CS4641-Team36

CS 4641 Project Final Report - Team 36

Skin Lesion Detection Model

Introduction

Skin cancer is one of the most common forms of cancer worldwide, and early detection is crucial for effective treatment. Traditional diagnostic methods rely on clinical examination and biopsy, which can be time-consuming, resource-intensive, and subject to variability by dermatologists. Recent advancements in machine learning, particularly deep learning, have shown significant promise in the automated classification of skin lesions. Thus our project aims to develop an ML model capable of predicting the diagnosis of a skin lesion from dermatoscopic images and patient metadata. We will be using the HAM10000 dataset, available on Kaggle (link to dataset); This dataset includes 10,015 dermatoscopic images of pigmented skin lesions, annotated with corresponding data such as the patient's sex, age, and the localization of the lesion on the body. The dataset categorizes seven types of common pigmented skin lesions: melanocytic nevi, melanoma, benign keratosis-like lesions, basal cell carcinoma, actinic keratoses, vascular lesions, and dermatofibroma (Tschandl et al., 2018).

Problem Definition

We propose to develop a model that predicts the type of skin lesion using the proposed dataset. Our approach will preprocess the image data, and apply a convolutional neural network (CNN) for classification, with additional layers that incorporate patient metadata (sex, age, and lesion localization). While existing literature has demonstrated the effectiveness of CNNs in classifying dermatoscopic images, many studies focus solely on image data without considering patient-specific metadata. By combining both images and patient data, our model aims to improve classification accuracy over these existing methods.

Methods

Data Preprocessing

For data processing, our group used a pre-trained deep learning model, ResNet-50, as a feature extractor, allowing our implemented ML models to run smoothly and efficiently. According to multiple healthcare professionals with expertise in using deep learning algorithms for disease detection and prevention, using a ResNet50 model yielded the greatest classification accuracy in comparison to other common pretrained models such as AlexNet, VGG-16, ResNet-18, and ResNet-34. According to a study completed by Muhammad Talo, Ozal Yildirim, Ulas Baran Baloglu, et. al. on brain disease detection and classification using MRI images, ResNet-50 obtained a classification accuracy of 95.23% ± 0.6%. Their team attributed the success of ResNet-50 to its "modern architecture" and its ability to transform high dimensional images that are very noisy into a 2048-dimensional feature vector that can capture high-level visual patterns. Because of its success in a case study regarding medical image classification, our group decided to implement this model as a feature extractor to transform our raw skin lesion images into a format that is more digestible for our machine learning algorithms.

In our data-preprocessing code, we iterated through each image, recoloring and resizing the image to meet the criteria for passing it through the ResNet-50 model (a 224 x 224 RGB image). From there, we converted the image to an array, added a dimension so it became a 4D vector to align with ResNet's required input parameters, and applied the ResNet-50 specific normalization. We then passed the preprocessed image through the ResNet model which outputs a 2048-dimensional feature vector that we flattened to a 1D vector with the dimension (2048,). Lastly, we appended the extracted feature vector to a list for use in our machine learning algorithms.

KMeans

We chose KMeans as our first ML algorithm to explore an unsupervised learning approach to classification, and to further understand our data - if it has any natural groupings, and to see how many visually similar images exist. This can give us an interesting comparison and contrast to when we use a supervised learning approach. KMeans was more promising compared to other unsupervised learning approaches as we already know the number of clusters. Since our dataset consists of 7 different types of skin lesion diagnosis, we experimented with around 7 clusters.

As outlined in our project proposal, we initially planned to implement DBSCAN alongside KMeans for comparison, as DBSCAN can detect clusters of arbitrary shape and handle noise, whereas KMeans has high noise sensitivity and assumes circular clusters which are not always present in medical image data. However, in practice, it performed poorly on our medical image dataset. As a result, we decided to pivot and use SVM and CNN as our remaining two ML algorithms.

SVM

The most crucial step in our SVM implementation was to address imbalances in the dataset and improve performance using minority class augmentation techniques and dimensionality reduction. In the skin lesion dataset, each lesion image is labeled with a diagnosis. Melanocytic nevi (labelled "nv" in the dataset) was the most common diagnosis, accounting for thousands of more samples than any of the other labels and diagnoses. Because of this supermajority, the model, without any preprocessing, would be able to predict "nv" for any new instance and achieve high accuracy without being able to distinguish between the rarer labels such as "mel" for melanoma or "bcc" for basal cell carcinoma. In a healthcare setting, this lack of nuance can have dangerous consequences and result in a misdiagnosis. After encoding the categorical variables in our dataset, splitting our data between testing and training sets, and standardizing the features, we decided to apply SMOTE and PCA to address the class weight imbalances in our dataset.

SMOTE is a technique used in machine learning to improve model performance for predicting and correctly classifying a minority class. Applying SMOTE allows for data augmentation of minority classes by selecting a minority class instance at random and finding its k-nearest minority class neighbors. From this, synthetic instances of the minority class are generated in a manner such that they are relatively close to existing samples in the feature space. PCA is a dimensionality reduction technique used to transform large datasets with many variables into smaller components that retain most of the variance of the original set. We applied PCA to our data, reducing dimensionality from 28 x 28 x 3 to 150 components. This allowed for more efficient performance and protected our model from overfitting.

After addressing the imbalances, we trained our SVM classifier. We decided to use a radial basis function (RBF) kernel, as this function allows the model to address nonlinear decision boundaries within our original data. The pixel values from the skin lesion images are not linearly separable, so utilizing an RBF kernel projects our input data into a higher dimensional space where it is easier to separate. Prior to training our model and separating our data into training and testing sets, we created a dictionary quantifying and encoding the importance of each label. Using this set in our SVM model helps address residual imbalance and penalizes the misclassification of minority classes more harshly.

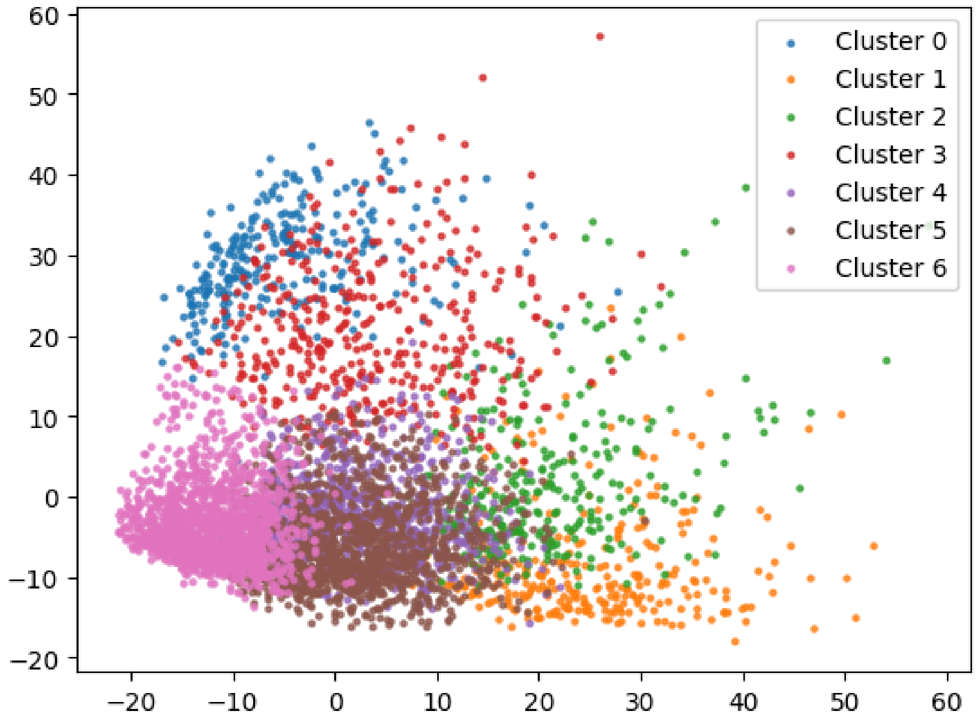
CNN

We chose to use a Convolutional Neural Network (CNN) because CNNs are designed to work with image data and capture spatial patterns like color, texture, and shape—key features, which is critical to our problem of differentiating types of skin lesions. A CNN can learn to recognize low-level features (ie. edges, color gradients) and higher-level features (i.e asymmetry, irregular borders) without needing manual feature extraction- which is a problem given that its difficult to scale and recognize more complex patterns as more images/new skin lesions are added. We found that CNN's have been widely used in medical imaging tasks, and have some of the highest accuracy, thus it made sense for this to be one of the models that we implement.

