# CZ-1 DL Alzheimer's

## Final Report

## CS 4850 - Sections 02 & 04 - Fall 2025

**Sharon Perry**

**December 3, 2025**

**Team Members:**

**Julia Johnson**

**Jordan Rainford**

**Website:** https://sites.google.com/view/ml-alzheimers/home

**GitHub:** https://github.com/CZ1-Alzheimers/Using-Machine-Learning-to-Diagnose-Alzheimers/tree/main

| | |
|---|---|
| **Number of lines of Code** | 1100 |
| **Number of Components** | 4 Total Components:<br>- Python Script<br>- Full Dataset (ADNI/Oasis/AIBL)<br>- SPM12<br>- CAT12 |
| **Total Man Hours (Estimate / Actual)** | 110 (Estimate) / 147.5 (Actual) |
| **Status** | 100% complete and working as designed |

# Table of Contents

## Abstract

Early detection of Alzheimer's disease (AD) remains a significant clinical challenge, as the changes associated with cognitive decline are often subtle and difficult to identify through visual assessment alone. This study investigates modern machine learning methodologies to improve the prediction of cognitive impairment using volumetric magnetic resonance imaging (MRI)–derived region-of-interest (ROI) features. We aim to classify normal control (NC), mild cognitive impairment (MCI), and AD. We constructed three binary classifiers [NC vs. AD, MCI vs. AD, and NC vs. MCI] and evaluated various algorithms, including logistic regression, random forests, neural networks, and support vector machines (SVM). Using measurements generated from eight anatomical brain templates, our models learned patterns indicative of normal cognition, mild cognitive impairment, and AD. Among all tested approaches, the radial basis function (RBF) SVM consistently achieved the highest performance, reaching accuracies of approximately 70–80% depending on the classification task. We discuss the implications of this model's dominance for future clinical applications and the continued development of machine learning–driven diagnostic tools.


## Introduction

AD is a neuro-degenerative disorder which is responsible for causing 60-80% of all dementia cases throughout the United States. Our current medical technology, while making significant enhancements in recent history, is still unable to accurately detect the subtle differences in the anatomy of the brain which can signal early AD development. The earlier that this disease is detected, the more effective treatment will be.

Our proposed method is to utilize three SVM binary classifiers (NC v. AD, NC v. MCI, AD v. MCI) trained on three databases of MRI data: Alzheimer's Disease Neuroimaging Initiative (ADNI), The Open Access Series of Imaging Studies (OASIS), and Australian Imaging Biomarkers and Lifestyle Study (AIBL). We processed each image in our dataset using MATLAB, SPM12, and CAT12. We believe our justification for choosing a SVM model is novel, and could give insight into potential scientific purposes for other machine learning (ML) models

## Related Work

Since ML tools have rapidly improved in their pattern recognition abilities, more scientists are looking to use this technology to recognize patterns too subtle for the human eye and possibly come to conclusions no human scientist could reasonably come to. In the last five years, there have been a plethora of papers which propose utilizing an AI model to analyze volumetric brain

data, be it through MRI or PETG scans, and form an understanding of the commonalities between brains that are NC, have MCI, and have AD.

Many of these methods involve utilizing computer vision technology to segment the brain, only then using binary classifiers to form profiles of NC, MCI, and AD brains. While novel and certainly impressive, we believe that enhancing our binary classifiers should be the primary focus of our study and therefore decided to use a standardized MRI pre-processing pipeline.

## Neuroimaging and Preprocessing

Detection of AD is typically based on a variety of clinical assessments, medical consultations, and the examination of brain scans. MRIs are the most widely utilized biomarker, employed in over 80% of single-modal AD detection studies. MRIs offer high spatial resolution necessary for analyzing anatomical structures of the brain. However, accurately detecting AD using algorithms is challenging due to inherent complexities in medical images, low image quality, and difficulties associated with preprocessing and segmentation steps. Effective diagnostic systems rely on strict preprocessing pipelines to handle discrepancies and variability across a wide variety of subjects and scans.

