Computer Aided Pigmented Skin Lesion Classification

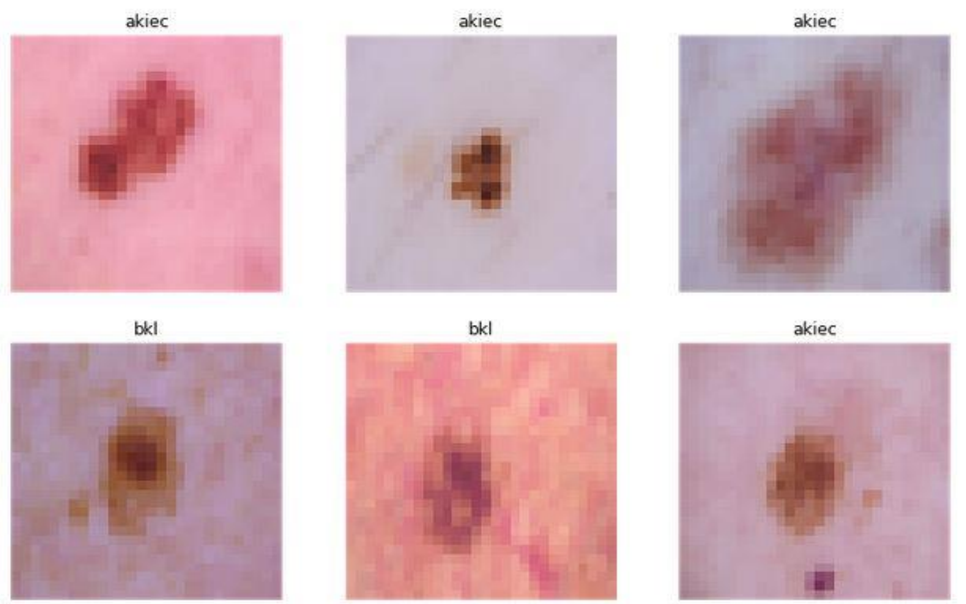
Technical Specifications

Ananalysis by Christopher Liedel

Data Used

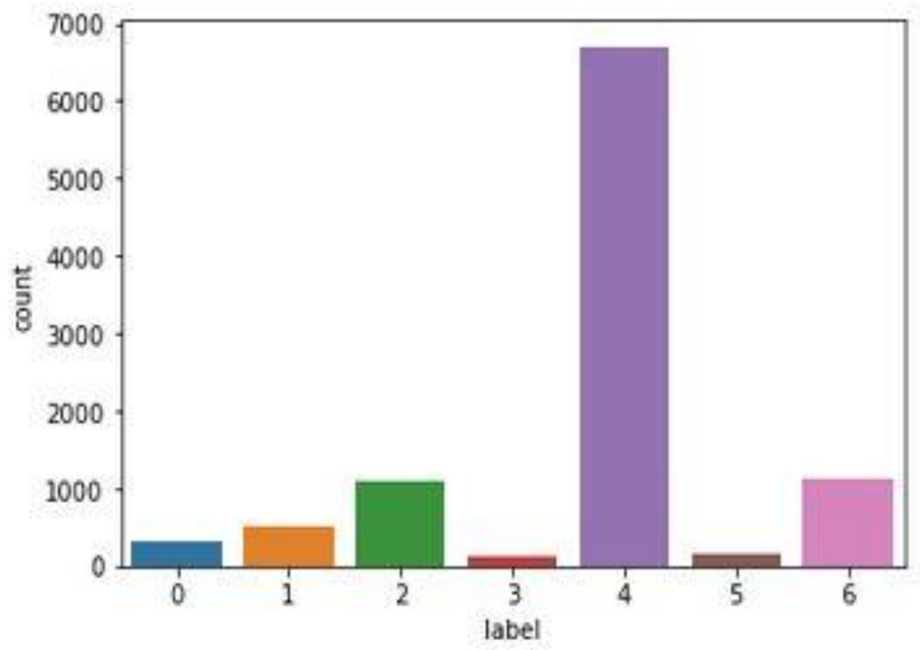
Dataset: HAM_10000 (hmnist_28_28_RGB.csv) for pigmented skin lesion diagnosis.

- Contains 10,000 images of skin lesions, both benign and malignant.



