APS360 PROJECT PROGRESS REPORT

Gauri Karajagi
Student# 1008949998
gauri.karajagi@mail.utoronto.ca

Myuri Thayalan
Student# 1009437600
myuri.thayalan@mail.utoronto.ca

Aneeqa Maham
Student# 1010087709
aneeqa.maham@mail.utoronto.ca

Deena Abrari
Student# 1010211557
d.abrari@mail.utoronto.ca

ABSTRACT

This project aims to develop a deep learning model capable of taking in dermatoscopic images and metadata and clasifying cases as benign or malignant. This is a sensitive field, and benefits greatly from accuracy and early detection. Our setup uses the HAM10000 dataset for training, creating a binary classification model meant to support medical practitioners' decision making.

So far, we have implemented a baseline SVM model and developed a primary model using transfer learning with ResNet50. The best ResNet50 model developed achieves 88.9% validation accuracy, with strong performance on benign cases. The group is working to improve malignant detection through class weighting and the planned integration of patient metadata using a multi-input CNN-MLP architecture. This report summarizes our progress, challenges, and next steps the group intends to take. —-Total Pages: 8

1 BRIEF PROJECT DESCRIPTION

The prevalence of skin cancer is well-documented, with approximately 132,000 cases of melanoma and 2-3 million cases of nonmelanoma skin cancer annually (Munjal et al. (2024)). Survival rate is highly correlated with identification, making early diagnosis vital for successful treatment. However, malignant lesions often go undetected or incorrectly identified due to the subjectivity of individual dermatologists' judgment (Munjal et al. (2024)).

Deep learning models are a promising solution to some of these challenges. Their ability to assess large samples of data allow them to recognize patterns that the human eye might not. Existing models have been shown to have over 20% greater accuracy in identifying malignant skin cancer, compared to the baseline accuracy of dermatologists (Esteva et al. (2017), Fink et al. (2019)). Thus, the goal of our project is to identify whether a given skin lesion is benign or malignant, given dermatoscopic images as input (as shown in the figure below), which can be used by health care providers as an additional form of validation, reducing false negatives and allowing for health care providers to more quickly and effectively identify risk for patients.

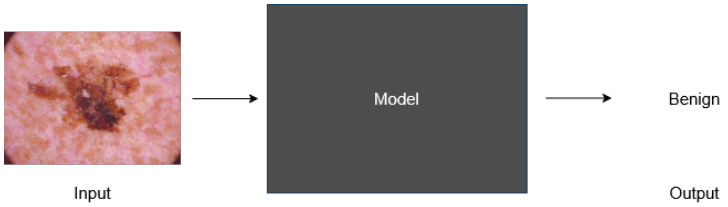


Figure 1: Input and output of model

2 INDIVIDUAL CONTRIBUTIONS AND RESPONSIBILITIES

According to the project proposal, the group has continued to allocate tasks and communicate responsibilities clearly with one another, as displayed in the table below.

Table 1: Team Member Responsibilities

Team Member	Responsibilities
Aneeqa Maham	• Primary model coding and testing • Collaborating on training strategy • Assisting with report writing • Latex formatting
Myuri Thayalan	• Baseline model coding and evaluation • Writing the baseline report section • Proofreading team deliverables
Deena Abrari	• Designing the primary neural network architecture • Implementing evaluation strategy • Contributing to final report
Gauri Karajagi	• Performing data processing • Baseline model coding and evaluation • Organizing and monitoring progress

Our team started the project by arranging the HAM10000 dataset, among other preliminary data processing tasks. Gauri was in charge of completing and examining the preprocessing code to make sure it was operating properly and yielding reliable results. Following this phase, she worked with Myuri to use a Support Vector Machine (SVM) to construct and evaluate the baseline model. Both members helped debug and enhance the model's functionality. Myuri was in charge of creating the baseline write-up, which summarized the procedures, findings, and insights, after the model was complete. The primary model was taken on by Aneeqa and Deena, who are now developing and testing the neural network architecture. They are also in charge of documenting the primary model's procedures, findings, and evaluation.

