UNVEILING ALZHEIMER'S: DEEP LEARNING MODEL FOR MRI-BASED DISEASE STAGE CLASSIFICATION

MSCA 31009: MACHINE LEARNING & PREDICTIVE ANALYTICS

SYDNEY PETERS

27 MAY 2023

PROBLEM STATEMENT

- The task was building a deep learning model to classify brain MRI images into four categories/labels of dementia:
  1. **MILD DEMENTIA**
  2. **MODERATE DEMENTIA**
  3. **NON-DEMENTIA**
  4. **VERY MILD DEMENTIA**
- The datasets train and test both contain a total of around **5,000 images** each.
- The images are labeled brain MRI images collected from patients with Alzheimer's disease and segregated into severity and progression of the disease in the brain.
- The goal was to train a model that can accurately classify new brain MRI images into the appropriate dementia category.

ASSUMPTIONS/HYPOTHESES ABOUT DATA AND MODEL

- **HYPOTHESES**

1. Datasets consist of representative brain MRI images for Alzheimer's disease classification, with reliable labels.
2. Relationship exists between the input MRI images and the presence or absence of Alzheimer's disease.
3. A convolutional neural network (CNN) can learn this relationship and be used on MRI brain scan images to determine the dementia level of a patient with Alzheimer’s disease.

- **ASSUMPTIONS**

1. Datasets consist of representative brain MRI images for Alzheimer's disease classification, with reliable labels.
2. The selected machine learning model is appropriate and capable of learning relevant features from the input images.
3. Training approaches such as adaptive learning rates, early stopping, and batch normalization are utilized to enhance the model's generalizability and performance.
4. The model's performance on the validation set serves as an indicator of its generalization capability and can be used to evaluate its performance on unseen data.

EXPLORATORY DATA ANALYSIS

train_ds

Found 5,121 files belonging to 4 classes. Used 4,097 files for training.

val_ds

Found 5,121 files belonging to 4 classes. Used 1,024 files for validation.

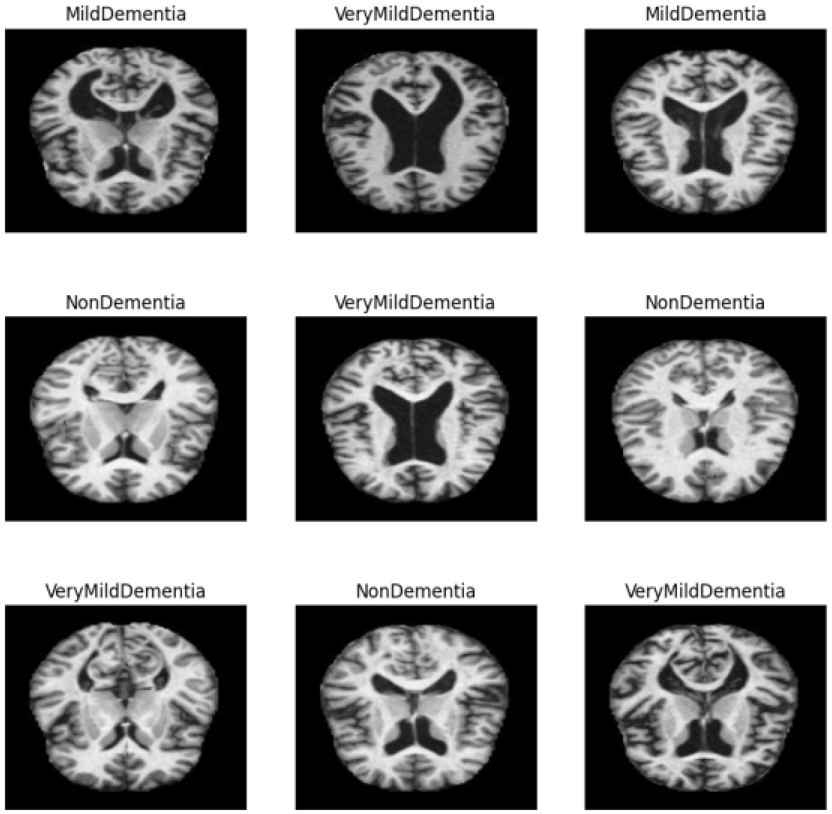
images_shape

Shapes of individual batches of images: (16, 176, 208, 3)

num_samples

Total # of Samples: 257

VISUALIZE DATA:



FEATURE ENGINEERING & TRANSFORMATIONS

DATA LOADING

- **tf.keras.preprocessing.image_dataset_from_director** function to load images from a directory
- **validation_split** parameter split training dataset into a training set and a validation set (20% to validation set, 80% to training set). Enabled evaluation of model's performance on unseen data during the training process.
- Function automatically preprocesses the images and resizes them to specified IMAGE_SIZE

FEATURE ENGINEERING

- **One-Hot Encoding** Converted labels for both train and validation sets into one-hot encodings using tf.one_hot
- **Caching & Prefetching** train_ds and val_ds) are cached and prefetched, improving training performance by caching the data in memory and overlapping data preprocessing with model training

METRIC SELECTION

- Determined accuracy could not be used for model selection because dataset was unbalanced
- **ROC AUC** was chosen metric for evaluation, indicated by specifying **tf.keras.metrics.AUC** as metric when compiling the model

MODEL ARCHITECTURE

- **build_model** creates a sequential model with various layers (i.e., convolutional, separable convolution layers, batch normalization, max pooling, dropout, and dense layers)
- Input shape of model set to (*IMAGE_SIZE, 3), indicates input images have three color channels (RGB)
- Final dense layer has NUM_CLASSES neurons with softmax activation function to produce class probabilities

PROPOSED APPROACHES (MODEL) WITH CHECKS FOR OVERFITTING/UNDERFITTING

PROPOSED MODEL APPROACHES

- Multiple convolutional blocks, followed by dense blocks, ending with softmax activation layer for **multi-class classification**
- Model incorporates techniques such as **separable convolution, batch normalization, max pooling, and dropout** to improve its performance

CHECKS FOR OVER/UNDERFITTING

- **Early Stopping** stops training if validation loss doesn't improve after a set # of epochs
- **Learning Rate Scheduling** decreases the learning rate over time to promote smooth convergence
- **Dropout** randomly deactivates neurons during training to prevent overfitting
- **Batch Normalization** normalizes activations to stabilize and speed up training
- **Model Checkpoint** saves the best model based on validation loss for evaluation (this wouldn’t run on my on my system for whatever reason; will work to get this implemented with more time)