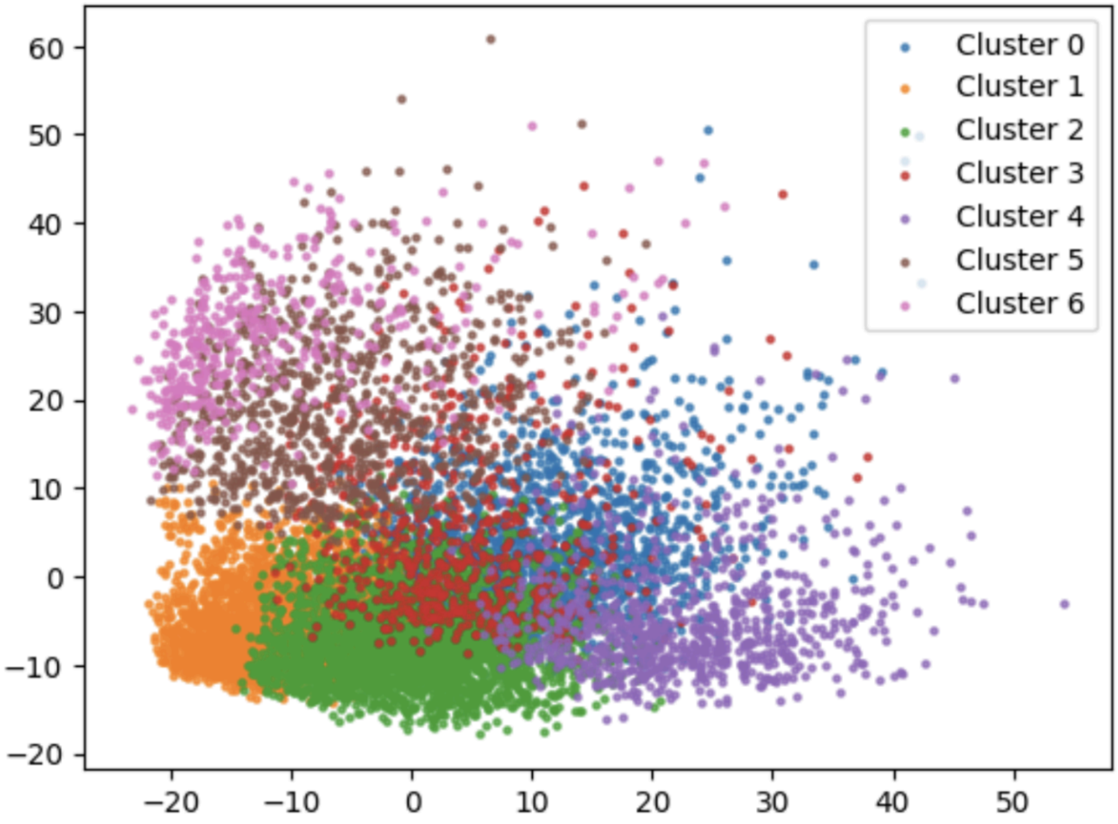
Results and Discussion

KMeans

Visualizations

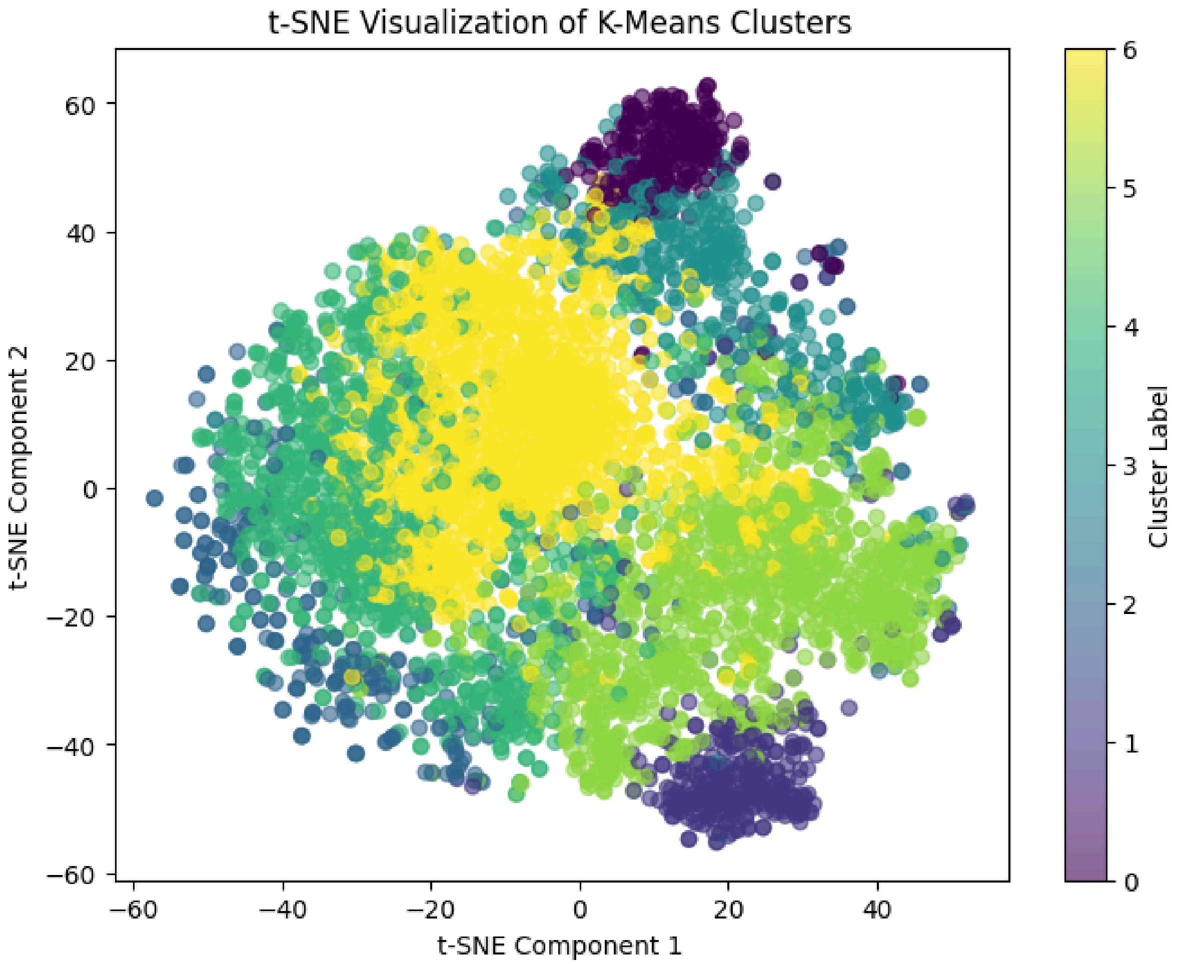


This shows a 2D scatterplot of our kmeans clusters trained on 5,000 images.

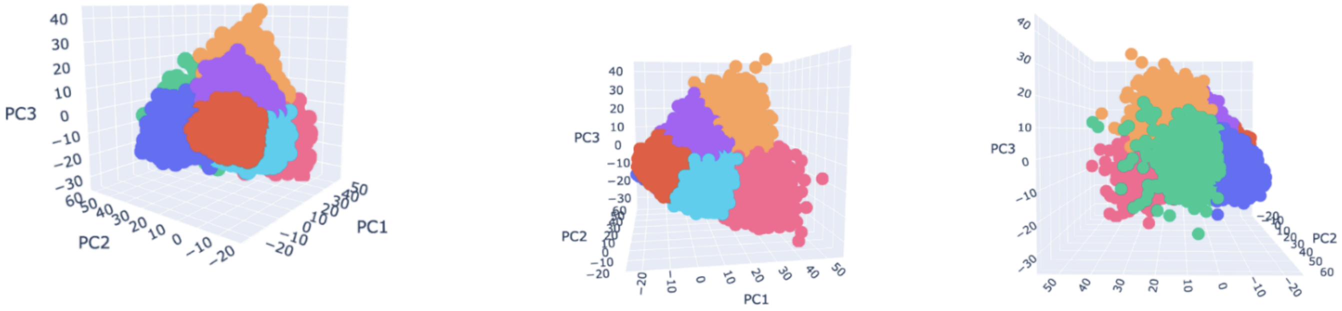


This shows a 2D scatterplot of our kmeans clusters trained on all 10,000 images.

First, we can see that the clusters became less defined with the full dataset of images compared to half the dataset, which suggests that the model struggles to differentiate clusters and it gets confused. In general, we can see that there is a lot of overlap, and the model is performing poorly. Since PCA emphasizes directions with the largest variance, this may not exactly align with the cluster boundaries. In order to see another perspective, we also plotted a t-SNE visualization of the clusters, as t-SNE groups based on points that are similar in higher dimensional space.



However, here we can see that there is also a lot of overlap. This can be due to the fact that all the skin lesion images look relatively similar.



This shows a 3D scatterplot of our clusters, with which there is some more definition. With 3 principal components we can retain more variance which is what helps with separating the clusters.

Quantitative Metrics

Our group utilized several clustering evaluation metrics to better understand our model's clustering performance, including **normalized mutual information, adjusted rand index, silhouette score**, and **Davies-Bouldin index.** We tried to optimize and set our n_components (which regulates variance) to .95 and our clusters to 7 for these metrics.

Normalized mutual information is a metric that measures the shared information between two data sets. In the context of k-means clustering, the NMI represents how closely each cluster label aligns with the true label, which gives us a good sense of how ideal the clustering is. Our NMI is 0.1055372260211378, which correlates to a pretty weak association between the true labels and the cluster labels.

Adjusted rand index simply accounts for the similarity between clusterings while also accounting for randomness by normalizing the Rand Index, making it an improved version of the regular rand index metric. Our ARI is 0.07711995078594112, which indicates clustering that is only slightly above random, which is a pretty weak correlation score.

Silhouette score represents how well formed each cluster is in relation to the other clusters. While NMI and ARI are metrics dedicated to showing how well a model's clusters align with true labels, silhouette score is holistically based on relative data points. More specifically, it measures how tightly packed a cluster's points are to each other and how far the cluster's points are from other clusters. The score we got is 0.016201116, which means that the clusters slightly overlap with one another.

Davies-Bouldin Index is similar to Silhouette score in that it also checks the compactness of the clusters and how separated they are from other clusters. However, this score focuses on the relations between all of the clusters rather than focusing on any individually. Our DBI is 3.3748358740964437, which means that clusters are pretty loose.

