Classifying Melanoma Cancer with Neural Networks

Presented By: Team Biscuit Boys
Claire Blackman, Vedika Birla, Max Van Fleet, William Petty, and Ryan Helms

Our Data

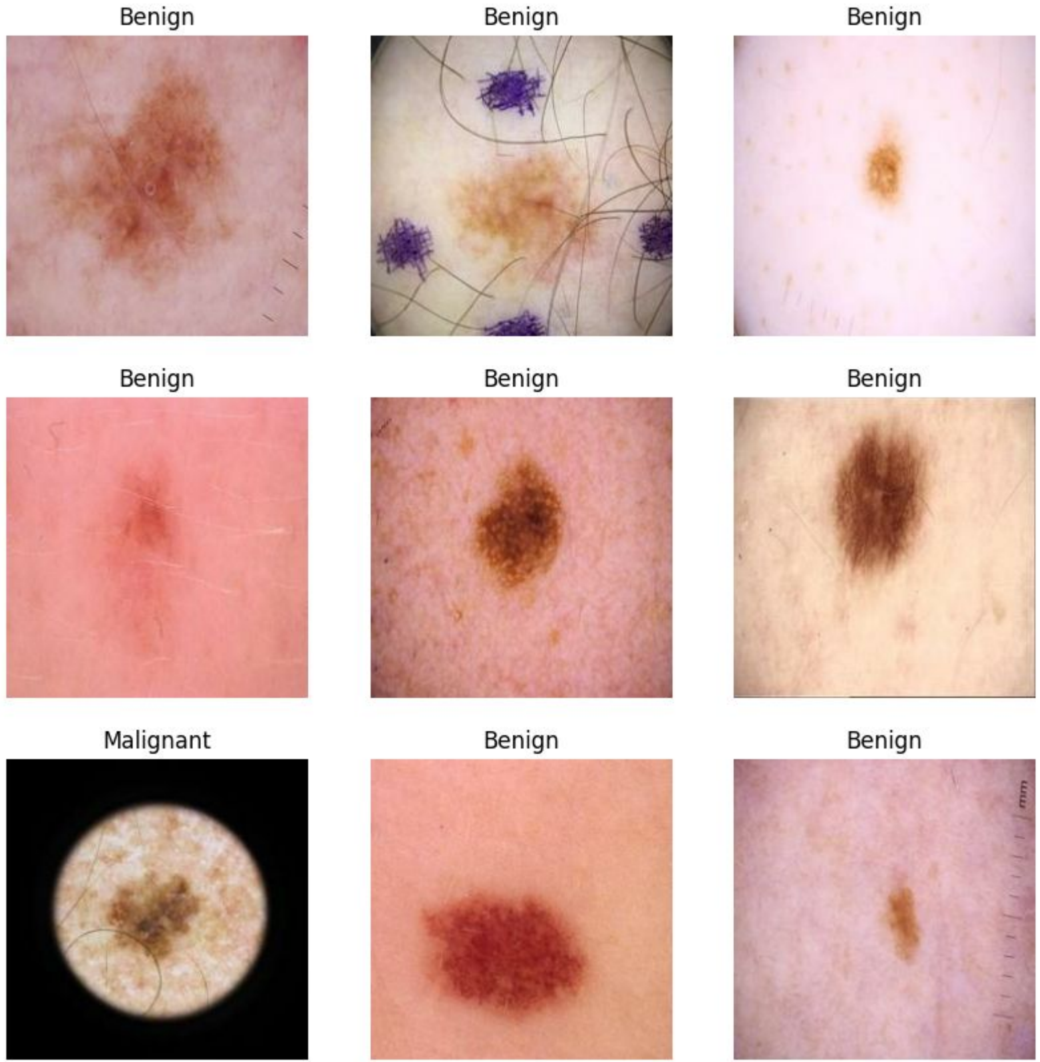
- Our data set is from Kaggle user Bhavesh Mittal titled *Melanoma Cancer Image Dataset*.
- The data contains 13,900 unique 224 by 224 pixel images of melanoma lesions.
- The data is separated into training and testing sets:
  - Each set contains a benign folder and malignant folder
  - Allows us to create a model on the training data and test our classification accuracy