Accurate diagnosis often depends on the quantitative analysis and segmentation of specific brain structures. Segmentation aims to divide images into defined regions like gray matter, white matter, and cerebrospinal fluid. Structural abnormalities, particularly tissue atrophy, are popular indicators used in diagnosing AD. Specifically, atrophy of medial temporal structures is considered a valid diagnostic marker even at the MCI stage. The hippocampus and the precuneus are identified as highly sensitive biological indicators of AD, particularly in the early stages.

## Classification Algorithms

Traditionally, the detection of AD using ML has employed a wide range of algorithms. Common classification algorithms in early ML research included Neural Networks (NN's) and SVMs applied to MRI scans and, occasionally, PETG scans. Traditional, established methodologies such as SVM still maintain relevance in AD classification studies; however, the field has seen a shift toward advanced deep learning methods since the rapid evolution of deep learning algorithms.

As for newer deep learning models, Convolutional Neural Networks (CNNs) have revolutionized outcomes when it comes to disease detection. CNNs merge feature extraction, feature selection, and classification all into a single process. The use of CNNs is prominent, with the *Journal of Medical Imaging* accounting for about 70% of AD detection methods utilizing CNN systems. While this technique makes sense for many other studies, we found it best to refine an SVM model for the sake of the accessibility of the technology. CNN's may be more efficient to train as they include the segmentation process within the pipeline, not all medical communities have the infrastructure or environment to dedicate hundreds of hours in compute time and thousands of

watts towards new technology to accomplish a task that MATLAB and SPM12 have already perfected the pipeline for.

## Datasets and Classification Targets

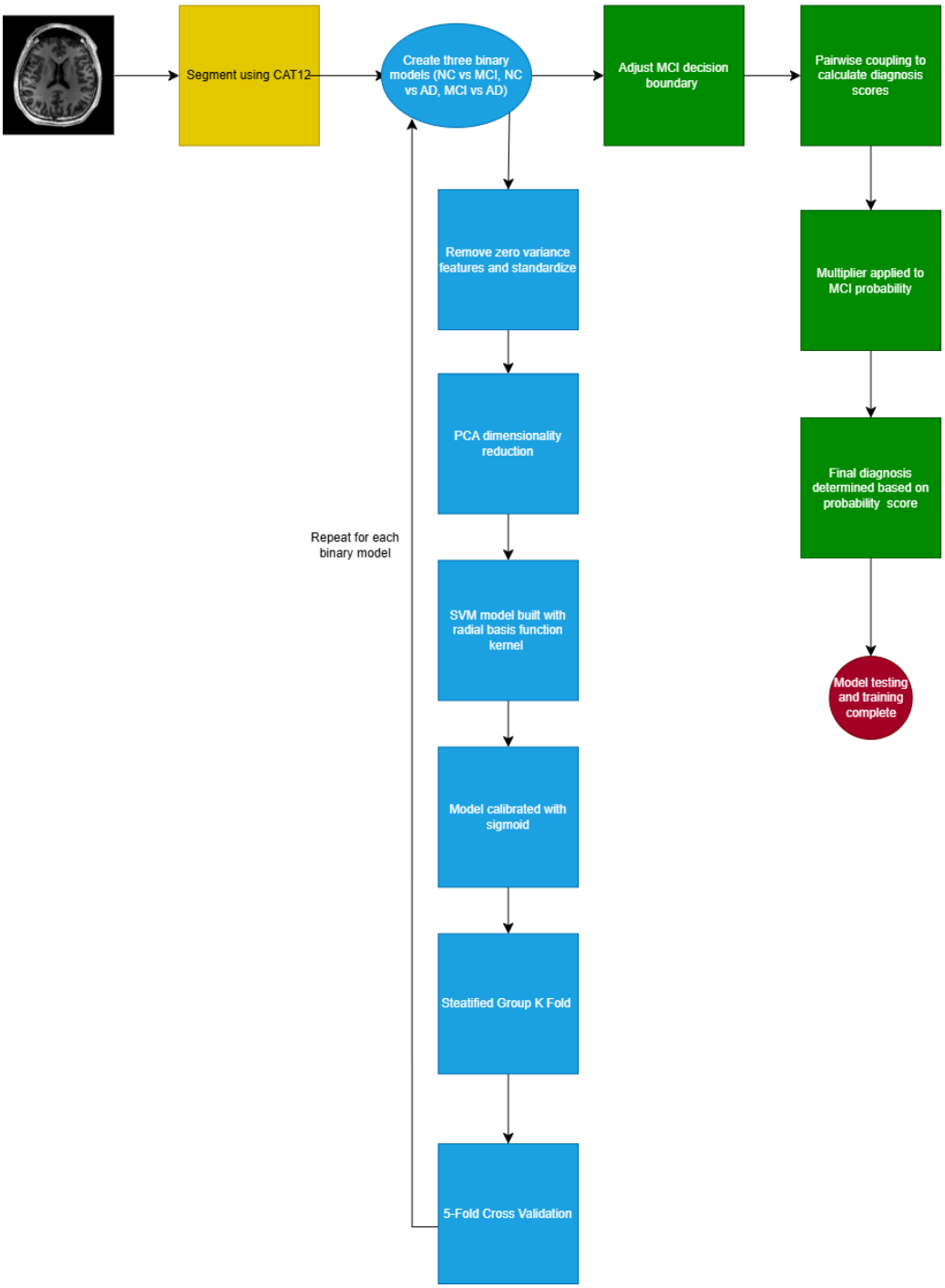
Publicly available datasets are crucial for developing and validating these methods. The ADNI dataset is the most extensively utilized database in this research domain. ADNI contains comprehensive information, including MRI, PET, and clinical data, aimed at identifying and validating biomarkers for tracking AD progression. The OASIS dataset is also commonly applied for brain MRI segmentation and AD diagnosis.

