Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας

Κατασκευή νευρωνικού δικτύου για τη μελέτη της ετερογένειας στη γήρανση του εγκεφάλου με χρήση γενετικών και απεικονιστικών δεδομένων

Μελέτη και εφαρμογή

Διπλωματικη Εργασια

του

ΑΪΔΙΝΗ Ν. ΓΕΩΡΓΙΟΥ

Επιβλέπουσα: Κωνσταντίνα Νικήτα
Καθηγήτρια

Αθήνα, Ιούνιος 2022

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας

Κατασκευή νευρωνικού δικτύου για τη μελέτη της ετερογένειας στη γήρανση του εγκεφάλου με χρήση γενετικών και απεικονιστικών δεδομένων

Μελέτη και εφαρμογή

Διπλωματικη Εργασια

του

ΑΪΔΙΝΗ Ν. ΓΕΩΡΓΙΟΥ

Επιβλέπουσα: Κωνσταντίνα Νικήτα
Καθηγήτρια

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 15η Ιουλίου 2022.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Κωνσταντίνα Νικήτα	Ανδρέας Γεώργιος Σταφυλοπάτης	Γεώργιος Στάμου
Καθηγήτρια	Καθηγητής	Καθηγητής

Αθήνα, Ιούνιος 2022

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ενυπογράφως ότι είμαι αποκλειστικός συγγραφέας της παρούσας Πτυχιακής Εργασίας, για την ολοκλήρωση της οποίας κάθε βοήθεια είναι πλήρως αναγνωρισμένη και αναφέρεται λεπτομερώς στην εργασία αυτή. Έχω αναφέρει πλήρως και με σαφείς αναφορές, όλες τις πηγές χρήσης δεδομένων, απόψεων, θέσεων και προτάσεων, ιδεών και λεκτικών αναφορών, είτε κατά κυριολεξία είτε βάσει επιστημονικής παράφρασης. Αναλαμβάνω την προσωπική και ατομική ευθύνη ότι σε περίπτωση αποτυχίας στην υλοποίηση των ανωτέρω δηλωθέντων στοιχείων, είμαι υπόλογος έναντι λογοκλοπής, γεγονός που σημαίνει αποτυχία στην Πτυχιακή μου Εργασία και κατά συνέπεια αποτυχία απόκτησης του Τίτλου Σπουδών, πέραν των λοιπών συνεπειών του νόμου περί πνευματικών δικαιωμάτων. Δηλώνω, συνεπώς, ότι αυτή η Πτυχιακή Εργασία προετοιμάστηκε και ολοκληρώθηκε από εμένα προσωπικά και αποκλειστικά και ότι, αναλαμβάνω πλήρως όλες τις συνέπειες του νόμου στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής άλλης πνευματικής ιδιοκτησίας.

(Υπογραφή)

.....
Γεώργιος Αϊδίνης

15 Ιουλίου 2022

Περίληψη

Η νόσος Αλτσχάιμερ αποτελεί αντικείμενο ολοένα και περισσότερων μελετών, αφού αποτελεί μια από τις σημαντικότερες νευροεκφυλιστικές ασθένειες. Η χρήση υπολογιστικών μεθόδων για την διάγνωση, την μελέτη αλλά και την αντιμετώπιση γνωρίζει ραγδαία ανάπτυξη, και εφαρμόζονται ως επί το πλείστον μέθοδοι Μηχανικής Μάθησης για την αποτύπωση των δεδομένων, την επεξεργασία και τον μετασχηματισμό τους, αλλά και την κατηγοριοποίηση τους. Οι σύγχρονες μέθοδοι χρησιμοποιούν πολυτροπικά δεδομένα, με την έμφαση να δίνεται στα απεικονιστικά και στα γενετικά δεδομένα. Στο πλαίσιο της παρούσας Διπλωματικής Εργασίας διερευνούνται εκτενώς διάφορες μέθοδοι ανάλυσης δεδομένων, μηχανικής μάθησης αλλά και βαθειάς νευρωνικής μάθησης, καθώς και οι μεταξύ τους συνδυασμοί. Το πρόβλημα που μελετάται είναι αυτό της κατηγοριοποίησης δεδομένων από το σύνολο δεδομένων Alzheimer's Disease Neuroimaging Initiative σε πάσχοντες από νόσο του Αλτσχάιμερ, άτομα με ήπια νοητική διαταραχή, και φυσιολογικά. Το σύνολο δεδομένων περιέχει απεικονιστικά αλλά και γενετικά δεδομένα από 1567 συμμετέχοντες. Οι μέθοδοι ανάλυσης δεδομένων που εξετάζονται είναι οι Deep Canonical Correlation Analysis, Multiple Correspondence Analysis, Orthonormal Projective Non-Negative Matrix Factorisation και Factor Analysis of Mixed Data. Οι μέθοδοι που χρησιμοποιήθηκαν για την κατηγοριοποίηση είναι τα Support Vector Machines καθώς και μέθοδοι Ensemble Classifiers. Για κάθε πιθανό συνδυασμό, τα μοντέλα αυτά αξιολογήθηκαν ως προς την απλή ακρίβειά τους, το F1 Score, και την εξισορροπημένη ακρίβειά τους. Παρατίθονται τα αποτελέσματα, σχολιασμός των συγκρίσεων, συμπεράσματα καθώς και μελλοντικές επεκτάσεις.

Λέξεις Κλειδιά

Νόσος Αλτσχάιμερ, Ήπια Νοητική Διαταραχή, Βαθειά Νευρωνική Μάθηση, Μηχανική Μάθηση, Data Analysis, Classification, Deep Canonical Correlation Analysis, Non-Negative Matrix Factorization, Correspondence Analysis

Abstract

Alzheimer's Disease (AD) is subject to an increasing number of studies, since it is one of the most important neurodegenerative diseases. The use of computational methods for the diagnosis, studying and treatment of the disease has enjoyed rapid growth, while presently, mostly Machine Learning are applied for the visualization, processing, transformation and classification of the data related to the disease. Modern methods utilize multi modal data, with the focus being on imaging and genetic modals. In this study, a multitude of Data Analysis, Machine and Deep Learning methods are extensively studied, as well as the combinations thereof. The subject of the task is that of classifying data from the Alzheimer's Disease Neuroimaging Initiative dataset into AD patients, Mild Cognitive Impairment patients, and Cognitive Normal people. The dataset contains imaging as well as genetic data from 1567 participants. The Data Analysis methods that were studied were those of Deep Canonical Correlation Analysis, Multiple Correspondence Analysis, Orthonormal Projective Non-Negative Matrix Factorisation and Factor Analysis of Mixed Data. The classification methods that were used were those of Support Vector Machines, and Ensemble Classifier methods. For every combination of the aforementioned methods, the models were evaluated on their Accuracy, their F1 Score and their Balanced Accuracy. The results of the study are presented, commented on, conclusions are drawn and future directions are discussed.

Keywords

Alzheimer's Disease, Mild Cognitive Impairment, Deep Learning, Machine Learning, Data Analysis, Classification, Deep Canonical Correlation Analysis, Non-Negative Matrix Factorization, Correspondence Analysis

στους γονείς μου

Acknowledgements

Θα ήθελα καταρχήν να ευχαριστήσω την καθηγήτρια κα. Νικήτα και τον καθηγητή κ. Νταβατζίκο, για την επίβλεψη αυτής της διπλωματικής εργασίας καθώς και για την δυνατότητα που μου έδωσαν να την εκπονήσω στο εργαστήριο Βιοϊατρικών Προσομοιώσεων και Απεικονιστικής Τεχνολογίας σε συνεργασία με το Center for Biomedical Image Computing and Analytics (CBICA) του UPenn. Χωρίς την συμβολή τους, η εργασία αυτή δεν θα ήταν δυνατή.Επίσης ευχαριστώ ιδιαίτερα την υποψήφια Διδάκτωρ Ιωάννα Σκαμπαρδώνη για την καθοδήγησή της, την πολύτιμη γνώση της και την εξαιρετική συνεργασία που είχαμε. Τέλος θα ήθελα να ευχαριστήσω τους γονείς μου, τον παππού μου και την γιαγιά μου, και την αδερφή μου και τους φίλους μου για την συμπαράσταση που μου προσέφεραν όλα αυτά τα χρόνια.

Αθήνα, Ιούνιος 2022

Γεώργιος Αϊδίνης

Περιεχόμενα

Κατάλογος Σχημάτων

Κεφάλαιο 1

Introduction

A rapidly growing cause of death in the developed countries are neurodegenerative diseases. Diseases such as the Alzheimer's, as well as disorders such as Mild Cognitive Impairment are most prevalent on older people, with 65 years of age and on wards being the group that's most affected by them. It is estimated that in the US alone, 6.5 million people are suffering from Alzheimer's Disease today, while globally that number can be as high as 35 million. Those estimations are expected to grow to 135 million globally, and many of them are undiagnosed and or untreated even today. (1,2)

During the normal course of aging, the human brain displays changes, both anatomical as well as functional, that seem to be accelerated in patients of such diseases. The human brain is developed until the age of 25 years, and after that it continuously loses neural mass, leading to brain atrophy, a condition that is studied extensively, thanks to advances the field of medical imaging. The atrophy effect is sped up in some patients quite significantly, and in the case of the Alzheimer's Disease, it leads to neuronal decay and eventually death.

The effects of the Alzheimer's Disease vary quite significantly, and are apparent not only clinically, but in imaging scans and genetic surveys as well. Because of low awareness, Mild Cognitive Impairment and the Alzheimer's disease are often mistakenly associated with getting older, while due to the need of an experienced practitioner for the diagnosis, many cases go undiagnosed. However, a great deal of research is being done on studying the symptoms, potential causes, as well as the treatment of the disease, with some estimates putting the total cost of Alzheimer's research in the tens of billions USD. Furthermore, for the total cost for the healthcare related to the Alzheimer's disease patients for the year 2020 in the US has been estimated to be around 300 billion USD. (2, 3).

Despite the considerable resources, no clear cause has been found for the Alzheimer's Disease, no definitive prevention method has been found, and no treatment method has been widely successful.

1.1 Contents of this Thesis

To tackle the problem of diagnosing Alzheimer's as well as MCI, and also predicting and modelling their respective courses, a plethora of studies have been published. A common characteristic that many have is the use of Machine and Deep Learning methods, especially on Neuroimaging, Genetic and Clinical data collected from patients of the diseases. The

main drivers behind this effort are the advances in medical imaging and the staggering growth of computational abilities in recent years, making more complex and better methods applicable and practical. (4, 5)

Most studies employ Neuroimaging data collected with the MRI and PET methods, that were collected from the Alzheimer's Disease Neuroimaging Initiative database. The imaging data are often accompanied with biomarkers, especially in the context of Single Nucleotide Polymorphisms, and even data from Clinical tests performed by licensed practitioners. (6,7,8).

While in general more complex models employing different modalities of the data performed better, there is no unified approach, and the diversity of models is intriguing. However, most studies focus on either classifying whether or not a subject is a patient of Alzheimer's Disease and predicting the course of the disease, while some papers focus on predicting if a patient with MCI will deteriorate to Alzheimer's. (4)

A common problem most studies had to overcome was the 'Curse of Dimensionality' problem that is associated with Biomedical data. The data is characterised by very large dimensions (especially in the case of the neuroimaging view), yet not enough samples. This is due to the imaging techniques post-collection data processing resulting in very high dimensions, while the subject count being very low, due to the difficult, sometimes inaccessible, expensive and lengthy nature of the technique. This has the adverse consequence of data not being easily visualised (since they cannot be interpreted in the three dimensional space humans are familiar with), along with the tendency of the models to overtrain and overfit on the low number of datapoints. To avoid the aforementioned problems, data analysis techniques have been used, with them being as simple as PCA, or as complex as employing deep learning, for example Neural Networks. (9,10, 11, 12)

This study attempts to create a comparative analysis of Machine Learning methods and algorithms being applied to the problem of predicting whether a subject is Cognitive Normal, has Mild Cognitive Impairment, or has Alzheimer's Disease. The data we used was collected through the ADNI dataset, and more specifically, imaging data taken from MRI scans, along with genetic biomarkers in the form of Single Nucleotide Polymorphisms.

In this thesis, we also explore Data Analysis techniques in order to tackle the Dimensionality Curse problem, as well as Transformations and Statistical Analysis methods. To convert the exceedingly dissimilar imaging and genetic views, we experimented with a novel technique, called Deep Canonical Correlation Analysis, where Neural Networks are used to learn a transformation of the different views' features into a hyperspace that is more linearly correlated, and thus potentially easier to perform the classification task.

Furthermore, we experimented with dimensionality reduction methods, such as Multiple Correspondence analysis, Orthonormal Projective Non-Negative Matrix Factorization, and Factor Analysis of Mixed Data. All of the possible combinations of the techniques were used, and their effect on the classification task was compared.

For the classification task in particular, we present the results from applying Support Vector Machines, which are widely regarded as a fairly simple, understandable, yet practical and capable model, as well as Ensemble Learning methods such as Bagging and Adaboost to the problem. For the ensemble methods, we experimented with Decision Trees and again

Support Vector Machines as base model classifiers, and the yielded results were compared not only between them, but also to the single-classifier results.

1.2 Structure of the Thesis

This work is divided into 6 chapters, apart from the introduction and the epilogue. Chapter 2 introduces some theoretical knowledge in order to tackle the problem described in the introductory chapter, pertaining to the human brain and some fundamentals about the Machine and Deep Learning methods that were used. Chapter 3 explains the methodology that was used, with a brief analysis of the dataset that was used, the parameters of the methods, as well as the metrics to evaluate them. On Chapter 4, we present some results from the attempt at optimizing the models that were later used for the classification tasks, and on Chapter 5 the classification results are presented. Finally, on Chapters 6 and 7, conclusions are drawn from observations made on the results, future extensions and directions are discussed, as well as the practical limitations of this work.

Κεφάλαιο 2

Theoretical Background

2.1 Human Brain

2.1.1 Aging

As the human body ages, all organs experience age-related effects, and so does the brain. Both physiologically and cognitively, there are several changes that can be observed as part of the normal brain aging process. (1)

Cognitively, memory (specifically Episodic and Semantic memory) is one of the core areas that are affected. Older people may be forgetting names of items or persons, having to repeat questions, misplacing items, getting lost, and having trouble recalling information in general. Additionally, language skills such as vocabulary and language skills may be affected, as well as the ability to learn new skills and multitask. (2)

Physiologically, the brain shrinks in the areas of the frontal lobe as well as the hippocampus, the areas that are generally thought to be linked with higher cognitive function and memory, at a rate of 5% per decade after the age of 40. This effect is due to the grey matter shrinking, which is attributable to neuronal cell death. Additionally, cortical density decline is observed, meaning that the outer surface of the brain is becoming thinner. This effect is more pronounced in the frontal and temporal lobes. White matter also declines, with the myelinating regions of the frontal lobe being most affected by white matter lesions. Finally, the levels of neurotransmitters such as dopamine and serotonin see a steep decrease, an effect that has been associated with declines in cognitive and motor performance. (3,4)

While the aforementioned symptoms are very much similar to the symptoms that Mild Cognitive Impairment and Alzheimer's Disease exhibit, normal aging patterns of decline are divergent from the ones of MCI and AD. The effects of normal brain aging are characterized by their occasional nature, while MCI's and AD's ones are more consistent, and gradually worsening in some cases, and they are accompanied by other dementia symptoms, such as confusion, mood changes and others. Furthermore, the physiological changes of the brain are much more pronounced and significantly more severe. MCI and AD have a much more noticeable effect on the person's daily life, and some cases need assistance in order to normal daily tasks. (5)

2.1.2 Mild Cognitive Impairment

Mild Cognitive Impairment is a state of a person that is characterized by problems with memory, language, thinking or judgment. It is usually observed between the stage of normal cognitive decline that happens to humans due to aging and the dramatic fall in cognition that is apparent to people with Dementia. People with MCI have memory loss or other cognitive ability loss, exceeding the normal decline due to age and are not demented.

MCI's symptoms can manifest in many different functionalities of the human brain, including weaker memory, poor reasoning and judgment skills, visual perception and others. Frequently, MCI coexists with other illnesses or emotions, such as depression, anxiety, irritability and aggression, or apathy.

The cause of the disorder is unknown; it is believed that MCI is caused by the same mechanisms that are thought to be responsible for the neuropathology of the early stages of Alzheimer's Disease, however that is unproven. Risk factors include age, family history of AD or dementia, genetic factors, and other medical conditions such as Diabetes, high blood pressure, smoking, obesity, depression etc.

People with diagnosed MCI have a significantly higher chance than that of cognitive normal population to develop Alzheimer's Disease or some form of Dementia. Despite the fact that there is no standardized test for MCI, clinical characterization is achieved through the results of various tests (such as mental tests, neurological exams, brain imaging and searching for biomarkers) and the informations that the patient provides.

It is not exactly clear how to prevent MCI, but studies show that engaging in frequent physical activity, maintaining a healthy and balanced diet, engaging socially with others, being mentally active, reducing alcohol and not smoking may be mitigating factors to the risk of developing the condition.

2.1.3 Alzheimer's Disease

Alzheimer's disease is a neurodegenerative disease that affects the brain both biologically and cognitively. It was first reported by A. Alzheimer in 1906, but described as a disease only after 1910, by E. Kraepelin. Alzheimer's Disease is the most common cause of Dementia, a term used to describe a group of symptoms that include decline in memory, reasoning, and other thinking skills. It is reported that AD is responsible for about 70% of Dementia cases. (1,2,3)

AD is a progressive condition, meaning the symptoms gradually appear and worsen over time. Early symptoms include short term memory loss, decline in conversational abilities and poor reasoning. As the disease progresses, patients have trouble recalling names, may have confusion and obsessive, repetitive or impulsive behaviour, serious problems with speaking and the use of language, and generally problems that require external assistance in their daily life. In later stages of the disease, the patients have trouble with even the most basic tasks, such as eating and moving, and require full time assistance. Gradually, the condition of the patients deteriorates, ultimately leading to death. (4)

The cause of AD is unknown, but genetic and environmental risk factors have been implicated. AD is linked with the formation and buildup of plaques (abnormal clusters of protein fragments) of the protein Amyloid β, and neurofibrillary tangles (twisted strands of protein) of the tau protein. There are several hypotheses as to the disease's origin, yet none of them have been confirmed. There are two perhaps significant hypotheses, the Amyloid and the Cholinergic Hypothesis. (5)

AD is a multifactorial disease, being associated with several risk factors, such as age and gender, genetic factors, life style, coexistent or previous diseases, head injuries and environmental factors. The most important however is age, with most AD cases having a late onset that starts after 65 years of age. Normal brain aging is characterized by a reduction in brain volume and weight, a loss of synapses, and the enlargement of ventricles. These changes appear in AD patients as well, but more profound in general. There are two categories of AD based on the age that it appears, Early Onset AD which is generally Familial and displays inheritance and has onset age that ranges from 30 to 60 years of age (1-6% of cases), and Late Onset AD, which is by far more common and has age of onset above 65 years. Genetic factors also play a significant role, with 70% of AD cases being related to genetic factors. The genes APP, PSEN-1, PSEN-2 and most importantly ApoE are associated with AD. (5,6)

In order to successfully diagnose the disease, a practitioner has to evaluate the person, with multiple tests if necessary. The diagnosis is based on medical history, advanced medical imaging of the brain (using CT or MRI or PET or SPECT), mental tests such as the MMSE, blood tests and psychological tests for depression, since depression can either be concurrent with Alzheimer's disease, an early sign of cognitive impairment, or even the cause. (6,7)

There's currently no cure for Alzheimer's disease, however there are certain medications available that can temporarily mitigate the symptoms. Since the cause of the disease is still unknown, there is no designated preventive roadmap. Despite that, frequent physical

exercise, a healthy and balanced diet as well as staying mentally and socially active all have been linked to lower rates of AD.

2.1.4 Fundamentals of MRI (Imaging - ROIs)

A big part of studying, understanding and diagnosing neurodegenerative diseases are brain imaging techniques, such as Magnetic Resonance Imaging (MRI), Positron Emission Tomography (PET), Computerized Tomography (CT) and others. This work focuses on data collected with the method of MRI, which were used to recognize signs of Mild Cognitive Impairment and Alzheimer's Disease.

The Magnetic Resonance Imaging technique works by measuring the energy signal of (typically) hydrogen nuclei, as a result of excitation by external radio frequency pulses. The MRI technique is frequently split into two different processes, based on the decay of the RF-induced nuclear magnetic resonance spin polarization, named T1 and T2, each producing different results depending on the signal and the tissue being imaged. Depending on the parameters of the process being used, MRI can imprint pictures of the anatomy of the human body, as well as its physiological processes. (1)

Oftentimes, to avoid examining every single point of space (commonly referred as a 'voxel' or volume-pixel), the scan is focused in specified regions of interest (ROIs). These regions are produced by segmenting the original image, either automatically, using Machine Learning methods or by employing previously computer computed brain atlases. Specifying however, the aforementioned regions, is quite complex, since there is a great variability of neuroanatomy between humans. It is believed that the brain's function is associated with structural and functional connectivities, and therefore identifying standardized and reliable is crucial for understanding the connection between the architecture of the brain and its function. (2, 3, 4)

Because the MRI method produces a tremendous amount of data for each scan, frequently before the main task, preprocessing is applied to capture the desirable information, while maintaining ease of data manipulation. Such methods include data augmentations, feature selection, dimensionality reduction, etc.

2.1.5 Fundamentals of genetics (SNPs)

DNA in humans is arranged in chromosome pairs, with each cell having in its nucleus 23 pairs, 22 of which are autosomes and 1 pair being the sex chromosomes. In each of the 22 pairs of chromosomes, DNA is stored in identical copies, with specific chunks being characterized as genes. A gene contains genetic information in the form of long sequences of nucleotides (A,T,G,C).

If a change or variation in one or more nucleotide positions of a chromosome in the DNA sequence is found, it is called a Single Nucleotide Polymorphism (SNP - pronounced "snip") . If most humans have a specific certain nucleotide in a specific position of the genome, and a SNP occurs in some individuals in that exact position, then this position is said to have more than one allele, meaning more than one variation. Because the DNA is stored in pairs of chromosomes, a person's DNA can contain in a specific position of the genome one SNP, two SNPs, or no SNPs at all. (1,2)

Observed SNPs can be associated with a disease, however, it may not always directly be the cause for that disease. An example of this is the APOE gene (chromosome 19 position q13.32), which has been determined to be a risk factor for Alzheimer's Disease, specifically the ε4 allele (variation). There are three versions of the gene in humans, ε2, ε3, and ε4, with ε3 being the most prevalent, the existence of the ε4 variant being a risk factor, while having two ε2 alleles being associated with lower probabilities of developing the disease. The disease however is also associated with other gene mutations, such as mutations in the genes APP, PSEN1, PSEN2 and others, which especially influence the early onset variant of the disease. (2,3,4)

Recognizing the different SNPs that are contained in the human genome is the topic of studies such as Genome Wide Association Studies (GWAS), which are a part of the field of Bioinformatics. Understanding the changes in the genome can help recognize how they translate in the phenotype, help with their treatment and their prevention. (5)

2.2 Fundamentals of the Machine Learning and Deep Learning methods

2.2.1 Canonical Correlation Analysis

Canonical Correlation Analysis is a standard tool of multivariate statistical analysis used to discover and quantify associations between two sets of variables. The aim of this method is to find a transformation (projection) of the two sets of variables, such that they are maximally associated (measured by correlation). The projections are found by performing a joint covariance analysis of the two variables. (1)

This concept was introduced by C. Jordan (1875), but the method was initially described by H. Hotelling (1936). It has been used extensively in many fields, such as Economics, Medicine, Psychology, etc., and has many extensions, such as the Kernel Canonical Correlation Analysis. (2,3,4)

Let $X \in \mathbb{R}^q$, and $Y \in \mathbb{R}^p$, two random vectors, and their respective covariances Σ_{11} and Σ_{22}, as well as the cross covariance Σ_{12}. The aim of CCA is to find vectors , such that the correlation $\rho(a, b) = corr(a^\intercal X, b^\intercal Y)$ is maximised.