PROPOSED SOLUTION (MODEL SELECTION) WITH REGULARIZATION

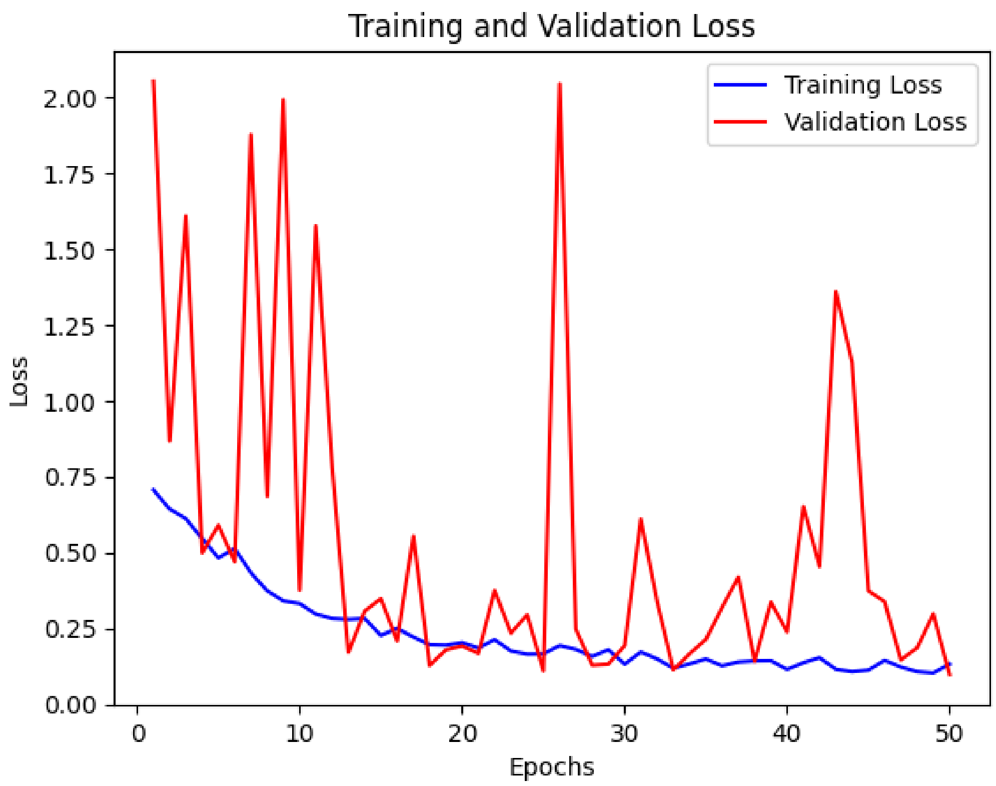
REGULARIZATION TECHNIQUES

- **Early Stopping**
  - Training process was monitored using validation data.
  - Early stopping was applied with a patience of 10; if there was no improvement in the validation loss for 10 consecutive epochs, training was stopped early.
  - Best weights from the training process were restored.
- **Dropout**
  - Dropout layers were inserted in the model architecture after specific convolutional blocks.
  - Specific dropout rate (portion of neurons deactivated) was 0.2, meaning that, during training, 20% of the neurons in those specific layers were randomly deactivated.
- **Learning Rate Scheduling**
  - Exponential decay was applied to the learning rate using a decay factor of 0.1.
  - The initial learning rate (lr0) was set to 0.01, and the decay was performed every 20 epochs.
  - This gradual reduction in the learning rate helped the model converge smoothly and avoid overshooting the optimal solution.

MODEL RESULTS (ACCURACY)