Importance

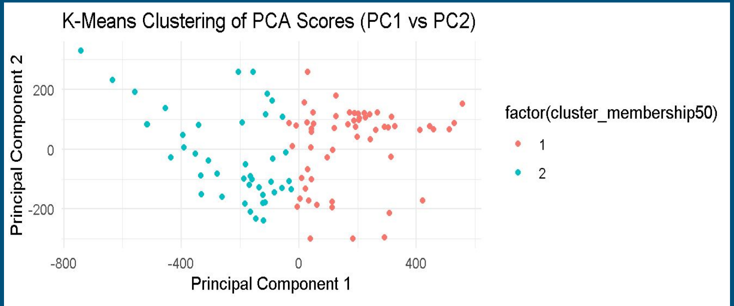
- Over 100,000 melanoma lesions are expected to be diagnosed in 2024
- 8000+ people are expected to die from melanoma in 2024
- Approximately 20% of Americans will be diagnosed with skin cancer by the age of 70
- Benign melanoma is not cancerous while malignant melanoma is cancerous and can spread throughout the body
- Early detection leads to a survival rate of 99%

Motivation

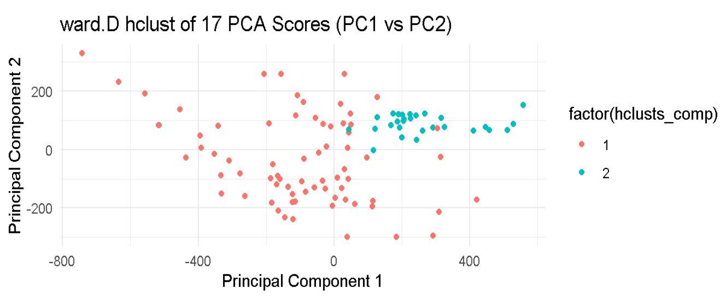
- We were interested in working with data related to healthcare due to its significance and wide applications
- By creating a model that can accurately classify melanoma lesions, we can take images of skin lesions and classify whether it is benign or malignant
- This has significant implications in healthcare, as it can help contribute to early detection
- Benign tumors typically have smoother and more defined borders while malignant tumors do not have regular borders leading to our models aiming to classify these differences


Benign
Benign
Benign
Benign
Benign
Benign
Malignant
Benign
Benign

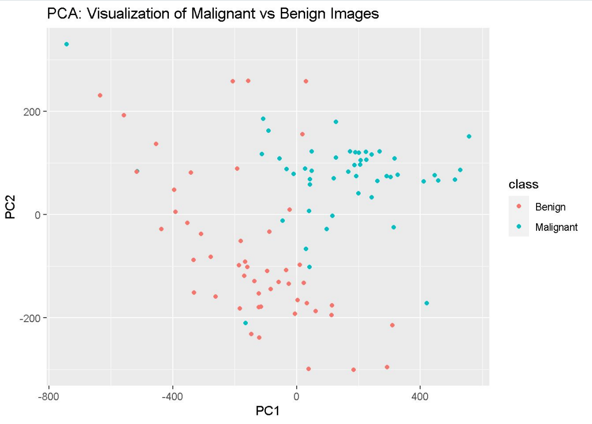
Exploratory Data Analysis



K-Means with a k = 2 showed the best results based on Adjusted Rand Index (0.4298)



Ward.D performed the best from the various hierarchical clustering techniques based on Adjusted Rand Index (0.30796)



Our EDA showed promising results for separating the images based on malignant and benign melanoma lesions based on the following plots: seen by the fairly clear separation by the PCA plots. First 3 PC values explained **65%** of the variation while first 17 PC values explained **90%** of the variation. PC is not the most appropriate method for this data, as the data is not linear.

Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a type of deep learning algorithm mainly designed for image classification tasks.

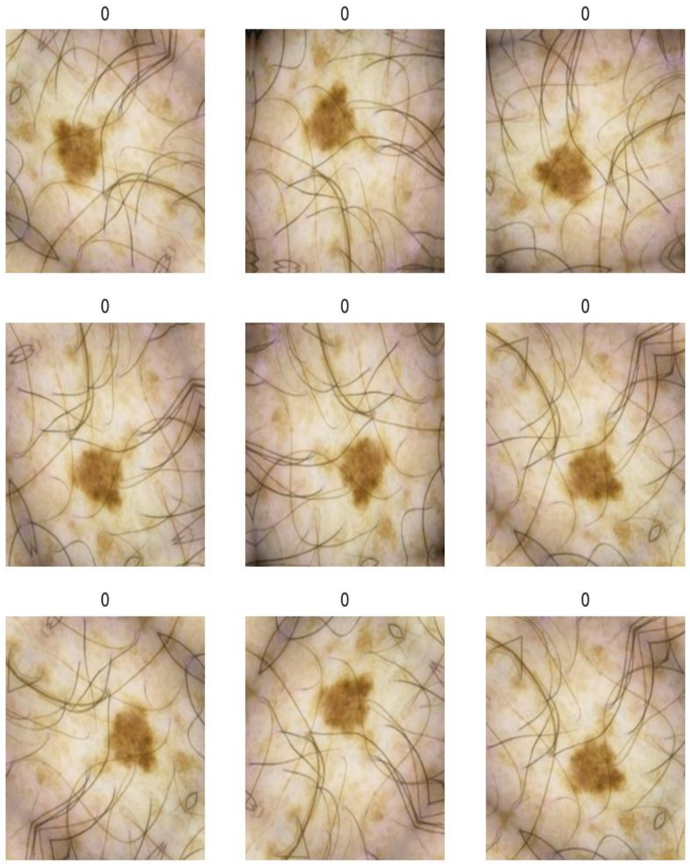
Key components of a CNN:

- **Convolutional layers:** These layers apply filters to input images to identify specific patterns like edges, textures, and shapes.
- **Pooling layers:** Pooling layers pull out the most important features from the convoluted matrix by reducing the dimension of the feature map. They also help prevent overfitting.
- **Activation functions:** Activation functions are applied to introduce an element of nonlinearity into the model, allowing it to learn complex patterns in the data. They also help mitigate the vanishing gradient problems.
- **Fully connected layers:** These layers are in the last layer of the CNN and are used to make final predictions.

CNNs are trained using backpropagation, where the model finetunes its internal parameters based on the error rate in order to minimize the difference between its predictions and the ground truth labels in the training data. CNNs have been highly successful in various applications, including image classification, object detection, facial recognition, and medical image analysis.

Pre-Processing

- Images were already scaled to 224 by 224 pixels
- Data Augmentation to generate more training examples:
  - Rotations
  - Reflections
- Xception expects values between [-1, 1], requiring an additional preprocessing step



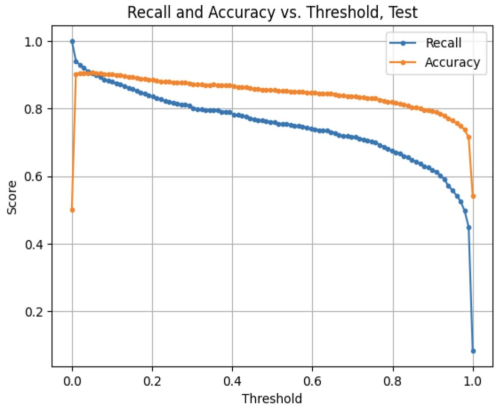
Our Models

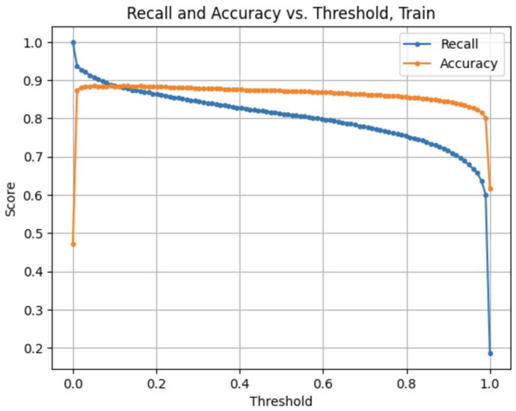
- Experimented with a few different pre-built models for feature extraction
  - Xception
  - EfficientNetV2B0

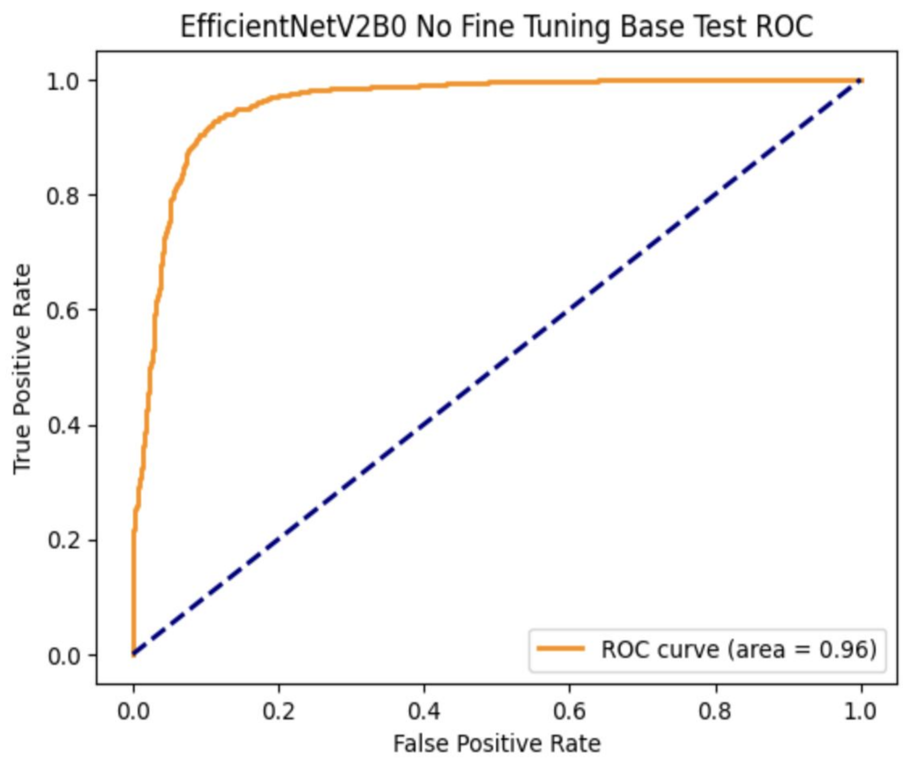
Model	Validation Accuracy
Xception w/ Fine Tuning	89.00%
EfficientNetV2B0 Feature Extraction only	87.45%
EfficientNetV2B0 w/ Fine Tuning	85.47%