The team uses Google Colab as our main platform for writing and testing code collaboratively, allowing multiple members to contribute and run experiments in real time. To provide convenient access and version control, we arrange and store all project files, reports, and datasets in a shared Google Drive folder to prevent overwriting others' code. Communication is maintained through frequent group messages on our Instagram group chat and meetings that are held Saturdays at 5pm. We discuss progress, clarify tasks, and provide help with coding and debugging issues as needed. To guarantee consistency, accuracy, and clarity across our deliverables, we have created a clear team dynamic where each member is encouraged to proofread each other's work. This practice has helped us maintain a professional standard of work and avoid duplication of efforts.

All team members have access to the entire codebase, shared files, and project notes, allowing anyone to intervene whenever necessary to reduce risks and ensure seamless development. Data processing, baseline implementation, model creation, and report writing are among the equally divided responsibilities, maintaining both support and accountability. We use shared to-do lists to keep track of assignments and due dates, and we use our group chat to share updates. Members update the group about the files, time, and branches they will be utilizing before beginning work, and follow up once tasks are finished. We are on pace to complete the project on schedule now that the baseline has been completed and the primary model is well underway.

3 NOTABLE CONTRIBUTIONS

This section describes the current progress of the team's project.

3.1 Data Processing

The team obtained the skin cancer dataset "Skin Cancer MNIST: HAM10000" from Kaggle to start data processing. The data set includes 10,015 images of different types of pigmented skin lesions, which are classified into 7 classes: akiec (actinic keratosis), bcc (basal cell carcinoma), bkl (benign lesions of type keratosis), df (dermatofibroma), mel (melanoma), nv (melanocytic nevi), vascular lesions (vasc). These classes represent the different diagnoses for the set of skin lesion images, with mel, akiec, bcc classified as malignant lesions and nv, bkl, vasc, df categorized as benign lesions.

To start, we first loaded the dataset by importing Kaggle and downloading the full dataset folder. Next, we build the full path to the metadata using os.path.join. Lastly, we turned the CSV file into a pandas frame named df.

After downloading the dataset, we used torchvision to define the transform function to process each image in the dataset. The parameters for 'transform' included resizing to a 224 by 224 image to ensure consistency, converting to a tensor, normalizing the images using their mean and standard deviation, and ensuring a diverse dataset by implementing horizontal flips, vertical flips, and rotation. Although most of the images in the HAM10000 dataset are the same size, it is better to use transforms.