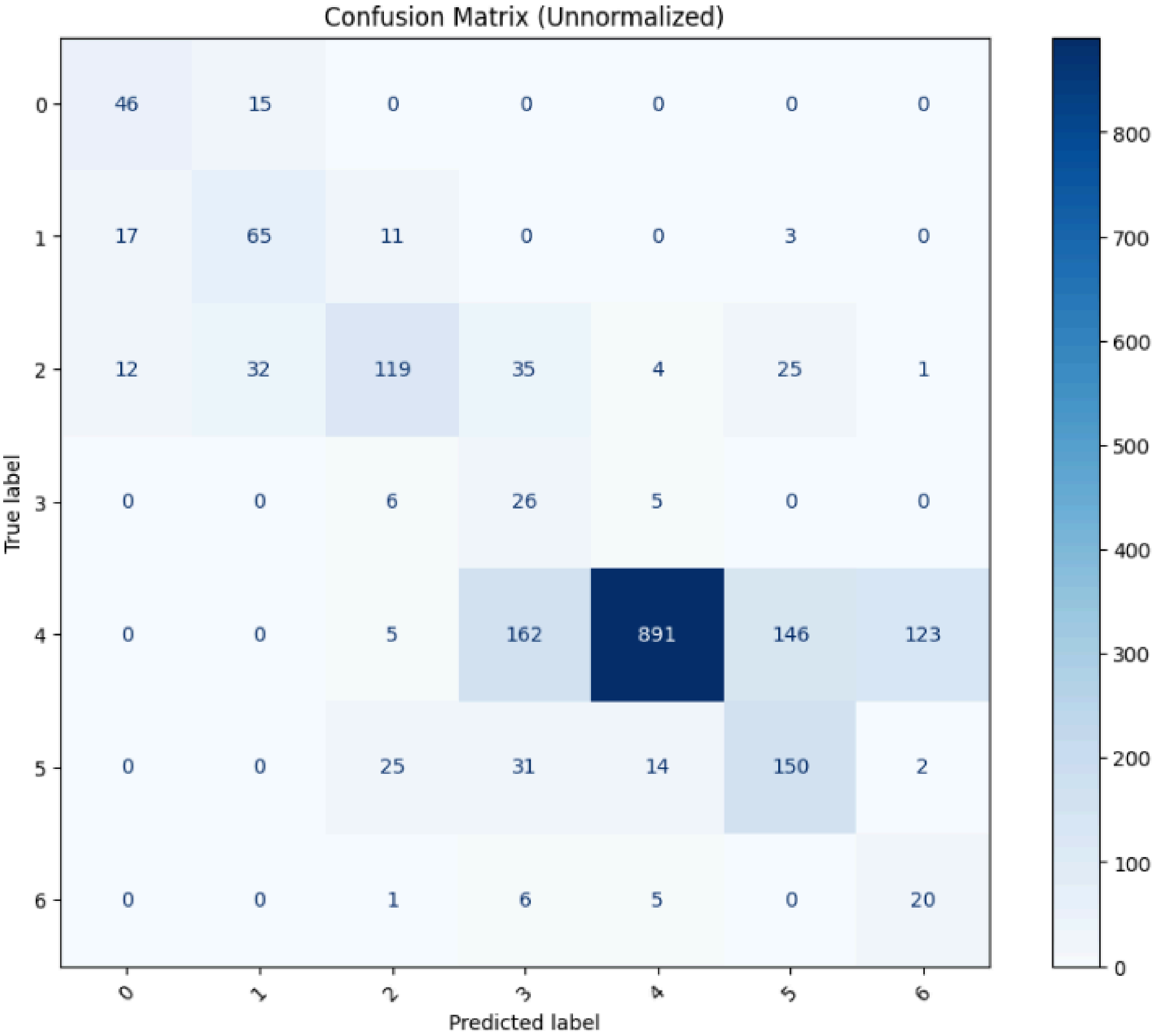
Analysis

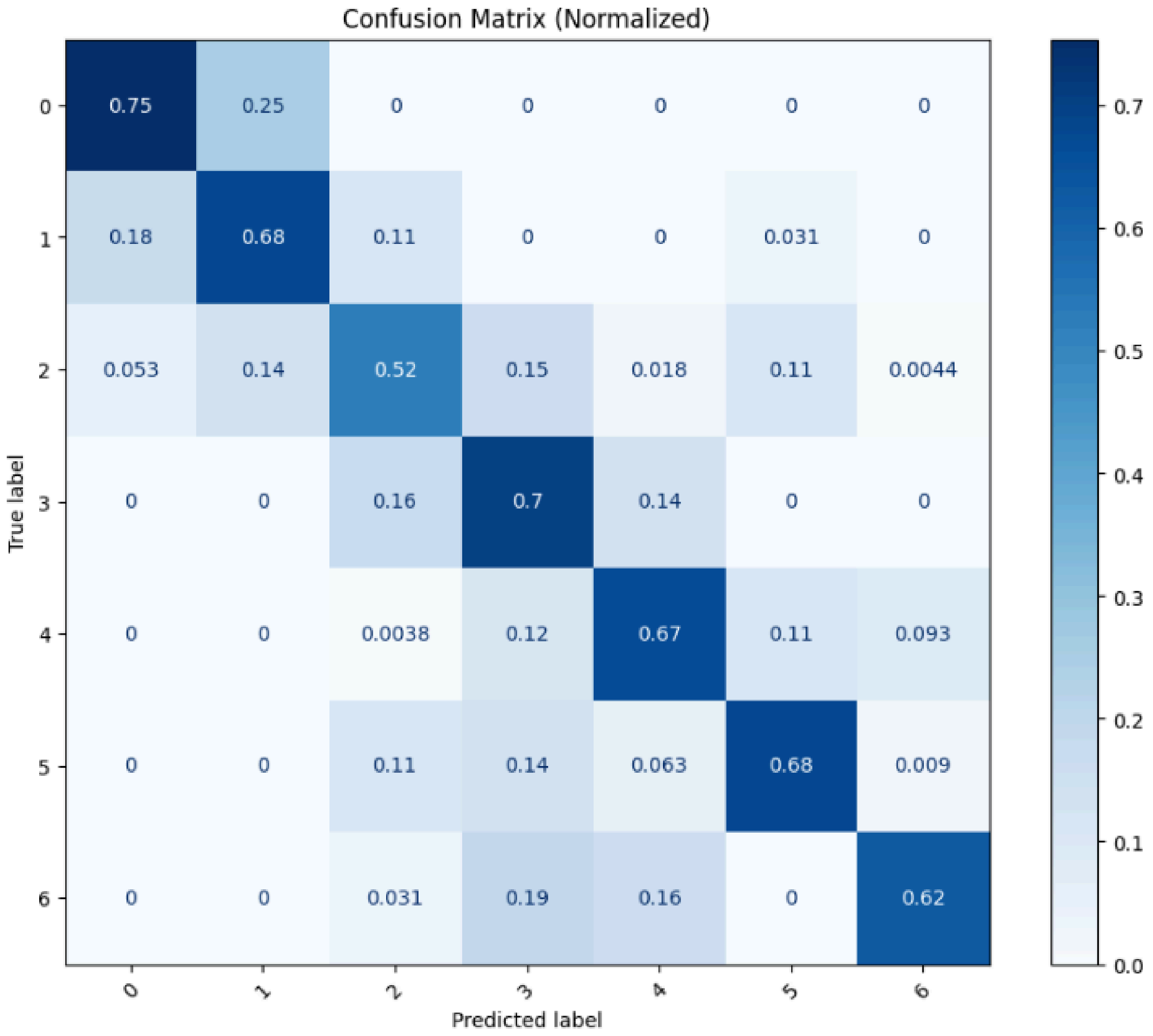
Prior research on challenges with distance-based clustering has shown that clustering on high-dimensional biomedical datasets (such as RGB images) "can yield arbitrary labels and often depends on the trial, leading to varying results" (Thrun, 2021). Two related reasons why KMeans clustering might yield low accuracy on a medical image dataset are high intra-class variability and low inter-class variability. Medical images may not have well-separated clusters in feature space and samples from the same class can look very different (e.g. different patient anatomy or imaging angles). At the same time, different diseases might appear visually similar, making clustering difficult.

High-dimensional image data can also make distance metrics like Euclidean distance (used for KMeans) less meaningful because, as dimensions increase, data points become increasingly sparse and all points tend to appear equidistant from each other. Therefore, clustering algorithms that rely on distance metrics like KMeans can struggle to distinguish between points meaningfully. Lastly, even with KMeans++, initialization may still lead to suboptimal results if the data is complex, and "small perturbations in the input space will lead to diverse clustering results since labels are absent in the unsupervised clustering task" (Thrun, 2021).