The correlation $\rho(a, b)$ can also be written as follows:

$\rho = \frac{a^\intercal \Sigma_{XY} b}{\sqrt{a^\intercal \Sigma_{XX} a}\sqrt{b^\intercal \Sigma_{YY} b}}$

This is achieved by setting the a, b parameters as follows:

$a = \Sigma_{XX}^{-\frac{1}{2}} c$, and

$b = \Sigma_{YY}^{-\frac{1}{2}} d$, where

c is an eigenvector of $\Sigma_{XX}^{-\frac{1}{2}} \Sigma_{XY} \Sigma_{YY}^{-1} \Sigma_{YX} \Sigma_{XX}^{-\frac{1}{2}}$,

and d is an eigenvector of $\Sigma_{YY}^{-\frac{1}{2}} \Sigma_{YX} \Sigma_{XX}^{-1} \Sigma_{XY} \Sigma_{YY}^{-\frac{1}{2}}$.

The vectors a, b are called canonical correlation vectors, and the indices $u = a^\intercal X$ and $v = b^\intercal Y$ are called canonical correlation variables.

This process may be repeated $min(q, p)$ times, and find subsequent projections. However, the new vectors are subject to the constraint that they are to be uncorrelated with the previous ones, that is $a_i \Sigma_{XX} a_j = b_i \Sigma_{YY} b_j = 0$, $\forall i < j$.

CCA is implemented using Singular Value Decomposition on the correlation matrix.

2.2.2 Deep Canonical Correlation Analysis

Deep Canonical Correlation Analysis is an extension of the standard Canonical Correlation Analysis, created by G. Andrew, R. Arora, J. Bilmes and K. Livescu. As with normal CCA, DCCA is a method that aims to discover and learn associations between two sets of variables.

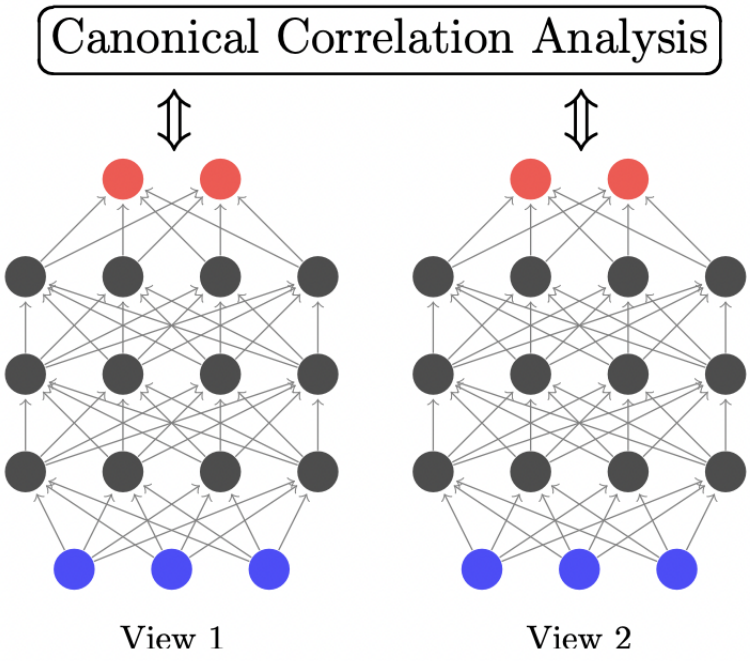
In the case of DCCA, the method can learn complex, nonlinear relations between the two random vectors, whereas the standard CCA cannot. The method is similar to the idea of Kernel CCA (2), where the optimal projections are found on the kernel-transformed random vectors, such that the resulting Reproducing Kernel Hilbert Space contains the variables in a manner that CCA can be impactful.

However, the problem with KCCA is the computation complexity, as the kernel matrices become very large for real-world datasets, meaning that since it is a nonparametric method, the time required to learn the transformation scales poorly with the size of the data (3). Additionally, the KCCA method is also limited to the choice of the fixed kernel, meaning they can't be flexible for different types of datasets.

To address these drawbacks, the use of Deep Neural Networks is proposed, in order to simultaneously learn two deep nonlinear mappings of two random variables. In their paper, they focus on the performance metric of achieved correlation, and comparing it to the correlation of the standard method.

The use of Deep Learning, meaning Neural Networks with more than two layers, is designated, since Deep Neural Networks have been proven to be capable of representing accurately and reliably nonlinear functions that model complex real world data. The method is used to correlate different views of the same dataset, for example different modalities of a biomedical dataset.

The method relies on passing each random vector through a neural network, designed and trained to transform the random vector nonlinearly, such that the mapping to a hyperspace that results is better correlated to the mapping of the respective (transformed) random vector.



Σχήμα 2.1: *The two parallel networks, along with the information path (arrows)*

In the figure above, the two Neural Networks are shown, consisting of 5 layers, with the red layer being the output layer, meaning the vectors are maximally correlated, and the blue layer being the input layer, meaning the original random vectors.

If θ_1 is the vector of all parameters (W_i^1, b_i^1) of the first network, for each layer i, and respectively θ_2 is the vector of all parameters (W_i^2, b_i^2) of the second network for each layer, then the training goal is equivalent to finding the optimum parameters such that the correlation of the output of the networks $f_1(X_1; \theta_1), f_2(X_2; \theta_2)$ given two random vectors (views) (X_1, X_2).

That is described as follows:

$$(\theta_1^*, \theta_2^*) = \text{argmax}_{\theta_1, \theta_2}\{corr(f_1(X_1; \theta_1), f_2(X_2; \theta_2))\}$$

Supposing that $H_1, H_2 \in \mathbb{R}^{o \times m}$ are the matrices that contain the respective outputs of the Neural Networks, for each of the training samples. The target then becomes $corr(H_1, H_2)$.

To train the Networks, the computation of the gradient is needed, and its backpropagation in order to tune the networks parameters. The target is found using the same steps as the standard CCA, while the computation of the gradient, as well as its backpropagation is facilitated through Singular Value Decomposition.

The authors of the original paper employed full-batch optimization, meaning that before every single weight update step, the network scanned the full dataset. They also used the Limited Memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) optimization method (4). In order to initialise the parameter optimization for the two networks, they utilised a Denoising Autoencoder for each layer of the networks (5). The network the method proposes uses a non-saturating nonlinearity activation function, in the form of: If g :

$\mathbb{R} \to \mathbb{R}$, and $g(x) = \frac{x^3}{3} + x$, then the function $s(x) = g^{-1}(x)$ is the activation function, maintaining a sigmoid shape, and unit slope at $x = 0$.

2.2.3 Multiple Correspondence Analysis

Multiple Correspondence Analysis is a data analysis technique used to analyse the structure of a number of dependent categorical variables in a dataset. It is an extension of simple Correspondence Analysis, and is similar to the well known method of Principal Component Analysis. The method is equivalent to methods such as optimal scaling, optimal or appropriate scoring, dual scaling, homogeneity analysis, scalogram analysis, and quantification method. (2)

MCA is used when a dataset contains variables that are described by nominal values, such as "Male" and "Female", or "Red", "Green", "Blue", etc. The variables can also contain quantitative values, split into categories. MCA is performed on an indicator matrix - also called a Complete Disjunctive Table - or on a Burt Table. It can also be viewed as the PCA method applied to the CDT. (1)

Suppose there is a Dataset containing only categorical variables, and its corresponding Complete Disjunctive Table, X. Let K be the number of the nominal variables, and each nominal variable has J_K levels and the sum of the J_K is equal to J. There are I observations. Then the Table X is actually the $I \times J$ indicator matrix.

We indicate the sum of all entries to be N, and compute the probability matrix $Z = fracXN$. We also use the special vectors r, and c, which are the vector of the row totals of Z, and the vector of column totals of Z respectively.

Then, if

$D_c = diag(c), D_r = diag(r),$

we have the factor scores of the MCA are obtained from the following singular value decomposition:

$M = D_r^{-\frac{1}{2}}(Z - rc^\intercal)D_c^{-\frac{1}{2}} = P\Delta Q^\intercal,$

where Δ is the diagonal matrix of the singular values, and the matrix of the eigenvalues is $\Lambda = \Delta^2$. MCA decomposes the matrix into coordinates (or scores) of the factor space, which can be found as follows:

$F = D_r^{-\frac{1}{2}}P\Delta$, for the row coordinates and

$G = D_c^{-\frac{1}{2}}Q\Delta$ for the column ones.

2.2.4 Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) is a unsupervised, multivariate, analytic method for the approximate factorization of a matrix V into two matrices W,H under the constraint that their elements are non-negative: $V = WH$, such that $H \geq 0$ and $W \geq 0$. (1)

The method was created by P. Paatero and U. Tapper, and further developed by D. Lee and H. Seung (2,3). It is used to dimensionally reduce data, performing clustering tasks and find underlying structures within the dataset. Because the resulting factorization contains non-negative elements, the method has the advantage of better interpretability, and its ability to produce parts-based representation of the data, it has been applied in many different fields, such as Machine Learning, Computer Vision, Signal Processing, Data Mining, Medical Imaging etc. (4,5,6,7)

Lee and Seung's multiplicative update rule is the basis of the method's computation of the W and H matrices, and has the characteristics of being iterative and element based. However there are other ways, and it can be supplemented with additional constraints or regularizations, leading to many extensions.

One notable extension is that of Orthonormal Projective Non-Negative Matrix Factorization (OPNMF), where the loading coefficients are estimated as the projection of the matrix onto the estimated components W ($H = W^\intercal V$), while maintaining orthonormality on the estimated components ($W^\intercal W = I$). As a result, all components participate in the reconstruction of all of the data samples, meaning that the overlap between the estimated components is significantly lower, having fewer parameters to be learned, while maintaining high sparsity. Additionally, this variant relies on the original update rule (and thus is computationally easier than the Projective NMF variant), and at the same time is able to generalize on unseen data without the need of retraining. (4)

2.2.5 Factor Analysis of Mixed Data

Similar to MCA, Factor Analysis of Mixed Data is a data analysis technique used to analyse the structure of mixed data, meaning both continuous numerical as well as categorical data. It is also used to in order to reduce the number of dimensions of the dataset, and improve interpretability. It is based on the methods of Multiple Correspondence Analysis and Principal Components Analysis. (1)

Suppose there is a Dataset containing both quantitative (numerical, standardized) and qualitative (categorical) variables. Let K_1 be the quantitative variables, Q the qualitative variables, and K_q the categories of the q^{th} variable. We can denote the overall number of categories of the qualitative variables as:

$K_2 = \sum_q K_q$

Let $K = K_1 + K_2$ the total number of quantitative variables and indicator variables.

We assume that individuals have the same weight, and the diagonal metric of the weights of the individuals is:

$D = \frac{1}{I} I_d$

The quantitative variables are represented by a vector of length 1, and the qualitative ones by a cloud N_q of its centered indicators. FAMD aims to look for a direction of v that maximizes the inertia of the $\mathbb{R}^I$ cloud. That goal is perfectly achieved by maximizing the following criterion:

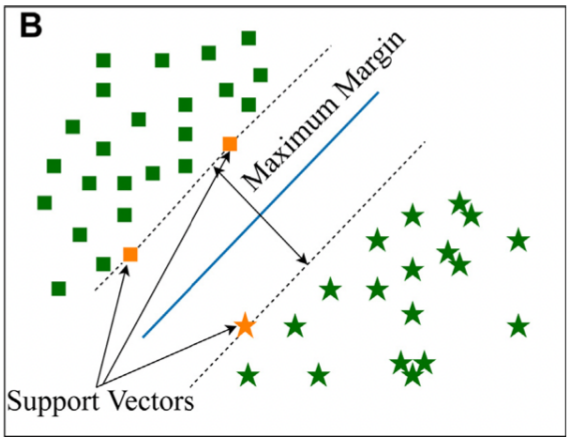
$\sum_{k \epsilon K_1} r^2(k, v) + \sum_{q \epsilon Q} \eta^2(q, v)$, where

$\eta^2(q, v)$ is the squared correlation ratio between q and v, and $r^2(k, v)$ is the squared projection coordinate of variable k on v. (2)

2.2.6 Support Vector Machines

Support Vector Machines are a family of models that are used for classification and regression analysis. The model was initially introduced by C. Cortes and V. Vapnik and has enjoyed considerable popularity since its creation, being one of the most widely used Machine Learning Techniques. They have been applied to many scientific fields, such as Pattern Recognition, Image Classification, Biomedical Research, Petroleum Exploration, etc.(1,2,3)

The goal of an SVM is to choose a hyperplane (e.g. a straight line in two-dimensional space, a plane in three-dimensional) that best separates a dataset consisting of samples that belong to one of two known classes. The method that SVMs rely on to achieve this goal is choosing two parallel hyperplanes that separate the two classes such that the distance between them (the margin) is maximal. The data points on the margin maximizing hyperplanes (supporting vectors) define the decision surface for the classification, as shown below: (4)



Σχήμα 2.2: *Support Vectors are shown in orange, the decision boundary in blue, the datapoints in green (squares for one class, stars for the other), and the dash lines are the margin maximizing hyperplanes. (4)*

SVMs can be used for multi-class classification as well, through a one-vs-one scheme or one-vs-rest approach, where decision boundaries are calculated between respective classes or a class and the rest of the dataset in each case. (5)

SVMs enjoy a variety of advantages, such as efficiency in both low and high dimensional spaces, memory efficiency, and being able to produce results in cases where the number of samples is less than the number of dimensions. (6)

However, one major drawback of SVMs is that the standard model does not work on datasets that are not linearly separable; that is one hyperplane cannot correctly divide the classes. To get around that, SVMs use kernels. Thus, the dataset is first non-linearly mapped through a kernel in a higher dimension space where the data is linearly separable, and then the original SVM algorithm is performed. In this approach, one can use the kernel trick to get around having to compute the transformations through the kernel function for the whole dataset, and only performing the minimum calculations needed. (4)

SMVs suffer from over-fitting issues, as well as being sensitive to parameter choices, such as the kernel function and regularization term choice, especially if the number of parameters is much greater than the number of available samples. Additionally, SVMs are not scale invariant, so scaling the dataset is highly recommended. Finally, k-fold Cross Validation is crucial, since SVMs do not inherently provide probability estimates. (6,7)

SVMs can also be used for clustering, where the process is called Support Vector Clustering and was created by H. Siegelmann and V. Vlapnik.(8)

2.2.7 Ensemble Learning

Ensemble Learning is a technique of combining a multitude of models to enhance the task to be performed, such as classification problems, regression or approximation tasks etc. This is achieved by applying the (perhaps different) models to the data available (or a subset thereof) and combine their outputs, in order to make a better attempt at solving the problem. (1)

An ensemble is created by combining either different models, or models with different parameter initializations and configurations. Such models can be relatively simple, such as Decision Trees, naive Bayes Classifiers, or Support Vector Machines, or more complicated, such as Multi Layer Perceptrons, or even other ensembles altogether. A key aspect of the base model selection is to create enough diversity of opinions, that is differentiation between the models themselves. (2)

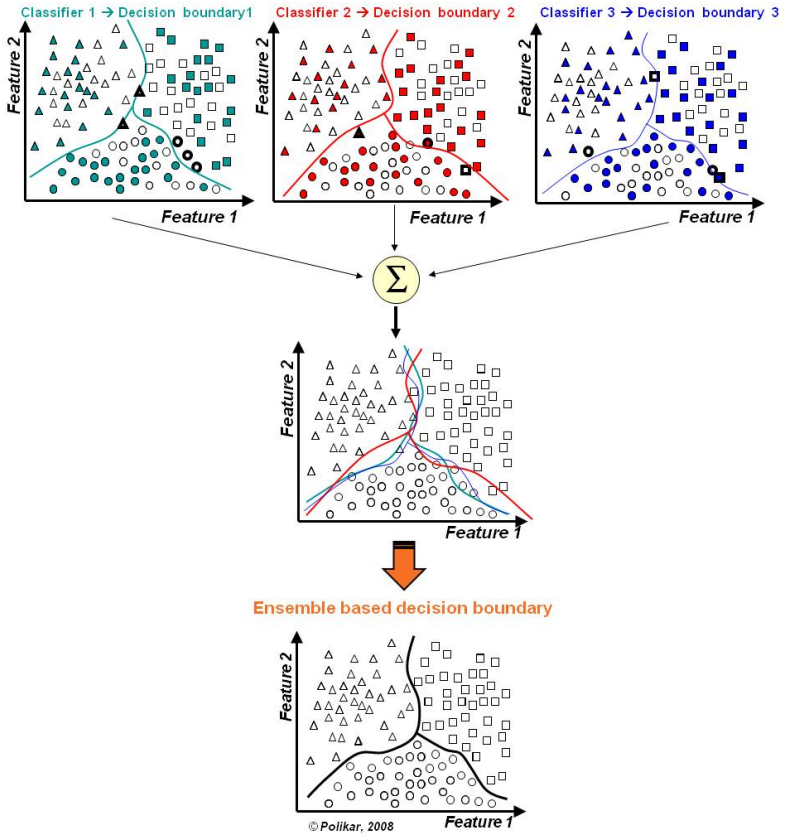
The Ensemble Learning technique relies on two concepts: the way the dataset is used to train the base models (how the data is introduced to the models) and the way that the outcome of each base model is considered towards the combined outcome.

The method that is employed for training the base models can be as simple as dividing the dataset by the number of models and feeding each subset to each model, or as strategically complex as to involve feature selection along with data augmentation during the phase of training the base models. Another approach might be introducing to different base models different views of the dataset. (2)

Respectively, the method employed for combining the outcomes of the base models can be as simple as simple majority voting (for example the most voted class in a classification problem) or algebraic combiners, or more sophisticated and tailored to a specific problem strategies.

The idea behind Ensemble Learning is to enhance the decision taken with group knowledge; that is to reduce the likelihood of an unfortunate selection. While that is not guaranteed, there is empirical evidence that Ensemble models achieve in general better worse case performance than that of single models, and in some case better than that of the average of their base models. (2)

Another motive for employing Ensemble methods is their ability to perform both in big data tasks and when there isn't adequate data for the successful training of a single model. The big data case is handled with dividing the dataset into many subsets, and training each model on a single subset, thus making the training phase much easier. On the other hand, with a strategy such as Bootstrapping, different base models can be trained on different combination of samples of data, taken from the initial dataset, with replacement, and treated as if they were independently drawn. (2,3)



Σχήμα 2.3: *Combining classifiers with different decision boundaries reduce error.*

One method of Ensemble Learning applied to the problem of classification is that of Bagging, or Bootstrap Aggregating, where the base models are trained on drawn samples from the initial dataset, with replacement, and the base models are classifiers of the same type. The individual classifiers' outcome is combined with the others in a simple majority voting strategy to determine the overall outcome of the Ensemble. Another notable example is that of Adaboost, a version of the Boosting Ensemble, adapted for the problem of multiclass classification. As previously, bootstrapped training data samples are drawn from an initially uniform but continuously evolving distribution, ensuring that samples that were previously mislabeled are seen more often, and therefore training the base classifiers to the most difficult instances. The base classifiers are combined in a weighted majority voting manner. (2,4)

Κεφάλαιο 3

Methodology

3.1 Data Pipeline Overview

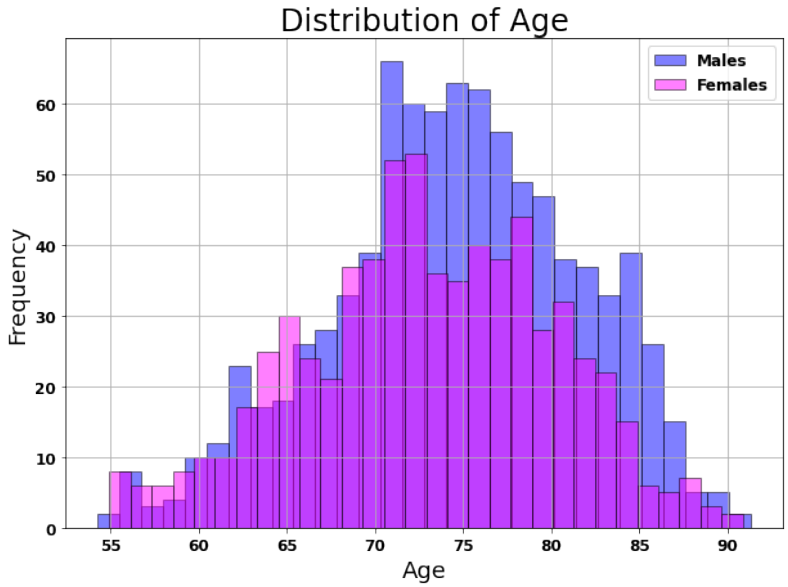
The source of the dataset has been the Alzheimer's Disease Neuroimaging Initiative (ADNI), a global research study that focuses on understanding better how to prevent or delay the disease, as well as supporting the investigation on methods of treatment. The study collects imaging, genetic, clinical, biospecimen data from people with MCI, Alzheimer's, as well as from people that are Cognitively Normal (CN). The data was collected in 4 phases, ADNI1, ADNI2, ADNIGO and ADNI3. This study uses data from all three phases, however only uses the genetic and imaging views of the dataset. For more information on the dataset, please visit .(1)

The imaging data is in the form of 145 Region Of Interest values, acquired from scanners either 1.5T or 3T, using T1- and dual echo T2-weighted sequences, depending on the phase. The images collected by the scanners have been filtered through Quality Control, and have been preprocessed through intensity normalization and gradient un-warping, either by Mayo Clinic or by the MR scanner vendors while the scan was performed (for the ADNI3 phase). In order to acquire the ROI intensity values, the method of MUlti-atlas region Segmentation utilizing Ensembles (MUSE) was used creating the regions of interest. ROIs values' magnitude is analogous to the regional distribution of brain tissue, depending on tissue type. The Regions that these values correspond to can be found in the appendix. The ADNI study had participants that were scanned in regular intervals in order to determine the effect of the diseases on their brain, however this study only uses the first scan, for simplicity and uniformity reasons. (2, 3)

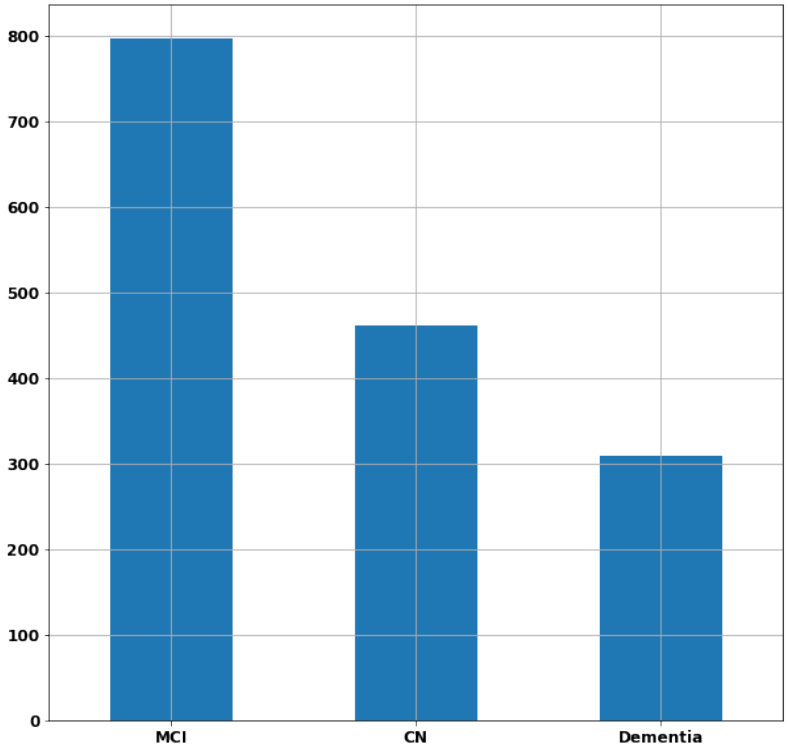
As for the genetic data, each participant has values for 54 susceptibility loci, in the form of Single Nucleotide Polymorphisms, that have been identified by AD genetics studies. These values have been filtered through Quality Control, and as previously mentioned are int the form of number of alleles (0,1,2) for each SNP. (4,5,6,7)

The dataset has 1567 participants (56.47% male), with mean age 74.50 y.o. for males and 72.91 y.o. for females. It is distributed in 3 classes, with 461 Cognitive Normal participants (29.41%), 797 Mild Cognitive Impairment patients (50.86%) and 309 Alzheimer's Disease patients (19.71%).