The classification objective often extends beyond simple AD versus NC distinctions to include multi-class categorization. Given that MCI is recognized as the prodromal phase of AD, efforts focus heavily on differentiating between NC, MCI (including subtypes like Early Mild Cognitive Impairment (EMCI) and Late Mild Cognitive Impairment (LMCI)), and AD states to facilitate early detection.

Although advanced deep learning models currently dominate the literature due to their superior performance in feature extraction and accuracy, traditional ML methods such as SVM still provide an important comparative baseline for understanding model performance, especially with new approaches focusing on specialized preprocessing methodologies.

# Methodology

## Method Overview



## Method Explanation

Each component in our methodology serves a vital purpose to accomplishing our goal. The following pipeline was implemented to create, train, and test our model.

### Segmentation using CAT12 of Input MRI Image

The purpose of this component is to convert raw T1 MRI images into usable imaging features. After the image is segmented, the data is mapped using eight templates, creating a total of 914 features. These features represent various aspects of the image, such as regional measurements or intracranial volume, and can be interpreted by the model. The segmentation extracts meaningful features, and the templates ensure that features across all images are represented in the same way.

### Creating Three Binary Models

Three binary models are used to make comparisons between each diagnosis label (NC vs AD, MCI vs AD, NC vs MCI). We use three binary models due to the ambiguous nature of an MCI diagnosis. Because MCI often resembles NC, predicting it can be difficult. Binary models allow for flexible processing when determining the final diagnosis. They also are better at determining boundary cases between each diagnosis. Each model is trained with identical pipelines, but different hyperparameters due to different needs between models.

### Removing Zero-Variance Features and Standardization

Features with no variance or unique information are dropped to reduce noise. The following equation is performed to convert each nonzero feature to have zero mean and unit variance to allow unique features to be comparable.

$$\tilde{x}_j = \frac{x_j - \mu_j}{\sigma_j}$$

This standardization is important before principal component analysis (PCA) and the model are applied to the data so that distances and dot-products are meaningful. Applying standardization to the dataset cleans the data and prepares it for feature scaling.

### PCA Dimensionality Reduction

PCA is applied to reduce the highly dimensional dataset into a more manageable, less dimensional dataset. Each binary model has a PCA value between 50-200 that is chosen based on a grid search. The dataset for that model is then reduced from 914 features to the determined PCA value. This reduction in dimensionality makes the differences between each label more obvious while preserving variance in the dataset.