SVM

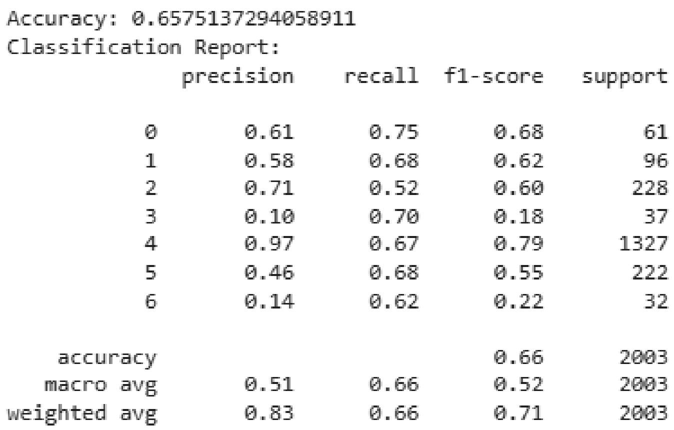
Visualizations


Confusion Matrix (Unnormalized)
True label
Predicted label
0
1
2
3
4
5
6
46 15 0 0 0 0 0
17 65 11 0 0 3 0
12 32 119 35 4 25 1
0 0 6 26 5 0 0
0 0 5 162 891 146 123
0 0 25 31 14 150 2
0 0 1 6 5 0 20
0
100
200
300
400
500
600
700
800



To visualize SVM we utilized a confusion matrix to see how many data points were categorized by their true label. We also included a normalization confusion matrix to visualize how many data points were accurately classified within their own label.

Quantitative Metrics



```
Accuracy: 0.6575137294058911
Classification Report:
              precision    recall  f1-score   support

           0       0.61      0.75      0.68        61
           1       0.58      0.68      0.62        96
           2       0.71      0.52      0.60       228
           3       0.10      0.70      0.18        37
           4       0.97      0.67      0.79      1327
           5       0.46      0.68      0.55       222
           6       0.14      0.62      0.22        32

    accuracy                           0.66      2003
   macro avg       0.51      0.66      0.52      2003
weighted avg       0.83      0.66      0.71      2003
```

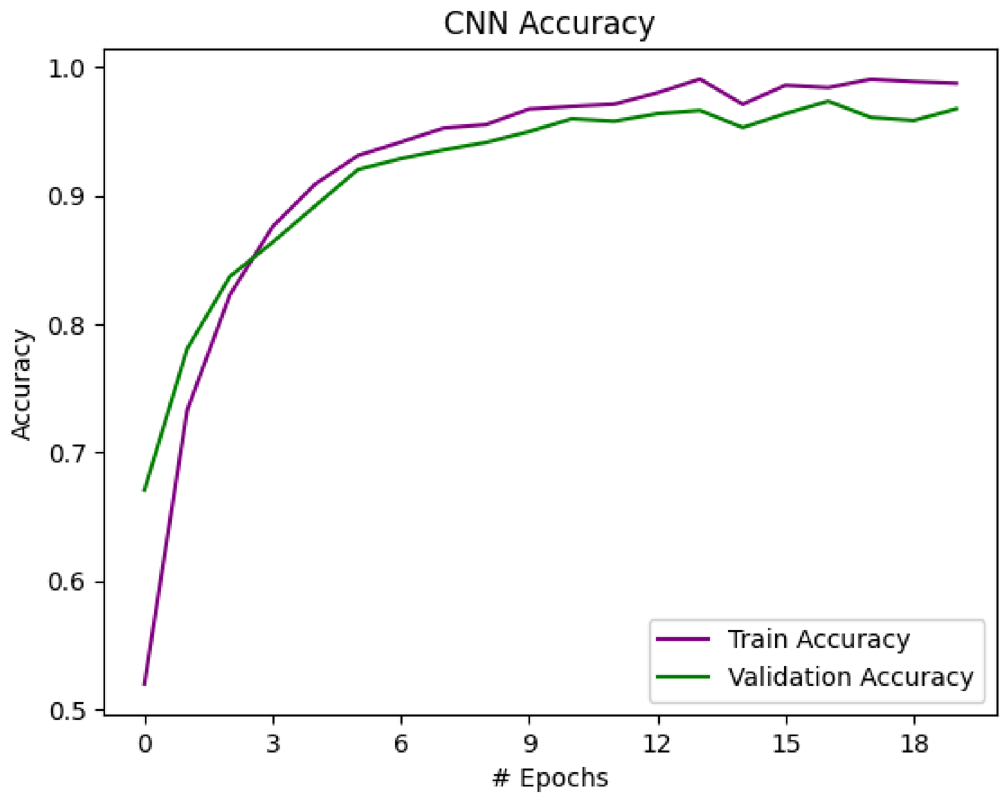
Running our SVM gives us an accuracy of ~65.75%, which is less than ideal but better than what we were seeing in k-means. Looking at the classification report also reveals some other significant trends within our data. The 'nv' labels (shown here as label 4) have a very high precision with 0.97, meaning that are model is very good at predicting 'nv' skin cysts, but given the support it has in comparison to the other classes this may just be a sign of data imbalance. This is further exemplified when looking at the vascular lesion and dermatofibroma labels (labels 3 and 6) which have a very low precision, but much higher recall. Despite the data imbalance, we can see that f1-scores are pretty decent across the board (with the exception of the aforemention labels 3 and 6), meaning that despite an overwhelming number of 'nv' data points SVM was able to decently classify all of the different skin cysts.

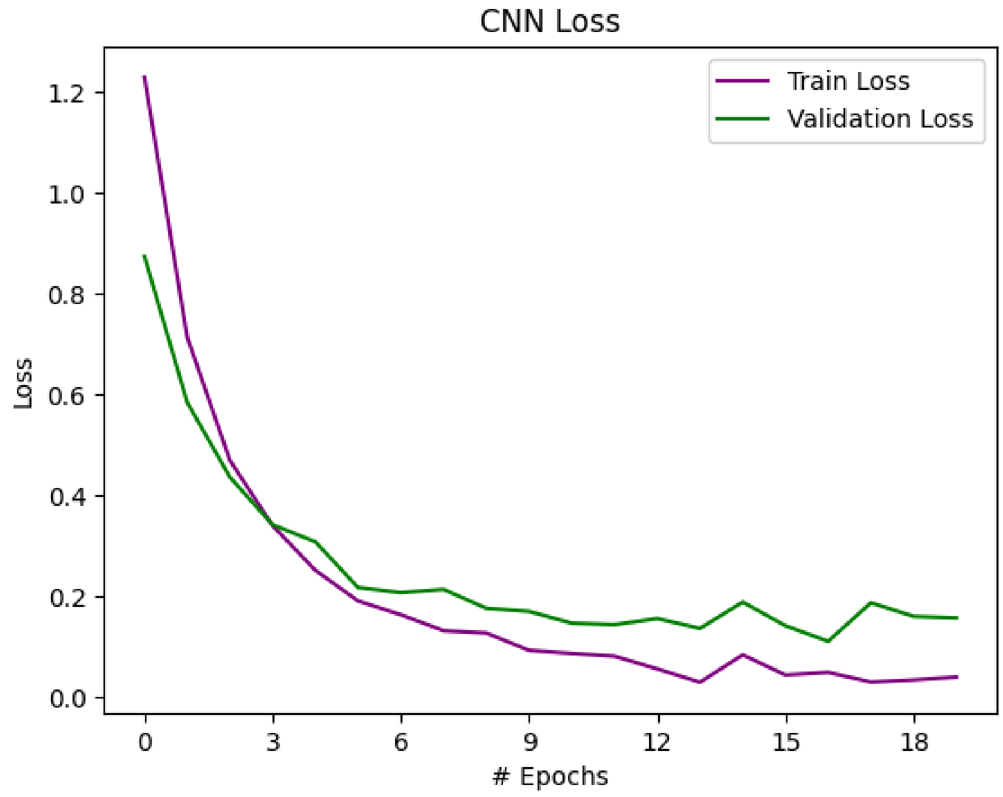
Analysis

SVM algorithms are widely used in the international medical community to solve classification and regression problems with biological data, particularly in determining diagnosis and prognosis conditions. This popularity is due to the model’s ability to overcome nonlinearity and high dimensionality, features common in medical records and data (Zhou et al, 2022). In instances where data is not linearly separable, SVM maps the inputs into higher-dimensional feature spaces using kernel functions. The radial bias function (RBF) kernel is the most widely used in SVM implementation because it yields superior accuracy in situations where the data is highly nonlinear and there is no prior knowledge of the data’s structure. This data imbalance is especially common in healthcare data, as there are often large numbers of patients with very few exhibiting rare diseases. Misclassification of minority classes has detrimental effects in healthcare, specifically in instances where positive cases of disease are misdiagnosed. SVM can be sensitive to imbalanced data, however results can be improved and validated using data preprocessing methods such as SMOTE, random oversampling, or random oversampling (Guido et al, 2024).