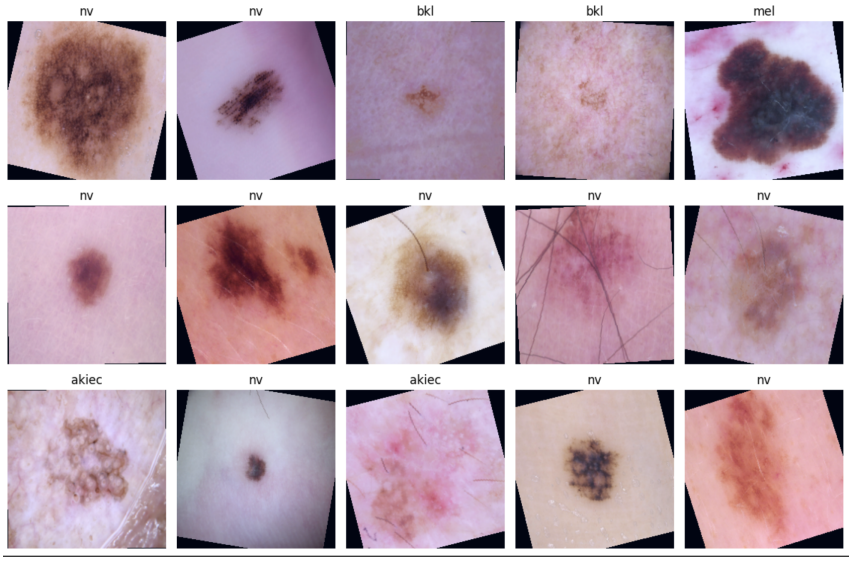


Figure 2: Dataset images after transforming

To make it easier to split the dataset, the team created folders in the `skin-cancer-mnist` directory to sort the classes into their respective folders. We first defined the source and output directory paths. The directories for the seven classes were created in the output directory. To move images into the class folders, we iterated through the dataframe and looked at the `image_id` and `dx` columns in each row and added `.jpg` to `image_id` (i.e., `image_filename = image_id + '.jpg'`) to construct the path for the image. In each iteration, a source path was created for the image and checked for existence in the source directory. If it existed, the image was moved to the appropriate class folder within the output directory.

The last part of data processing is to split the dataset into training, validation, and testing sets. For this code block, the team used the functions `get_relevant_indices(dataset, classes, target_classes)` and `get_data_loader(target_classes, batch_size)` from one of the course labs. The `get_relevant_indices` function returns a list of indices that correspond to images with labels from one of the target classes. The target classes for this project are `akiec`, `bcc`, `bkl`, `df`, `mel`, `nv`, and `vasc`, since we want the model to learn to identify all types of skin lesions. The `get_data_loader` function returns the training, validation, and testing datasets according to the specified batch size. First, we set the variable `trainset` to the skin lesion dataset using `ImageFolder` (pointing to the directory with the sorted classes). Next, we obtained the

relevant indices for each image and its class, and set a random seed to ensure reproducibility during the splitting process. Using the total number of indices, we split the dataset into three sets: training (70%), validation (15%), and testing (15%).

The relevant indices were split into training, validation, and testing indices, which were then passed to `SubsetRandomSampler` to create samplers for each subset. The samplers draw data samples from their respective sub-datasets. Using `DataLoader`, we then iterated through the dataset with each sampler to create the final training, validation, and testing datasets. The total split was 7010 indices for training, 1502 for validation, and 1503 for testing.

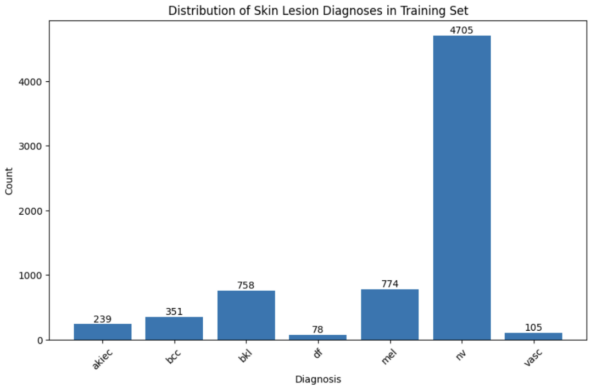


Figure 3: Class distribution of training dataset

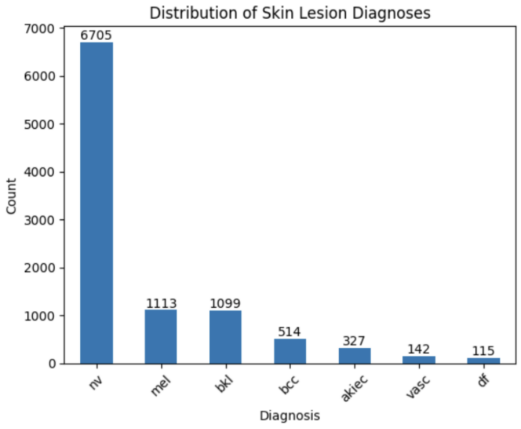


Figure 4: Class distribution of dataset

Our current challenge is the imbalance in the dataset. The original dataset has a very high number of images for class `nv`, while other classes have significantly fewer samples—some with nearly one-third the number, with `df` being the smallest. This skewed data distribution can negatively affect model training and performance. However, this issue can be mitigated using data augmentation for minority classes, employing weighted loss functions, and analyzing the confusion matrix for each class to better understand and address classification imbalances.