HAM_10000_metadata.csv file

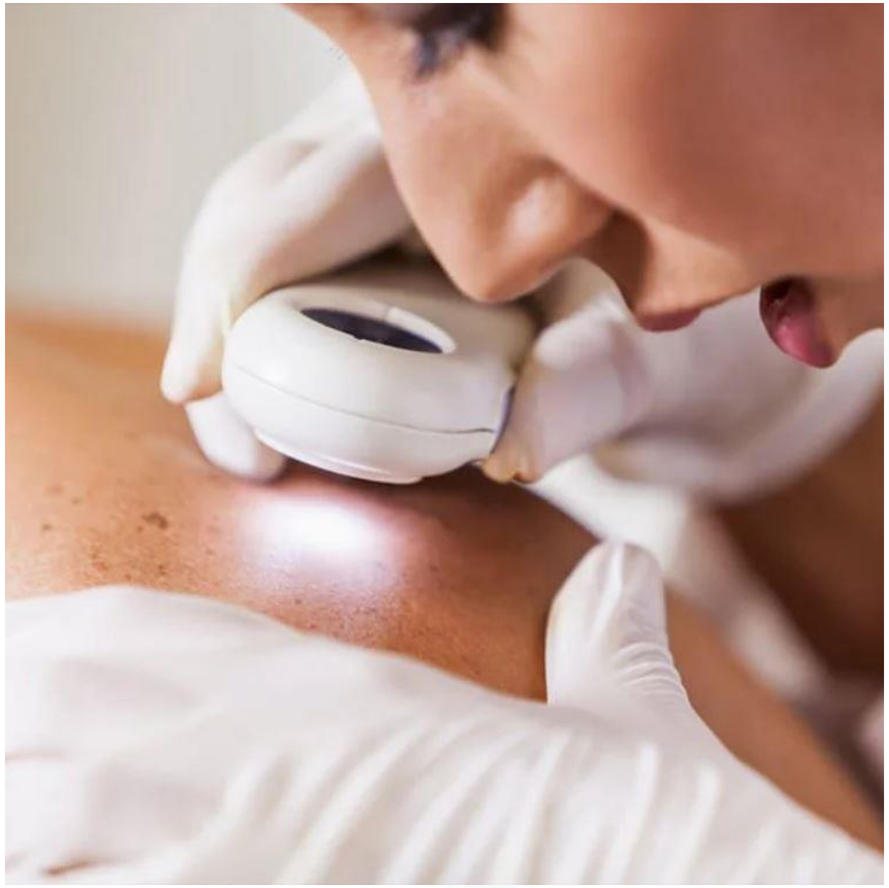
- Includes image/lesion ID, dx_type, sex, age and location of lesion.



Use Case: Multi-Class Classification

Skin cancer is the most common human malignancy

- Early detection of skin cancer allows for a more successful and simpler cure.
- Initial Diagnosis is performed in a clinical setting limiting access to medical care for some.
- Automating skin lesion classification can be used as diagnostic aid in a clinical setting.
- Automation of skin lesion classification through mobile devices can potentially expand needed low-cost medical care outside the clinic.



Technology

- IBM Watson Cloud and Jupyter notebooks.
- Python with data science libraries: pandas, scikit-learn, seaborn, matplotlib, numpy and keras.
- Data set and metadata file in CSV format.

Data Assessment

Assessment and Visualizations with pandas, matplotlib and seaborn.

- **Data types:** Conversion of age from float to int and that columns match content
- **Range:** Check value distribution using stats. *df.describe()*
- **Emptiness:** Check null values; impute missing values
- **Uniqueness:** Are duplicates present where undesired? (e.g. lesion ID)
- **Set Memberships:** Are only allowed values chosen for categorical or ordinal fields? (e.g. location of lesion, sex…)