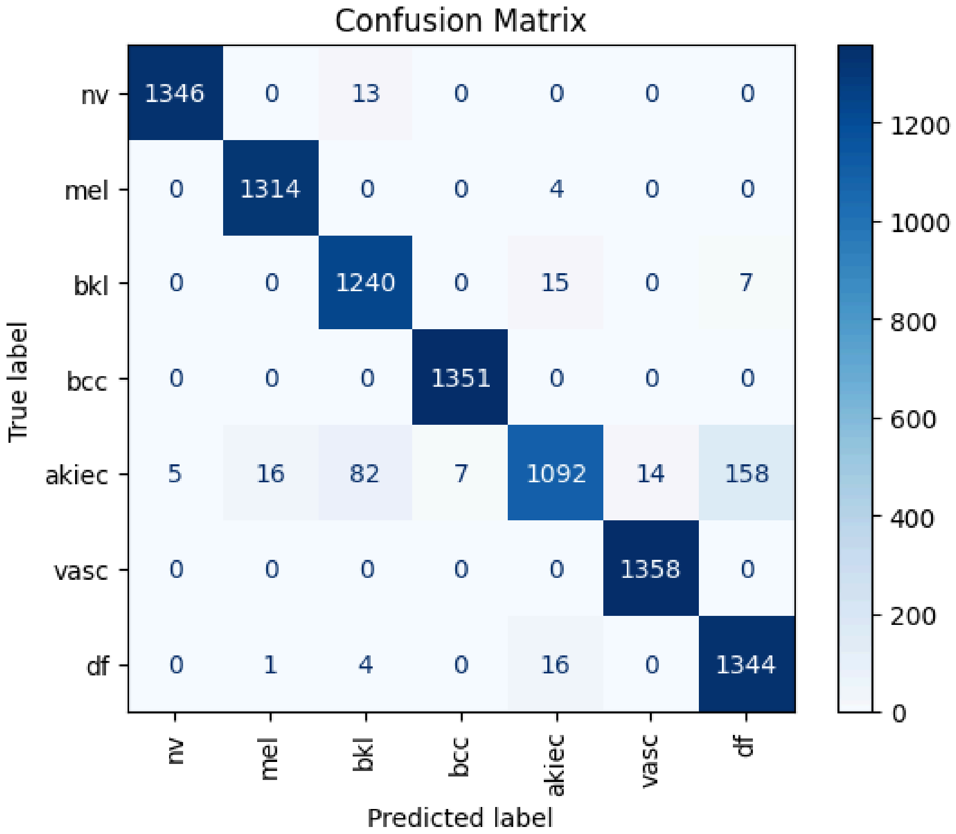
CNN

Quantitative Metrics & Visualizations



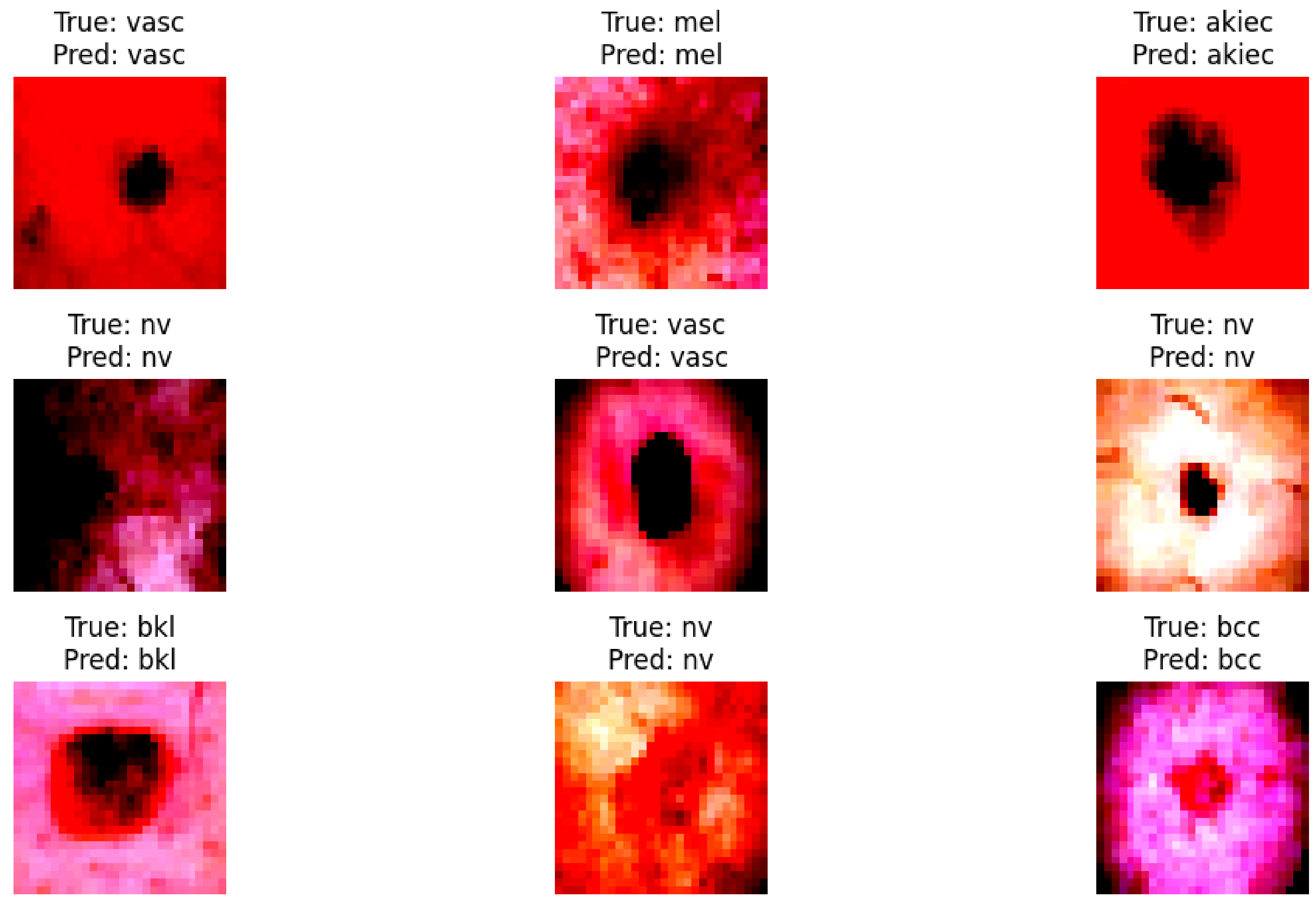


The first plot shows the accuracy over epochs reaches around 98%, showing that the model learned very well and doesn't seem to be overfitting much (training and validation lines stay close). And for the Loss plot, it goes down quickly and stabilizes after about 5-6 epochs, indicating good learning behavior and performance in classifying the skin lesions.



	nv	mel	bkl	bcc	akiec	vasc	df
nv	1346	0	13	0	0	0	0
mel	0	1314	0	0	4	0	0
bkl	0	0	1240	0	15	0	7
bcc	0	0	0	1351	0	0	0
akiec	5	16	82	7	1092	14	158
vasc	0	0	0	0	0	1358	0
df	0	1	4	0	16	0	1344

This confusion matrix shows that most predictions are on the diagonal, with the highest concentrations of images being classified correctly.



Here we can see a visualization of the skin lesions, specifically how the model learned the visual patterns for all the different skin lesions. All examples shown here are correctly labeled (true labels match predicted labels), indicating that the model was successful.

Analysis

Some prior research on medical image classification by CNN has “achieved performances rivaling human experts” (Yadav et al, 2019). For example, when CheXNet, a CNN with 121 layers, was trained on a dataset with 100,000+ chest X-ray images, it achieved “a better performance than the average performance of four radiologists” (Yadav et al, 2019). There are several reasons why CNN is an especially useful machine learning method for medical image datasets, including its ability to capture spatial hierarchies, automatic feature extraction, and translation invariance.

Convolution layers in CNNs form a hierarchical representation of the input image, where low-level features are captured in the initial layers and high-level features (complex layers and textures) are learned in the deeper layers. This hierarchical representation “enables CNNs to learn complex patterns in medical images, making them capable of differentiating between normal and abnormal findings” (Singh et al, 2023). Unlike other machine learning models including SVM, kNN, Decision Trees, and Random Forest, CNNs automatically learn hierarchical feature representations directly from raw image data. This automatic feature extraction is particularly beneficial in medical imaging, where subtle differences in images can be critical points for accurate diagnosis. Lastly, CNNs are often beneficial for their ability to achieve translation invariance, meaning they can recognize patterns regardless of their location in an image. This stems from two primary architectural features:

convolutional layers and pooling layers. Convolutional layers apply the same filters across different regions of the input image, enabling the network to detect specific features (e.g. edges or textures) regardless of their position. Pooling operations help reduce the spatial dimensions of the image, summarizing the presence of features in a region without emphasizing their exact locations.

Given these advantages, our CNN implementation achieved high accuracy on the medical image classification task. The model's ability to automatically extract relevant features and generalize across spatial variations contributed to its strong performance, reinforcing the suitability of CNNs for complex image-based diagnostic applications.

Comparison

Our project implemented both supervised and unsupervised learning, with SVM and CNN being supervised methods and KMeans being an unsupervised method. KMeans requires manual feature extraction and cannot work as effectively with raw image pixels. Compared to CNN and SVM, it lacks the capacity to model complex image patterns, which is evident in the less than satisfactory quantitative metric values (NMI, ARI, Silhouette Score, and Davies-Bouldin Index). Although SVM also relies on manually engineered features (e.g. pixel intensities, shape descriptors), it performs better than kMeans on structured feature datasets because it benefits from being a supervised model and knowing ground truth labels. However, it still cannot match the automatic hierarchical learning of CNNs as they perform automatic feature extraction directly from raw images and can learn low-level to high-level features across layers. Therefore, CNN offers a clear advantage over both kMeans and SVM in handling complex medical images.

With regards to scalability and computational cost, CNN requires high computational resources (GPU, TPU) and larger training time but scales extremely well and benefits from large image datasets. KMeans has the lowest computational cost of the three methods and scales well to large datasets, but it is limited in predictive power due to the fact that it blindly clusters data based on distance to a centroid without any access to the true labels. In the middle of the two, SVM is computationally expensive for non-linear kernels (better for image data) and large datasets but is well suited for smaller datasets with high-quality features.

While KMeans and SVM offer value under specific circumstances and constraints, CNNs stand out due to their ability to automatically extract meaningful features and handle complex image patterns with high accuracy. Their high accuracy and superior performance in both scalability and feature learning make them the most effective model for medical image classification in our project.