The final testing plan for our project will include using a test dataset from the ISIC Challenge Datasets archive, which includes multiple testing datasets from 2016 to 2020. This dataset will help to test the accuracy of our model during final testing. Currently, the baseline model uses the split testing data to test its accuracy.

3.2 BASELINE MODEL

For this project, a Support Vector Machine (SVM) with a radial basis function (RBF) kernel was selected as the baseline model to classify dermatoscopic images as benign or malignant. This model was chosen because it is simple, well-understood, and effective for binary classification tasks with limited computational overhead. It serves as a reasonable starting point to evaluate the feasibility of applying machine learning to skin cancer detection.

We used the HAM10000 dataset, which consists of pictures tagged by diagnosis type, to prepare the dataset. These were mapped to binary labels, with all other classes being regarded as benign (0) and "mel," "bcc," and "akiec" as malignant (1). To balance the number of samples in each class and reduce bias during training, the Synthetic Minority Over-sampling Technique was used in the training data, given the class imbalance (about 4:1 benign to malignant).

Grayscale images were converted to HOG (Histogram of Oriented Gradients) feature representations after being reduced to a consistent 128x128 pixel size. Important texture and edge information is captured by these features, which are useful for identifying patterns of skin lesions.

To further address the imbalance, the `class_weight="balanced"` parameter was used when training the model on the resampled data. The SVM model obtained an overall accuracy of roughly 0.78 on the holdout test set, with a precision of 0.83 and recall of 0.91 for the benign class. On the other hand, performance on the malignant class was significantly worse, with an F1-score of 0.31, precision of 0.41, and recall of 0.25. A summary table and categorization report are used to illustrate these findings.

The discrepancy in the performance between the classes points to a significant issue. The classifier still had trouble correctly identifying malignant lesions even when methods like SMOTE were used to balance the dataset during training. There could be other reasons for this poor performance. First, hand-crafted descriptors such as the Histogram of Oriented Gradients (HOG) are difficult to represent the subtle and varied visual aspects that are frequently present in malignant tumors. Malignant cases can differ greatly in size, color, and border irregularity, in contrast to benign lesions, which may have more uniform shapes and textures. Traditional characteristics have a harder time generalizing across samples because of this visual heterogeneity. Additionally, some malignant lesions may share overlapping characteristics with benign ones, further increasing the likelihood of misclassification. This implies that although the baseline model shows promise, more advanced feature extraction or deep learning techniques could be required to attain consistent performance in identifying cancerous instances.

The algorithm does well on benign samples but has trouble identifying malignant instances, according to one important finding from the baseline results. The malignant class trailed behind (0.41 precision, 0.25 recall), resulting in a low F1-score of 0.31. In contrast, the benign class got good scores (0.83 precision, 0.91 recall). These figures show that the model has a tendency to forecast benignly with more confidence, most likely as a result of the initial class imbalance and the HOG characteristics' limits. accurately from picture data. This implies that malignant lesions are difficult for fixed, hand-crafted features to capture because they are frequently more varied and subtle in appearance. A more expressive model, such a deep neural network that can directly learn intricate patterns from visual input, is required, and the results demonstrate the importance of validating a baseline model in exposing these limits.

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

           0       0.83      0.95      0.88      1202
           1       0.51      0.23      0.32       301

    accuracy                           0.80      1503
   macro avg       0.67      0.59      0.60      1503
weighted avg       0.77      0.80      0.77      1503

