CLASSIFICATION CHALLENGE :

CADx PROJECT: SKIN LESION CLASSIFICATION CHALLENGE USING CLASSICAL MACHINE LEARNING APPROACH

Presentation by:

Daniel Tweneboah Anyimadu & Taofik Ahmed Suleiman

Supervised by:

Dr. Xavier Llado & Dr. Arnau Oliver

ABSTRACT

- This study focuses on the classification of skin lesions through a classical machine-learning (ML) approach employing a range of ML classifier models.

- Our approach addresses both two-class and three-class problems, improving classification accuracy despite huge class imbalances and lesion variations.

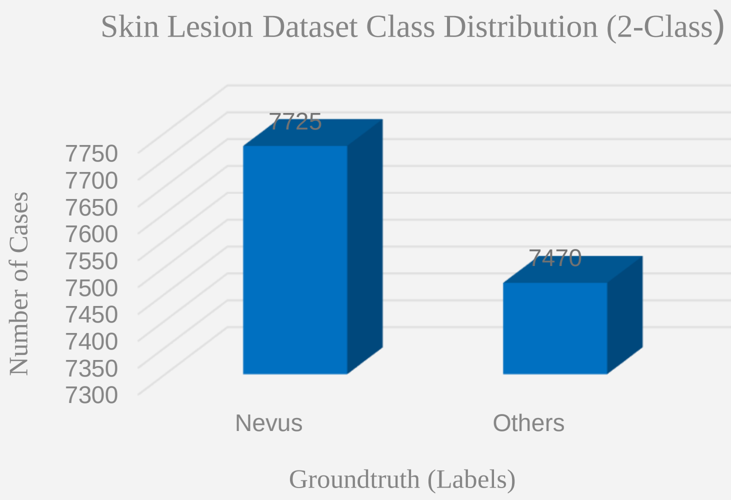
- The challenge dataset was provided by our course organizers and includes images from the HAM10000, BCN_20000, and MSK datasets, offering a rich variety of dermatological samples for analysis. The project involves exploratory data analysis, data preprocessing, feature extraction, feature engineering, model training, classification, and performance evaluation.

- Two-class problems are assessed using overall accuracy, while three-class problems are evaluated using kappa values on a test dataset.

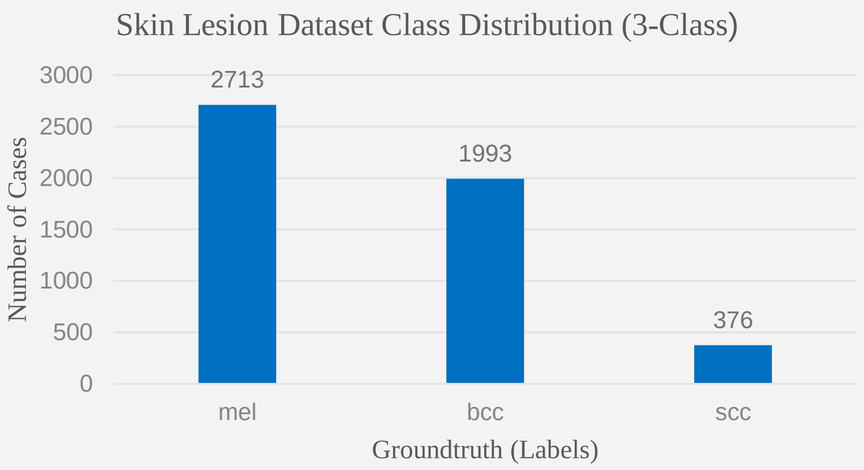
- The Ensemble model (RF + XGBoost) achieved an accuracy of 0.83 on the two-class problem where as MLP achieved a kappa value of 0.71 on the 3-class

Exploratory Data Analysis (EDA)

- 2 Class Distribution Plot :



- 3 Class Distribution Plot :



- A visualization of class imbalance shown in the class distribution plots.