Next Steps

Due to the high accuracy of our CNN implementation and understanding of the limitations of KMeans when it comes to image datasets, our next steps primarily focus on tuning hyper-parameters within SVM. SVM has been shown to be one of the ”best machine learning models for solving several real-life classification problems” (Guido et al, 2023); however, this claim does not seem to be well supported by our data. Therefore, future work on this dataset could include:

1. **Better preprocessing and feature extraction**: Complete a more disciplined approach to preprocessing & feature extraction, as several papers have shown that SVM performance depends heavily on having well-engineered, low-redundancy feature vectors before model tuning begins (Guido et al, 2023 & Liu, 2021).

2. **Using an imbalance aware, cost-sensitive SVM**: Applying class weighting or cost-sensitive variant to keep minority-class errors from being swamped by majority examples, which we indicated as one of our goals in the problem definition. Guido et al. demonstrated that pairing cost weights with smarter metrics such as G-Mean gives more reliable results.
3. **Implementing a hybrid model**: It may be possible to achieve higher accuracy by combining “multivariate empirical modal decomposition (MEMD) and adaptive differential evolution (ADE) algorithms with a support vector machine” (Zulfiqar et al, 2022). MEMD simultaneously decomposes multichannel data and allows for effective extraction of unique information over different time frequencies to ensure high computational efficiency. The ADE algorithm is beneficial for finding the optimal values of hyperparameters of SVM for improved accuracy.

By using more disciplined feature engineering, adopting cost-sensitive training, and exploring optimizations with MEMD and ADE, we can improve the accuracy of SVM.

References

An Q, Rahman S, Zhou J, Kang JJ. A Comprehensive Review on Machine Learning in Healthcare Industry: Classification, Restrictions, Opportunities and Challenges. Sensors (Basel). 2023 Apr 22;23(9):4178. doi: 10.3390/s23094178. PMID: 37177382; PMCID: PMC10180678.

E. Jana, R. Subban, and S. Saraswathi (2018, November 8). Research on Skin Cancer Cell Detection Using Image Processing. February 20, 2025, https://ieeexplore.ieee.org/document/8524554/

Guido, R., Groccia, M.C. & Conforti, D. A hyper-parameter tuning approach for cost-sensitive support vector machine classifiers. Soft Comput 27, 12863–12881 (2023). https://doi.org/10.1007/s00500-022-06768-8

Liu, H. (2021). An Evaluation of Hyperparameter Tuning Methods in SVM (Undergraduate honors thesis, University of California San Diego, Department of Mathematics)

Maruyama T, Hayashi N, Sato Y, Hyuga S, Wakayama Y, Watanabe H, Ogura A, Ogura T. Comparison of medical image classification accuracy among three machine learning methods. J Xray Sci Technol. 2018;26(6):885-893. doi: 10.3233/XST-18386. PMID: 30223423.

Rao, S. (2024, August 20). Deep Learning Illustrated, Part 3: Convolutional Neural Networks. February 20, 2025, https://towardsdatascience.com/implementing-convolutional-neural-networks-in-tensorflow-bc1c4f00bd34/

Singh, Y., Farrelly, C., Hathaway, Q. A., Choudhary, A., Carlsson, G., Erickson, B., & Leiner, T. (2023). The role of geometry in convolutional neural networks for medical imaging. Mayo Clinic Proceedings: Digital Health, 1(4), 519–526. https://doi.org/10.1016/j.mcpdig.2023.08.006OUCI+1PMC+1

Talo M, Yildirim O, Baloglu UB, Aydin G, Acharya UR. Convolutional neural networks for multi-class brain disease detection using MRI images. Comput Med Imaging Graph. 2019;78:101673.

Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi:10.1038/sdata.2018.161 (2018).

Yadav, S.S., Jadhav, S.M. Deep convolutional neural network based medical image classification for disease diagnosis. J Big Data 6, 113 (2019). https://doi.org/10.1186/s40537-019-0276-2

Zulfiqar, M., Kamran, M., Rasheed, M. B., Alquathami, T., & Milyani, A. H. (2022). Hyperparameter optimization of support vector machine using adaptive differential evolution and multivariate empirical mode decomposition for load forecasting in smart grid [Preprint]. SSRN. https://doi.org/10.2139/ssrn.4115269

Gantt Chart

Team 36's Gantt chart is linked here.

Contribution Table

Name	Final Report Contributions
Cas Copeland	CNN Implementation & Visualizations
Sadhana Kumar	CNN analysis, Comparison, Next Steps
Lydia Lazor	CNN Implementation & Writeup
Daniel Noelle	SVN Implementation & Analysis
Amara Rangwala	SVM Implementation & Writeup

CS 4641 Project Midterm Report - Team 36

Skin Lesion Detection Model

Introduction

Skin cancer is one of the most common forms of cancer worldwide, and early detection is crucial for effective treatment. Traditional diagnostic methods rely on clinical examination and biopsy, which can be time-consuming, resource-intensive, and subject to variability by dermatologists. Recent advancements in machine learning, particularly deep learning, have shown significant promise in the automated classification of skin lesions. Thus our project aims to develop an ML model capable of predicting the diagnosis of a skin lesion from dermatoscopic images and patient metadata. We will be using the HAM10000 dataset, available on Kaggle (link to dataset); This dataset includes 10,015 dermatoscopic images of pigmented skin lesions, annotated with corresponding data such as the patient's sex, age, and the localization of the lesion on the body. The dataset categorizes seven types of common pigmented skin lesions: melanocytic nevi, melanoma, benign keratosis-like lesions, basal cell carcinoma, actinic keratoses, vascular lesions, and dermatofibroma (Tschandl et al., 2018).

Problem Definition

We propose to develop a model that predicts the type of skin lesion using the proposed dataset. Our approach will preprocess the image data, and apply a convolutional neural network (CNN) for classification, with additional layers that incorporate patient metadata (sex, age, and lesion localization). While existing literature has demonstrated the effectiveness of CNNs in classifying dermatoscopic images, many studies focus solely on image data without considering patient-specific metadata. By combining both images and patient data, our model aims to improve classification accuracy over these existing methods.

Methods

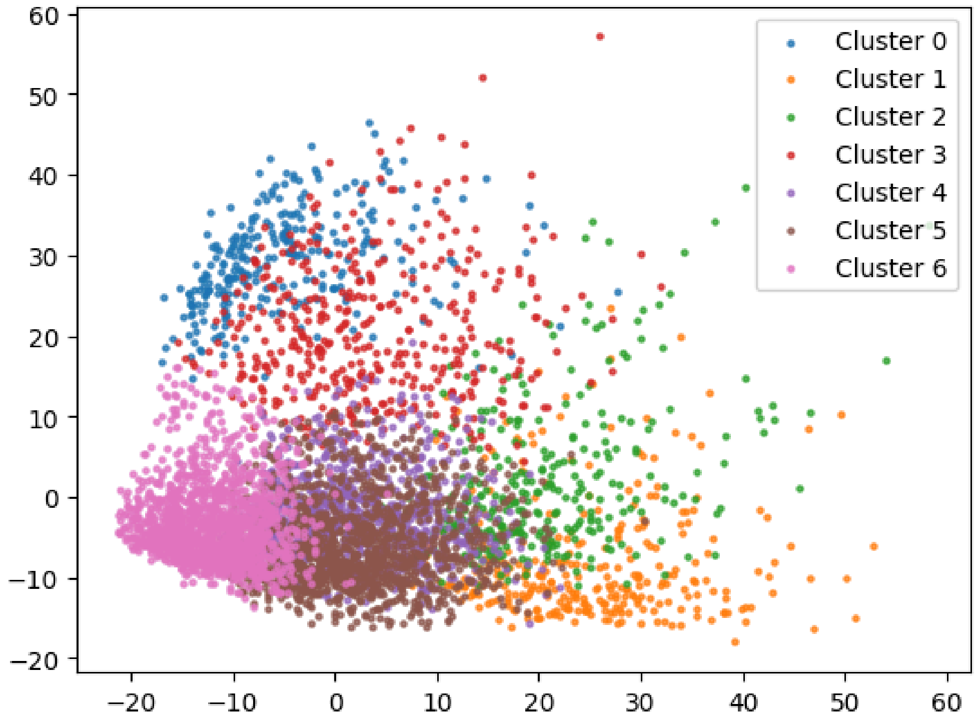
For data processing, our group used a pre-trained deep learning model, ResNet-50, as a feature extractor, allowing our implemented ML models to run smoothly and efficiently. According to multiple healthcare professionals with expertise in using deep learning algorithms for disease detection and prevention, using a ResNet50 model yielded the greatest classification accuracy in comparison to other common pretrained models such as AlexNet, VGG-16, ResNet-18, and ResNet-34. According to a study completed by Muhammad Talo, Ozal Yildirim, Ulas Baran Baloglu, et. al. on brain disease detection and classification using MRI images, ResNet-50 obtained a classification accuracy of 95.23% ± 0.6%. Their team attributed the success of ResNet-50 to its “modern architecture” and its ability to transform high dimensional images that are very noisy into a 2048-dimensional feature vector that can capture high-level visual patterns. Because of its success in a case study regarding medical image classification, our group decided to implement this model as a feature extractor to transform our raw skin lesion images into a format that is more digestible for our machine learning algorithms.

In our data-preprocessing code, we iterated through each image, recoloring and resizing the image to meet the criteria for passing it through the ResNet-50 model (a 224 x 224 RGB image). From there, we converted the image to an array, added a dimension so it became a 4D vector to align with ResNet’s required input parameters, and applied the ResNet-50 specific normalization. We then passed the preprocessed image through the ResNet model which outputs a 2048-dimensional feature vector that we flattened to a 1D vector with the dimension (2048,). Lastly, we appended the extracted feature vector to a list for use in our machine learning algorithms.