Accuracy:  0.801729873586161
```

Figure 5: Classification report of baseline model

In conclusion, the SVM baseline offered a useful starting point, was computationally efficient, and was straightforward to apply. Its poor performance on the malignant class, however, highlights the need for more sophisticated algorithms that can identify intricate patterns in data on skin lesions. This encourages switching to a more adaptable neural network model as the primary approach.

3.3 Primary Model

As the primary model for this project, the group chose to use a ResNet50 model, pretrained on ImageNet, which we chose due to the effective use of CNNs like ResNet50 for medical imaging in past research (Mengfang Li, Yuanyuan Jiang, Yanzhou Zhang, Haisheng Zhu (2023)). The use of transfer learning through ImageNet reduces resources needed and training time. The team's early approach involved freezing the early layers of the model (such as conv1, bn1, and layer1) to retain general features, then replacing the final classification with a single-neuron linear layer with a dropout rate of 0.3. The team had initially planned to use 0.5, but reduced it to improve execution speed; future tests will revisit the original value. This architecture reform allowed the model to perform binary classification, distinguishing images as either benign or malignant. The use of binary classification was chosen to simplify potential classification issues, while also being easily interpretable by intended users (health care providers). The output is a single logit, which is then interpreted using sigmoid activation during evaluation.

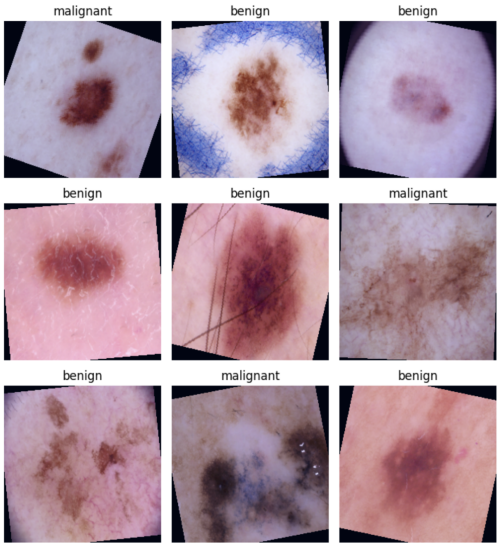


Figure 6: Binary classification example outputs from dataset

The model had a total of 23,510,081 parameters, with 23,284,737 being the trainable ones left after freezing the layers. The training was done with BCEWithLogitsLoss, the Adam optimizer, 10 epochs, and a batch size of 16. The group experimented with different batch sizes to discern which would result in the highest accuracy, and 16 won out between them, with the 8 batch size run being widely inaccurate and the 32 batch size run experiencing strong overfitting.