The data pipeline is as follows. Since quality control, data cleaning and preprocessing had already been performed, those steps weren't needed. First, any duplicates were

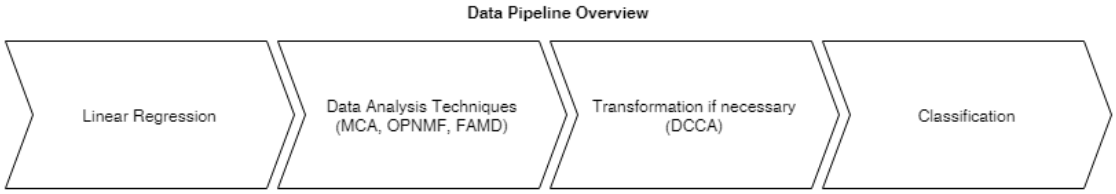


Σχήμα 3.1: *Distribution of age for participants*

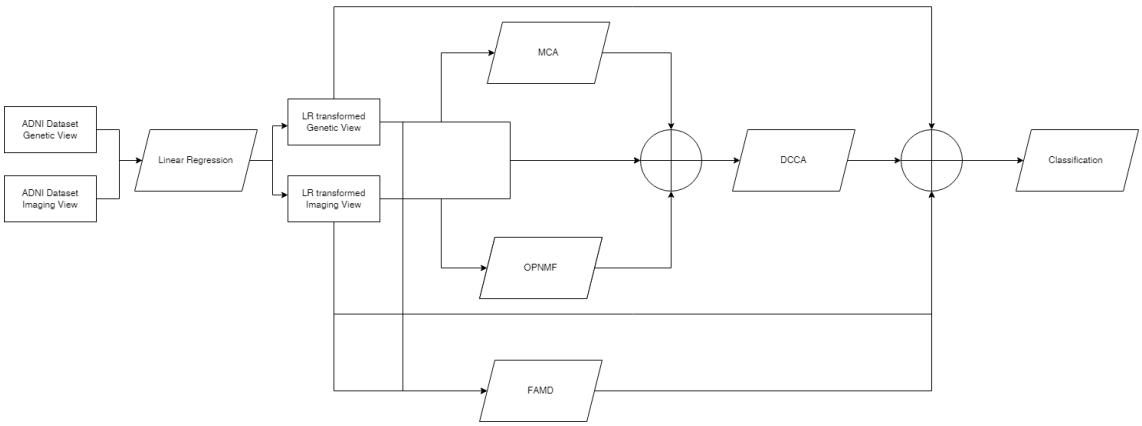


Σχήμα 3.2: *Distribution of classes for participants*

removed, and an appropriate age group was selected (65 y.o. to 85 y.o.) was selected, in order to reduce any extreme values at the lower and higher age groups, which reduced the dataset to 1302 participants, (56.91% male), with mean age 75.20 y.o. for males and 74.36 y.o. for females. The dataset now has 433 Cognitive Normal participants (33.25%), 626 Mild Cognitive Impairment patients (48.07%) and 243 Alzheimer's Disease patients (18.66%). Following that, Linear regression was performed to remove any unwanted age, sex, or brain size related effect. The regressor was fitted on the Cognitive Normal group, and the transformation was applied to every group. Subsequently, the data was transformed through experimentation, and the output of the method was stored, in order to be compared with the raw data. Afterwards, data analysis techniques were applied, such as OPNMF to the imaging data, MCA to the genetic data, and FAMD to the whole dataset. These methods were applied both to the raw data, as well as the output of the DCCA method, in order to be compared later. The result of each one of those techniques was then saved for later tasks. Finally, each one of the results of combinations of the methods was fed to SVM as well as Ensemble classifiers, to measure the impact of each method to the data on the classification task.



Σχήμα 3.3: *Data Pipeline Overview*



Σχήμα 3.4: *Data Pipeline Diagram*

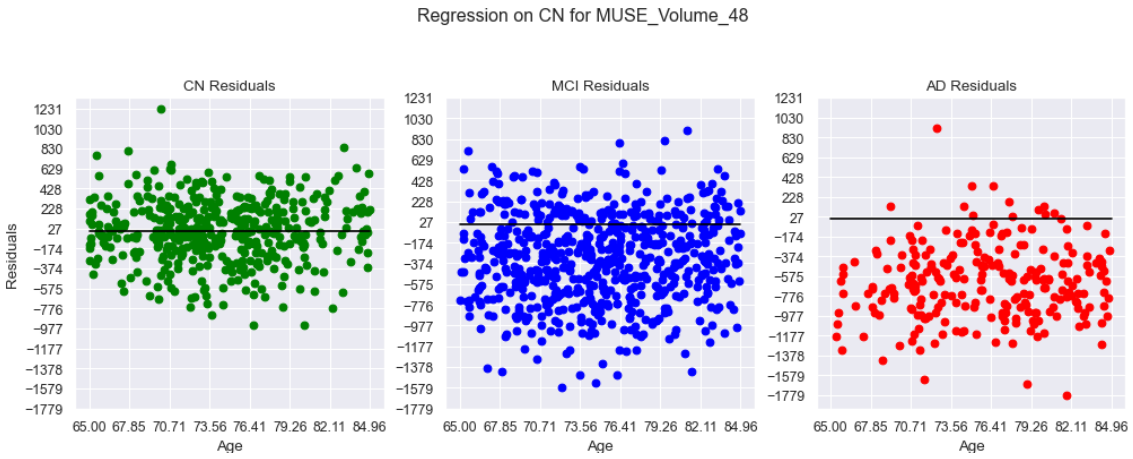
3.2 Linear Regression

Linear Regression is a statistical method that aims to learn the relationship between dependent and independent variables. In our case, the goal is to study the shrinkage effect that AD and MCI have on brain size, and therefore to study it properly we need to isolate that effect. In order to do that, any effects that age, sex and different cranium sizes have, must be removed. Finding out the pattern between the brain size, which is the 145 ROI values, and the age the person has, as well as his/her gender and cranium size, is necessary if we want to remove it.

Linear regression learns the trend between the independent variables (age, gender, cranium size) and the dependent variable (brain size), so calculating the difference from the computed trend values and the real values is called the residual values. We use those residual values as a means to recognize how intense the effect of the disease on the participant's brain size is, since the other independent variables' effects have been subtracted, meaning that if there was no effect, the trend between age, gender and cranium size would accurately predict the brain size and there would be no error.

The model is trained on Cognitive Normal participants only, however all participants brain sizes are predicted using the trend line learned, to find out the difference between the real values and the predicted ones. We only keep the residual values, as they signify the difference from the trend line, which is equivalent to the intensity of the disease's effect.

The trend lines can be observed in the below figure for the ROI MUSE_Volume_48 which is translated to the Left Hippocampus area. It is clear that while the Cognitive Normal (green) population is centered around the x axis, meaning the brain size values are well predicted, the MCI population (blue) is lower than predicted, and the AD population (red) is much lower. This is expected, as the real brain size values are lower, since MCI and AD both cause a shrinkage of the brain.



Σχήμα 3.5: *Linear Regression on CN for MUSE_ Volume_ 48*

3.3 DCCA model training

In addition to the previously mentioned data analysis techniques, in order to transform the two uncorrelated, different views into two that are more linearly correlated, the method of DCCA was used. The choice of hyperparameters, the methods and class functions, as well as the parameter optimization was subject to our experimentation.

As stated before, the DCCA model relies on two parallel networks, each taking as input a view of the dataset, and producing the output that are the views but nonlinearly transformed in order to use as subject for classification.

The dataset for the model optimization was split into three sets, the training, validation and the test set, with the last one being kept hidden from the model during the training phase, in order to ensure an accurate prediction on unseen data. The split between the sets was 75% for the training set, 15% for the validation set, and 10% for the test set, split randomly. This was done in order to ensure that the model had enough training samples, as well as validation samples to achieve good accuracy scores without overfitting, and enough test scores as to not skew the results.

The hyperparameters include number of hidden layers as well as hidden layer size, output layer size, the regularization parameter, the learning rate, as well as the batch size. These hyperparameters were chosen after extensive testing with each one, with the best values stored and trained on the data, in order to produce the best results possible. Other hyperparameters such as the activation function, the error metric, and the optimizer function were not experimented with, as this study followed the original DCCA method paper as closely as possible. The epoch number was kept at 100 epochs along all experimentation, which was enough for all cases for the validation accuracy score to stabilize.

We experimented with 3 or 4 hidden layers, as it became apparent that due to the complexity of the problem, a big enough network was needed for both views. The original paper used the same architecture for both of the networks, a logic we followed in our study as well. The size of the hidden layers ranged from 256 neurons per layer to 1024 neurons per layer, with all hidden layers having the same number of neurons. As for the output layer size, we experimented with sizes of [10,50,100,150]. One attempt was made with output layer size of 300 in order to observe how the trend continues, however the computation was extremely time-consuming. Furthermore, the learning rate in our tests ranged from 10^{-4} to 10^{-2}, and the regularization parameter being in the range of 10^{-4} to 10^{-2}. The batch size that we used was either 500 samples or 1000 samples per batch.

The activation function was kept the same as the original paper, which was a Sigmoid function, and the error metric was the error metric as defined from the paper, a version of CCA using a derivative-free optimization method. The optimizer function of the paper was the L-BFGS second-order optimization method, however due to it being more difficult to compute, the optimizer RMSProp was used, with similar results. The original paper initialized the neural network using a denoising autoencoder, but this was out of the scope of this study. (1)

The DCCA implementation we used was made with python by Zhanghao Wu. The method is implemented with pytorch, which supports for multi-GPU training, however

this study employed only CPU training. (2,3)

3.4 Data Analysis Techniques

In our experimentation with the dataset, it was noticed early on that the nature of the different types of views of the dataset was an obstacle to the methods and the goals we wanted to achieve. We hypothesized that the difficulty to achieve better classification scores stemmed from the fact that the two views of data were different not only in type (numerical vs categorical), but in the difference of the dimensions as well, making the algorithms employed inefficient and/or not well suited for the task.

To remedy this situation, we experimented with data analysis techniques. Initially we tried OPNMF, to bring down the number of imaging dimensions, to more closely match the number of genetic dimensions. Then, we tried using MCA to transform the genetic data from categorical to numerical, and finally we tried creating a combining transformation of the two views using FAMD, as a benchmark for the other methods. All possible combinations of the suitable methods were tested, in order to find the best possible mix.

A data analysis technique that this study explores is that of Orthonormal Projective Non-Negative Matrix Factorization. This method is used to dimensionally reduce a dataset, while still maintaining interpretability, due to the non-negative nature of the matrix decomposition. For this dataset, we applied OPNMF to the imaging data, in order to dimensionally reduce the 145 ROIs into 30 components.

The number of the resulting components was chosen in a way that was close to the number of the number of dimensions of the genetic data, close to the (later referenced) number of MCA components, while not too small in order to retain most of the information and minimize approximation error, and not too big in order for the method to have any use. It is worth noting that the OPNMF method rescales the data, a much needed action, since different ROIs have orders of magnitude different intensities, a feature that the SVM classifiers benefit from.

The method was performed with the help of the OPNMF code from the CBICA Lab at UPenn. (1)

Another data analysis technique that is used in this study is Multiple Correspondence Analysis, which, as previously mentioned, analyses the structure of a number of dependent categorical variables, and performs dimensionality reduction if necessary. For this dataset, we applied MCA to the genetic data, in order to dimensionally reduce the 54 SNPs into 10 components. This had the added benefit of transforming the categorical data into numerical, which we hypothesized would greatly enhance the outcome of the classification.

The number of the resulting components was chosen based on trial and error, with fewer components (2, then 5) producing worse accuracy results, and more components (15) not improving the scores achieved, while being significantly harder to compute via the MCA method employed (more than twice the running time of MCA code in our tests).

The method was performed using mca, a package for python which is intended to be used along with pandas. (2)

The third and final technique we explored is FAMD, since it can combine multiple views of the dataset, to create a transformation along with a dimensionality reduction. This in theory is the most desired effect, but in practice its merits are limited, since there can be

a big information loss, and our experimentation wasn't as extensive as it could be, again due to limitations in our computational ability.

The number of the resulting components was chosen based on the fitting time, however it was quickly observed that the method regardless of the choice for number of components was producing subpar results to other methods.

The method was based on the implementation of the python package prince, an open-source package developed by Max Halford. (3)

3.5 Classification

The task we tried to enhance the models for was the one of classification. To solve that, we chose the Support Vector Machine family of models, and optimized it through parameter Grid Search, which is exhaustive search of all the different parameter combinations.

As mentioned previously, SVMs can have different kernels, in order to accommodate for non-linear datasets. We experimented with Linear, Polynomial, and Radial Basis Function kernels, which are among the most commonly used. For each kernel, its specific parameters were optimized through Grid Search, and the results were evaluated on their ability to generalize through Cross Validation.

For all of the different cases, we used the Python Library Scikit-Learn, and specifically the sklearn.svm module. (1)

In particular, for the Linear Kernel we experimented with L2 normalization penalty, [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10] C (regularization parameter) values, and One-Vs-Rest multi class classification strategy. As for the Polynomial kernel, we experimented with polynomial degrees of [2,3,4,5], independent term values of [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10], C values of [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10], and finally kernel coefficient values (g values) of [0.0001, 0.001, 0.01, 0.1, 1]. Finally, as for the Radial Basis Function kernels, we experimented with C values of [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10], and kernel coefficient values of [0.0001, 0.001, 0.01, 0.1, 1].

Each kernel was given 1000 iterations in order to converge, and the Cross Validation was done with 5 folds. The data was split into train and test splits, with respective sizes of 80% and 20% of the initial dataset, without shuffling, as the data was already ordered randomly. All of the combinations were run on all of the potential different combinations on views of every data analysis techniques (Imaging + Genetic views, only Imaging, only Genetic).

Furthermore, in addition to the simple method of the SVM models, we attempted to use the method of Ensemble Learning to further enhance the classification outcome. For that reason, we experimented with the methods of Bagging (Bootstrap Aggregating) and Adaboost. For both of those methods, we experimented with the base classifier being a Decision Tree or a Linear Support Vector Machine. The relatively simple choices were taken purely because of computational limitations, as well as time constraints.

For all of the different combinations, we use the Python Library Scikit-Learn, and specifically the sklearn.ensemble module.

The parameter tuning for both of those models as well as their base classifiers was done with Grid Search along with Cross Validation, using 5 folds. Specifically, for the Bagging classifier Ensemble model, the parameters that we experimented with were the number of estimators, with values of [5,10,15], the maximum samples of the dataset that an estimator could train on, with values of [60%,80%,100%]. As for the Adaboost Ensemble classifier model, the parameters we experimented with were again the number of estimators, with those being [5,10,15,50], and SAMME and SAMME.R for the boosting algorithm. The learning rate for this model was kept at 1.0.

For the Decision Tree base classifier, the parameters we experimented with were the

Estimator Criterion, with it being either Gini Impurity or Entropy, along with the max depth, with its values being [1,2,5]. Finally, for the Linear SVM base classifier, we experimented with the C parameter, with its values being [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10].

Before the classification task, if needed, balancing and scaling was applied. Balancing the dataset was done through random undersampling, while sampling was performed utilizing scikit-learn's preprocessing module, and more specifically the StandardScaler function. This function standardizes features by removing the mean and scaling to unit variance.

The metrics we chose for the classification task were those of Accuracy, Balanced Accuracy, and F1 score. The implementation of the metrics that was used was again from Scikit-Learn. As for Accuracy, it is the ratio of:

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + False\ Positive + True\ Negative + False\ Negative}$$

As for the Balanced Accuracy, it is the ratio of:

$$Balanced\ \ Accuracy = Avg(\frac{True\ Positive}{True\ Positive + False\ Negative} + \frac{True\ Negative}{True\ Negative + False\ Positive})$$

And as for the F1 score, it is the ratio of:

$$F1\ \ Score = \frac{True\ Positive}{True\ Positive + \frac{1}{2}(False\ Positive + False\ Negative)}$$

Κεφάλαιο 4

DCCA Optimizations

Before the classification, we must optimize the DCCA network parameters, and find the ideal values and combinations. We do that for the different cases of the views.

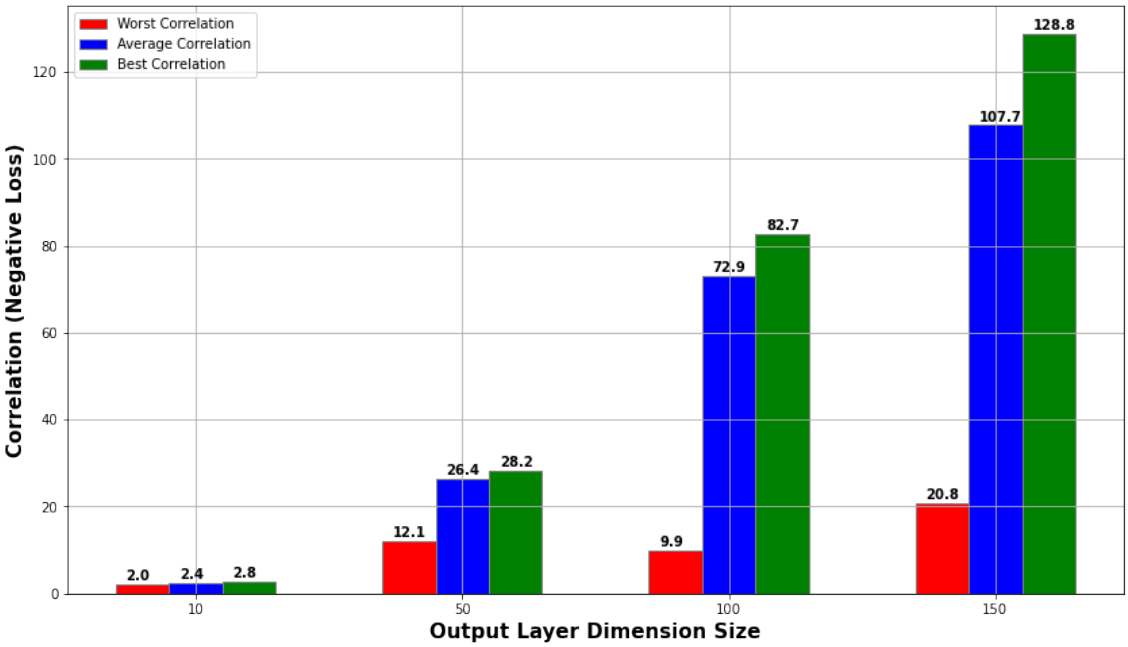
First of all, we train the DCCA networks on the raw data, meaning the 145 ROI values for the imaging view, and the 54 SNPs for the genetic view. After that, we take the MCA-transformed genetic data (10 genetic components), and pair them with the original imaging data, meaning the 145 ROI values. Finally, we experiment with the opposite combination, which is the OPNMF-transformed imaging data (30 imaging components) and pair them with the original genetic data, meaning the 54 SNPs.

As mentioned in the previous chapter, the parameter combinations we explored affect the number and sizes of the hidden layer, the size of the output layer of the network, the learning rate, the regularization parameter and the batching size. The following results are after exhaustive search of the different parameter combinations.

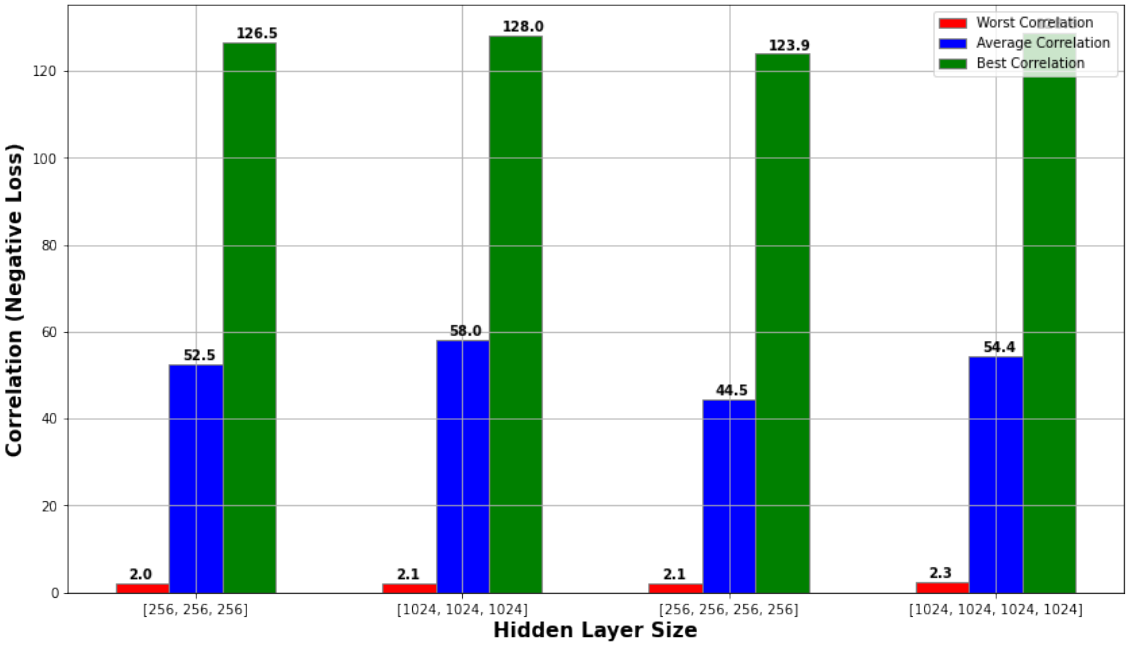
The metric we optimized for is the correlation between the two views, after being transformed by their respective DCCA network. Essentially we pass the data through their own trained network, transforming them and measuring how linearly correlated they have become. We chose to optimize for this metric instead of the classification accuracy, since we wanted to evaluate the suitability of the DCCA method on the specific problem as a transformation mechanism. The total number of parameter combinations for each different case of the views is 288.

For each parameter, we plotted for the different values the correlation achieved by the worst combination of the parameter's value with all the other possible parameters values, the average correlation, as well as the best correlation. Since the implementation of the DCCA network employed the correlation as a Loss function (assigned with a negative signum, in order to be able to be minimized), we used that as the metric directly.