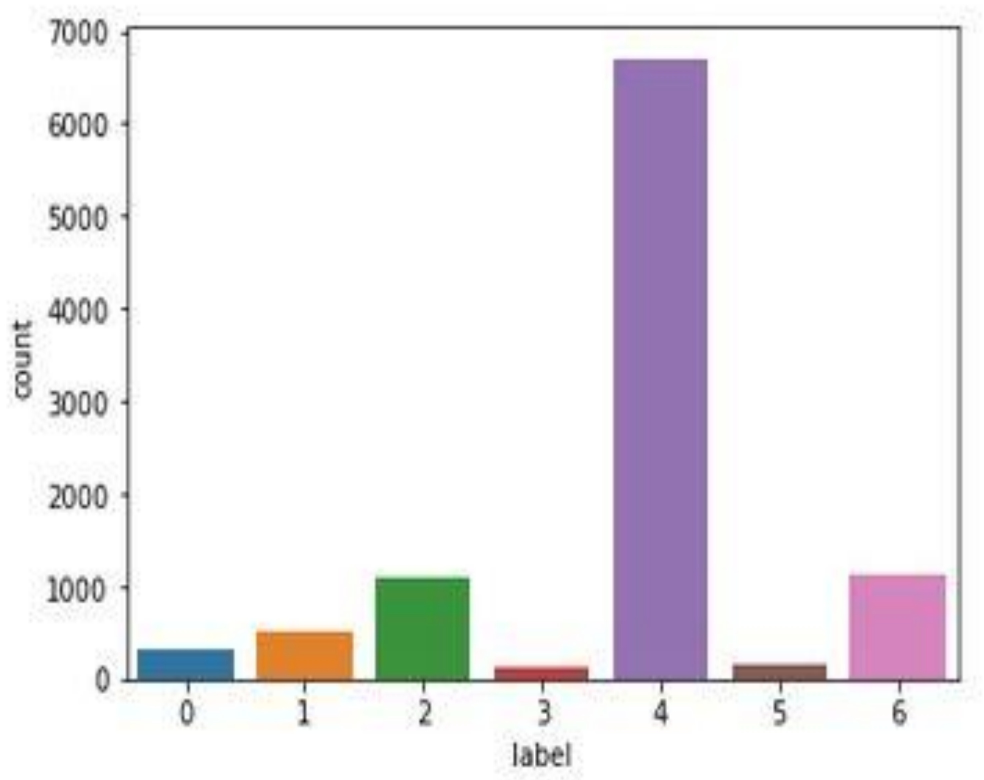
Data Analysis – diagnosis

Lesions

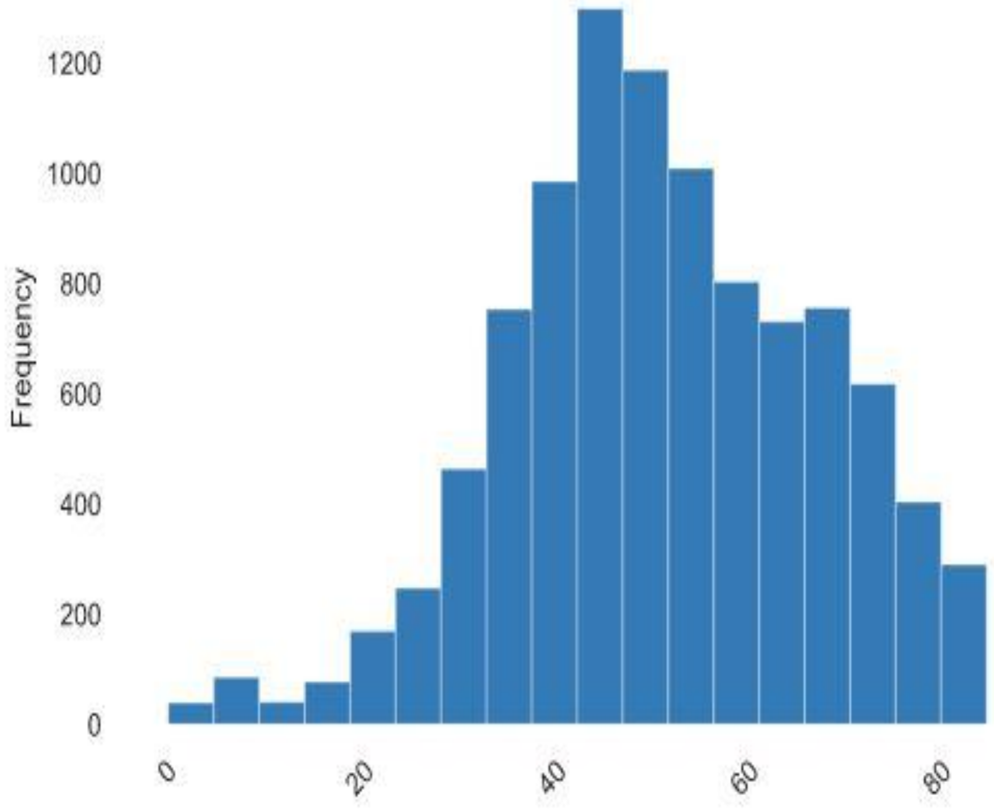
There are 7 types of lesions

- 0: nv - Melanocytic nevi
- 1: mel - Melanoma
- 2: bkl - Benign keratosis-like lesions
- 3: bcc - Basal cell carcinoma
- 4: akiec - Actinic keratoses
- 5: vasc - Vascular lesions
- 6: df - Dermatofibroma

Distribution



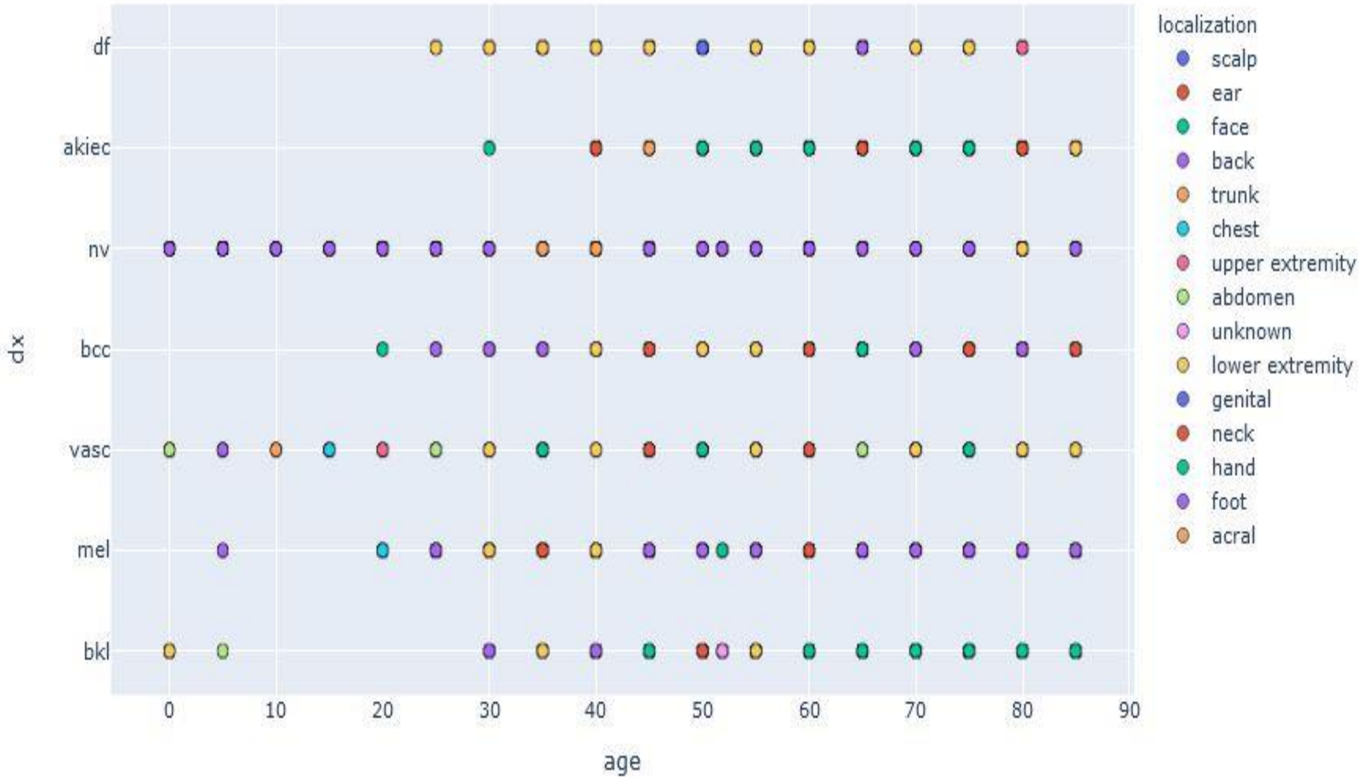
Data Analysis – age & frequency



Histogram with fixed size bins (bins=18)