EfficientNetV2B0 Feature Extraction Only

	Actual Benign	Actual Malignant
Predicted Benign	949	240
Predicted Malignant	51	760

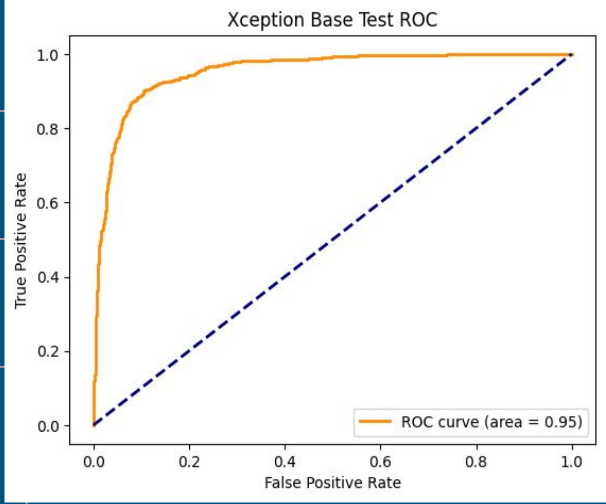


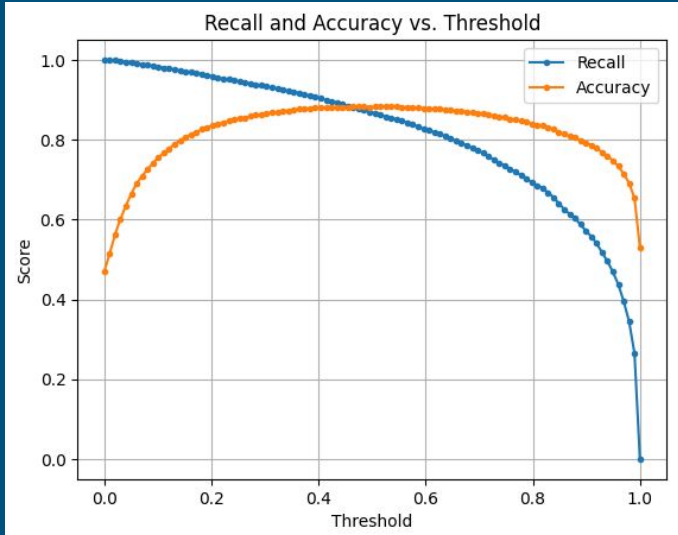




Xception

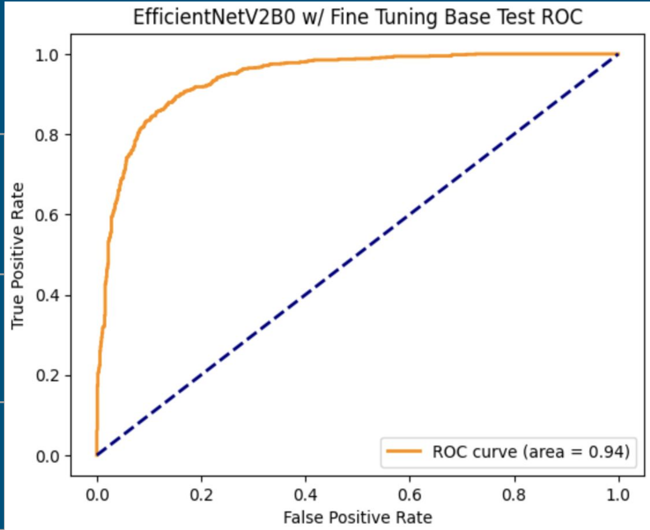
	Actual Benign	Actual Malignant
Predicted Benign	937	170
Predicted Malignant	63	830