This is performed by computing the correlation matrix of standard features, performing eigen decomposition, and projecting data onto the leading principal components. The top *k* components are kept to represent a fraction of variance, where *k* is the chosen PCA value.

## SVM Model with Radial Basis Function Kernel

Each binary model is built as an SVM model with a radial basis function (RBF) kernel. An SVM model is used because they are good at creating nonlinear decision boundaries between classes. The RBF kernel maps the inputs into a feature space with infinite dimensions so that complex boundaries can become linear using this equation:

$$K(x, x') = \exp\left(-\gamma \|x - x'\|^2\right)$$

The decision function utilizes this equation:

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b$$

This type of model is good for this purpose due to its ability to handle nonlinear data with moderate dimensionality. It is well equipped to handle a small dataset, such as ours.

## Calibrate Model with Sigmoid

Calibration via sigmoid is helpful to convert the decisions from SVM models into probability scores. These probability scores can be combined across each model, which is necessary to predict a final diagnosis. The sigmoid function maps decision scores to class probabilities using the following equation:

$$P(y = 1 \mid x) = \frac{1}{1 + \exp(Af(x) + B)}$$

In the equation, *A* and *B* are determined within cross-validation by minimizing the likelihood of negative logs. Many operations further in the model's pipeline require probabilities in place of raw SVM decision scores, so calibration is a vital step to ensuring accurate predictions.

## Stratified Group K-Fold

The purpose of Stratified Group K-Fold is the split data across training and testing folds while ensuring that data from the same subjects stay in the same group. This prevents inaccurate high-performance predictions from the model interpreting data it has seen before.

The stratification portion of this component ensures that each fold has approximately similar class proportions. This step ensures that folds and class evaluations are fair and balanced, while avoiding data leakage.

## 5-Fold Cross Validation

5-fold cross validation is a technique used to tune hyperparameters and evaluate the performance on each binary model. This works by taking the five folds generated by the Stratified Group K-Fold component and dividing them into four training folds and one validation fold. This process is repeated five times, one for each fold.

This process produces an evaluation of the model based on the average performance across all five folds. Overall accuracy, sensitivity, specificity, area under the curve (AUC), and F1 scores are evaluated and stored. Receiver operating characteristic (ROC) curves are also evaluated and plotted.

This component serves to reduce variance in model evaluation and determine the best hyperparameters to use in each model via grid search based on the calculated metrics. The hyperparameters it tunes are PCA number, selector k, model C, model gamma, and custom class weights to boost MCI predictions.

## Pairwise Coupling to Calculate Diagnosis Scores

When the models have finished training, predicting the final diagnosis on an MRI image is the next goal. To accomplish this, we utilize pairwise coupling to combine binary probabilities and convert them into multiclass probabilities. The multiclass probability vector is represented as:

$$\mathbf{p} = (p_{\text{NC}}, p_{\text{MCI}}, p_{\text{AD}})$$

We utilized the Wu-Lin-Weng pairwise coupling algorithm to perform this task. The algorithm works by assigning pairwise probabilities from each classifier, then determining the class probability that best identifies the image using this equation:

$$\sum_{i<j} \left(p_{ij} - \frac{p_i}{p_i + p_j}\right)^2$$

This technique is used because the generated probabilities are more consistent and more easily interpreted compared to inconsistent voting algorithms.

### Adjusting MCI Decision Boundary and Applying MCI Multiplier

A common issue with this and other similar models is MCI prediction. MCI, by nature, closely resembles NC and has more subtle features. Because of this, MCI prediction would be disproportionately low compared to NC and AD prediction. To fix this, adjust the MCI decision boundary and apply a multiplier to the MCI probability.

By adjusting the decision boundary, the threshold for what the model considers MCI is lowered, making predictions more likely. Applying the multiplier also increases the overall probability of an MCI diagnosis. Through testing, we determined that shifting the decision boundary by 0.18 and declaring a small multiplier of 1.01x produced the best results.

These changes in MCI probability give the diagnosis label the boost it needs to be accurate without sacrificing AD and NC diagnosis accuracy.

### Final Diagnosis based on Probability Score

This final component converts the multiclass probabilities in a final determined diagnosis label for clinical use. All label probabilities are compared after MCI boosting is complete, and the label with the highest probability score is determined to be the final diagnosis.