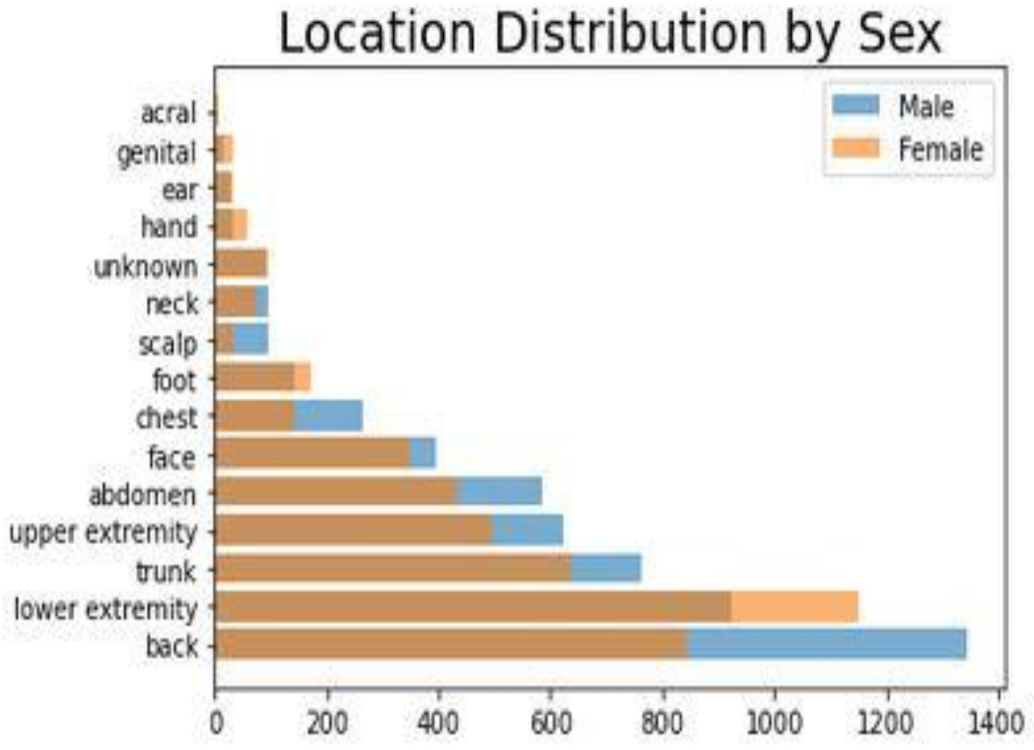
Value	Count	Frequency (%)
45	1299	13.0%
50	1187	11.9%
55	1009	10.1%
40	985	9.8%
60	803	8.0%
70	756	7.5%
35	753	7.5%
65	731	7.3%
75	618	6.2%
30	464	4.6%
Other values (8)	1353	13.5%