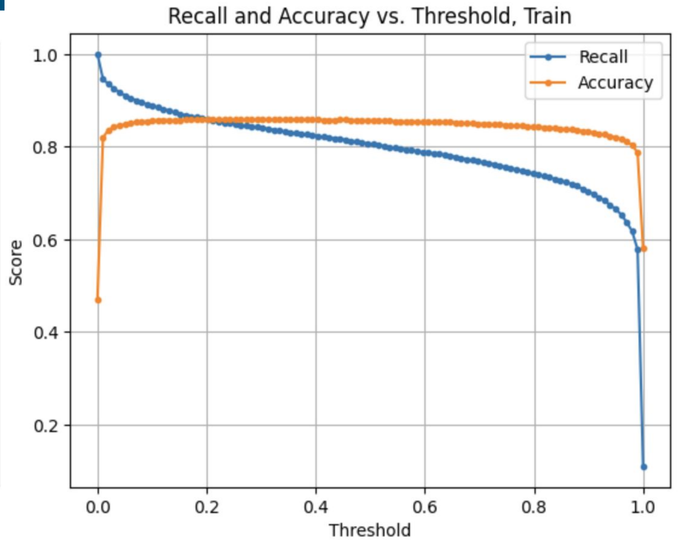




EfficientNetV2B0 W/ Fine Tuning

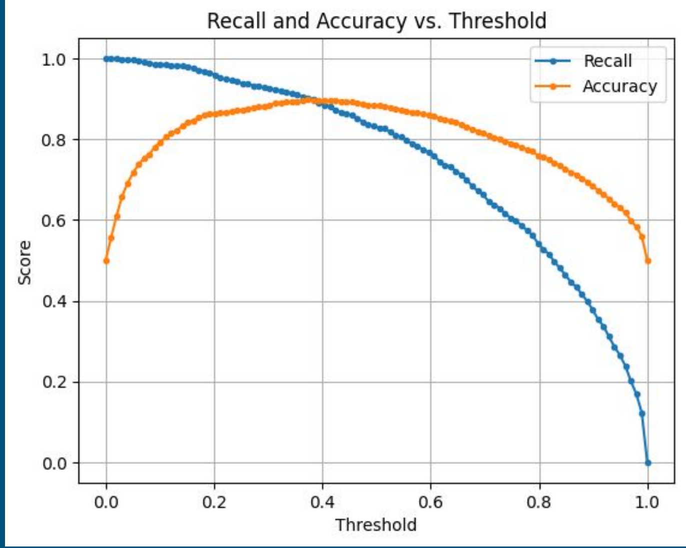
	Actual Benign	Actual Malignant
Predicted Benign	946	261
Predicted Malignant	54	739





Conclusion

- Overall our study was pretty limited
  - Doctors already have good tools for diagnosis
- Additional areas to improve:
  - Pre-processing and data augmentation
  - Experimentation with other training parameters, i.e. batch size
  - Freezing Layers for model fine tuning
  - Resource consumption management
- Model Generalizability:
  - Other topical skin conditions
  - Could be more useful when there is more ambiguity



Works Cited

1. Bhavesh Mittal. (2024). 🔬*Melanoma Cancer Image Dataset.* Kaggle. https://www.kaggle.com/datasets/bhaveshmittal/melanoma-cancer-dataset
2. American Cancer Society. (2024, January 17). *Key Statistics for Melanoma Skin Cancer.* American Cancer Society. https://www.cancer.org/cancer/types/melanoma-skin-cancer/about/key-statistics.html#:~:text=About%20100%2C640%20new%20melanomas%20will,5%2C430%20men%20and%202%2C860%20women
3. Skin Cancer Foundation. (2024, February). *Skin Cancer Facts & Statistics*. Skin Cancer Foundation. https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/
4. City of Hope. (2023, January 12). *What's the difference? Benign vs. malignant tumors.* City of Hope. https://www.cancercenter.com/community/blog/2023/01/whats-the-difference-benign-vs-malignant-tumors#:~:text=While%20benign%20tumors%20generally%20don,They%20also%20grow%20faster
5. Cleveland Clinic. (2021, August 10). *Benign Tumor*. Cleveland Clinic. https://my.clevelandclinic.org/health/diseases/22121-benign-tumor
6. Keita, Z. (2023, November 14). *An introduction to Convolutional Neural Networks.* datacamp. https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns
7. Keras. *Keras Applications.* Keras. https://keras.io/api/applications/