## Evaluation Metrics

To evaluate this method, we use a variety of  techniques to ensure that our standards are being met and to determine weak points in the model.

### Confusion Matrix

Because accurate predictions are vital to our goal, we utilized a confusion matrix (CM) to evaluate how often true and false predictions occur for each diagnosis label. For each binary model, a mean CM was calculated and displayed to help us determine weak points in each binary model.

After predicting the diagnosis on our testing set, made with 20% of our dataset, a 3x3 CM is calculated and displayed. The CM displays true NC, MCI, and AD predictions compared to false predictions, and displays how many times one label was confused for another. An overall accuracy is then calculated as well based on the number of correct predictions in the CM.

### Classification Metrics

For each binary model, various classification metrics are calculated and averaged across all five folds. The metrics we focus on are accuracy, sensitivity, specificity, AUC, and F1 scores when evaluating the binary models. These metrics help us determine overall performance and help us find any weaknesses in the datasets. Standard deviations are also calculated for each metric to help us determine model reliability.

### ROC Curves

ROC curves are used to graph the model's ability to make accurate predictions. The ROC curve of each binary plot is generated to evaluate and compare each model's performance and help us identify ideal sensitivity thresholds.

# Experimental Results

## Data and Experiment Settings

The methodology above was implemented using a dataset of multiple MRI images containing subjects from ADNI, OASIS, and AIBL databases. Each image is processed through CAT12 for segmentation, then saved with all features in a single dataset file. Each image is then labeled with its proper diagnosis. The dataset is then randomly split into 80% training and 20% testing, with Stratified Group K-Fold ensuring that images from the same subject are grouped together. Our data set consists of 1551 images.

During model training, Stratified Group K-Fold is utilized to prevent data leakage and keep images from the same subject in the same fold. An outer loop of five folds is used to estimate performance, while an inner loop of three folds is used to tune hyperparameters.

Three binary classifiers (NC vs MCI, NC vs AD, MCI vs AD) are made and evaluated. These classifiers are used to build a pairwise system that is then combined using the above methodology to perform multiclass probability estimates.

## Implementation Details

All MRI feature vectors move through the same preprocessing pipeline. Zero-variance features are removed to clean the data, then standardization is applied to convert all features to zero mean and unit variance. Dimensionality is then reduced with PCA. The number of dimensions it is reduced to is determined by each classifier during hyperparameter tuning.

After preprocessing, each classifier is modeled using an SVM with an RBF kernel. Hyperparameters are optimized using grid search and cross validation. Synthetic Minority Over-Sampling Technique (SMOTE) oversampling is also applied to each training fold to aid with class imbalance. When training is complete, sigmoid based calibration is used to convert the classifiers into probability predictors.

The three calibrated binary probability predictors are combined with pairwise coupling to determine the overall most probable diagnosis for any given image. MCI decision boundary shifting and multipliers are applied to aid with MCI ambiguity.

## Major Results

As seen in Table 1, each binary classifier performs well with metrics being in their predicted range.

| Comparison | Accuracy | AUC | Sensitivity | Specificity | F1-Score |
|---|---|---|---|---|---|
| **NC vs MCI** | 0.667872 | 0.722778 | 0.395709 | 0.859942 | 0.477875 |
| **NC vs AD** | 0.804809 | 0.885175 | 0.763742 | 0.841388 | 0.787457 |
| **MCI vs AD** | 0.704832 | 0.753465 | 0.772687 | 0.617057 | 0.747717 |

*Table 1- Binary Classifier Comparisons*

Of all three classifiers, NC vs MCI performs the worst. This is due to the high similarity between NC and MCI images and MCI's ambiguous nature. Our goal for the NC and MCI comparison was at least 65% accuracy, which the model achieved. Though the F1-score is relatively low, the accuracy and AUC scores meet expectations signifying that model is useful and usable.

The goal for the NC vs AD and MCI vs AD comparisons was to reach at least 70% accuracy, which both models achieved. MCI vs AD has the same issue as NC vs MCI, which is MCI's ambiguous nature. When compared to AD, however, the results are more accurate due to more obvious differences between MCI and AD.