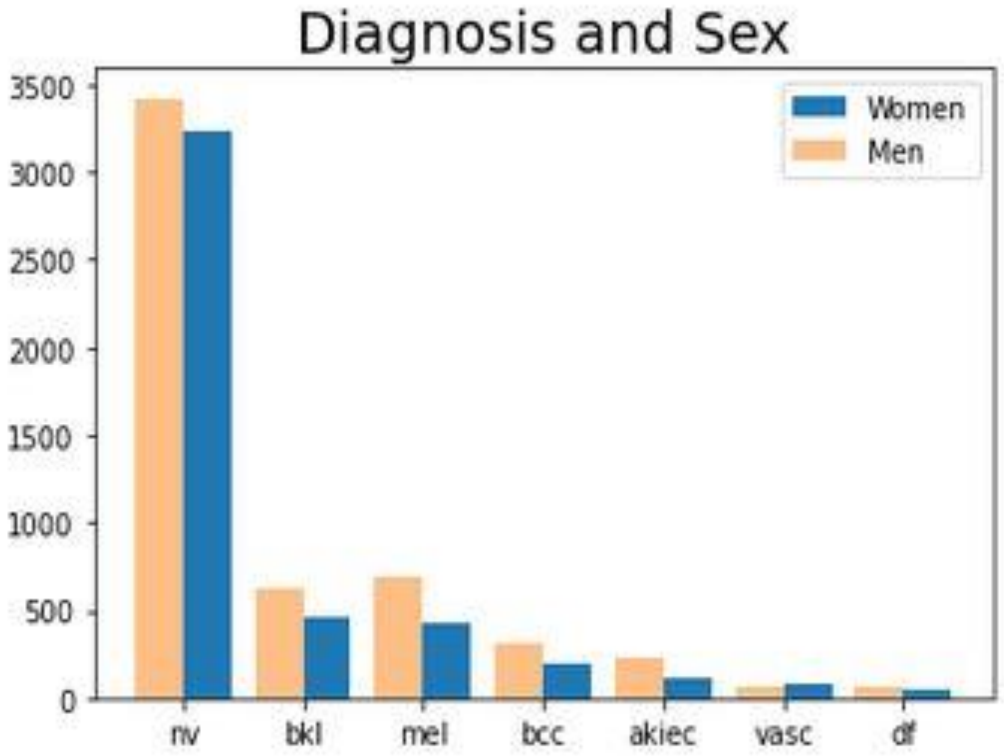
Data Analysis – age/diagnosis/location



Value	Count	Frequency (%)
extremity	3195	24.2%
back	2192	16.6%
lower	2077	15.7%
trunk	1404	10.6%
upper	1118	8.5%
abdomen	1022	7.7%
face	745	5.6%
chest	407	3.1%
foot	319	2.4%
unknown	234	1.8%
Other values (6)	497	3.8%

Data Analysis – sex/diagnosis/location





Data Preparation

- Work on copies of the dataset (keep original intact)

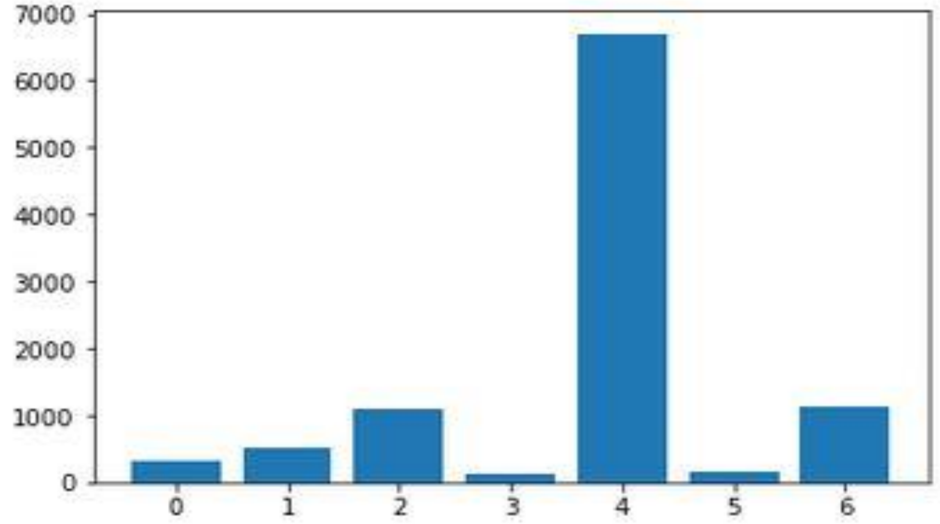
- **Data Cleaning**:
  - Fill in missing values (e.g., mean, median, zero…) or drop the rows or columns.
  - Fix or remove outliers (optional).
- **Feature Selection** (optional):
  - Drop attributes that provide no usefulness for the task. (e.g. lesion/image ID)
  - Drop duplicates.
- **Feature Engineering** where appropriate:
  - Discretize features
  - Decompose features (e.g., categorical).
  - Add transformations of features.
- **Feature Scaling**:
  - Standardize or normalize features

Feature Transformation:

Imbalanced Data

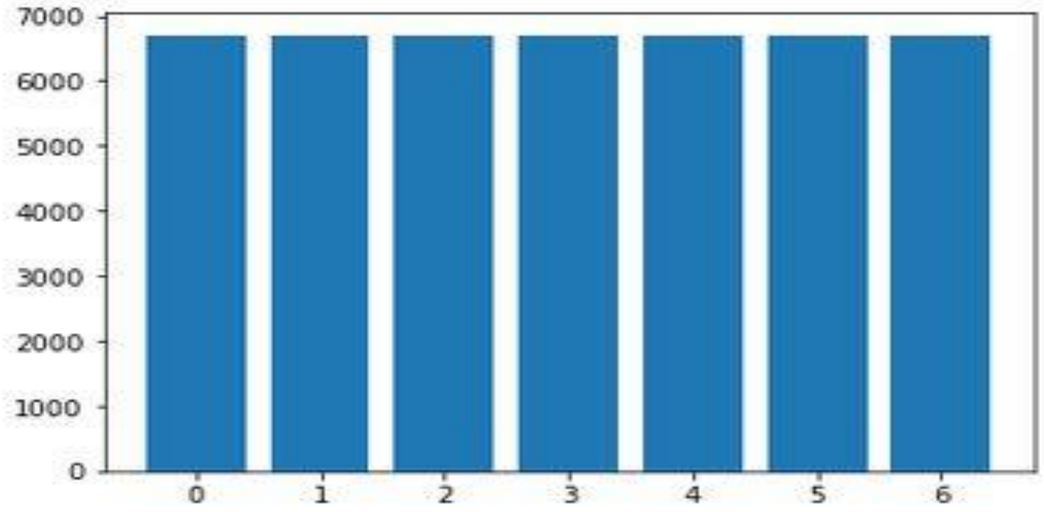
Class Distribution

```
Class=2, n=1099 (10.974%)
Class=4, n=6705 (66.950%)
Class=3, n=115 (1.148%)
Class=6, n=1113 (11.113%)
Class=5, n=142 (1.418%)
Class=1, n=514 (5.132%)
Class=0, n=327 (3.265%)
```



Result of Using Synthetic Minority Oversampling Technique: SMOTE

```
Class=2, n=6705 (14.286%)
Class=4, n=6705 (14.286%)
Class=3, n=6705 (14.286%)
Class=6, n=6705 (14.286%)
Class=5, n=6705 (14.286%)
Class=1, n=6705 (14.286%)
Class=0, n=6705 (14.286%)
```



Performance Metric

We have a multi-class classification project so **accuracy** will be chosen for our performance metric, along with **precision**, **recall** and **specificity**.