The group is well underway in further developing this model to further hone its accuracy. For one, the group intends to implement the class weights it has already prepared for the loss function, only left out of the current results due to minor coding errors. This will help balance out the class imbalance between the benign and malignant tumours. Additionally, the group intends to include metadata features (patient sex, age, region of lesion) to further hone the model, providing data that is not just pixel-based, as shown in Kwon et al. (2022). During training, the group will process the metadata into vectors using a multilayer perceptron (MLP), then concatenate it image vector, passing this through the final layer and sigmoid to factor both metadata and image data into the prediction. This architecture was intentionally chosen to allow for easy expansion and alterations throughout the remainder of the project.

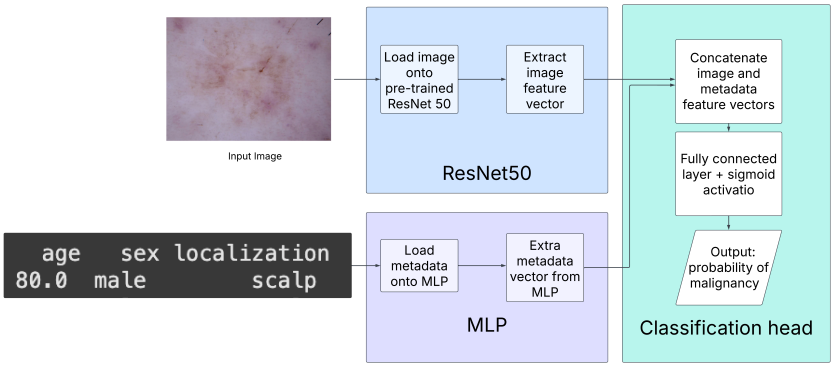


Figure 7: Diagram of planned primary model architecture

After running training loops with the team's current progress, the team found the training and validation errors trending downwards (albeit with some bumps on the validation side), with the training error going down from roughly 17% to 8%, while the validation error went from 15% to 11%. The learning curves displayed this steady decrease, as well as the bumpiness in validation. The team has noticed the slight overfitting, but believes it will be much reduced after the team makes changes like applying class weights, using metadata concatenation, and potentially unfreezing some layers to improve learning and balance out the severely skewed data.

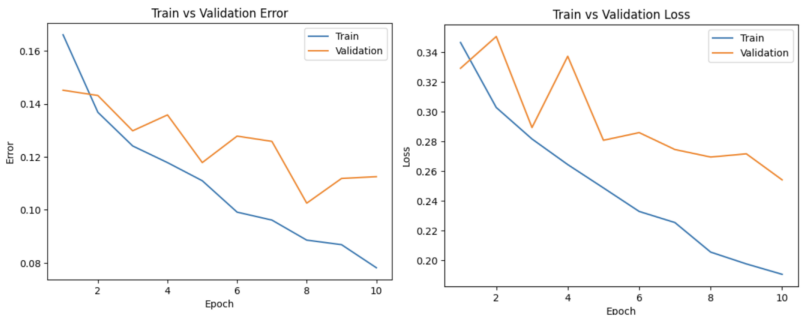
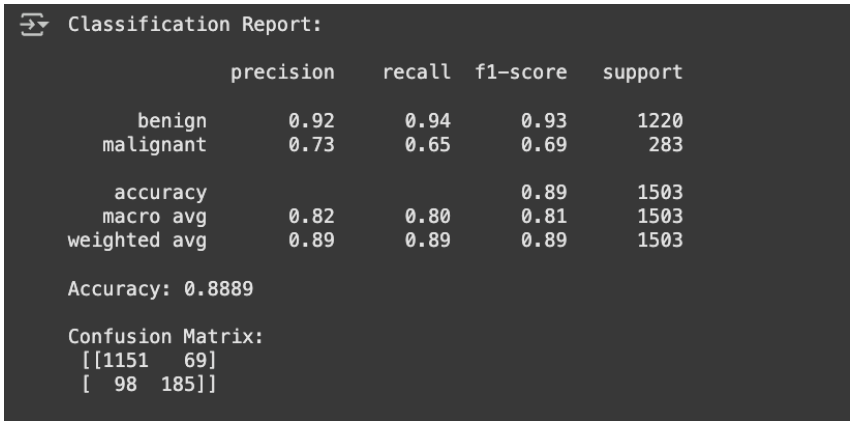


Figure 8: Training/validation loss and error curves for primary model

The team also did a breakdown of the model's features, as seen in the figure below, including its precision and confusion matrix. The F1 score for malignant lesions was 0.69, while for benign lesions it was 0.92. The benign lesions scored generally higher than the malignant one, contributed to by the class imbalance. Again, this is something the group intends to rectify in further iterations. The best validation accuracy reached 88.9%, marking a relatively high value that the group will only further improve upon. Analyzing the confusion matrix also painted a clearer picture of the model's performance. It shows 98 false negatives for malignant lesions, and 69 false positives for benign lesions. The false negatives are an especially notable number (given its medical significance), so the group will keep an eye out for this in future iterations of the model.