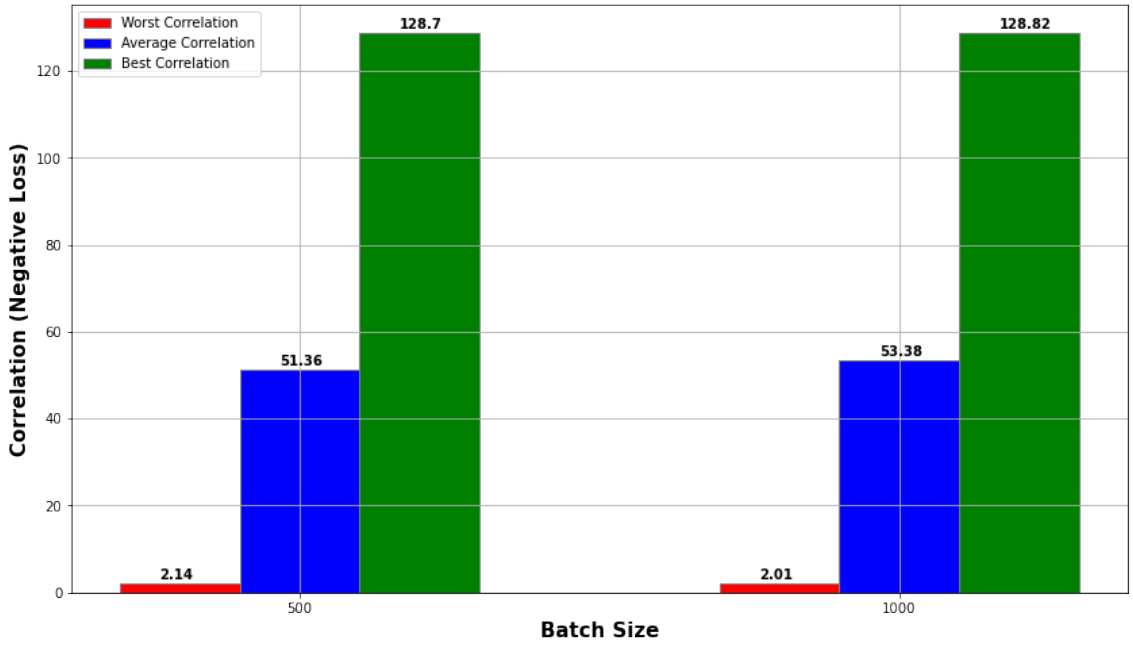
4.1 Original data: 145 ROI (Imaging) + 54 snp (Genetic)



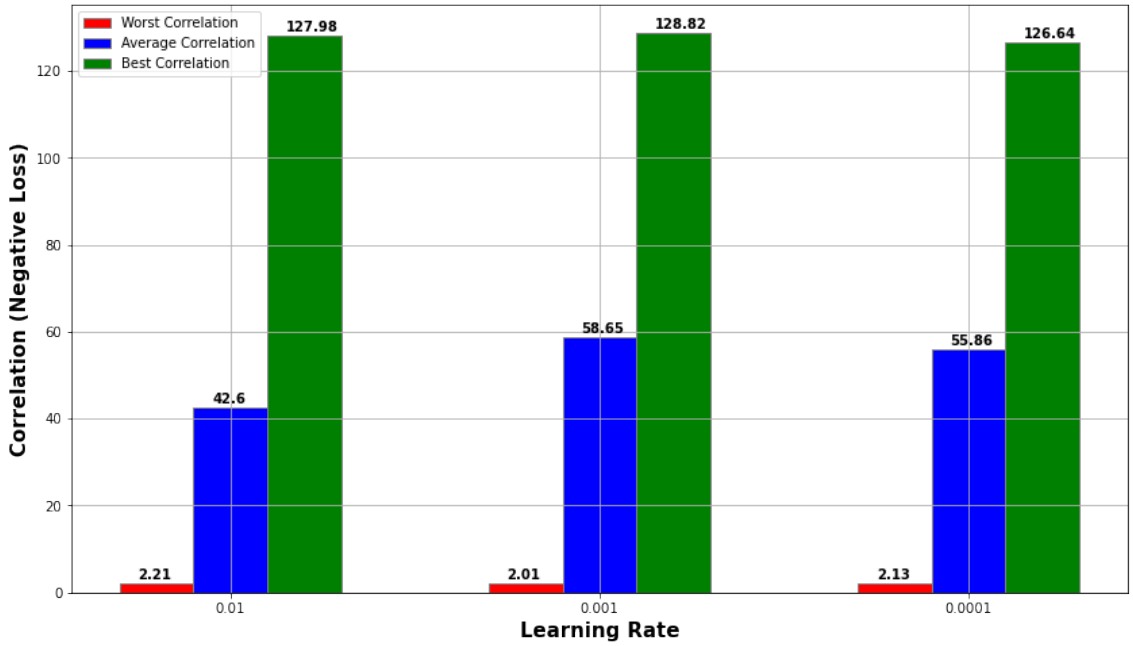
Σχήμα 4.1: *Output Layer Dimension size vs Achieved Correlation*



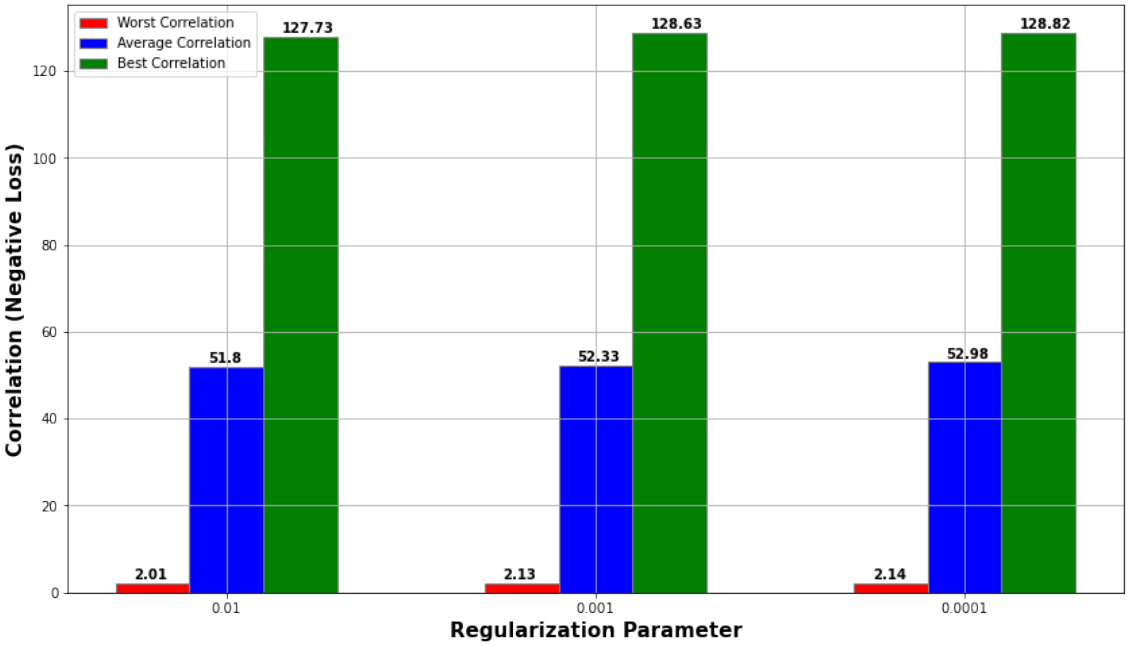
Σχήμα 4.2: *Hidden Layer size vs Achieved Correlation*



Σχήμα 4.3: *Batch size vs Achieved Correlation*



Σχήμα 4.4: *Learning Rate vs Achieved Correlation*



Σχήμα 4.5: *Regularization Parameter vs Achieved Correlation*

Based on the above figures, it is clear that the more nodes the output layer has, the better the correlation of the transformed data is, not only on the best case, but on average as well as on the worst case.

Furthermore, it is clear that hidden layer size and number have little effect on the output correlation, since not only the best, but also the average and the worst cases, the correlation numbers seem to be the same. It can be noted that more hidden layers and more nodes per hidden layer do seem to be achieving better results, but the difference is minor.

As for the batch size parameters, in our experiments the results stay basically identical; the only change being noticed in terms of training time, since the lower the batch size, the more time the model takes per epoch to run through the dataset, hence more training time for the same number of epochs.

The same effect can be noticed with the Learning Rate, since the best case is more or less achieving the same results, however here we can observe that the average case benefits from a medium Learning Rate value of .

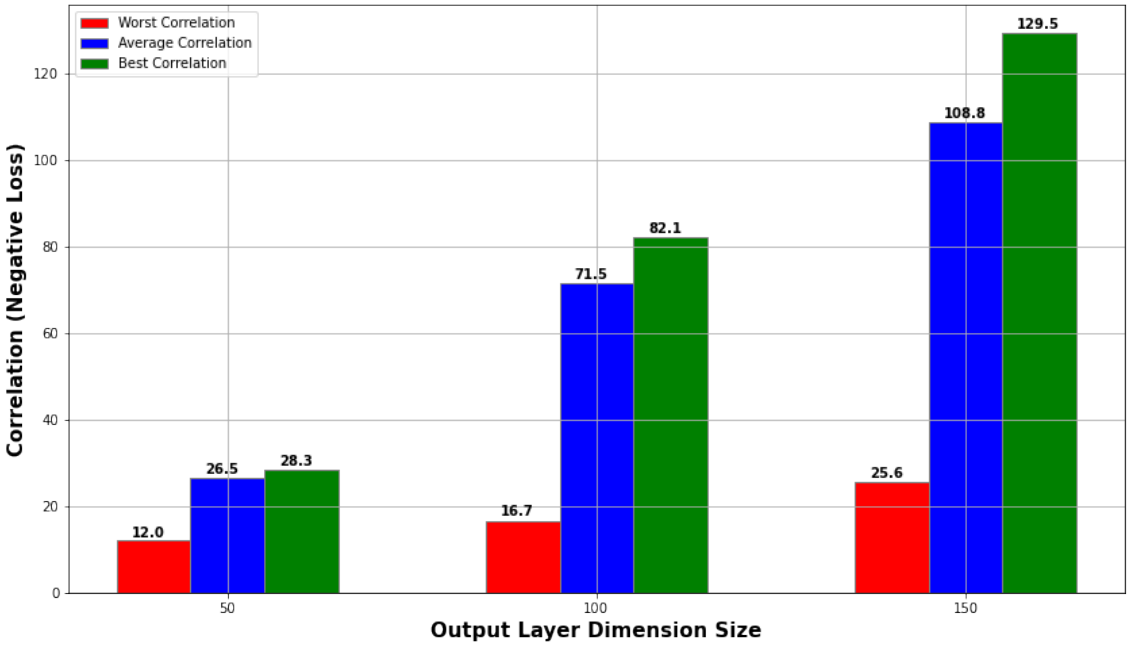
Finally, the Regularization parameter follows the same logic, with its changing not making a substantial difference, and on all accounts the results being identical.

To summarize the parameters' effects, we can create the following table:

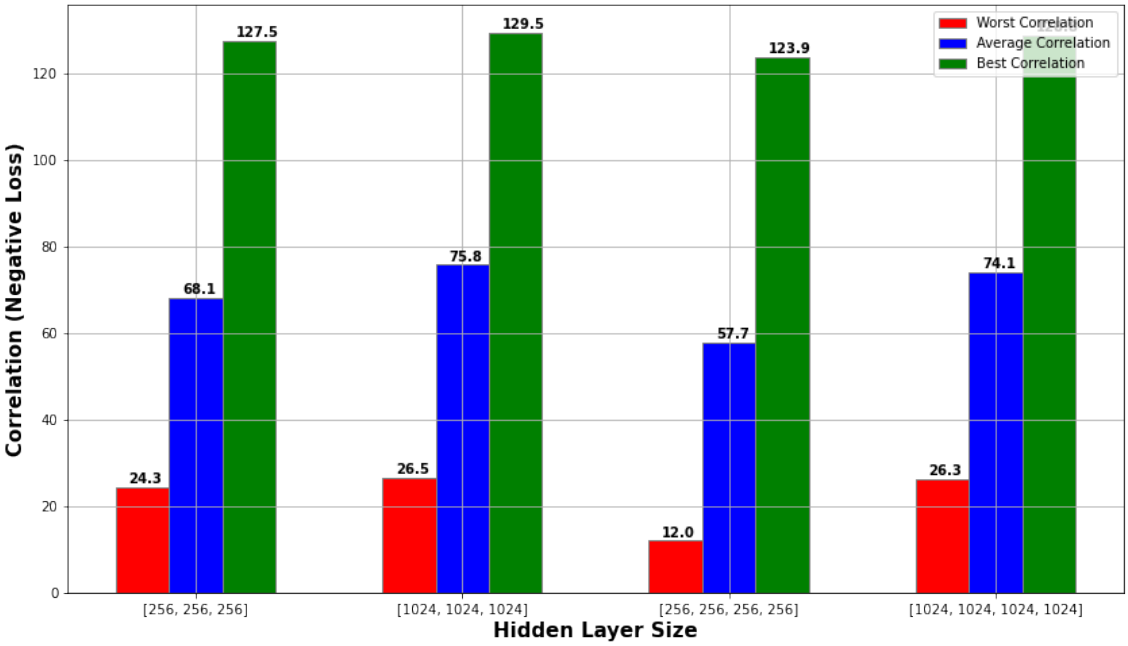
Parameter Action	*Correlation (Negative Loss)*
Output Dimension Size ↑	↑
Hidden Layer Size ↑	↑
Learning Rate	Medium to low LR is best
Batch Size	Stays basically the same
Regularization Parameter	Stays basically the same

Σχήμα 4.6: *Learned Conclusions from DCCA optimizations on 145 ROI (Imaging) and 54 SNPs (Genetic)*

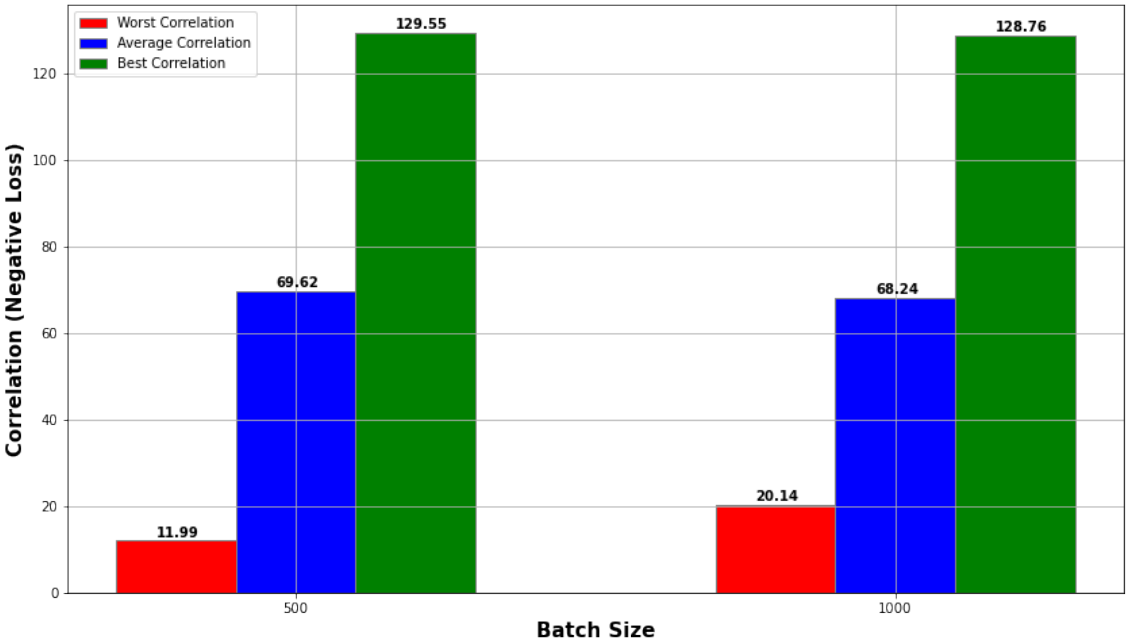
4.2 Transformed Genetic data: 145 ROI (Imaging) + 10 MCA components (Genetic)



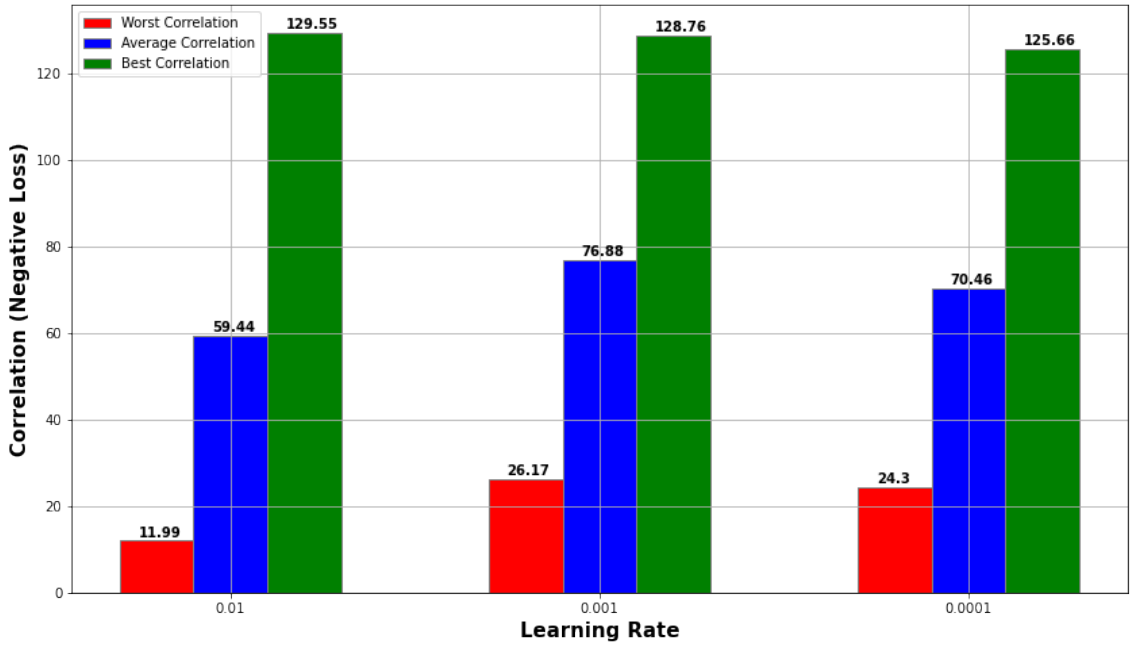
Σχήμα 4.7: *Output Layer Dimension size vs Achieved Correlation*



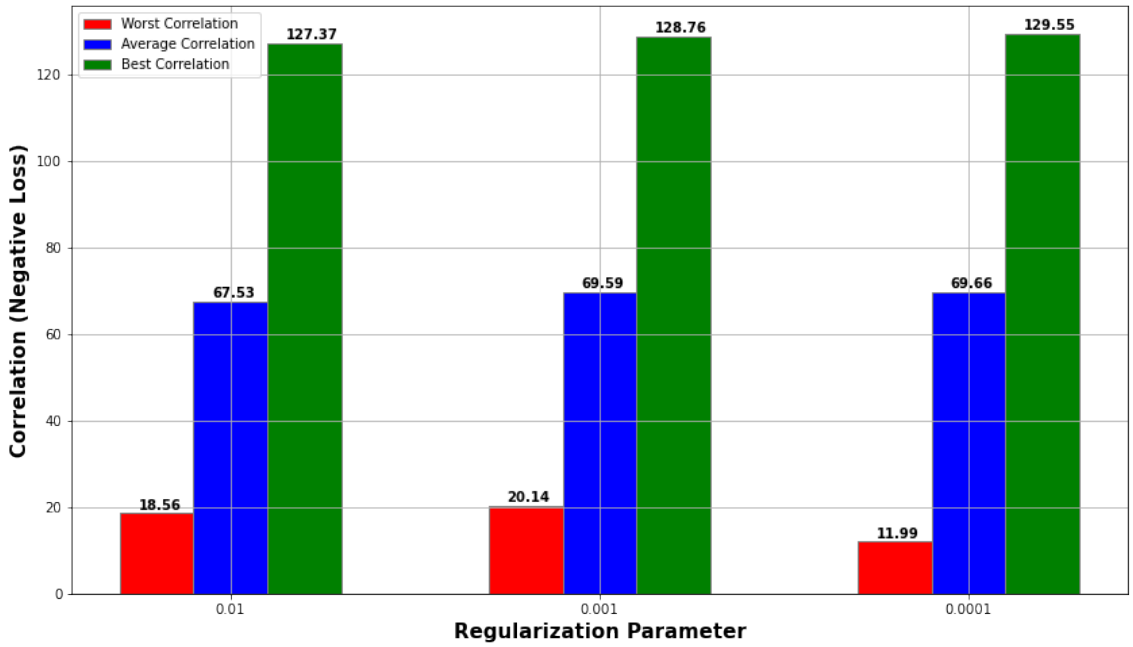
Σχήμα 4.8: *Hidden Layer size vs Achieved Correlation*



Σχήμα 4.9: *Batch size vs Achieved Correlation*



Σχήμα 4.10: *Learning Rate vs Achieved Correlation*



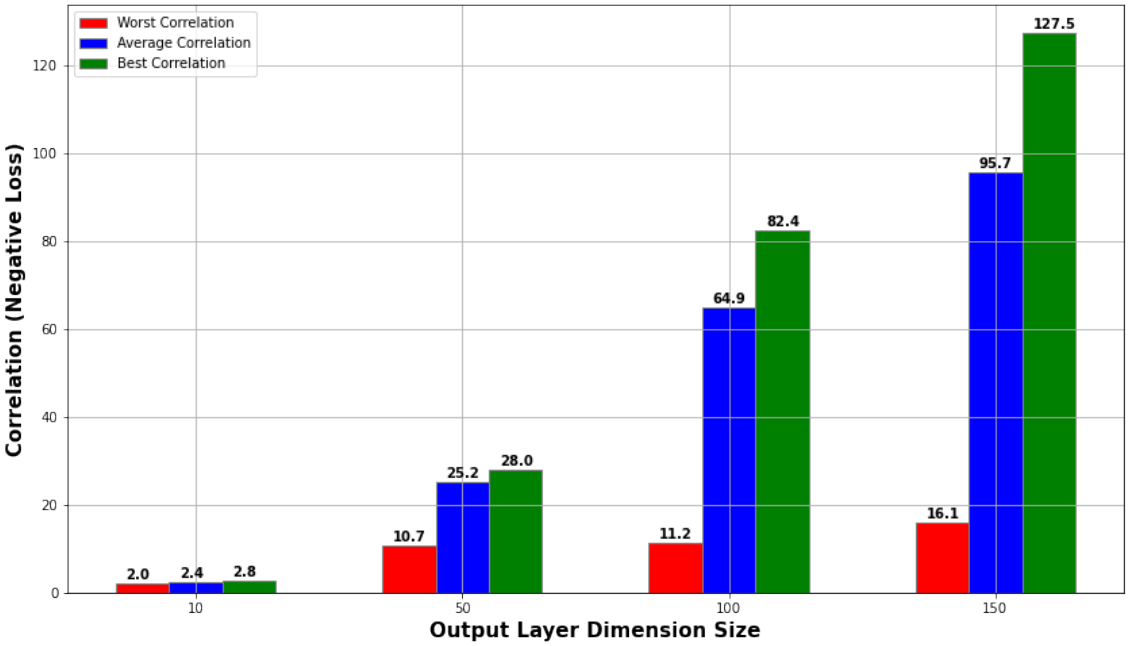
Σχήμα 4.11: *Regularization Parameter vs Achieved Correlation*

As before, we notice the same patterns. Increasing the output layer size results in an increase in output correlation, hidden layer size and number of hidden layers seems to make a small difference, and for the rest of the parameters the effect seems to be negligible. We can sum up the parameter behaviour in the following table:

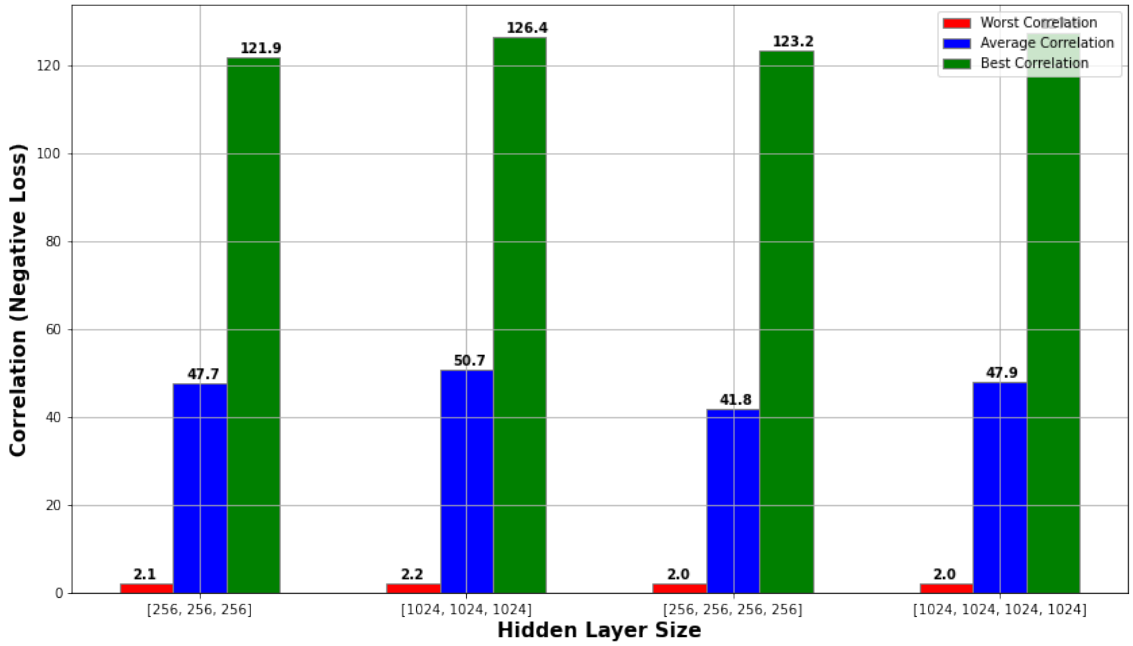
Parameter Action	*Correlation (Negative Loss)*
Output Dimension Size ↑	↑
Hidden Layer Size ↑	↑
Learning Rate	Medium to low LR is best
Batch Size	Stays basically the same
Regularization Parameter	Stays basically the same

Σχήμα 4.12: *Learned Conclusions from DCCA optimizations on 145 ROI (Imaging) and 10 MCA Genetic components*

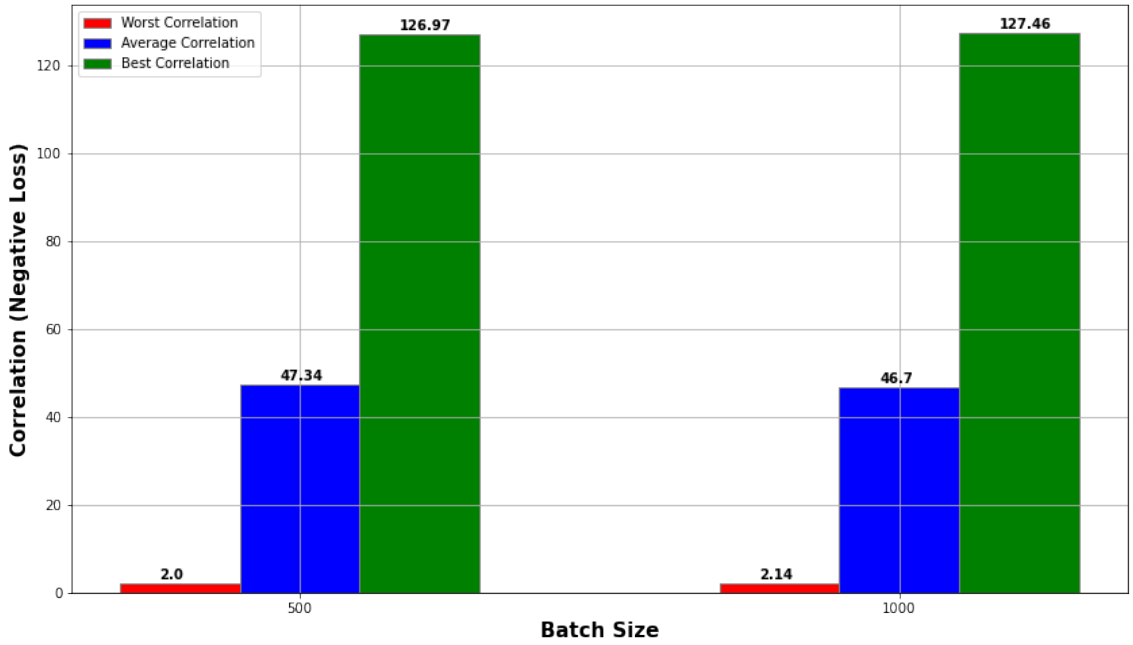
4.3 Transformed Imaging data: 30 OPNMF components (Imaging) + 54 SNP (Genetic)



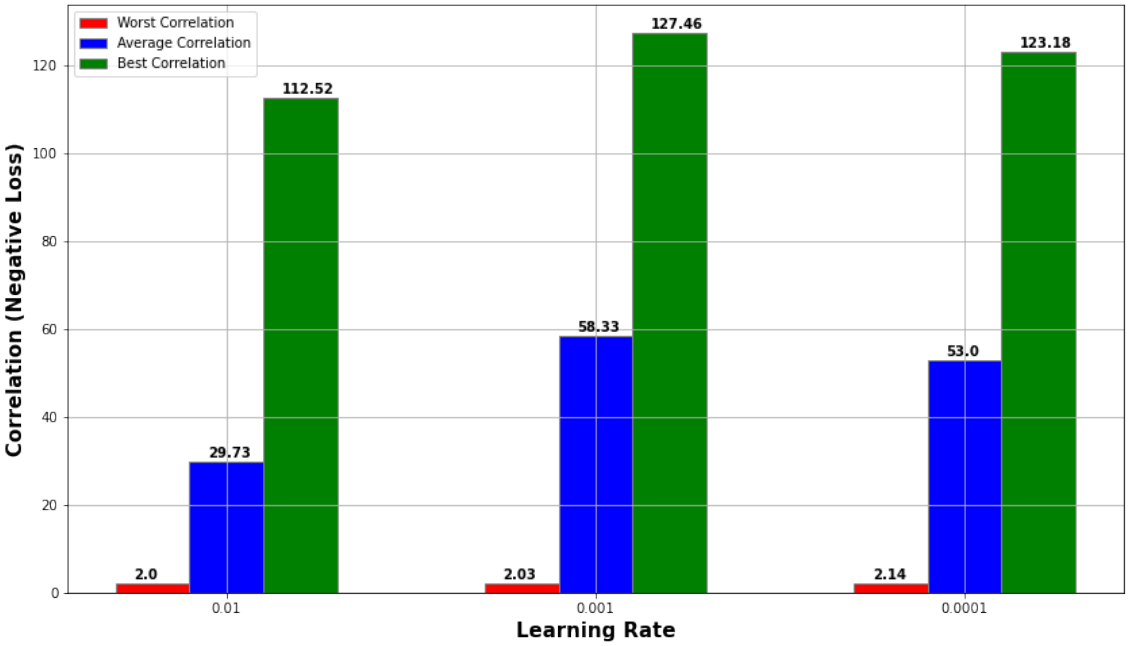
Σχήμα 4.13: *Output Layer Dimension size vs Achieved Correlation*



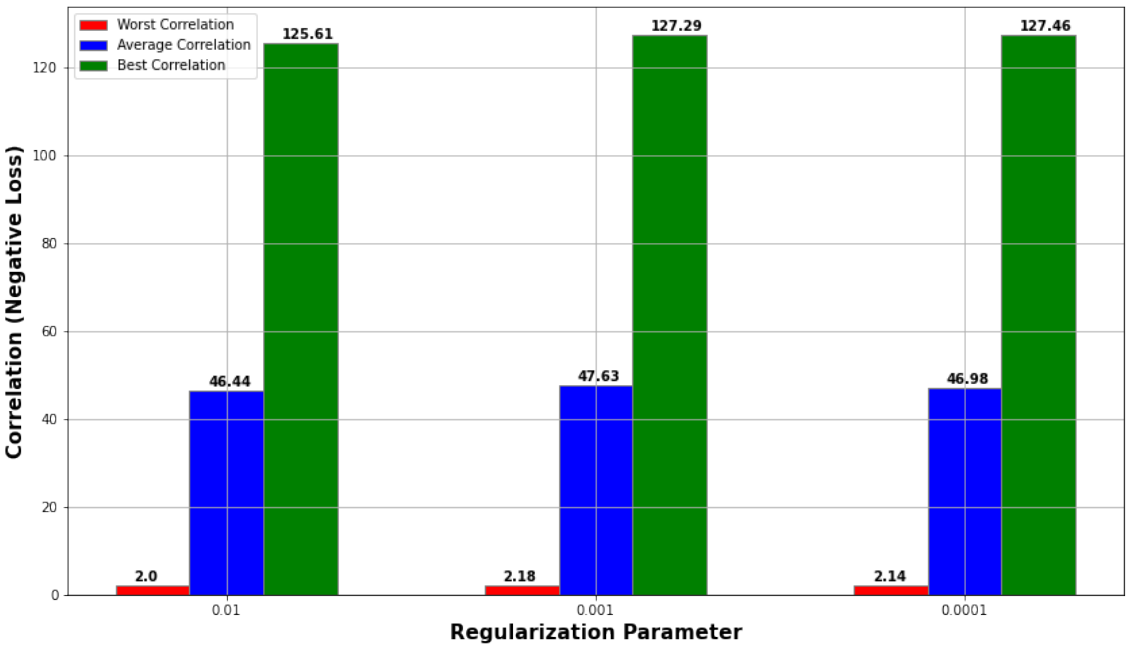
Σχήμα 4.14: *Hidden Layer size vs Achieved Correlation*



Σχήμα 4.15: *Batch size vs Achieved Correlation*



Σχήμα 4.16: *Learning Rate vs Achieved Correlation*



Σχήμα 4.17: *Regularization Parameter vs Achieved Correlation*

Finally, in the case of the transformed through OPNMF imaging data combined with the raw genetic data, we can see that the parameter behaviour is again the same. Once again, output layer size increase correlates with better results, bigger hidden layer size, along with increasing the number of hidden layers improves the output correlation but only slightly, learning rate should be kept at a value of , and altering the other parameters has little to no effect.

Parameter Action	*Correlation (Negative Loss)*
Output Dimension Size ↑	↑
Hidden Layer Size ↑	↑
Learning Rate	Medium to low LR is best
Batch Size	Stays basically the same
Regularization Parameter	Stays basically the same

Σχήμα 4.18: *Learned Conclusions from DCCA optimizations on 54 Imaging components and and 54 SNPs (Genetic)*

Κεφάλαιο 5

Results

In this chapter, we introduce the layout and the specifics of the various results the methods and combinations thereof we experimented with achieved.

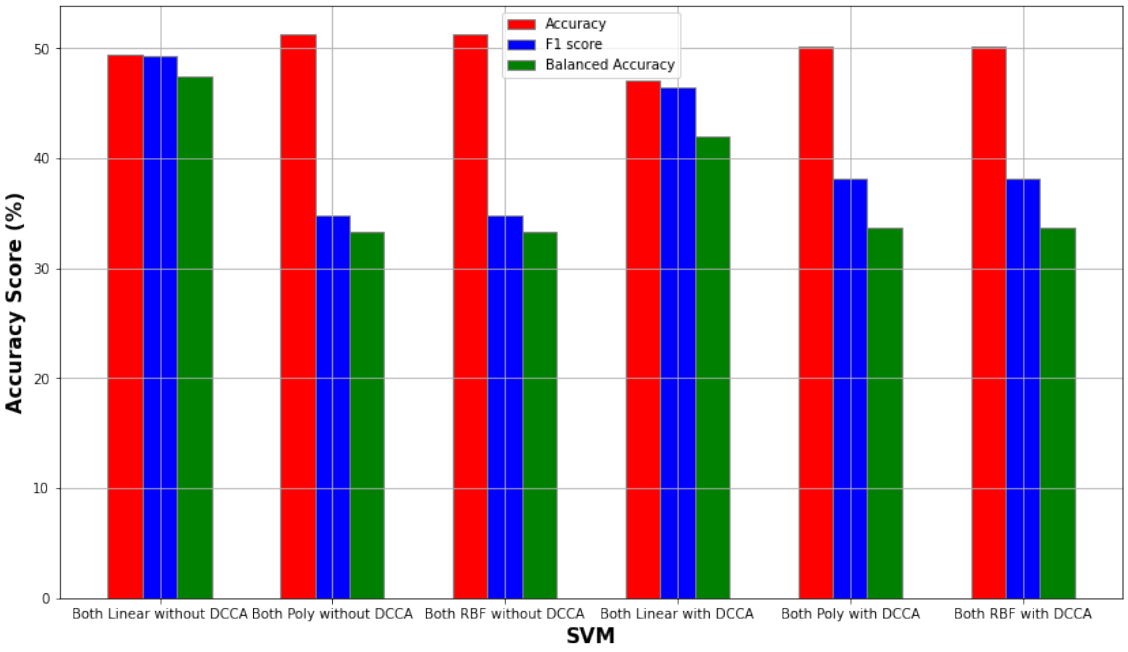
First, the classification results of the raw views are presented, meaning the imaging data as is (145 ROIs) and the genetic data as is (54 SNPs), as a baseline. Those results are contrasted with the results of applying DCCA to those data. Following that, the results of the data with MCA transformed genetic data (10 genetic components) are presented, and contrasted with DCCA applied on top of that. Afterwards, the respective results are presented after OPNMF (30 imaging components) and then with DCCA on top of that, using the combination of MCA and OPNMF, and finally after FAMD.

For all of the aforementioned combinations of methods, the classification results are presented before and after scaling (if needed) and balancing, to highlight the effect those techniques have on the task. Finally, to ensure that the effect of having both views is properly documented, we perform the same task with both views, as well as keeping only one view, testing imaging and genetic for every method.

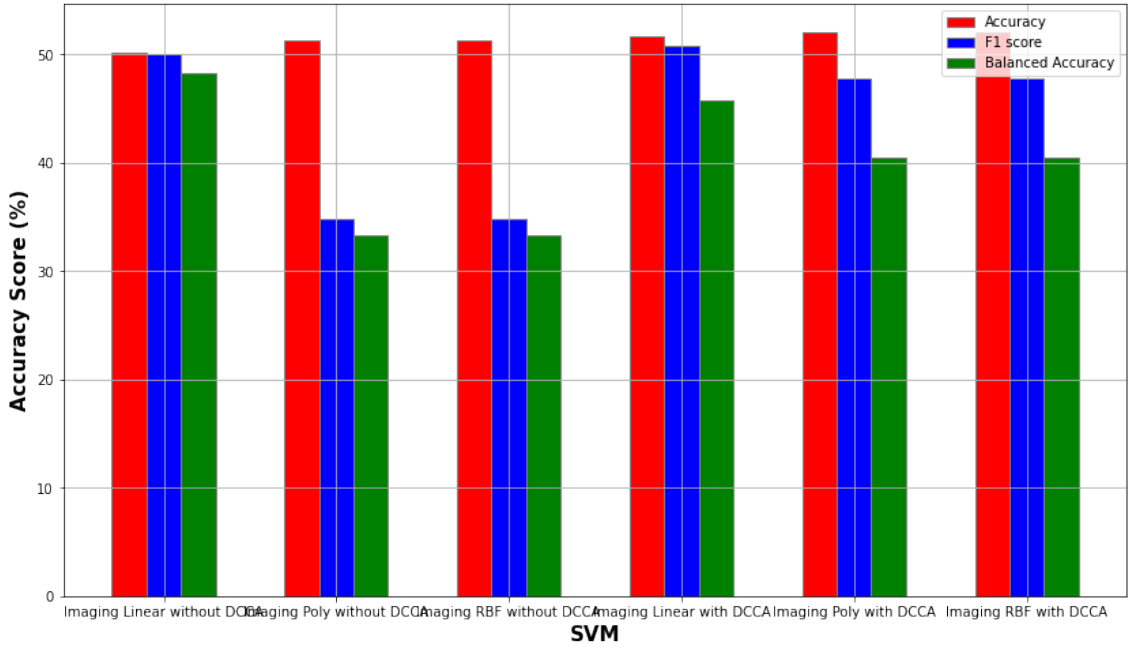
5.1 Raw data vs DCCA

As mentioned before, we consider the SVM classification on the imaging and genetic data taken directly after Linear Regression on the ADNI dataset the baseline results, to be compared with the methods we experimented with.

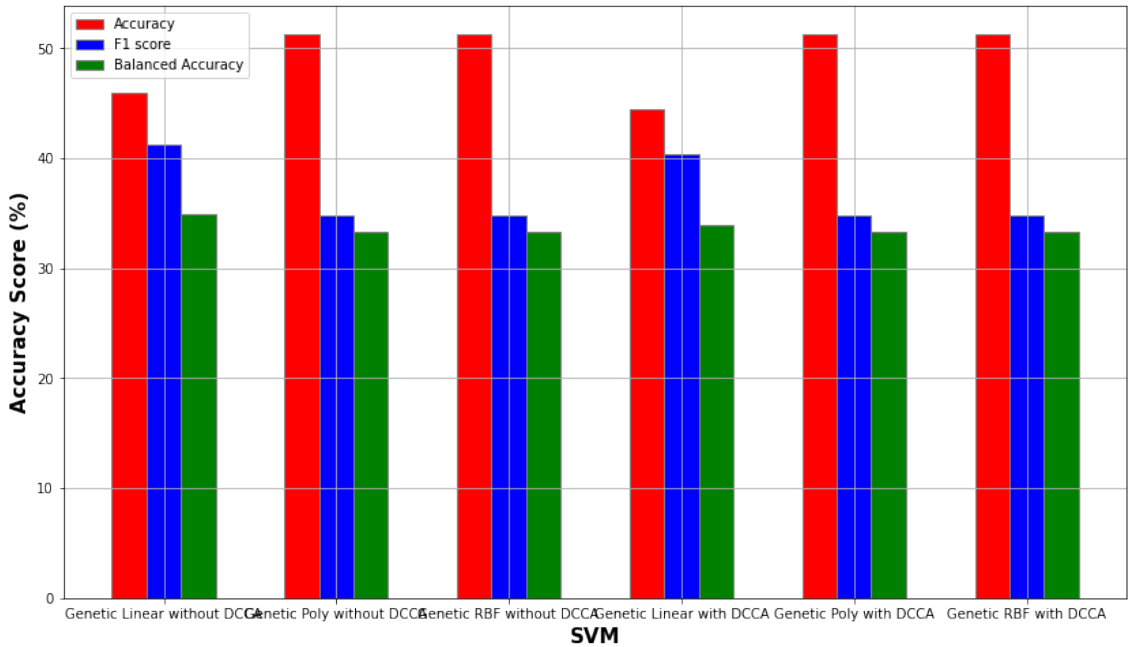
5.1.1 Without scaling or balancing:



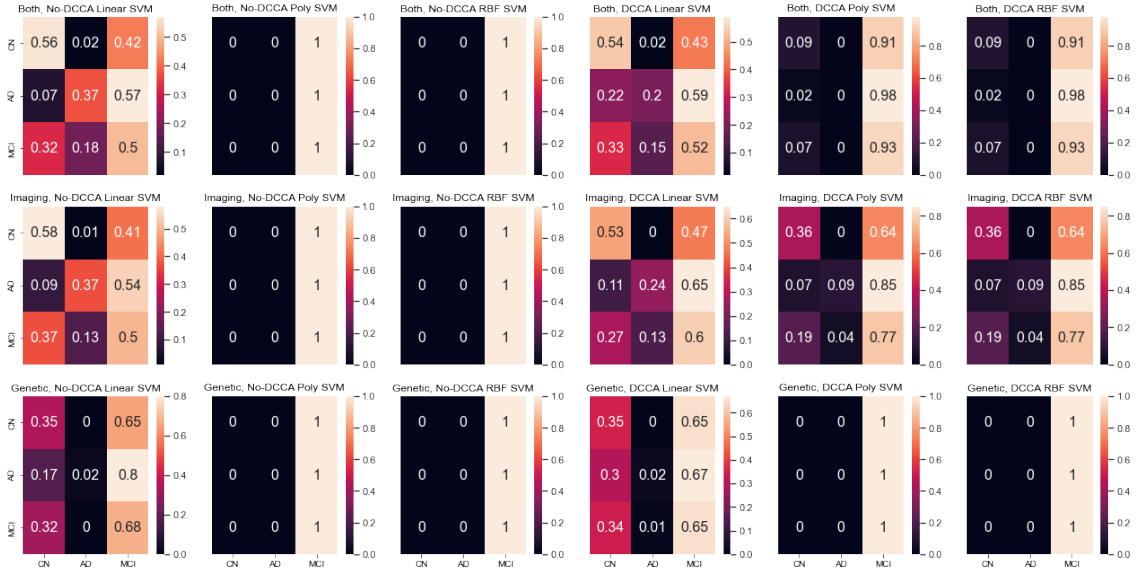
Σχήμα 5.1: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA (3 right bar groups)*



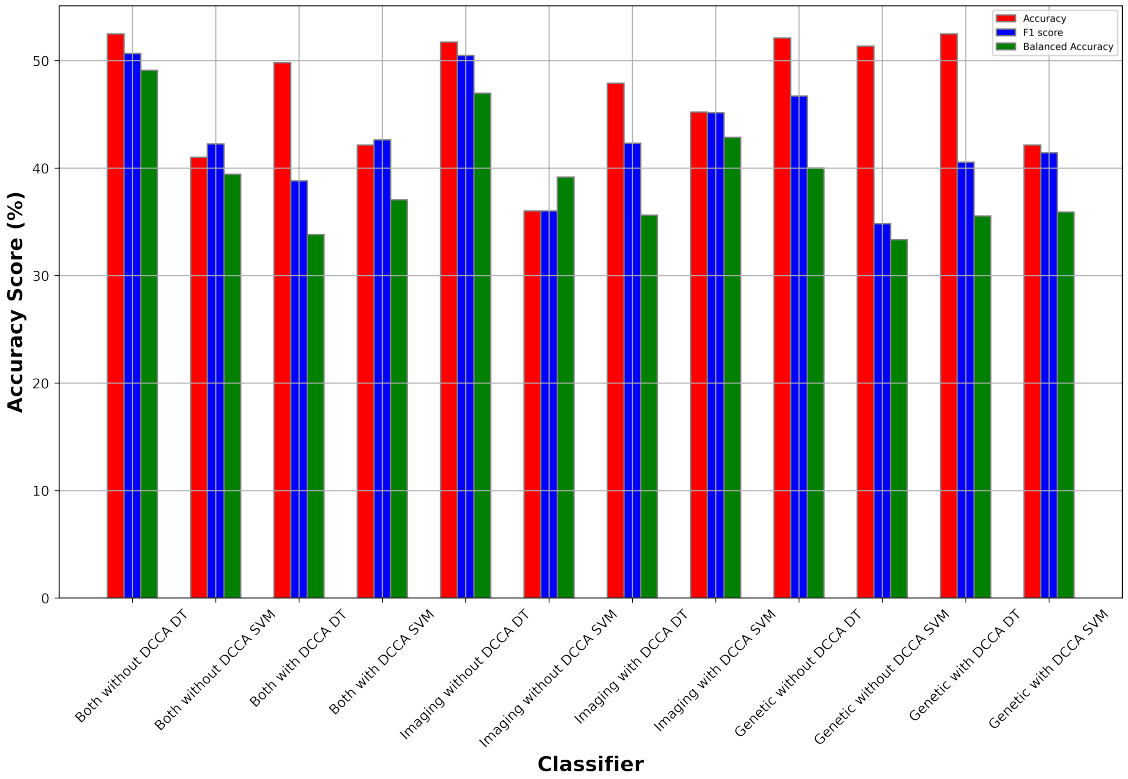
Σχήμα 5.2: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA transformed imaging data, trained on both views (3 right bar groups)*



Σχήμα 5.3: *Classification metric scores using only the Genetic view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA transformed genetic data, trained on both views (3 right bar groups)*