Macro precision and recall used for balanced dataset.

Weighted precision and recall used for imbalanced dataset.

90/10 Split to Train and Test sets. Models evaluated on test sets.

Model Evaluation performed using both balanced and imbalanced data.

Precision — Out of all the examples that predicted as positive, how many are positive?

Recall — Out of all the positive examples, how many are predicted as positive?

Used in Medicine

Specificity — Out of all the people that do not have the disease, how many got negative result?

Sensitivity — Out of all the people that have the disease, how many got positive test results?

Tested Algorithms

Algorithm	Scores With Imbalanced Data	Scores With Balanced Data
Dummy Classifier	Acc .49 Weighted: Prec .49 / Rec .49 Specificity 0.861	Acc .15 Macro: Prec ..15 / Rec .15 Specificity 0.858
Logistic Regression	Acc .69 Weighted: Prec .63 / Rec .69 Specificity 0.898	Acc .64 Macro: Prec .64 / Rec .64 Specificity 0.941
Decision Tree	Acc .63 Weighted: Prec .63 / Rec .62 Specificity 0.908	Acc .85 Macro: Prec .85 / Rec .85 Specificity 0.974
Random Forest Classifier	Acc .68 Weighted: Prec .62 / Rec .67 Specificity 0.892	Acc .92 Macro: Prec .92 / Rec .93 Specificity 0.986
XGBoost Classifier	Acc .73 Weighted: Prec . 68 / Rec .73 Specificity 0.909	Acc .95 Macro: Prec .95 / Rec .95 Specificity 0.991
Neural Network with one hidden layer	Acc .73 Weighted: Prec .71 / Rec .73 Specificity 0.925	Acc .971 Macro: Prec .97 / Rec .97 Specificity 0.995
Neural Net with additional hidden layer of 64 neurons and dropout at (0.20)	Acc .73 Weighted: Prec .72 / Rec .74 Specificity 0.921	Acc .979 Macro: Prec .98 / Rec .98 Specificity 0.996
Neural Net with additional hidden layer of 64 neurons, dropout at (0.20) Tuned using Grid Search	NA	Acc .979 Macro: Prec .98 / Rec .98 Specificity 0.996