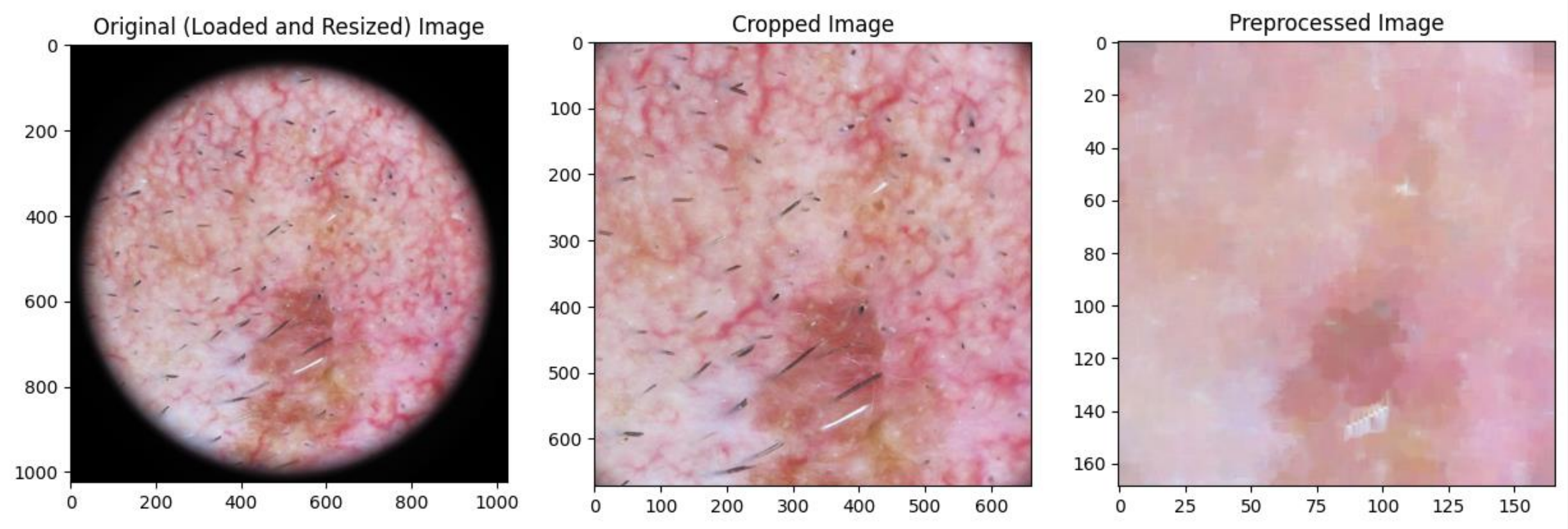
Data Preprocessing: Vignette Frame and Hair Removal – 2 Class

Vignette Frame Removal:

- Define a function that takes an image and a threshold as input parameters. It then detects the threshold where darkening occurs and crops the image accordingly.

Hair Removal:

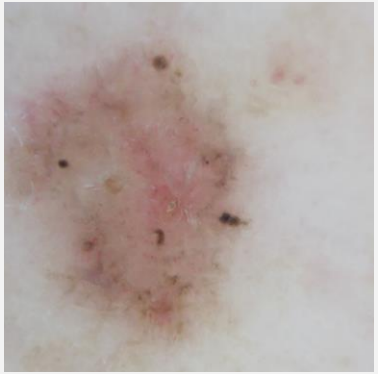
- Two input parameters: an image and a structuring element. The function applies morphological black-hat filtering and thresholding operations to remove black hairs.

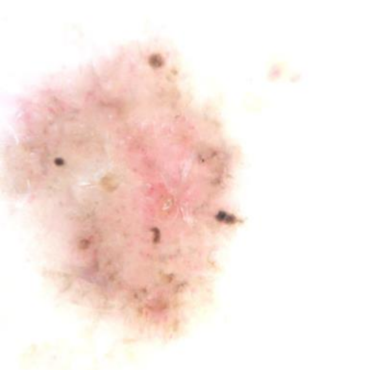


Data Preprocessing: Handling Imbalanced Data – 3 Class

❑ Data Augmentation:

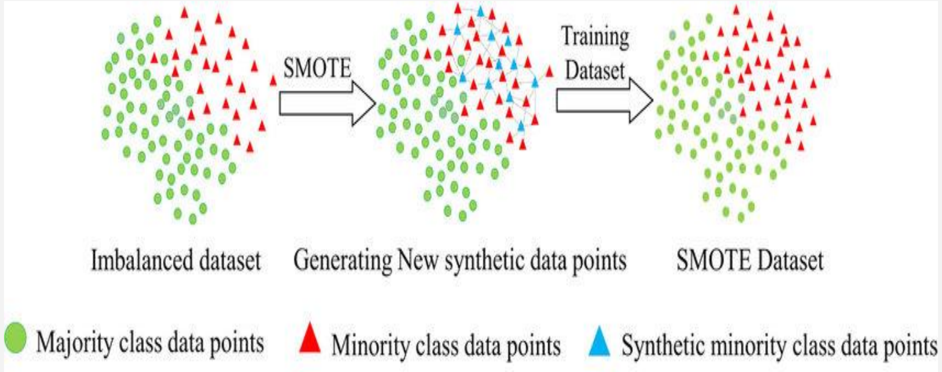
- Central cropping, rescaling, and contrast enhancement techniques are used to create augmented images for both "bcc" and "scc" classes to improve the robustness and diversity of the dataset while increasing sample numbers.