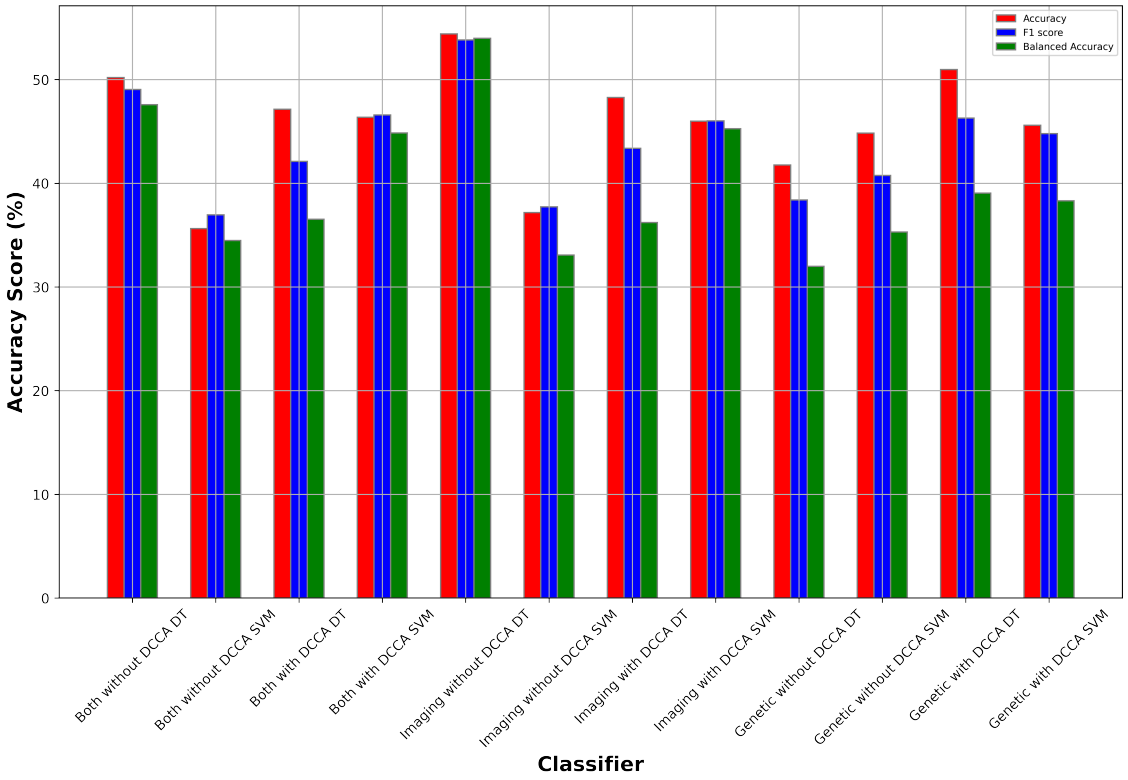


Σχήμα 5.4: *The Confusion Matrices for each class, per model, using both views (top row), only the imaging view (middle row), and only the genetic view (bottom row). The three left columns represent the CM of the raw data classification, while the three right columns represent the CM of the DCCA transformed data classification.*



Σχήμα 5.5: *Classification metric using Bagging on the imaging and genetic data.*

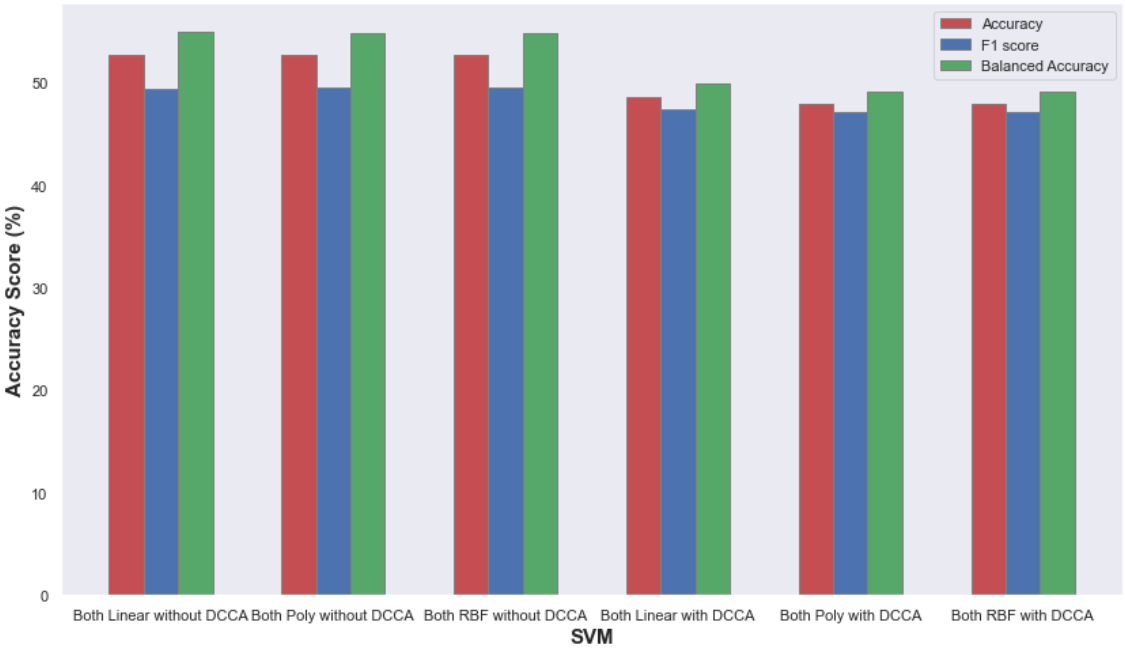
Σχήμα 5.6: *The Confusion Matrices for each class, with Bagging, for the imaging and genetic data.*



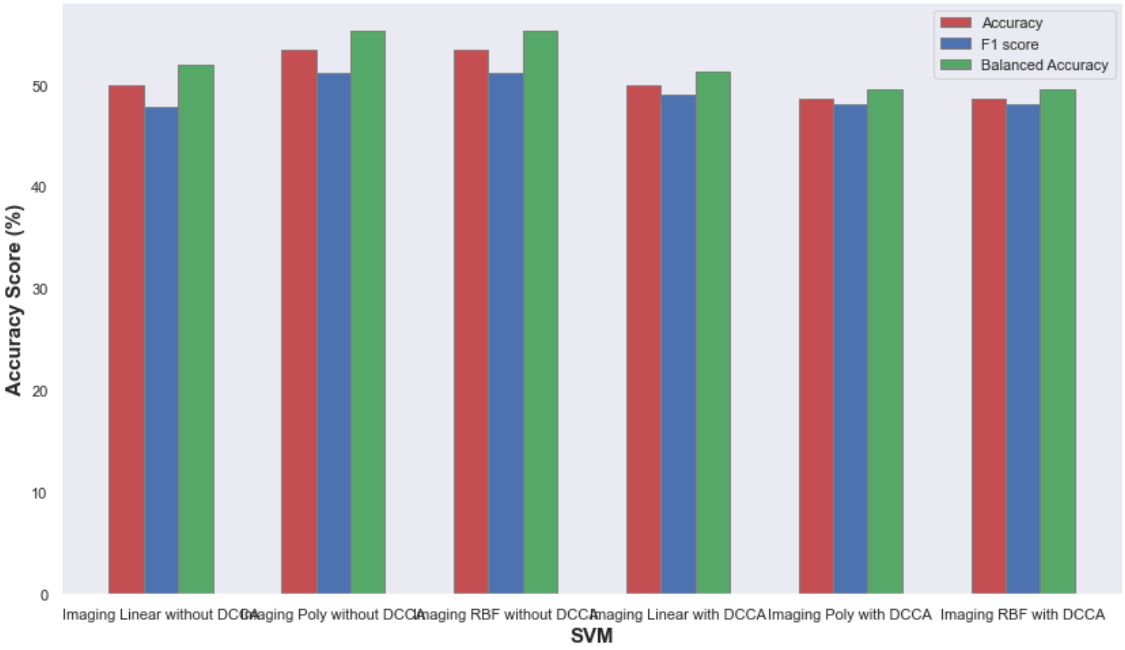
Σχήμα 5.7: *Classification metric using AdaBoost on the imaging and genetic data.*

Σχήμα 5.8: *The Confusion Matrices for each class, with AdaBoost, for the imaging and genetic data.*

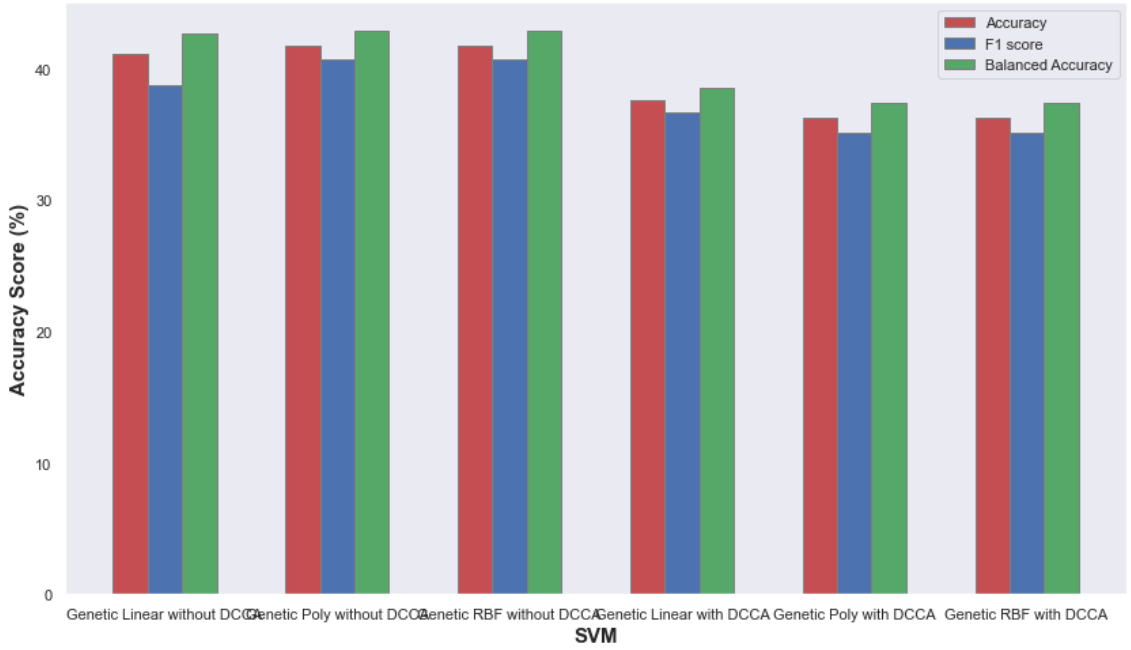
5.1.2 With scaling and balancing:



Σχήμα 5.9: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA (3 right bar groups)*

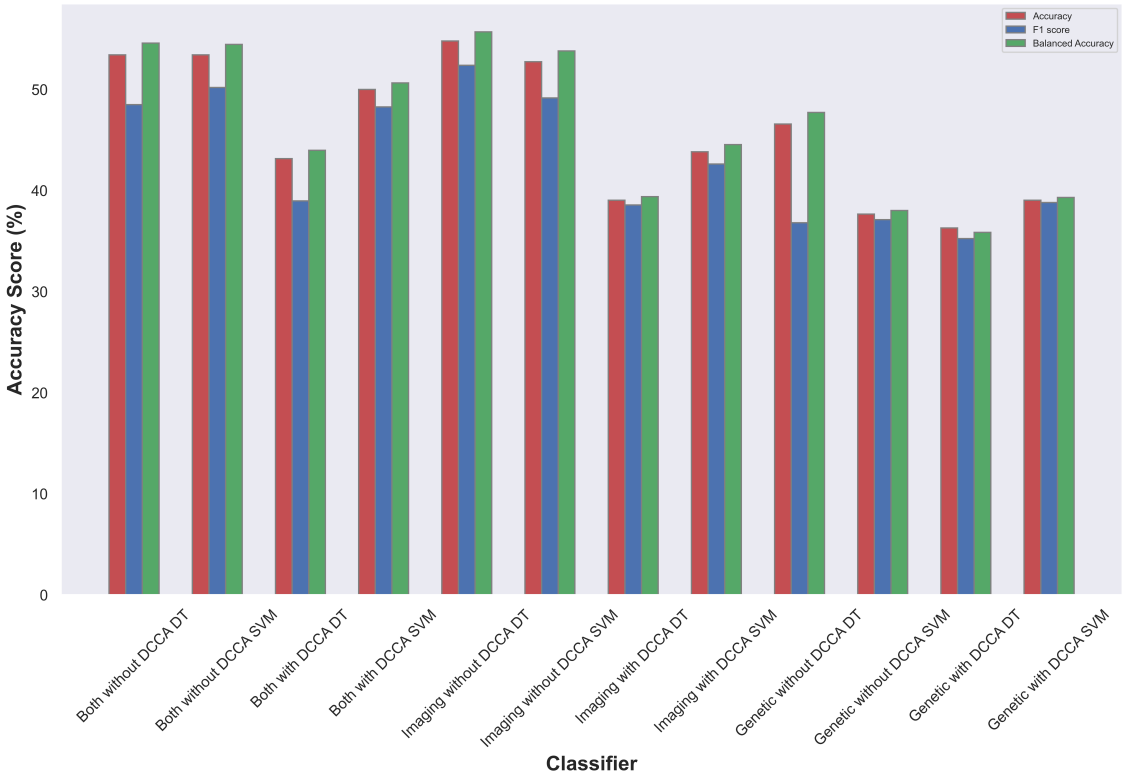


Σχήμα 5.10: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA transformed imaging data, trained on both views (3 right bar groups)*

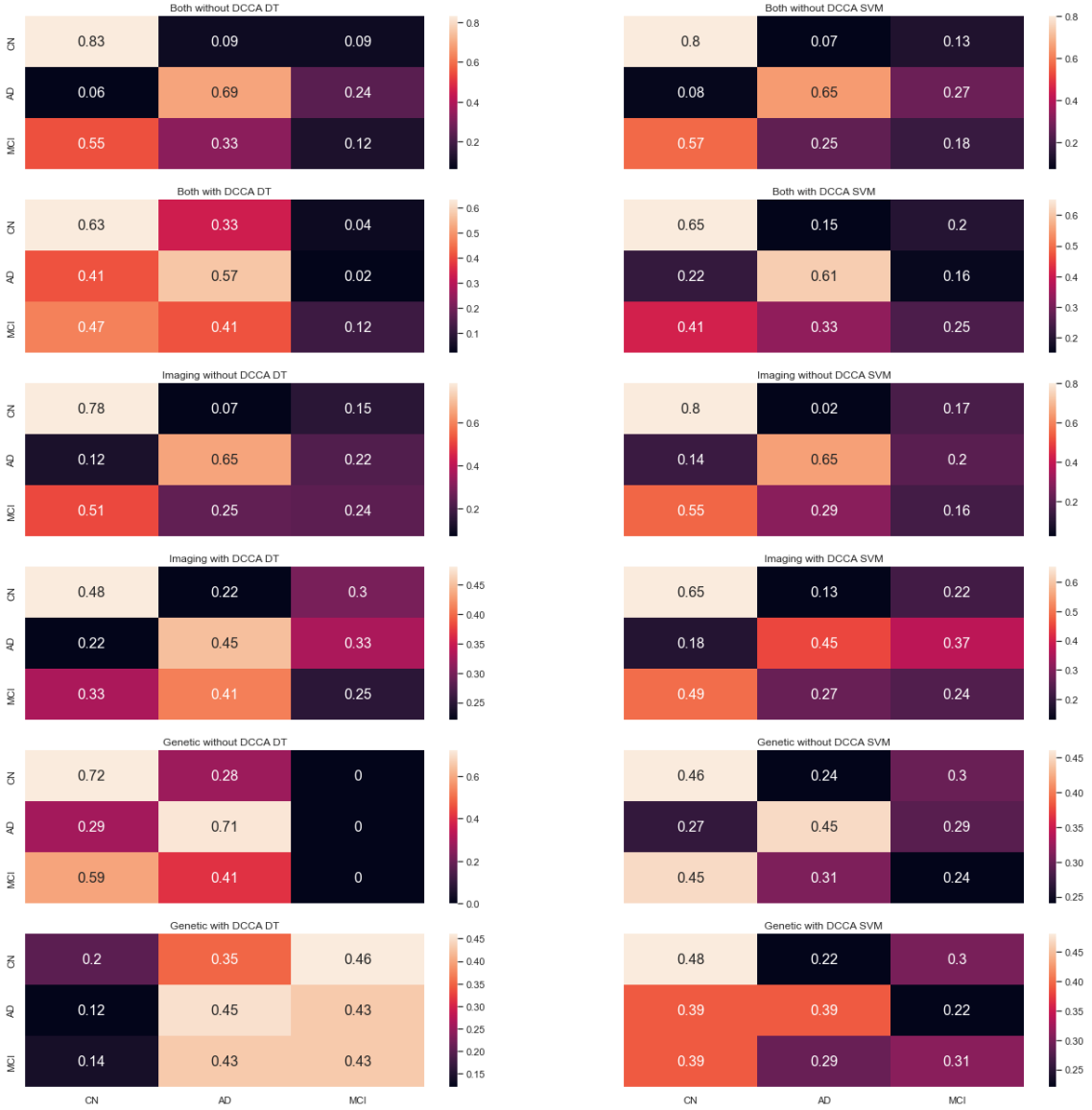


Σχήμα 5.11: *Classification metric scores using only the Genetic view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw data (3 left bar groups) vs using DCCA transformed genetic data, trained on both views (3 right bar groups)*

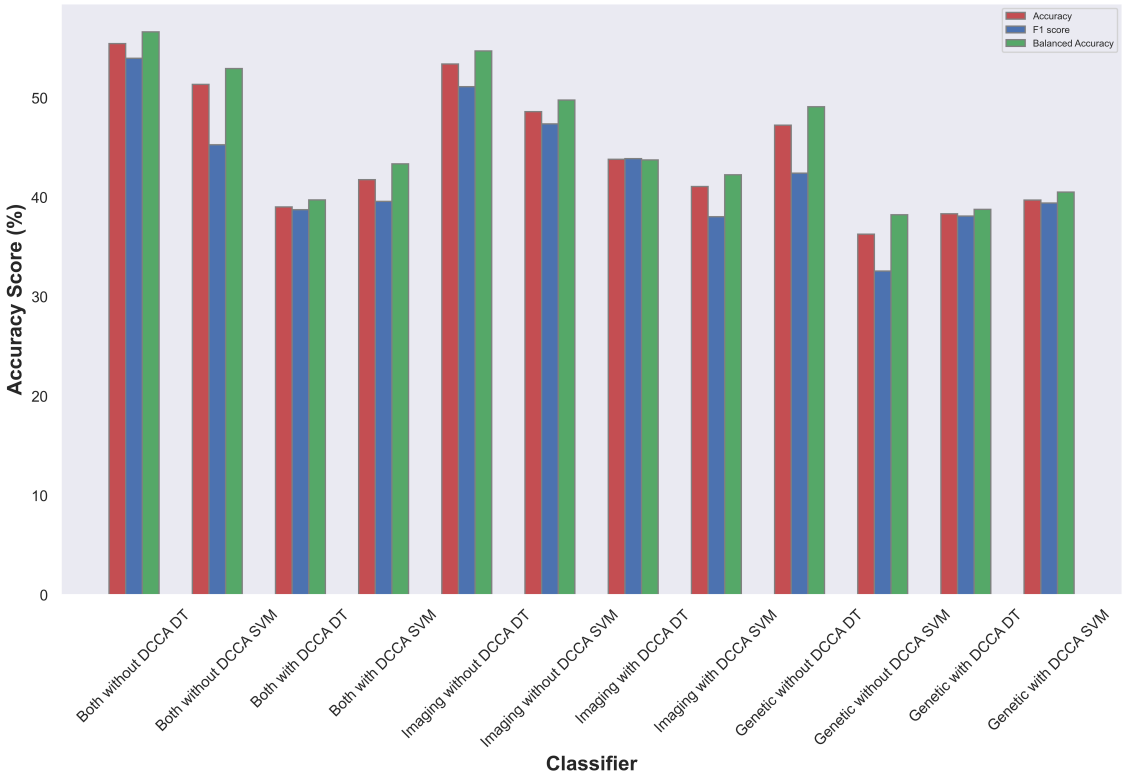
Σχήμα 5.12: *The Confusion Matrices for each class, per model, using both views (top row), only the imaging view (middle row), and only the genetic view (bottom row). The three left columns represent the CM of the raw data classification, while the three right columns represent the CM of the DCCA transformed data classification.*



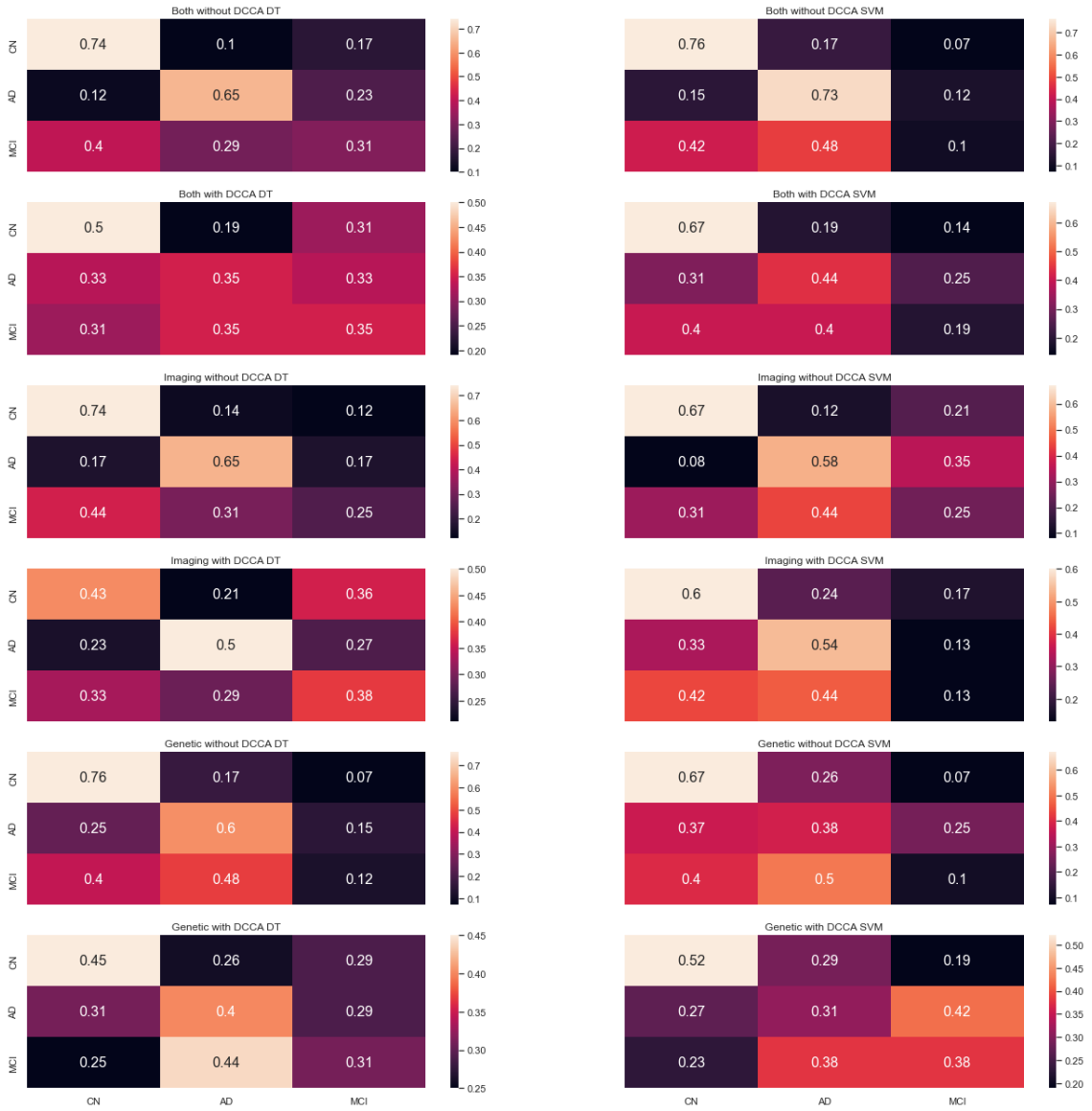
Σχήμα 5.13: *Classification metric using Bagging on the imaging and genetic data.*