Best Model: Neural Network
Summary

Tuned, Fully Connected **Neural Network**

Accuracy, **Precision** and **Recall** of **0.98**
Specificity of **0.996**

Epochs: 100

Batch size: 850

1st hidden layer: 512 neurons

2nd hidden layer: 64 neurons

1st activation: relu

2nd activation: relu

Dropout: 0.2

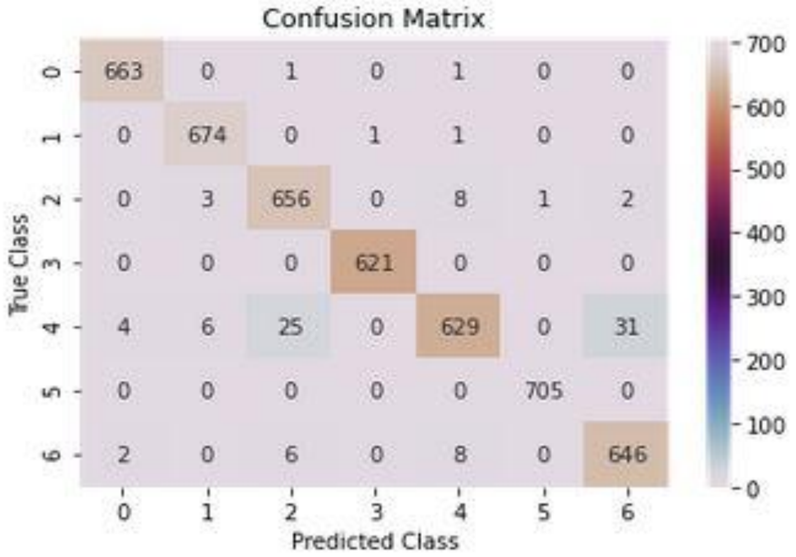
Output layer activation: softmax

Optimizer: adam

Loss: sparse_categorical_crossentropy

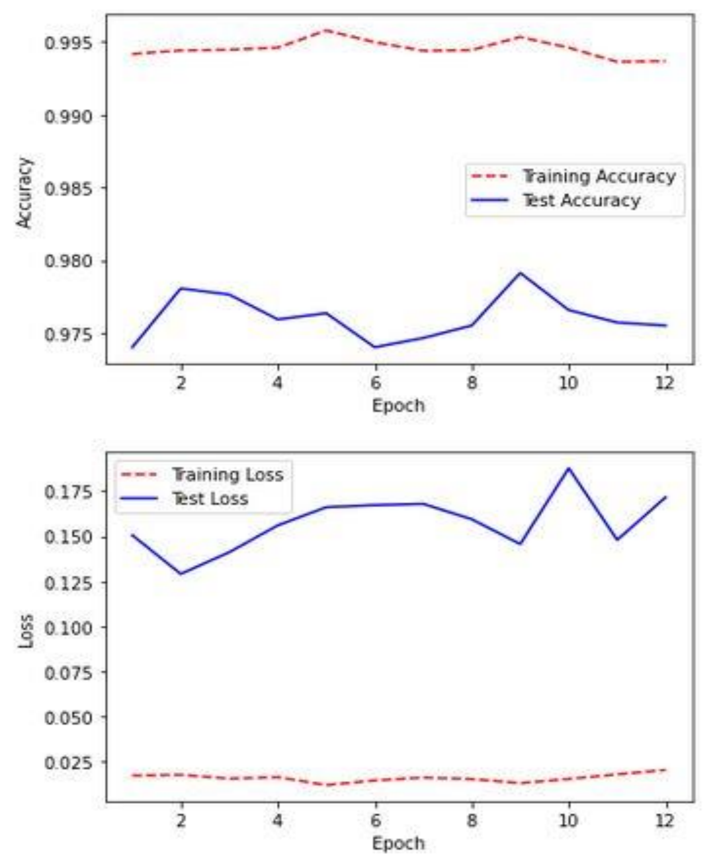
Classification Report

	precision	recall	f1-score	support
0	0.99	1.00	0.99	665
1	0.99	1.00	0.99	676
2	0.95	0.98	0.97	670
3	1.00	1.00	1.00	621
4	0.97	0.91	0.94	695
5	1.00	1.00	1.00	705
6	0.95	0.98	0.96	662
accuracy			0.98	4694
macro avg	0.98	0.98	0.98	4694
weighted avg	0.98	0.98	0.98	4694

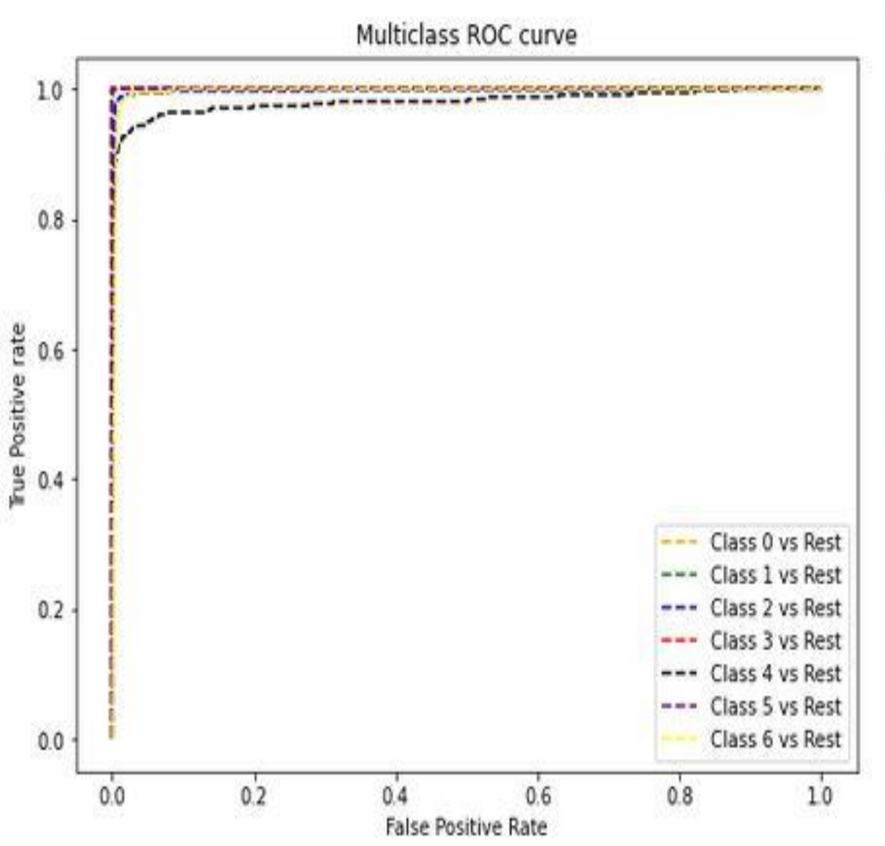


Performance Charts

Train and Test Performance



Probability Curves



Example of a prediction made using our best model

- The image used was selected from an external source and not taken from the dataset used to train and evaluate the model
- Model predicted actinic keratoses : class 4. This is the label given from the source as well.

```
from keras.preprocessing import image

img_path = ('./Desktop/acct2.jpg')
img = image.load_img(img_path, target_size=(158,158))
img
```

```
# Classify image not in the dataset
import cv2
img = cv2.imread('./Desktop/acct2.jpg',0)


img = cv2.resize(img,(58,58))
cv2.waitKey(0)
cv2.destroyAllWindows()

print(class_label)

prediction_cplx = tuned_model.predict_classes(img)

print('The predicted class of the lesion is:', prediction_cplx[1])
```

```
{0: 'nv: melanocytic nevi', 1: 'mel: melanoma', 2: 'bkl: benign keratosis-like lesion', 3: 'bcc: basal_cell_carcinoma', 4: 'akiec: actinic_keratoses', 5: 'vasc: vascular lesion', 6: 'df: dermatofibroma'}
The predicted class of the lesion is: 4
```

XGBoost

Summary

Second best performing model is **XGBoost** with **accuracy**, **precision** and **recall** of **0.95**; **Specificity** of **0.991**