❑ Synthetic Minority Over-Sampling Technique:

- SMOTE is applied to balance the number of samples in "bcc" and "scc" by generating synthetic examples.

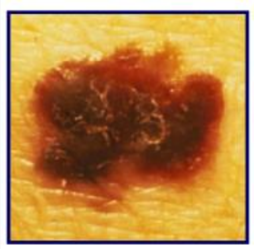


Feature Extraction

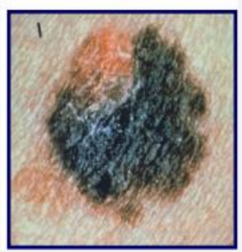
❑ ABCD RULE:

- We applied manual feature extraction to the ABCD rule, encompassing image asymmetry, border characteristics, color properties, and image diameter. These features were derived from shape, color, and texture.

asymmetry



border



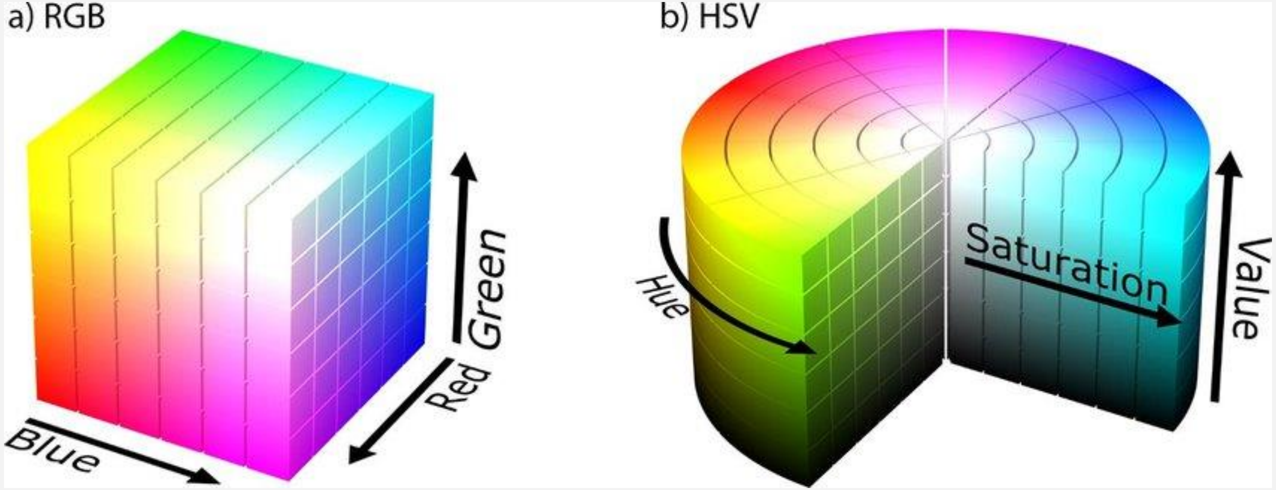
color

diameter

Feature Extraction Cont'd: Color Features

❑ CHANNEL-WISE COLOUR FEATURES :

- Various statistics were computed to characterize color and intensity information in the image using both RGB (standard color representation for image display) and HSV (color representation based on color dimensions).
- Mean
- Std
- Min
- Max
- Skewness
- Kurtosis
- Entropy



Feature Extraction Cont'd: Shape Features

- Grayscale images simplify contour detection. Key techniques for extracting shape features include:

- Canny Edge Detection: Identifies edges by suppressing noise using a Gaussian filter, finding gradients and directions with Sobel filters, and suppressing non-maximum values.

- Hu Moments: These shape descriptors are translation, rotation, and scale-invariant. Computed from central moments, they consist of 7 moments for a concise shape representation.