NC vs AD performed the best overall, achieving 80% accuracy. This classifier was able to perform the best due to the drastic differences between NC images and AD images.

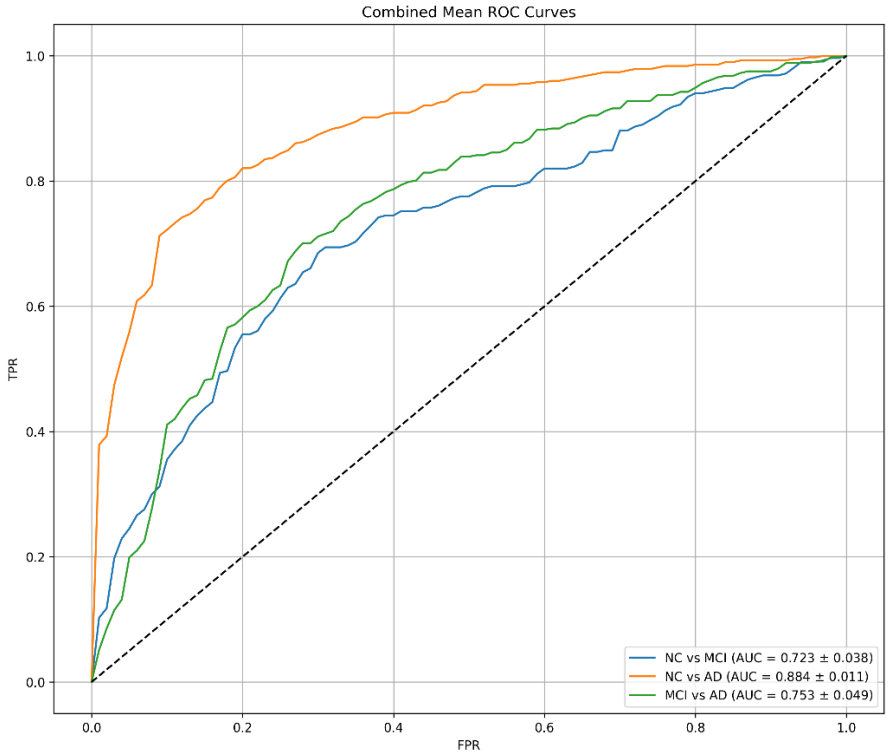



*Figure 1- Classifier ROC Curves*

As seen in Figure 1, each classifier performs well above the base 0.5 AUC rate. NC vs MCI and MCI vs AD have similar curves due to MCI's nature. The NC vs AD model predictably performs much better than models involving MCI due to stark differences in images and reduced ambiguity.