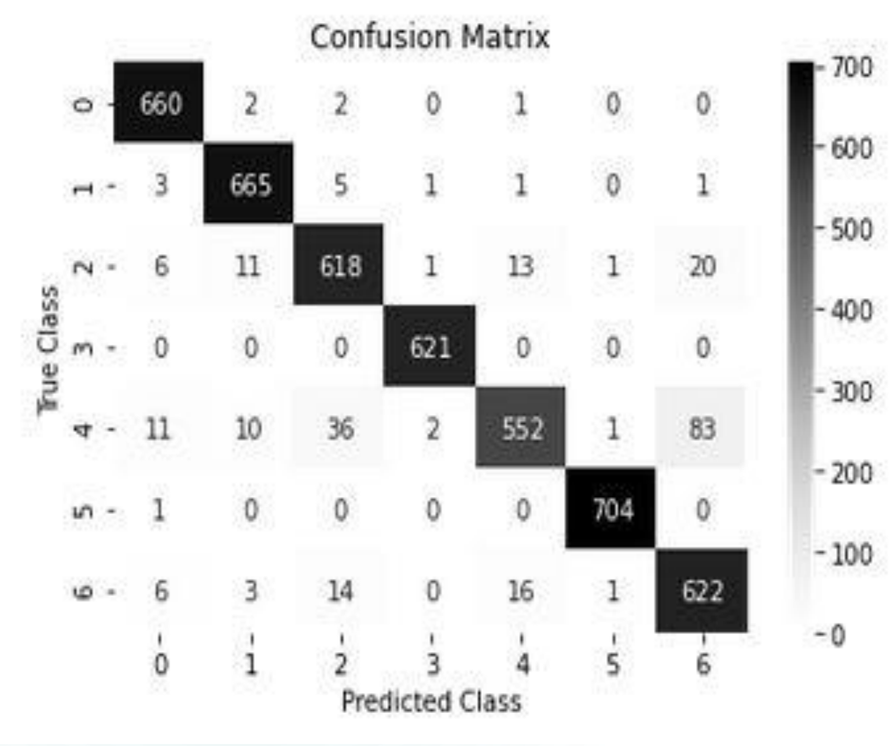
- max_depth : 7
- booster: "gbtree"
- num_classes: 7
- eval_metric: "mlogloss"
- objective: "multi:softprob"

Classification Report

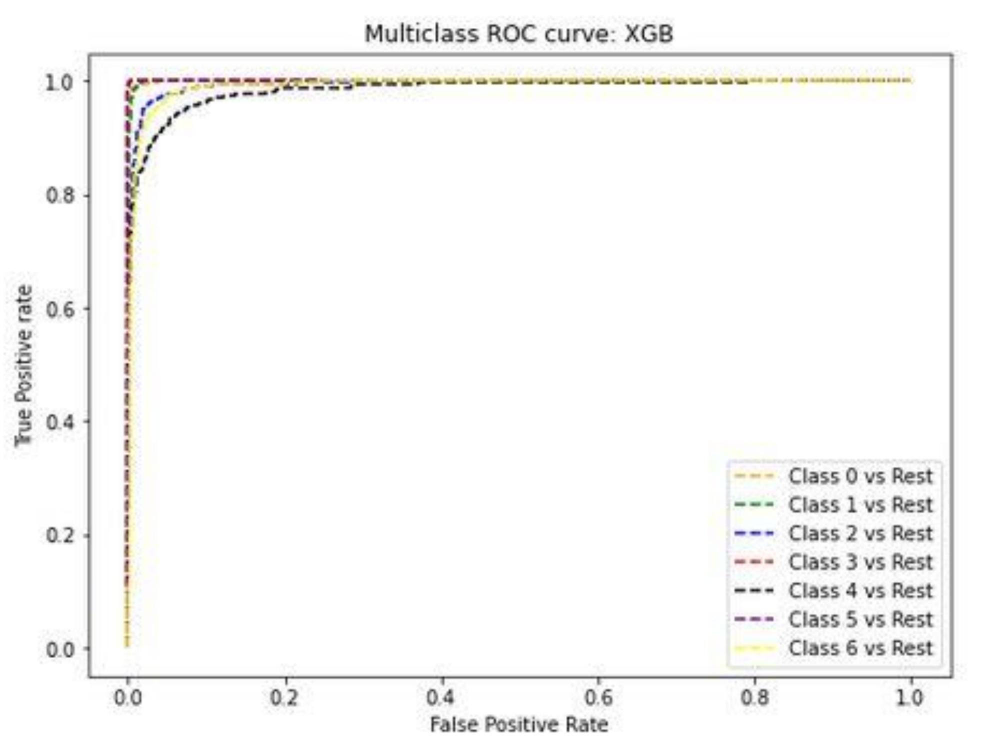
xgb:	precision	recall	f1-score	support
0	0.96	0.99	0.98	665
1	0.96	0.98	0.97	676
2	0.92	0.92	0.92	670
3	0.99	1.00	1.00	621
4	0.95	0.79	0.86	695
5	1.00	1.00	1.00	705
6	0.86	0.94	0.90	662
accuracy			0.95	4694
macro avg	0.95	0.95	0.95	4694
weighted avg	0.95	0.95	0.95	4694

XGBoost Performance

Prediction Results



Probability Curves



Using the model

Trained model can be used in an application placed on a tablet device or smart-phone.

- A digital image is taken of the lesion in question.
- Image enters the transformation pipeline.
- The prepared image is fed into the pre-trained model.
- Model predicts the lesion class.
- The model prediction can be used as and aid to diagnosis.

Can be used outside of a clinical setting expanding access to needed low-cost medical care.

Thank You for Viewing!