```
Classification Report:

                precision    recall  f1-score   support

        benign       0.92      0.94      0.93      1220
     malignant       0.73      0.65      0.69       283

      accuracy                           0.89      1503
     macro avg       0.82      0.80      0.81      1503
  weighted avg       0.89      0.89      0.89      1503

  Accuracy: 0.8889

  Confusion Matrix:
   [[1151   69]
   [  98  185]]
```

Figure 9: Classification report for the current best primary model

In conclusion, the primary model is well on its way, and while there is still work to be done, it is showing good promise and the group has a clear idea of how to proceed.

REFERENCES

Hiam Alquran, Isam Abu Qasmieh, Ali Mohammad Alqudah, Sajidah Alhammouri, Esraa Alawneh, Ammar Abughazaleh, and Firas Hasayen. The melanoma skin cancer detection and classification using support vector machine. In *2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pp. 1–5, 2017. doi: 10.1109/AEECT.2017.8257738.

I. D. Apostolopoulos, N. D. Papathanasiou, S. K. Rengarajan, H. A. Hesham, M. A. Tzani, A. T. Azar, and E. I. Papageorgiou. Explainable artificial intelligence (xai) for the detection of covid-19 in chest x-rays using transfer learning and a new class activation mapping technique. *Diagnostics*, 12:2921, 2022. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9730580/.

Abhishek Bandyopadhyay, Rishav Ghosh, Vaibhav Kumar, Raju Ghosh, and Hiranmay Saha. A comprehensive survey on biomedical image analysis and deep learning techniques: Recent trends and future directions. *Bioengineering*, 10:1293, 2023. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10662291/.

Yoshua Bengio and Yann LeCun. Scaling learning algorithms towards AI. In *Large Scale Kernel Machines*. MIT Press, 2007.

Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, January 2017. doi: 10.1038/nature21056. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC8382232/.

C. Fink, A. Blum, T. Buhl, C. Mitteldorf, R. Hofmann-Wellenhof, T. Deinlein, W. Stolz, L. Trennheuser, C. Cussigh, D. Deltgen, J.K. Winkler, F. Toberer, A. Enk, A. Rosenberger, and H.A. Haenssle. Diagnostic performance of a deep learning convolutional neural network in the

differentiation of combined naevi and melanomas. *Journal of the European Academy of Dermatology and Venereology*, 34(6):1355–1361, December 2019. doi: 10.1111/jdv.16165. URL https://pubmed.ncbi.nlm.nih.gov/31856342/.

GeeksforGeeks. Implementing svm from scratch in python, 2023. URL https://www.geeksforgeeks.org/machine-learning/implementing-svm-from-scratch-in-python/.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

International Skin Imaging Collaboration. Isic challenge datasets, 2019. URL https://challenge.isic-archive.com/data/#2019. Accessed: 2025-07-11.

Klaus-Robert Müller and Philipp Tschandl and Noel Codella and Veronica Rotemberg and others. Skin cancer mnist: Ham10000 dataset. https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000, 2018. Accessed: 2025-07-11.

Dohyun Kwon, Jaemyung Ahn, Chang-Soo Kim, Dong Ohk Kang, and Jun-Young Paeng. A deep learning model based on concatenation approach to predict the time to extract a mandibular third molar tooth. *BMC Oral Health*, 22(1), December 2022. doi: 10.1186/s12903-022-02614-3. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC9730580/#:~:text=The%20deep%20learning%20model%20was,the%20CNN%20part%20were%20concatenated.

Mengfang Li, Yuanyuan Jiang, Yanzhou Zhang, Haisheng Zhu. Medical image analysis using deep learning algorithms, 2023. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC10662291/#:~:text=Deep%20learning%20techniques%2C%20notably%20Convolutional,advanced%20healthcare%20solutions%20(26).

Geetika Munjal, Paarth Bhardwaj, Vaibhav Bhargava, Shivendra Singh, and Nimish Nagpal. Skinsage xai: An explainable deep learning solution for skin lesion diagnosis. *Health Care Science*, 3 (6):438–455, November 2024. doi: 10.1002/hcs2.121. URL https://doi.org/10.1002/hcs2.121.

scikit-learn developers. Support vector machines — scikit-learn documentation, 2024. URL https://scikit-learn.org/stable/modules/svm.html.

Alquran et al. (2017) Apostolopoulos et al. (2022) Bandyopadhyay et al. (2023) Bengio & LeCun (2007) Goodfellow et al. (2016) Klaus-Robert Müller and Philipp Tschandl and Noel Codella and Veronica Rotemberg and others (2018) Hinton et al. (2006) International Skin Imaging Collaboration (2019) Mengfang Li, Yuanyuan Jiang, Yanzhou Zhang, Haisheng Zhu (2023) scikit-learn developers (2024) GeeksforGeeks (2023)