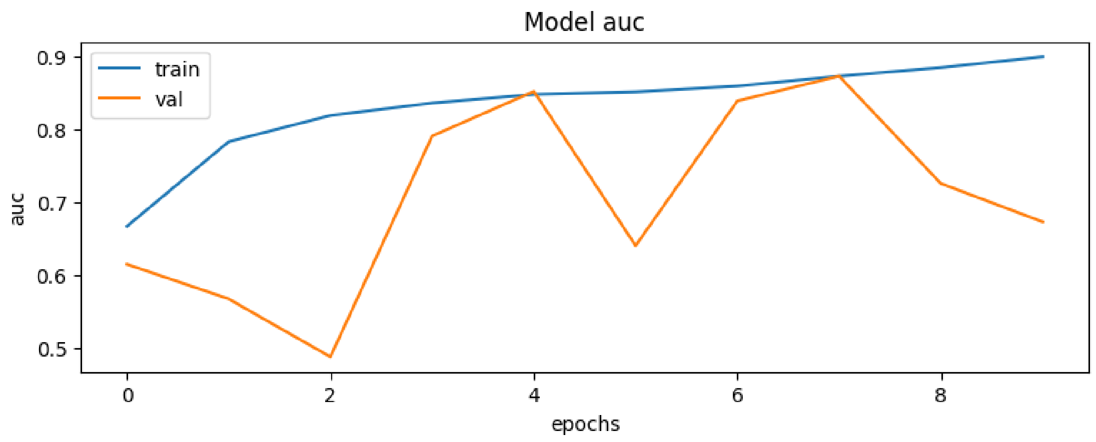
TRAIN & VALIDATION

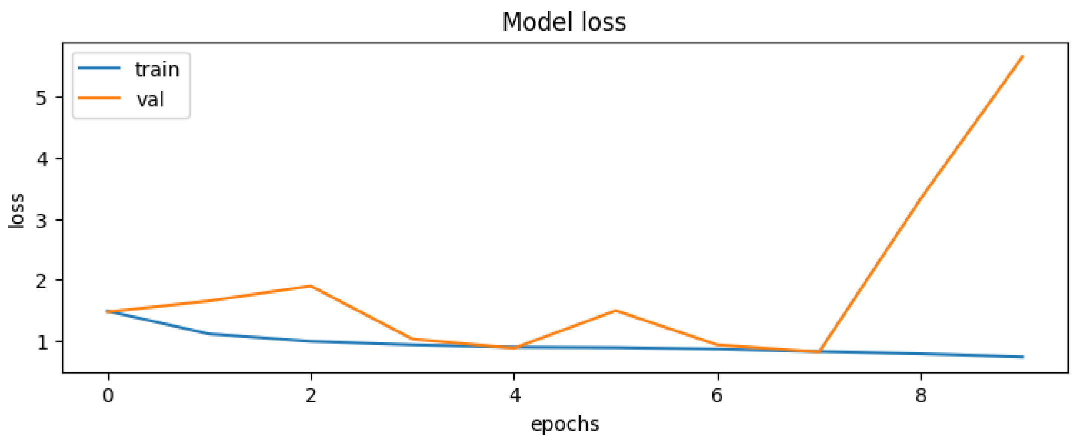
- **loss 0.1320**
- **auc 0.9954**
  - Suggests model has achieved a high level of classification performance on the training data, indicating strong predictive accuracy and successful learning of underlying patterns.
- **val_loss 0.0968**
- **val_auc 0.9979**
  - Also indicates that model performs exceptionally well on the validation data, demonstrating its ability to generalize and make accurate predictions on new, unseen instances.



MODEL RESULTS (ACCURACY)

GRAPH OF ROC AUC METRIC & LOSS AFTER EACH EPOCH – TRAINING AND VALIDATION

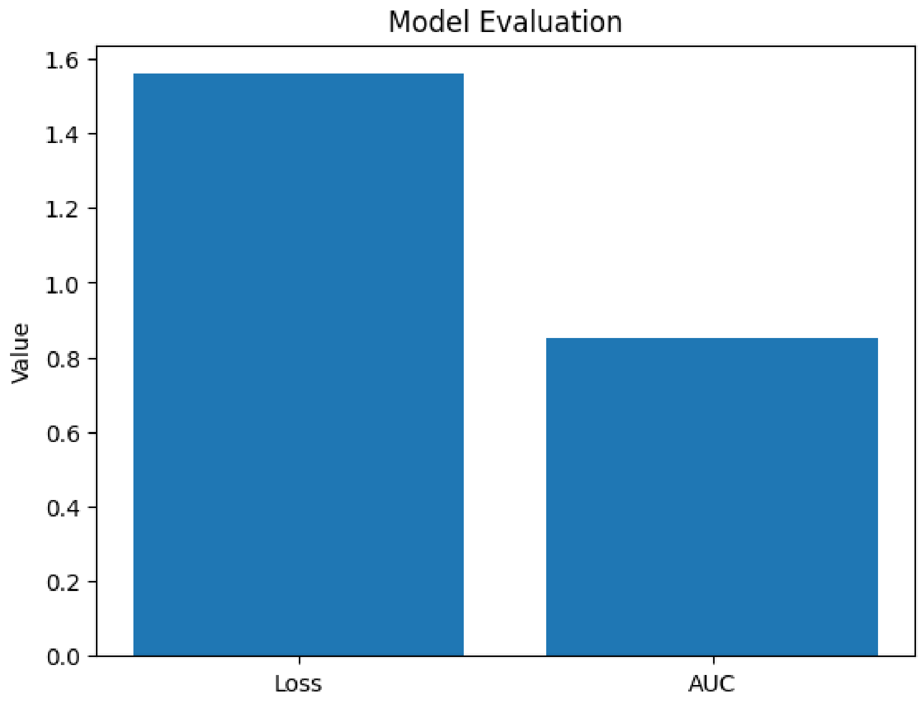




MODEL RESULTS (ACCURACY)

TEST

- **loss 1.5586**
  - Indicates average discrepancy between predicted and actual values on the test data.
- **auc 0.8529**
  - Suggests relatively high level of classification performance in distinguishing between different classes in test dataset.
- Model has reasonable predictive capability on the test data, but further analysis and evaluation is needed.



