5. Dense layer (128 units) + ReLU + Dropout (0.4)
6. Output layer (1 unit) + Sigmoid activation

- **Parameters:** 683,329 trainable parameters
- **Training:** 10 epochs with Adam optimizer, binary cross-entropy loss

4.2.2 Model 2: Transfer Learning with ResNet50

To leverage pre-learned features and potentially improve performance:

- **Base Model:** ResNet50 pre-trained on ImageNet (frozen weights)
- **Custom Head:**
  1. Global Average Pooling 2D
  2. Dense layer (128 units) + ReLU + Dropout (0.4)
  3. Output layer (1 unit) + Sigmoid activation
- **Parameters:** 23,850,113 total parameters (262,401 trainable)
- **Training:** 5 epochs with Adam optimizer, binary cross-entropy loss

4.2.3 Model Selection Rationale

- The custom CNN was designed for efficiency and interpretability with the specific image size.
- ResNet50 was chosen for transfer learning to benefit from features learned on large-scale image datasets.
- Both models use sigmoid activation for binary classification (IDC vs. non-IDC).

4.3 DSO3: Identify risk factors in healthy patients

Several machine learning models were evaluated to identify patients at higher risk of breast cancer based on clinical variables. The selected models cover different learning paradigms, including tree-based, margin-based, and ensemble approaches, allowing a comprehensive comparison under identical experimental conditions.

4.3.1 Model 1: Random Forest

Model Rationale: Random Forest is an ensemble learning method based on multiple decision trees. It is particularly well suited for structured medical data, as it can capture non-linear relationships and interactions between clinical variables while remaining robust to noise.

Parameters Used:

- Number of trees (*n_estimators*): 100
- Class weight: balanced
- Maximum depth: unconstrained
- Random state: 42

Strengths:

- Handles heterogeneous clinical features effectively
- Reduces overfitting through bootstrap aggregation
- Provides feature importance for medical interpretability

4.3.2 Model 2: Support Vector Machine (SVM)

Model Rationale: Support Vector Machines aim to find an optimal separating hyperplane that maximizes the margin between classes. SVMs are effective for complex, non-linear decision boundaries when combined with kernel functions.

Parameters Used:

- Kernel: Radial Basis Function (RBF)
- Regularization parameter: default
- Class weight: balanced
- Probability estimation: enabled

Strengths:

- Strong theoretical foundations
- Effective in high-dimensional feature spaces
- Suitable for non-linear separations

4.3.3 Model 3: Gradient Boosting

Model Rationale: Gradient Boosting builds a strong classifier by sequentially combining weak learners, where each new model focuses on correcting the errors of the previous ones.

Parameters Used:

- Number of estimators: 100
- Learning rate: default
- Maximum depth: default

Strengths:

- Capable of modeling complex non-linear patterns
- Good bias-variance trade-off
- Frequently used in structured data problems

4.3.4 Model 4: XGBoost

Model Rationale: XGBoost is an optimized gradient boosting algorithm designed for efficiency and scalability. It is widely used in medical and tabular data competitions due to its strong predictive performance.

Parameters Used:

- Number of estimators: 100
- Learning rate: 0.1
- Evaluation metric: log-loss
- Random state: 42

Strengths:

- High predictive power
- Efficient handling of non-linear relationships
- Built-in regularization to reduce overfitting

4.3.5 Training Strategy

Prior to modeling, the dataset was balanced using SMOTETomek to address class imbalance. The balanced dataset was then split using stratified sampling:

- **Training set:** 75%
- **Test set:** 25%

All models were trained using medically reasonable hyperparameters to preserve interpretability and avoid overfitting.

5

Evaluation

5.1 DSO1: Classify tumor type

5.1.1 Evaluation Metrics

All models were evaluated using the same metrics to allow fair comparison:

- **Accuracy:** Overall percentage of correct predictions
- **TPR (Sensitivity):** Ability to detect true positives (malignant cancers)
- **TNR (Specificity):** Ability to identify true negatives (benign cancers)
- **FPR:** False positive rate
- **FNR:** False negative rate
- **Confusion Matrix:** Detailed analysis of error types

5.1.2 Comparative Results

lightgray **Algorithm**	**Accuracy**	**TPR**	**TNR**	**Article Comparison**
Linear Regression	96.49%	90.62%	100.00%	+0.40%
Nearest Neighbor (L1)	95.91%	93.75%	97.20%	+2.34%
Nearest Neighbor (L2)	94.15%	90.62%	96.26%	-0.58%
Softmax Regression	98.83%	96.87%	100.00%	+1.17%
SVM (L2)	98.25%	95.31%	100.00%	+2.15%
MLP	96.49%	92.19%	99.07%	-2.55%
GRU-SVM	96.49%	90.62%	100.00%	+2.74%

Table 5.1: Algorithm Performance on WDBC Dataset

5.1.3 Performance Analysis

Top Performers

- **Softmax Regression:** Best overall accuracy (98.83%) with excellent TPR/TNR balance
- **SVM:** Performance very close to Softmax (98.25%) with perfect specificity
- **GRU-SVM:** Significant improvement over article results (+2.74%)

Moderate Performance

- **Linear Regression:** Solid performance despite its simplicity, exceeding article results
- **Nearest Neighbor:** Good performance, particularly with Manhattan distance