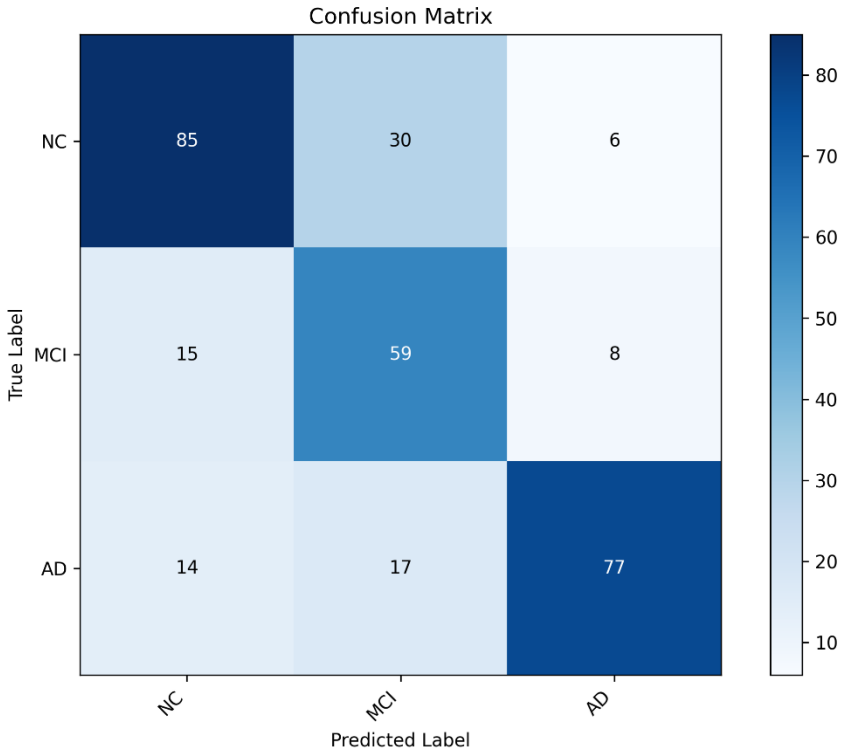



*Figure 2 - Testing Confusion Matrix*

Figure 2 shows the confusion matrix of the testing data. NC and AD are easiest to diagnose and are most often confused for MCI. MCI sits between NC and AD, so it understandable that ambiguity would cause it to be falsely predicted. NC is most often falsely predicted as MCI because of their strong similarities, while AD is confused as NC or MCI an almost equal amount. Probability adjustments for MCI increased its prediction accuracy, but may have had some minor negative impacts on NC detection.

Based on the CM in Figure 2, the overall accuracy of the model is 71%.

## Discussion

Our goal for this project was to reach at least 70% overall accuracy when predicting diagnosis labels of MRI images. We were able to reach this goal using an accurate, calibrated SVM model.

Through our research, we determined that separating each comparison into three binary models was highly effective and made creating decision boundaries between labels much stronger. Pairwise modeling and coupling was highly effective for separating each label, then combining their probabilities to calculate a final diagnosis label.

We also determined that MCI remains highly ambiguous, which was expected prior to experimentation. Adjusting the boundary and boosting MCI's probability significantly helped detection without compromising NC or AD detection. Because we used a standard SVM model, MCI prediction accuracy for our model can likely not be improved without implementation of more complicated models.

Overall, our model demonstrates strong potential to diagnose AD in clinical patients. While our model is not accurate enough to completely declare a diagnosis, it could serve as a way to support clinicians catch AD and MCI development early. Predictions of MCI and AD in patients through our model would encourage clinicians to take second looks at MRIs or reconsider factors that might have been missed in patients thought to have no impairment.

# References


1. Ebrahimi, Amir, and Suhuai Luo. "Convolutional neural networks for Alzheimer's disease detection on MRI images." *Journal of Medical Imaging*, vol. 8, no. 2, 29 Apr. 2021, p. 024503. DOI: 10.1117/1.JMI.8.2.024503.
2. Shahbaz, Muhammad, et al. "Classification of Alzheimer's Disease using Machine Learning Techniques." *In Proceedings of the 8th International Conference on Data Science, Technology and Applications (DATA 2019)*, 2019, pp. 296-303.
3. Sarraf, Saman, and Ghassem Tofighi. "Classification of Alzheimer's Disease Structural MRI Data by Deep Learning Convolutional Neural Networks." *arXiv preprint arXiv:1607.06583*, 2016.
4. Miltiadous, Andreas, et al. "DICE-Net: A Novel Convolution-Transformer Architecture for Alzheimer Detection in EEG Signals." *IEEE Access*, vol. 11, 12 July 2023. DOI: 10.1109/ACCESS.2023.3294618.
5. Yamanakkanavar, Nagaraj, Jae Young Choi, and Bumshik Lee. "MRI Segmentation and Classification of Human Brain Using Deep Learning for Diagnosis of Alzheimer's Disease: A Survey." *Sensors*, vol. 20, no. 11, 7 June 2020, p. 3243. DOI: 10.3390/s20113243.
6. Alsubaie, Mohammed G., Suhuai Luo, and Kamran Shaukat. "ConvADD: Exploring a Novel CNN Architecture for Alzheimer's Disease Detection." *[Source title not explicitly provided in excerpt]*, vol. 15, no. 4, 2024.
7. El-Assy, A. M., et al. "A novel CNN architecture for accurate early detection and classification of Alzheimer's disease using MRI data." *Scientific Reports*, 4 Feb. 2024. DOI: 10.1038/s41598-024-53733-6.