LEARNINGS FROM THE METHODOLOGY

- **Accuracy may be the most conventional metric for model performance evaluation, but it cannot be used for imbalanced datasets.** It can be misleading on unbalanced datasets because it calculates the % of correctly classified samples without considering the class distribution. **If one class dominates a dataset, a model that predicts the majority class for every sample can achieve high accuracy without effectively learning about the minority classes.**
  - Since the number of images in each class for the training data was imbalanced ([717, 52, 2560, 1792]), **we had to consider other evaluation metrics and ended up using ROC-AUC.**
  - ROC-AUC is a method used very often for multi-label classification, as we performed here.
- **Proper data preprocessing**, including data transformations on inputs, improves model performance.
- **Utilizing callbacks** optimizes training and monitoring of model performance (e.g., early stopping, learning rate schedular, and model checkpoint).
- **Batch normalization** helps stabilize and accelerate the training process by normalizing the inputs.
  - This leads to faster convergence and improved generalization.

MODEL APPLICATIONS & IMPORTANCE

- Image classification in healthcare has a myriad of applications and is extremely important for the future health of our population.
  - Advancing potential for early detection and treatment of Alzheimer’s by **accurately classifying the disease stage** based on MRI scans.
  - Tool for researchers in mission to **uncover the underlying causes and manifestations** of Alzheimer’s.
  - Aids in **efficient allocation of healthcare resources** (e.g., identifying early-stage patients who may benefit most from specific interventions and support services).
  - Streamlines process of analyzing MRI scans; improved efficiency can save radiologists’ and clinicians’ time and **reduce diagnostic turnaround times**.
- **Most importantly, outcomes and quality of life can be improved by models like this – for Alzheimer’s patients and their caretakers alike.**

FUTURE WORK TO ENHANCE MODEL PERFORMANCE AND GENERALIZE TO UNSEEN DATA

REGULARIZATION TECHNIQUES

- There is much room to implement more regularization techniques in the model – some ideas:
  - **L1 and L2 Regularization:** Add penalty terms based on magnitudes of weights to loss function.
  - **Data Augmentation:** Apply random transformations to training data to increase input diversity and improve generalization.
  - Explore **spatial dropout or variational dropout** to drop feature maps or apply different dropout mask apart from the standard dropout technique.
  - Consider **layer normalization or instance normalization** as alternatives to standard batch normalization.

FUTURE WORK & PERSONAL GOALS

- Fix MacOS system errors + incompatibility errors (e.g., model checkpoint) – I used 10 instead of 100 epochs because of these, and I would use 100 next time for improved ROC-AUC.
- Input more diverse and representative data to increase the model's exposure to various Alzheimer's cases and improve generalization.
  - There are many neuroimaging open-access databases that have been created to foster clinical neuroscience discoveries (e.g., http://www.oasis-brains.org/ and https://adni.loni.usc.edu/).
- Train a model on non-Alzheimer's patients' data to help the model learn features and patterns that specific to healthy brain function.
  - This model's broader exposure could be leveraged to make predictions on both Alzheimer's and non-Alzheimer's cases.

APPENDIX

- I am motivated to one one day work on machine learning and AI-applications for the management and treatment of Alzheimer's, as this topic is close to my heart, and this was a perfect introduction to that!
- Given time and bandwidth constraints, I was not able to dig into building my own model as much as I'd hoped for this project. I faced several incompatibility errors as I was making a simpler model based off Assignment 7. I have had a lot of issues with tensorflow on my Mac OX. Nonetheless, I gleaned a lot of insights and overlaps in learnings from going through this Kaggle model and adjusting based on my project goals. I hope there is an opportunity to work on a similar project for my Capstone.
  - **ALZHEIMER'S DATASET:** https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images?datasetId=457093&sortBy=voteCount
  - **ALZHEIMER MRI MODEL + TENSORFLOW 2.3 DATA LOADING:** https://www.kaggle.com/code/amyjang/alzheimer-mri-model-tensorflow-2-3-data-loading