- Perimeter

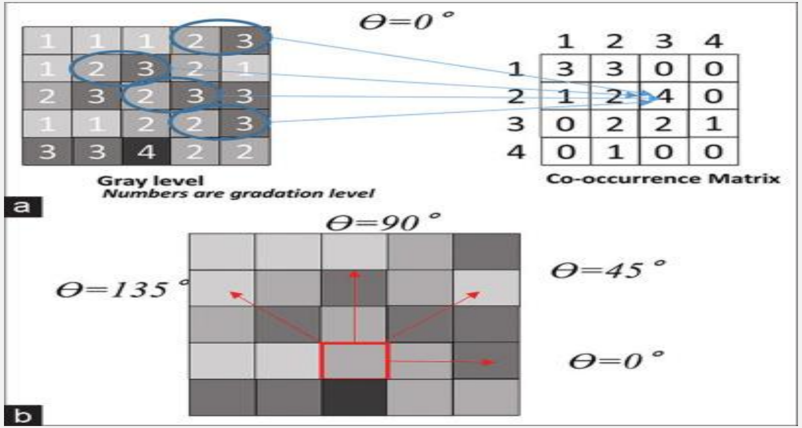
- Area

- Circularity

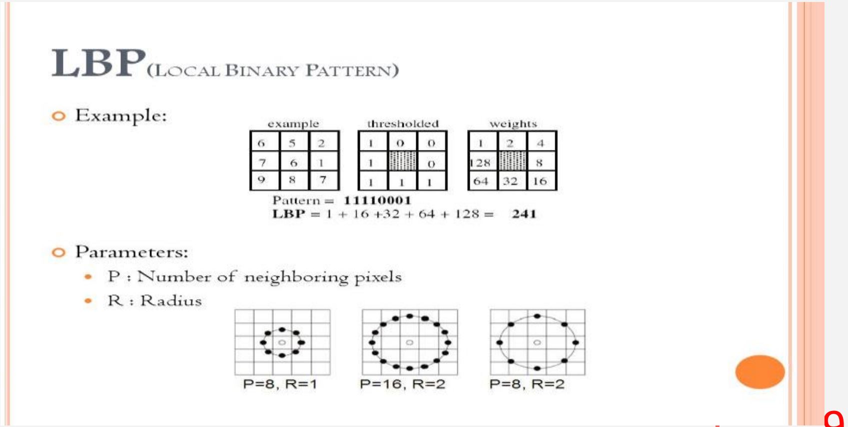
- Compactness

Feature Extraction Cont’d: Texture Features

- GLCM: Analyzes pixel intensity spatial arrangement using parameters of pixel pair distance and orientation.
  - Distances = [2, 5, 7, 10, 15]
  - Angles = [0, 45, 90, 135]



- LBP: Utilizes radius and point count to compute texture information.
  - Radius = [1, 2, 3, 4, 5, 6, 7, 8, 9]
  - Number of points = 8 * r



Feature Engineering: Scaling & Selection

- Feature Scaling :
  - Extracted classical ML features are normalized using standardization to achieve consistent feature distributions. This transforms the features to have a mean of 0 and a standard deviation of 1.

- Feature Selection:
  - We leveraged Principal Component Analysis (PCA) to select the top 70 components, reducing noise and multicollinearity.
  - Additionally, we explored SelectKBest with k = 100 for feature selection.
  - A comparison shows that PCA outperforms SelectKBest.

Machine Learning (Model Training)

❑ Machine Learning Models:

- A subfield of AI which involves developing models to analyze massive datasets and extract meaning patterns and insights for automatically making predictions.

❑ Ensemble model (Random Forest + XGBoost)

❑ XGBoost:

- Combines strength of decision trees with a gradient boosting framework.

❑ Random Forest:

- This model leverages decision trees to provide robust predictions.

❑ K-Nearest Neighbors:

- K-NN relies on proximity to neighbors for classifying skin lesion images.

❑ Support Vector Machine:

- SVM excels in finding optimal hyperplanes to separate different classes effectively.

❑ Multi-Layer Perceptron Classifier:

- A network of interconnected neurons that learns by adjusting connections between neurons.

PERFORMANCE EVALUATION (ACCURACY & KAPPA)

- TWO-CLASS PROBLEM:

MODEL (ML)	10-FOLD CROSS (TRAINING)	ACCURACY
Ensemble model (Random Forest + XGBoost)	0.8309	0.8314
XGBoost	0.8195	0.8280
Random Forest	0.8126	0.8280
K-Nearest Neighbors	0.7816	0.7830
Support Vector Machine	0.8030	0.8087
Logistic Regression	0.7841	0.7882
AdaBoost	0.8063	0.7858

PERFORMANCE EVALUATION (ACCURACY & KAPPA)

- ❑ THREE-CLASS PROBLEM:

MODEL (ML)	5-FOLD CROSS (TRAINING)	KAPPA
Ensemble model (Random Forest + XGBoost)	0.836660842	0.6705
XGBoost	0.83499672	0.6705
Random Forest	0.824372922	0.6705
K-Nearest Neighbors	0.762031794	0.5512
Support Vector Machine	0.764080534	0.6012
Multi-Layer Perceptron Classifier	0.833205126	0.7136

CONCLUSION

- Based on the results, the combination of preprocessing, feature extraction (color, shape, and texture), feature engineering (scaling, PCA), and the use of both random forest and XGBoost classifiers as ensemble model enhanced the accuracy performance of our binary classification model.

- Additionally, leveraging the same methodology + data augmentation with the Multi-layer Perceptron classifier led to improved kappa performance in our multi-classification model.

- **THANK YOU FOR YOUR ATTENTION!!!**