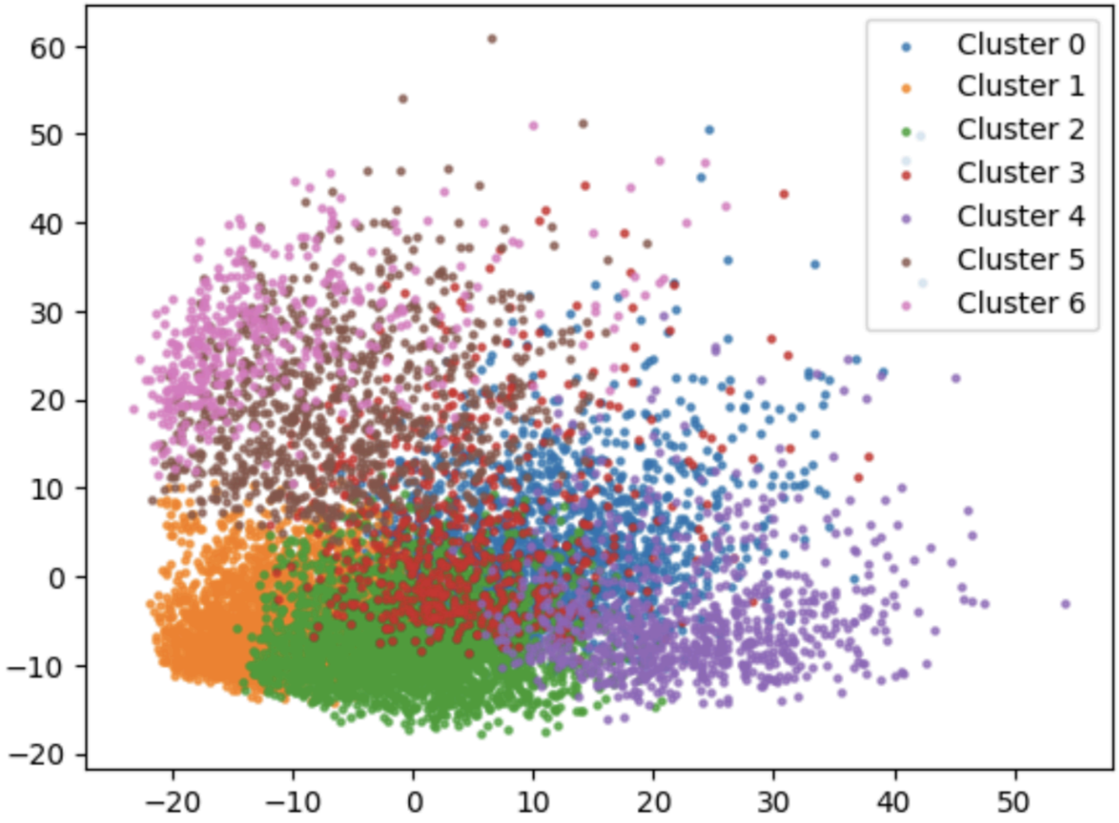
We chose KMeans as our ML algorithm to explore an unsupervised learning approach to classification, and to further understand our data- if it has any natural groupings, and to see how many visually similar images exist. This can give us an interesting comparison and contrast to when we use a supervised learning approach. KMeans was more promising compared to other unsupervised learning approaches as we already know the number of clusters. Since our dataset consists of 7 different types of skin lesion diagnosis, we experimented with around 7 clusters.

Results and Discussion

Visualizations

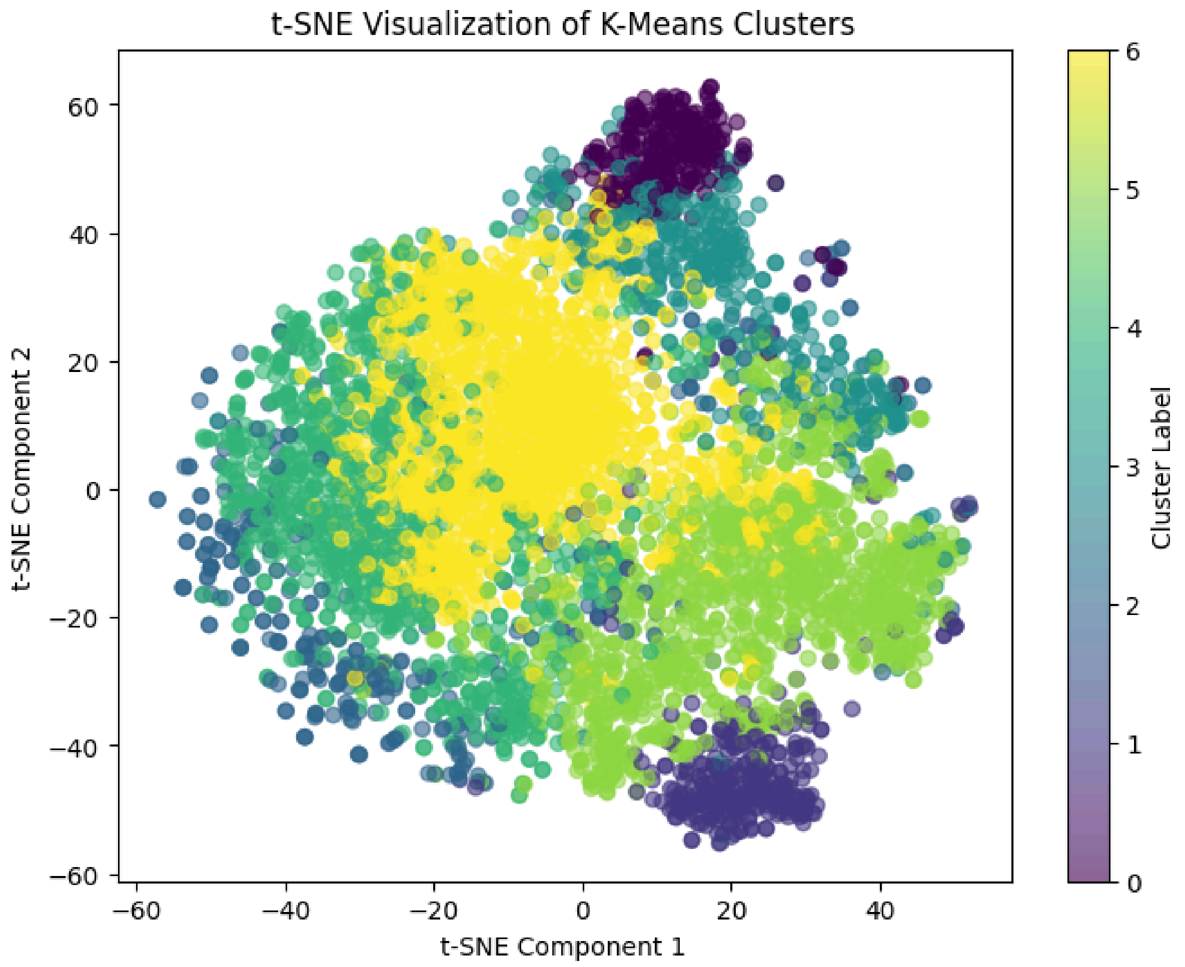


This shows a 2D scatterplot of our kmeans clusters trained on 5,000 images.

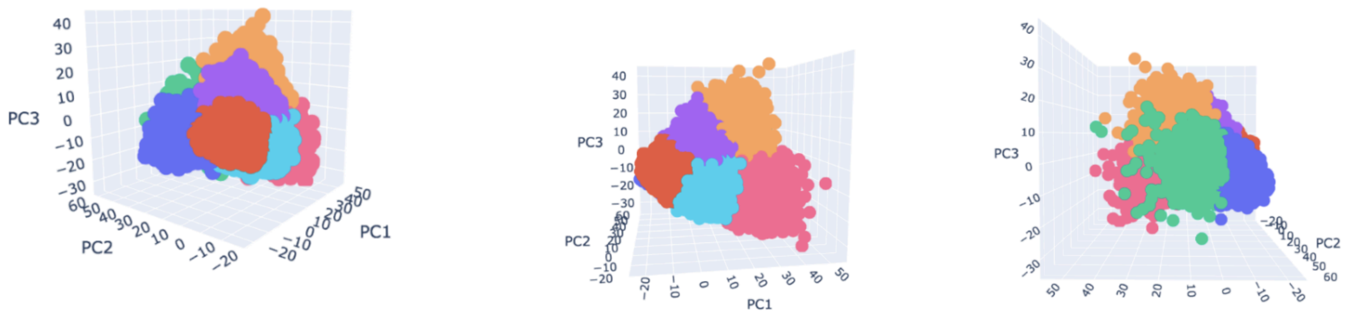


This shows a 2D scatterplot of our kmeans clusters trained on all 10,000 images.

First, we can see that the clusters became less defined with the full dataset of images compared to half the dataset, which suggests that the model struggles to differentiate clusters and it gets confused. In general, we can see that there is a lot of overlap, and the model is performing poorly. Since PCA emphasizes directions with the largest variance, this may not exactly align with the cluster boundaries. In order to see another perspective, we also plotted a t-SNE visualization of the clusters, as t-SNE groups based on points that are similar in higher dimensional space.