Error Analysis

- **False Negatives:** Critical in oncology, minimized by Softmax and SVM
- **False Positives:** Less critical but generate anxiety and additional tests
- **MLP:** Relative underperformance possibly due to overfitting despite deep architecture

5.1.4 Comparison with Reference Article

Overall, 5 out of 6 algorithms exceeded the performance reported in the reference article:

- **Improvements:** +0.40% to +2.74% depending on algorithm
- **Decline:** Only MLP performed worse (-2.55%)
- **Consistency:** Relative trends between algorithms are generally preserved

5.1.5 Clinical Implications

- **Softmax and SVM:** Recommended for their high accuracy and low false negative rate
- **Interpretability:** Linear regression offers the best clinical interpretability
- **Computation Time:** Simple models (regression, k-NN) are preferable for real-time deployment

5.2 DSO2: Histopathology Image Analysis

5.2.1 Evaluation Metrics

Both models were evaluated using comprehensive metrics suitable for medical image analysis:

- **Accuracy:** Overall correctness of predictions
- **Loss:** Binary cross-entropy loss during training and testing
- **Confusion Matrix:** Detailed breakdown of true positives, false positives, true negatives, false negatives
- **ROC-AUC:** Area under the Receiver Operating Characteristic curve

5.2.2 Model Performance Results

Custom CNN Model:

- **Test Accuracy:** 85.69%
- **Test Loss:** 0.3918
- **Training Time:** 500 seconds per epoch

ResNet50 Transfer Learning Model:

- **Test Accuracy:** 85.78%
- **Test Loss:** 0.3516
- **Training Time:** 700 seconds per epoch (slower due to larger architecture)

5.2.3 Comparative Analysis

- **Performance:** Both models achieved similar accuracy (85.7-85.8%), with ResNet50 showing slightly better loss reduction.
- **Efficiency:** The custom CNN trained faster (10 epochs in 5000s vs. ResNet50's 5 epochs in 3500s).
- **Interpretability:** The custom CNN is more interpretable with fewer parameters and simpler architecture.
- **Clinical Relevance:** High recall is crucial for cancer detection to minimize false negatives (missed IDC cases).

5.2.4 Key Insights

- Patient-wise splitting prevented optimistic bias that could occur with random image splitting.
- Class imbalance was addressed through appropriate loss functions and evaluation metrics.
- Both architectures demonstrated capability in learning discriminative features from histopathology images.

5.3 DSO3: Identify risk factors in healthy patients

5.3.1 Evaluation Metrics

Given the medical application, evaluation focused on both predictive performance and clinical relevance:

- Accuracy
- Precision
- Recall (Sensitivity)
- F1-score
- ROC-AUC
- Confusion Matrix

Recall was prioritized to minimize false negatives, which correspond to missed high-risk patients.

5.3.2 Comparative Results

lightgray **Algorithm**	**Accuracy (%)**	**F1-score**
Random Forest	63.92	0.636
Support Vector Machine (SVM)	56.85	0.569
XGBoost	56.43	0.569
Gradient Boosting	55.80	0.533

Table 5.2: Performance comparison of machine learning models for DSO3

5.3.3 Model Performance Comparison

Among all tested models, the Random Forest classifier achieved the best overall performance:

- **Accuracy:** 63.9%
- **ROC-AUC:** 0.691
- **Sensitivity (Recall):** 0.633
- **Specificity:** 0.646

Other models such as SVM and XGBoost showed competitive accuracy but lower ROC-AUC.

5.3.4 ROC Curve Analysis

ROC curve analysis confirmed that Random Forest provides the strongest discrimination capability between healthy individuals and patients compared to other models.

5.3.5 Clinical Interpretation

Feature importance analysis revealed that metabolic indicators such as Glucose, BMI, HOMA-IR, and Insulin are key contributors to breast cancer risk. These findings are consistent with established medical research linking metabolic syndrome to cancer development.

6

Deployment

To make the developed models accessible and usable in a real-world context, a web-based deployment was implemented.

A web application was developed using the **Flask** framework for the backend and **HTML, CSS, and JavaScript** for the frontend. The application allows users to input patient data through a graphical interface, which is then processed by the trained machine learning model.

The backend handles data validation, preprocessing, and model inference, while the frontend provides clear and intuitive feedback regarding tumor presence predictions. This deployment approach demonstrates how machine learning models can be integrated into practical clinical decision-support systems.

For text-based models (DSO1 and DSO3):

- Forms with data verification
- Real-time preprocessing (data preparation pipeline)
- Model confidence scores

For image-based model (DSO2):

- Image upload functionality for histopathology patches
- Real-time preprocessing (resizing, normalization)
- Model confidence scores

Sources

- Scikit-learn Documentation:
  https://scikit-learn.org
- TensorFlow Documentation:
  https://www.tensorflow.org
- World Health Organization (WHO):
  Breast Cancer Fact Sheets
- Histopathologic images:
  https://www.kaggle.com/datasets/paultimothymooney/breast-histopathology-images
- Clinical data:
  https://www.kaggle.com/code/yasserhessein/breast-cancer-coimbra-classification-with-eda-ml/input
- WDBC:
  https://www.kaggle.com/code/nancyalaswad90/analysis-breast-cancer-prediction-dataset/input