However, here we can see that there is also a lot of overlap. This can be due to the fact that all the skin lesion images look relatively similar.



This shows a 3D scatterplot of our clusters, with which there is some more definition. With 3 principal components we can retain more variance which is what helps with separating the clusters.

Quantitative Metrics

Our group utilized several clustering evaluation metrics to better understand our model's clustering performance, including **normalized mutual information, adjusted rand index, silhouette score**, and **Davies-**

Bouldin index. We tried to optimize and set our n_components (which regulates variance) to .95 and our clusters to 7 for these metrics.

Normalized mutual information is a metric that measures the shared information between two data sets. In the context of k-means clustering, the NMI represents how closely each cluster label aligns with the true label, which gives us a good sense of how ideal the clustering is. Our NMI is 0.1055372260211378, which correlates to a pretty weak association between the true labels and the cluster labels.

Adjusted rand index simply accounts for the similarity between clusterings while also accounting for randomness by normalizing the Rand Index, making it an improved version of the regular rand index metric. Our ARI is 0.07711995078594112, which indicates clustering that is only slightly above random, which is a pretty weak correlation score.

Silhouette score represents how well formed each cluster is in relation to the other clusters. While NMI and ARI are metrics dedicated to showing how well a model's clusters align with true labels, silhouette score is holistically based on relative data points. More specifically, it measures how tightly packed a cluster's points are to each other and how far the cluster's points are from other clusters. The score we got is 0.016201116, which means that the clusters slightly overlap with one another.

Davies-Bouldin Index is similar to Silhouette score in that it also checks the compactness of the clusters and how separated they are from other clusters. However, this score focuses on the relations between all of the clusters rather than focusing on any individually. Our DBI is 3.3748358740964437, which means that clusters are pretty loose.

Analysis of KMeans

Prior research on challenges with distance-based clustering has shown that clustering on high-dimensional biomedical datasets (such as RGB images) "can yield arbitrary labels and often depends on the trial, leading to varying results" (Thrun, 2021). Two related reasons why KMeans clustering might yield low accuracy on a medical image dataset are high intra-class variability and low inter-class variability. Medical images may not have well-separated clusters in feature space and samples from the same class can look very different (e.g. different patient anatomy or imaging angles). At the same time, different diseases might appear visually similar, making clustering difficult.

High-dimensional image data can also make distance metrics like Euclidean distance (used for KMeans) less meaningful because, as dimensions increase, data points become increasingly sparse and all points tend to appear equidistant from each other. Therefore, clustering algorithms that rely on distance metrics like KMeans can struggle to distinguish between points meaningfully. Lastly, even with KMeans++, initialization may still lead to suboptimal results if the data is complex, and "small perturbations in the input space will lead to diverse clustering results since labels are absent in the unsupervised clustering task" (Thrun, 2021).

Next Steps

As outlined in our project proposal, we initially planned to implement DBSCAN alongside KMeans for comparison, as DBSCAN can detect clusters of arbitrary shape and handle noise, whereas KMeans has high noise sensitivity and assumes circular clusters which are not always present in medical image data. However, in practice, it performed poorly on our medical image dataset. As a result, we decided to pivot and use SVM and CNN as our remaining two ML algorithms.

We chose SVM because it is a supervised machine learning algorithm that has been described as “one of the most effective machine learning algorithms for pattern recognition” (An et al., 2023). Because it is a supervised ML algorithm, it will likely mitigate the sensitivity present in KMeans. CNN was selected because it has been shown to be “more accurate than conventional machine learning methods that utilize the manual feature extraction”, such as KMeans and DBSCAN (Maruyama et al., 2018). Our next steps include implementing these models and comparing their accuracy to that of KMeans.

References

An Q, Rahman S, Zhou J, Kang JJ. A Comprehensive Review on Machine Learning in Healthcare Industry: Classification, Restrictions, Opportunities and Challenges. Sensors (Basel). 2023 Apr 22;23(9):4178. doi: 10.3390/s23094178. PMID: 37177382; PMCID: PMC10180678.

Maruyama T, Hayashi N, Sato Y, Hyuga S, Wakayama Y, Watanabe H, Ogura A, Ogura T. Comparison of medical image classification accuracy among three machine learning methods. J Xray Sci Technol. 2018;26(6):885-893. doi: 10.3233/XST-18386. PMID: 30223423.

Rao, S. (2024, August 20). Deep Learning Illustrated, Part 3: Convolutional Neural Networks. February 20, 2025, https://towardsdatascience.com/implementing-convolutional-neural-networks-in-tensorflow-bc1c4f00bd34/

Talo M, Yildirim O, Baloglu UB, Aydin G, Acharya UR. Convolutional neural networks for multi-class brain disease detection using MRI images. Comput Med Imaging Graph. 2019;78:101673.

Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi:10.1038/sdata.2018.161 (2018).

Gantt Chart

Team 36's Gantt chart is linked here.

Contribution Table

Name	Midterm Report Contributions
Cas Copeland	Code & Visualizations
Sadhana Kumar	Analysis & Next Steps
Lydia Lazor	Visualizations & Methods
Daniel Noelle	Quantitative Metrics
Amara Rangwala	Methods Analysis

CS 4641 Project Proposal - Team 36

Skin Lesion Detection Model

Introduction

Skin cancer is one of the most common forms of cancer worldwide, and early detection is crucial for effective treatment. Traditional diagnostic methods rely on clinical examination and biopsy, which can be time-consuming, resource-intensive, and subject to variability by dermatologists. Recent advancements in machine learning, particularly deep learning, have shown significant promise in the automated classification of skin lesions. Thus our project aims to develop an ML model capable of predicting the diagnosis of a skin lesion from dermatoscopic images and patient metadata. We will be using the HAM10000 dataset, available on Kaggle (link to dataset); This dataset includes 10,015 dermatoscopic images of pigmented skin lesions, annotated with corresponding data such as the patient's sex, age, and the localization of the lesion on the body. The dataset categorizes seven types of common pigmented skin lesions: melanocytic nevi, melanoma, benign keratosis-like lesions, basal cell carcinoma, actinic keratoses, vascular lesions, and dermatofibroma (Tschandl et al., 2018).

Problem Definition

We propose to develop a model that predicts the type of skin lesion using the proposed dataset. Our approach will preprocess the image data, and apply a convolutional neural network (CNN) for classification, with additional layers that incorporate patient metadata (sex, age, and lesion localization). While existing literature has demonstrated the effectiveness of CNNs in classifying dermatoscopic images, many studies focus solely on image data without considering patient-specific metadata. By combining both images and patient data, our model aims to improve classification accuracy over these existing methods.

Methods

Our preprocessing methods include: Data cleaning, dimensionality reduction, and sampling data.

Dimensionality reduction aims to reduce the dimensionality of image data. By reducing image dimensionality, we can expect to experience more efficient computation time and potential clustering performance improvement. Data sampling is a method to reduce the size of the dataset by eliminating specific data points that minimally impact performance or accuracy. Since computation is a potential bottleneck, allowing quicker iteration without losing impactful diversification of lesions is crucial. Data transformation, specifically pixel value normalizations and augmentations (ie: flips and rotations), will ensure that the CNN model identifies consistent inputs while simultaneously increasing data variability for generalization.

Our machine learning algorithms include CNNs, K-Means, and DBSCAN:

"Convolutional Neural Networks (CNNs) are specialized models designed for image recognition tasks" (Rao, 2024). In other words, CNNs are great at extracting high level features from raw pixel information, allowing images to accurately be understood and categorized. K-Means showcases how the data naturally groups into clusters, revealing underlying patterns (valuable for demographic information) or mislabeled samples. DBSCAN detects outliers, a critical consideration in the medical field where atypical cases may need special attention/further investigation.

From these models and methods, we will be able to create a quality, effective, and necessary unsupervised and supervised machine learning pipeline for the realm of dermatology.

(Potential) Results and Discussion

To evaluate model performance, we will use:

- **ROC-AUC Score:** Measures the model's ability to distinguish between malignant and benign lesions, crucial for prioritizing high-risk cases.
- **Recall:** Ensures positive cases of malignant skin cancer are accurately detected, reducing the risk of missed diagnoses.
- **Precision:** Ensures flagged malignant cases are truly malignant, preventing an excessive number of false positives.

"When evaluating performance of SVM with Quadratic, Polynomial, and Gaussian RBF kernels...it is clear the best achieved classification accuracy (85.3%) is with polynomial kernel in SVM" (Jana et. al, 2018). Given research supporting SVM's effectiveness in melanoma detection, we aim to compare SCM with CNNs to determine which is better suited for skin cancer classification. We expect them to outperform SVMs in both recall and precision along with better accuracy for image-only classifiers.

References

Rao, S. (2024, August 20). Deep Learning Illustrated, Part 3: Convolutional Neural Networks. February 20, 2025, https://towardsdatascience.com/implementing-convolutional-neural-networks-in-tensorflow-bc1c4f00bd34/

Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi:10.1038/sdata.2018.161 (2018).

E. Jana, R. Subban, and S. Saraswathi (2018, November 8). Research on Skin Cancer Cell Detection Using Image Processing. February 20, 2025, https://ieeexplore.ieee.org/document/8524554/

Gantt Chart

Team 36's Gantt chart is linked here.

Contribution Table

Name	Proposal Contributions
Cas Copeland	Methods and GANTT Chart
Sadhana Kumar	GitHub Pages and dataset
Lydia Lazor	Intro and Problem Def
Daniel Noelle	Results and Discussion
Amara Rangwala	Lit Review & References