Σχήμα 5.14: *The Confusion Matrices for each class, with Bagging, for the imaging and genetic data.*



Σχήμα 5.15: *Classification metric using AdaBoost on the imaging and genetic data.*

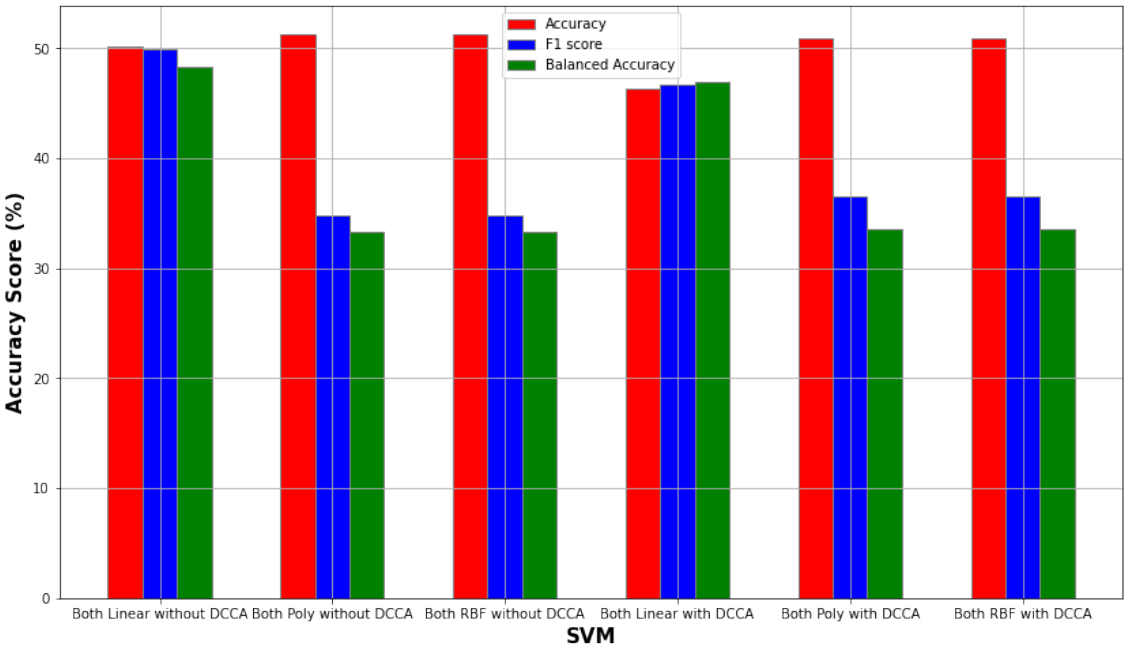


Σχήμα 5.16: *The Confusion Matrices for each class, with AdaBoost, for the imaging and genetic data.*

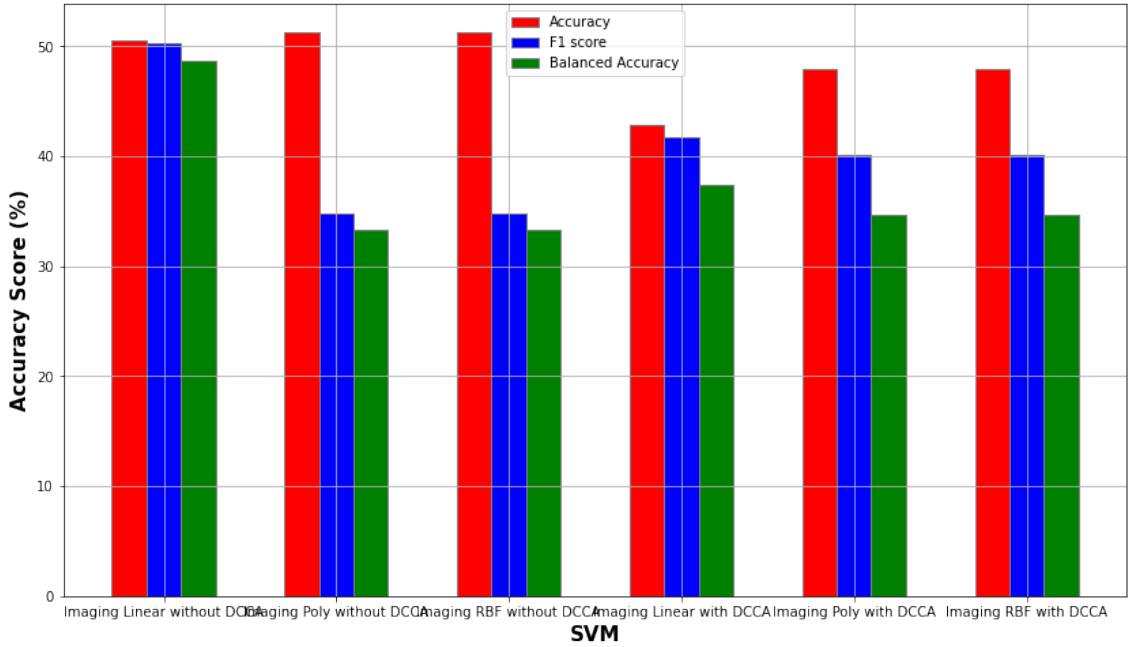
5.2 MCA vs MCA - DCCA

Moving on from the baseline classification of the raw data versus the DCCA transformed data, we explore the effect that MCA has, and introduce the results of the classification. In this part, we compare the classification results, (a) of the data after the genetic view has been transformed through MCA, and (b) of the DCCA transformed data after the genetic view has been transformed through MCA.

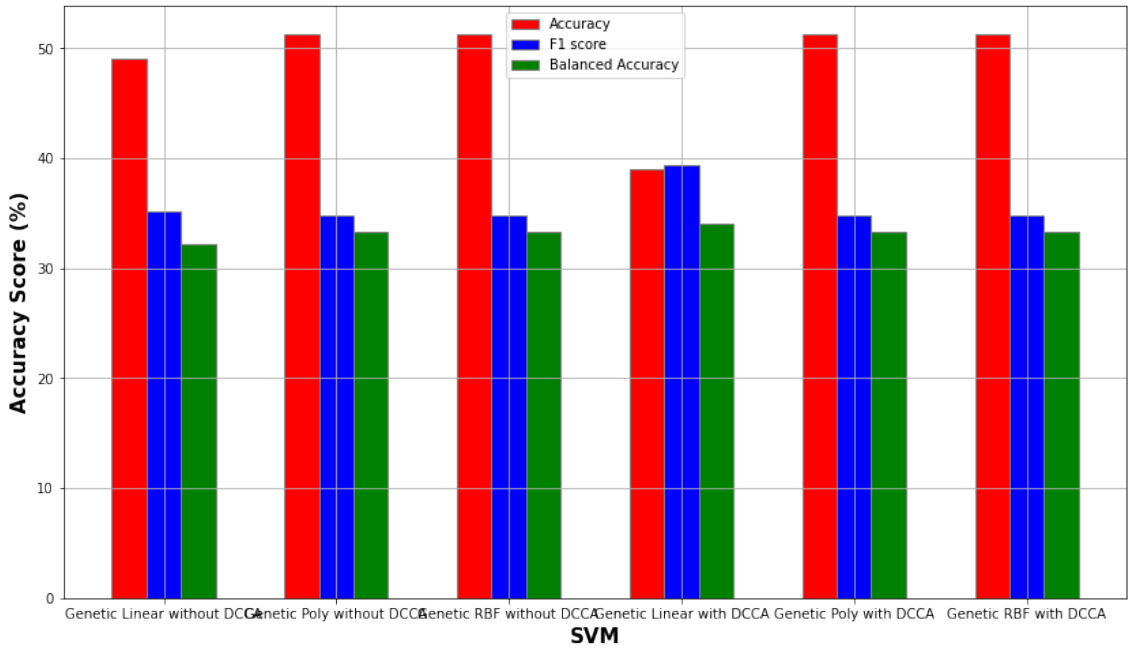
5.2.1 Without scaling or balancing:



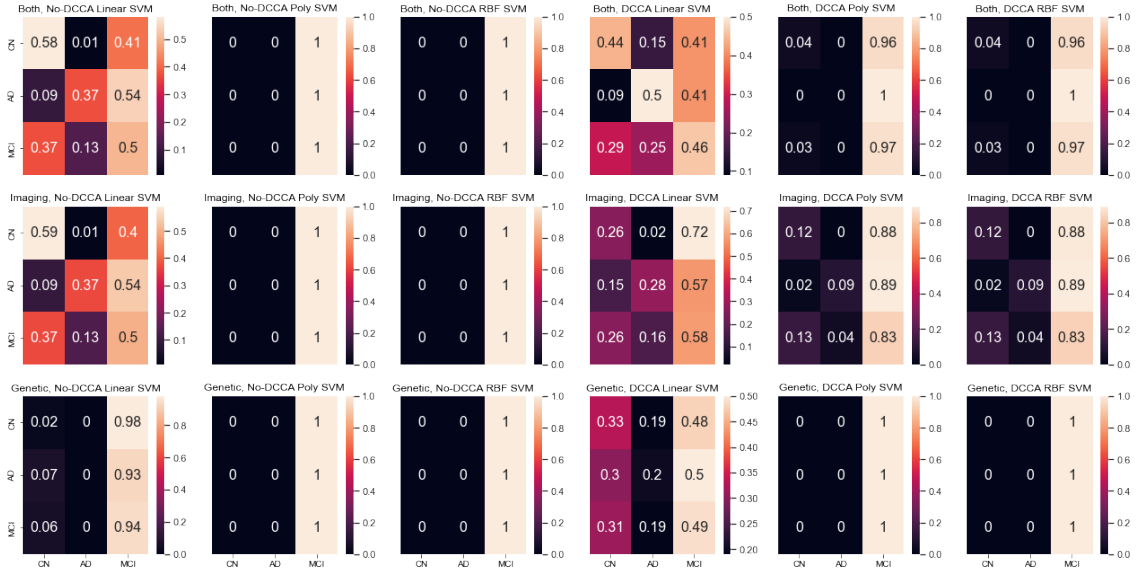
Σχήμα 5.17: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw imaging and MCA transformed genetic data (3 left bar groups) vs using DCCA transformed raw imaging and MCA transformed genetic data (3 right bar groups)*



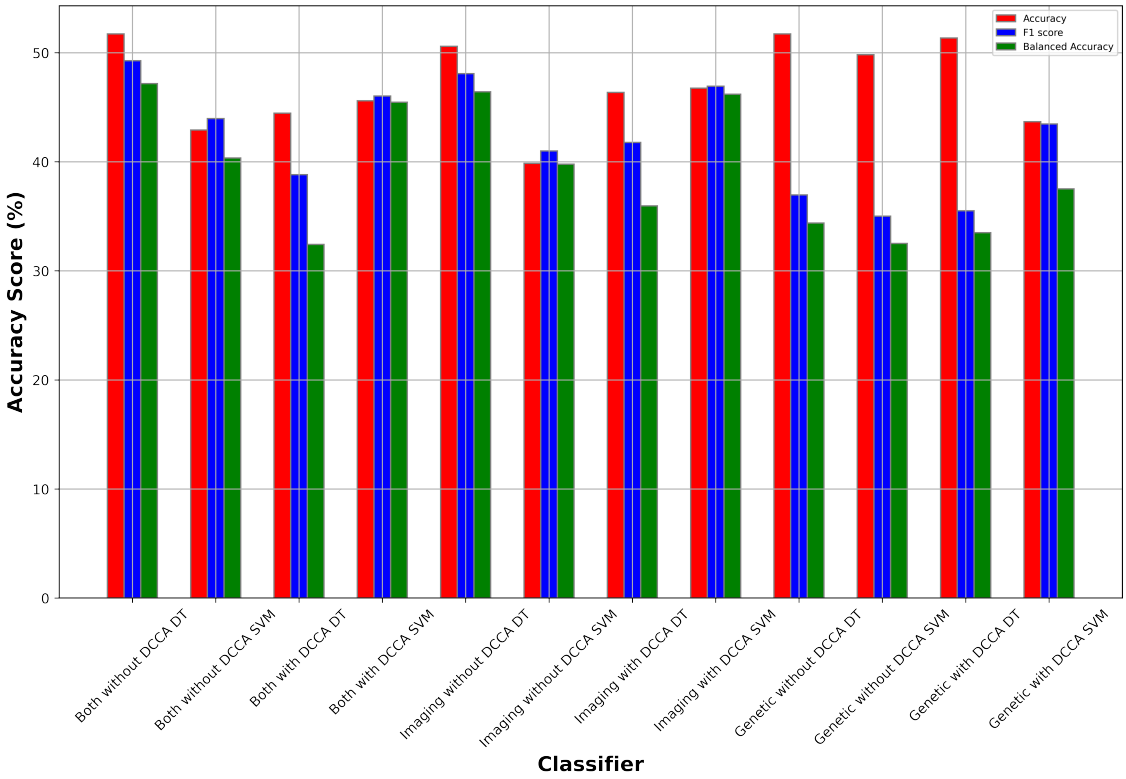
Σχήμα 5.18: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw imaging data and MCA transformed genetic data (3 right bar groups).*



Σχήμα 5.19: *Classification metric scores using only the genetic view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using MCA transformed genetic data (3 left bar groups) vs using the DCCA transformed genetic data, trained on raw imaging data and MCA transformed genetic data (3 right bar groups).*

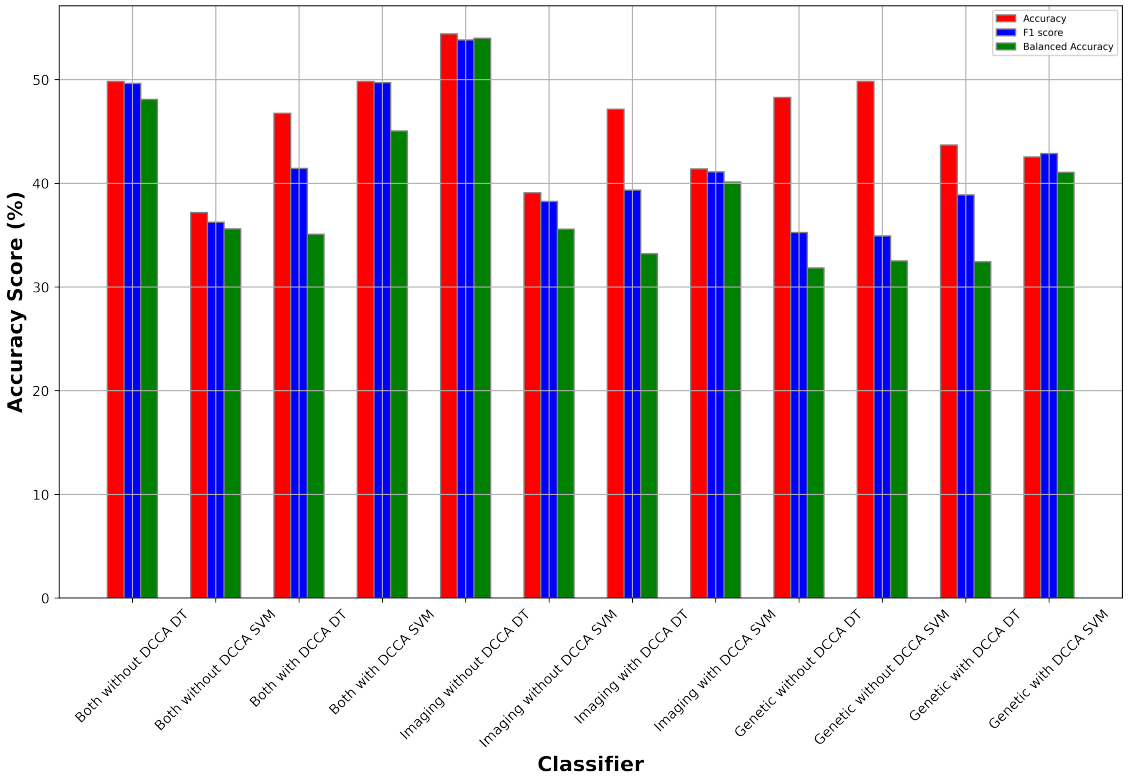


Σχήμα 5.20: *The Confusion Matrices for each class, per model, using both views (top row), only the imaging view (middle row), and only the genetic view (bottom row). The three left columns represent the CM of the raw imaging and MCA transformed genetic data classification, while the three right columns represent the CM of the DCCA transformed data classification.*



Σχήμα 5.21: *Classification metric using Bagging on the MCA transformed imaging and genetic data.*

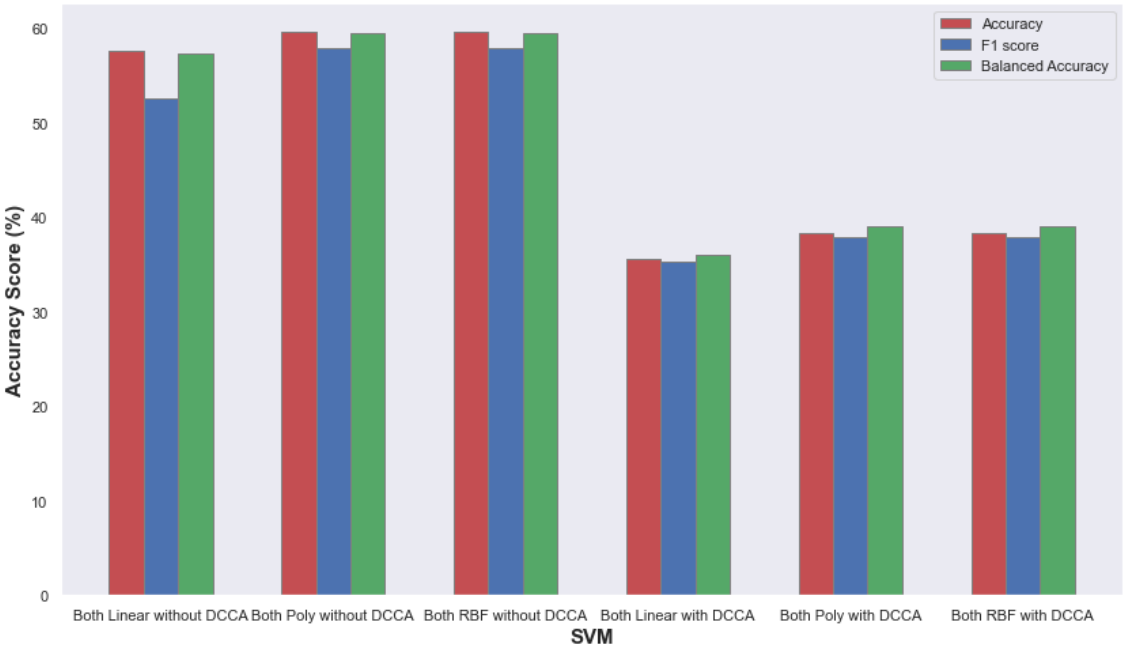
Σχήμα 5.22: *The Confusion Matrices for each class, with Bagging, for the MCA transformed imaging and genetic data.*



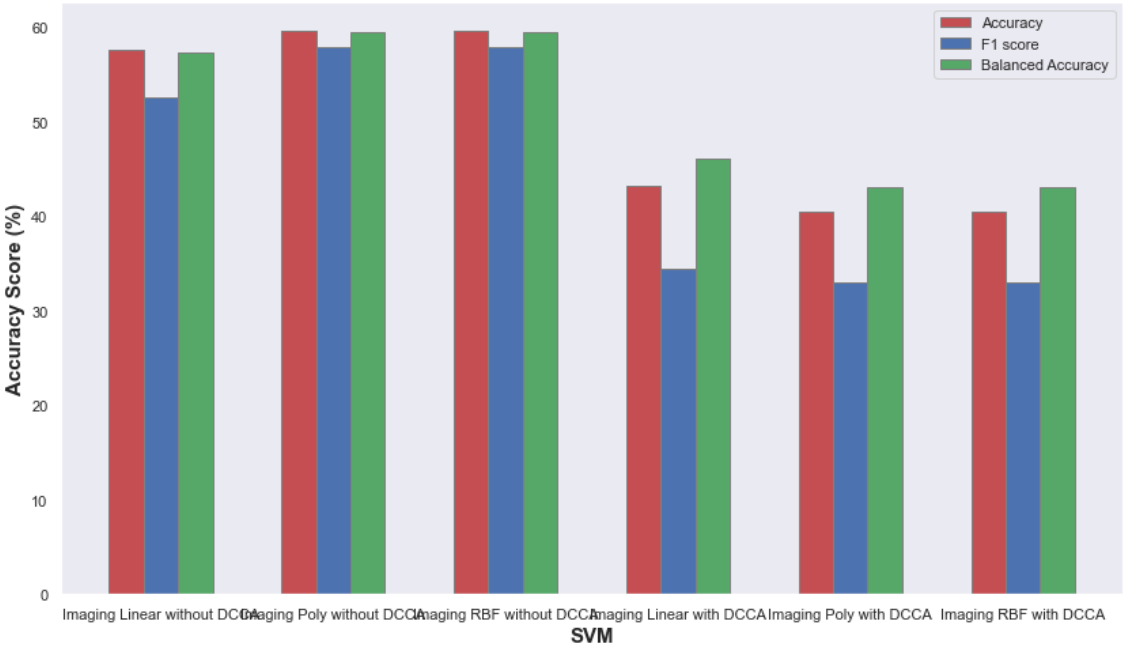
Σχήμα 5.23: *Classification metric using AdaBoost on the MCA transformed imaging and genetic data.*

Σχήμα 5.24: *The Confusion Matrices for each class, with AdaBoost, for the MCA transformed imaging and genetic data.*

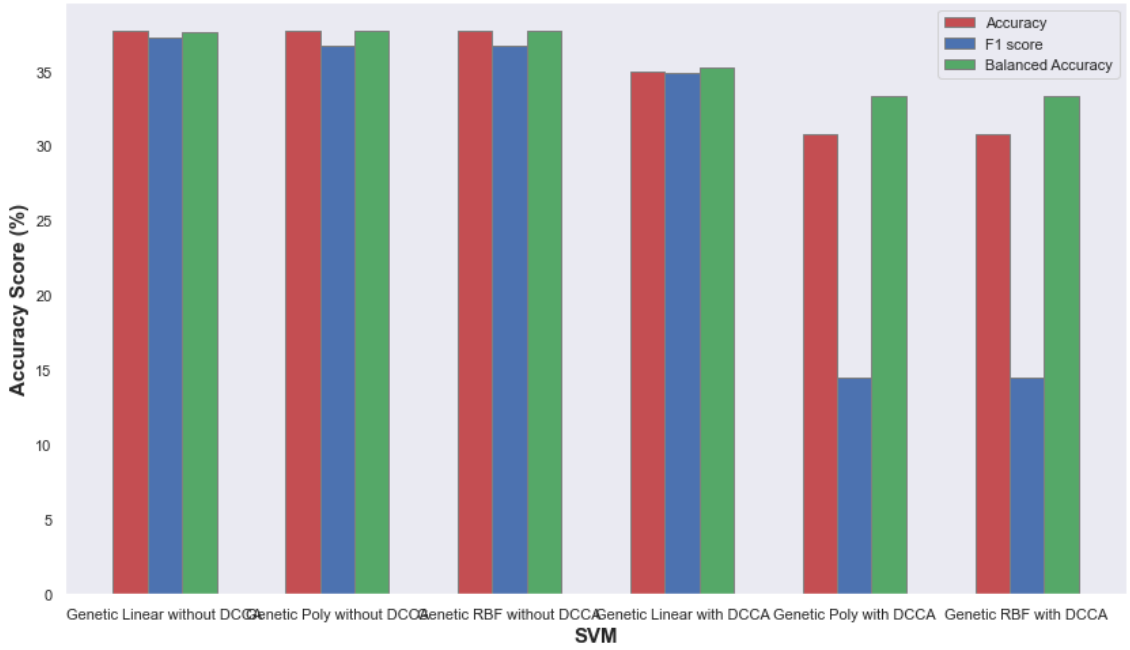
5.2.2 With scaling and balancing:



Σχήμα 5.25: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw imaging and MCA transformed genetic data (3 left bar groups) vs using DCCA transformed raw imaging and MCA transformed genetic data (3 right bar groups)*

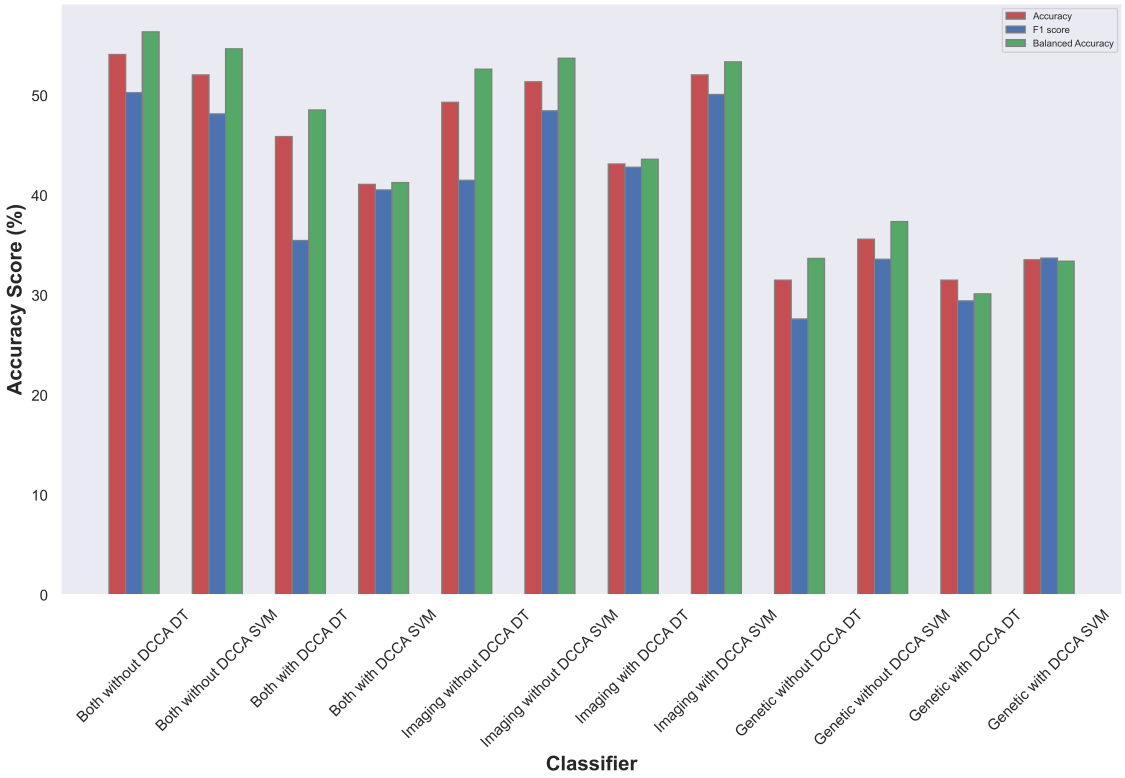


Σχήμα 5.26: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw imaging data and MCA transformed genetic data (3 right bar groups).*



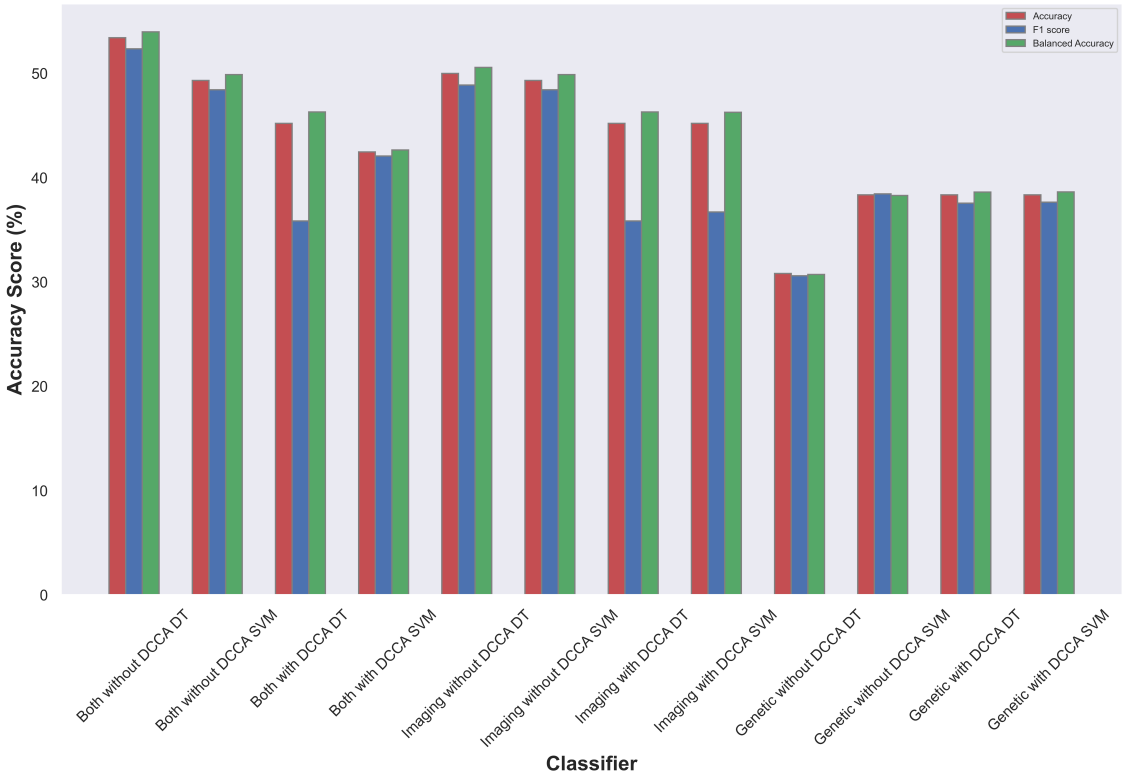
Σχήμα 5.27: *Classification metric scores using only the genetic view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using MCA transformed genetic data (3 left bar groups) vs using the DCCA transformed genetic data, trained on raw imaging data and MCA transformed genetic data (3 right bar groups).*

Σχήμα 5.28: *The Confusion Matrices for each class, per model, using both views (top row), only the imaging view (middle row), and only the genetic view (bottom row). The three left columns represent the CM of the raw imaging and MCA transformed genetic data classification, while the three right columns represent the CM of the DCCA transformed data classification.*

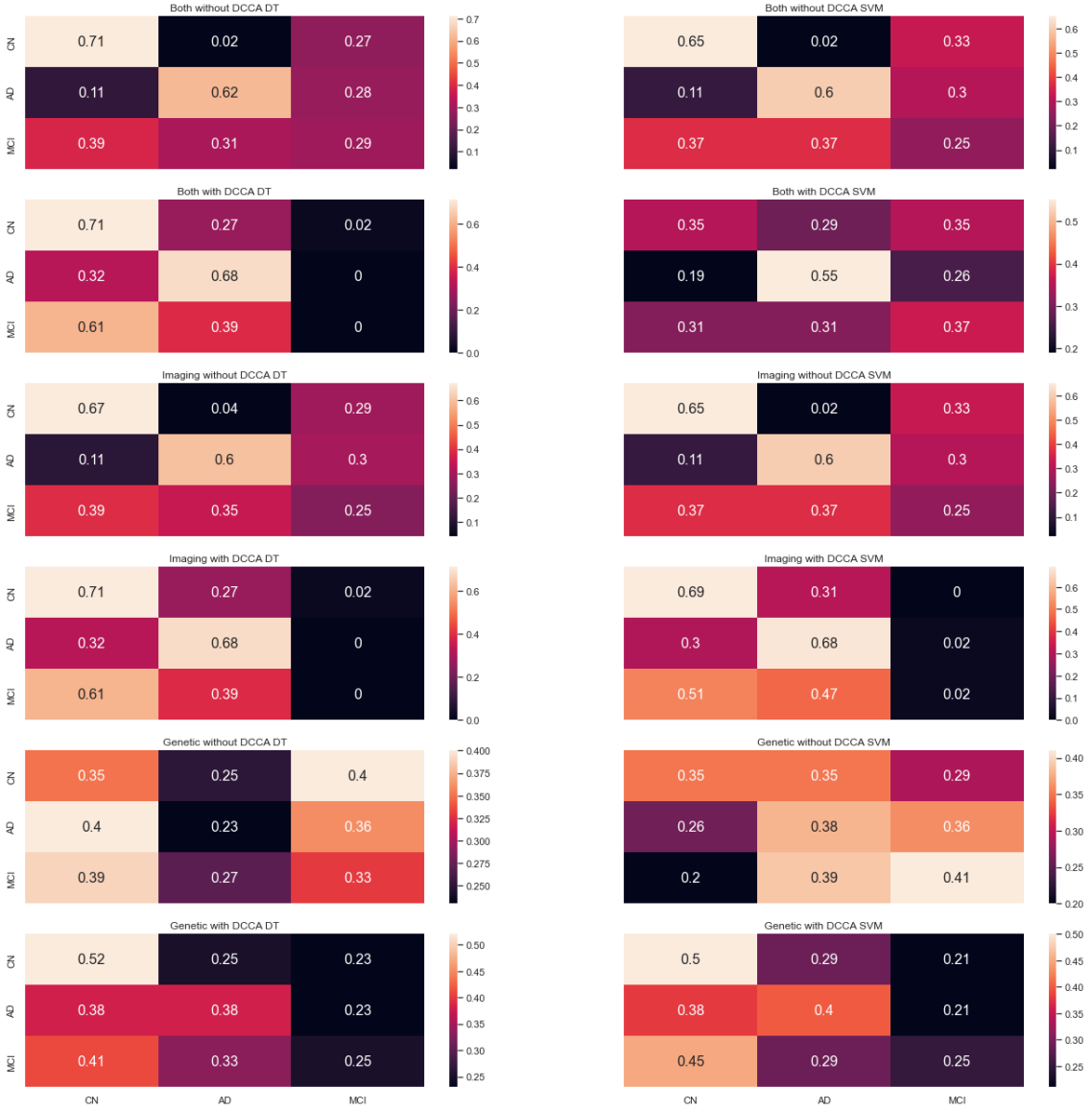


Σχήμα 5.29: *Classification metric using Bagging on the MCA transformed imaging and genetic data.*

Σχήμα 5.30: *The Confusion Matrices for each class, with Bagging, for the MCA transformed imaging and genetic data.*



Σχήμα 5.31: *Classification metric using AdaBoost on the MCA transformed imaging and genetic data.*

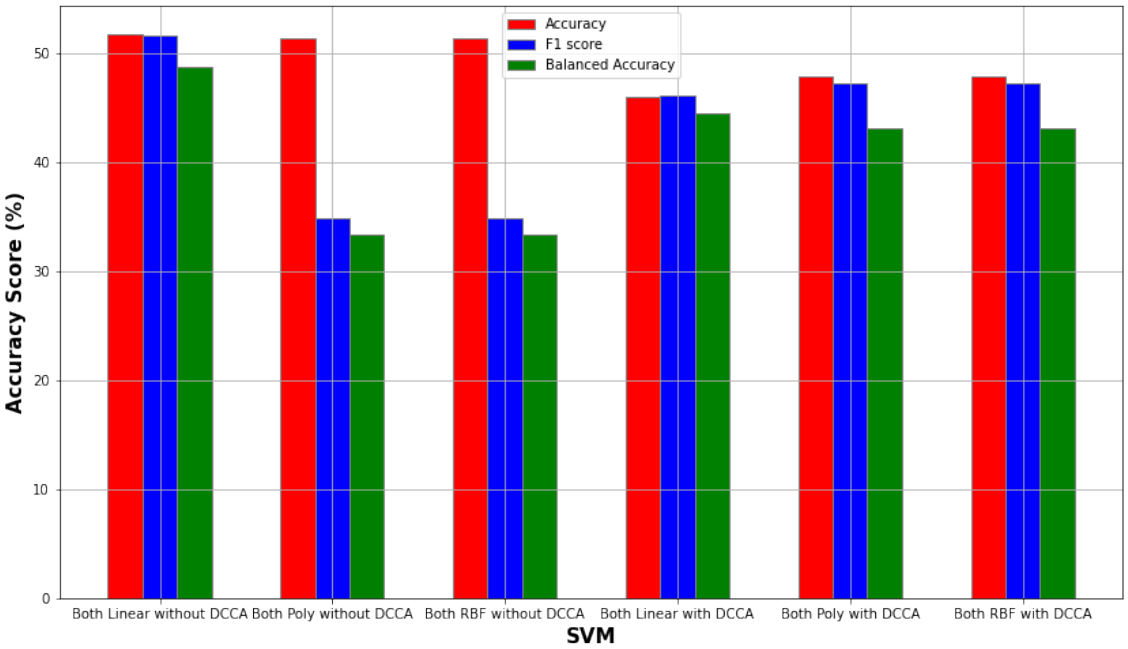


Σχήμα 5.32: *The Confusion Matrices for each class, with AdaBoost, for the MCA transformed imaging and genetic data.*

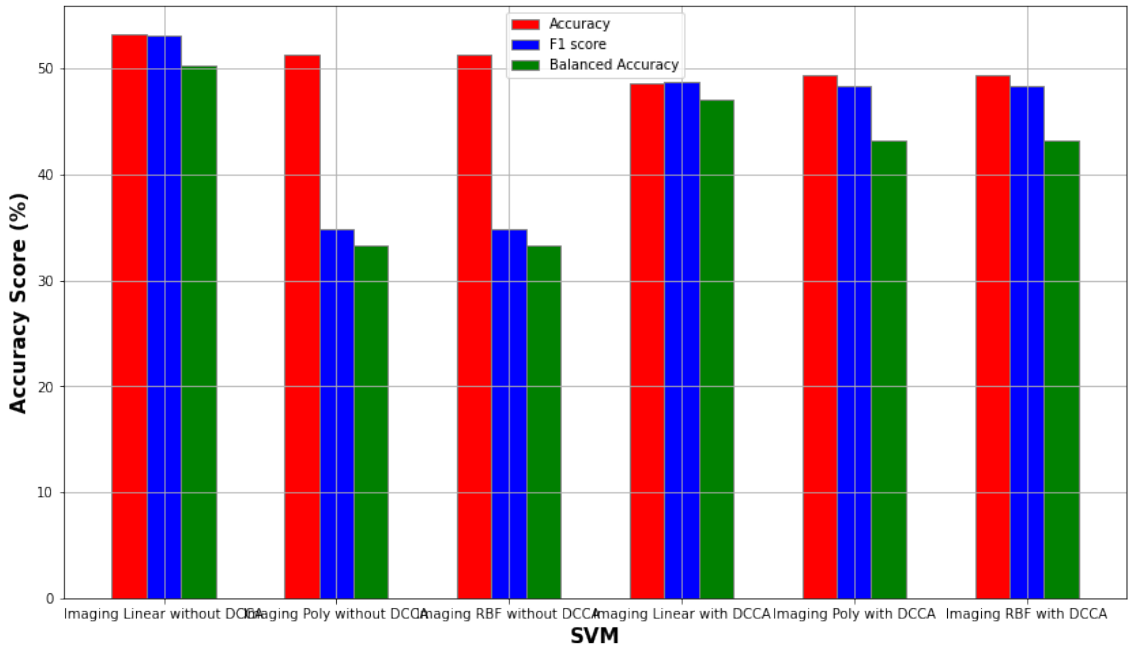
5.3 OPNMF vs OPNMF - DCCA

Furthermore, the same logic was applied to exploring the effect that OPNMF has to the task of classification. To that extent, we compare the classification results of (a) the data after the imaging view has been transformed with OPNMF and (b) the DCCA transformed data, trained on the raw genetic data and the OPNMF transformed imaging data.

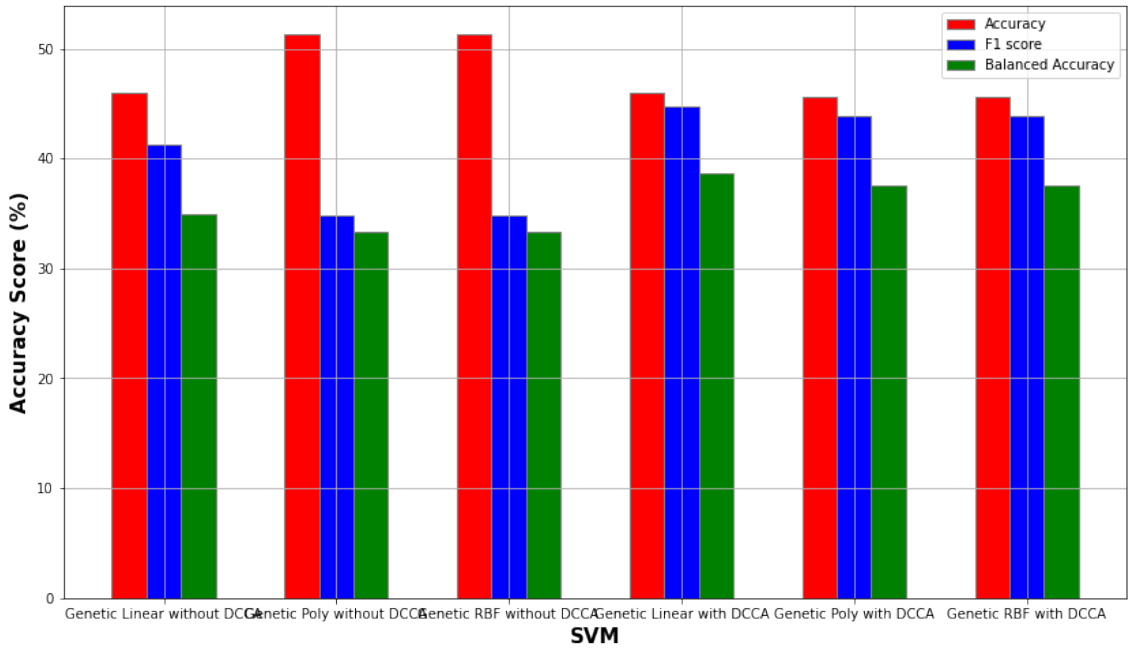
5.3.1 Without scaling or balancing:



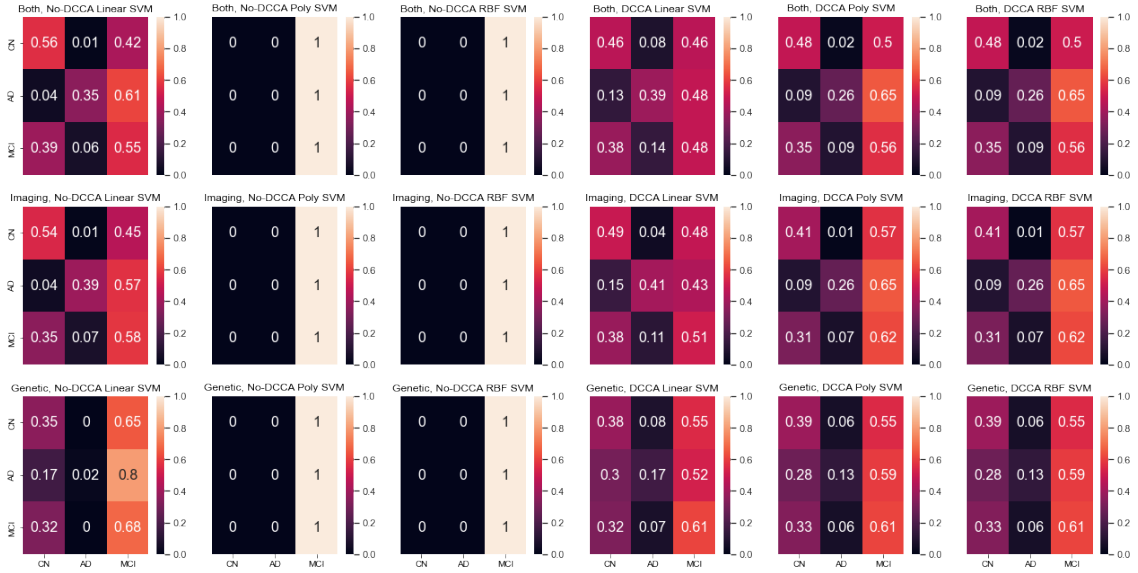
Σχήμα 5.33: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw genetic and OPNMF transformed imaging data (3 left bar groups) vs using DCCA transformed raw genetic and OPNMF transformed imaging data (3 right bar groups)*



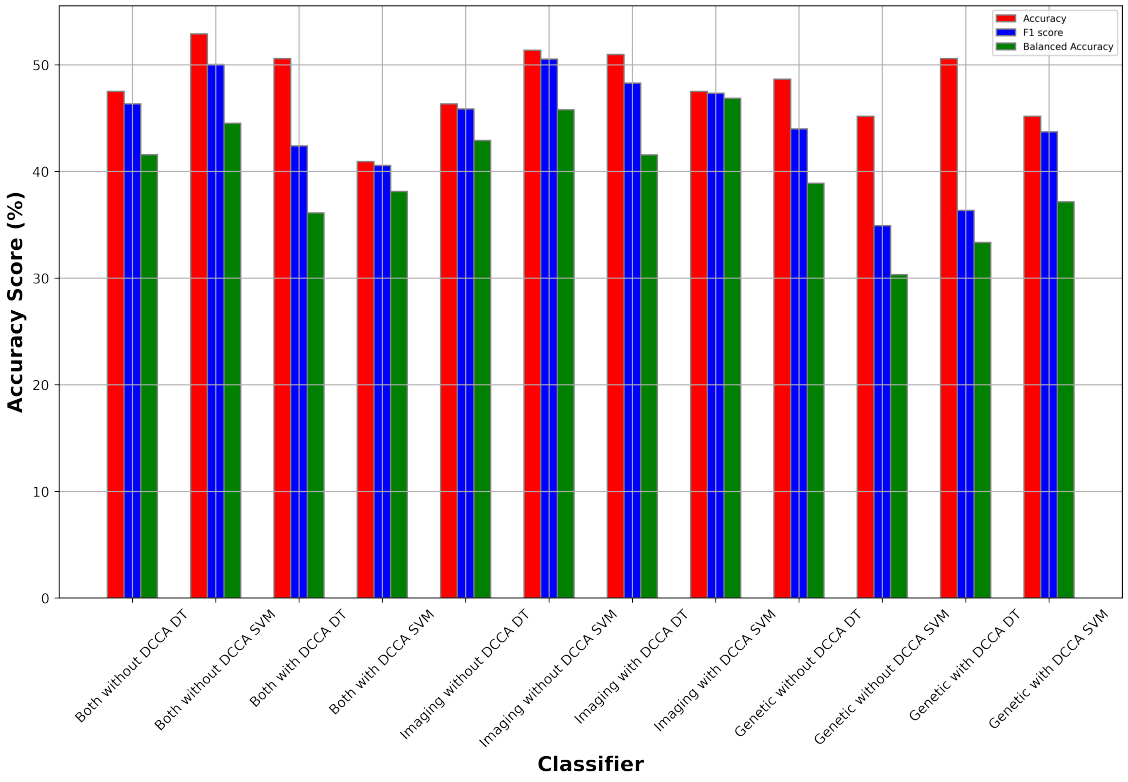
Σχήμα 5.34: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using OPNMF transformed imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw genetic data and OPNMF transformed genetic data (3 right bar groups).*



Σχήμα 5.35: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using OPNMF transformed imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw genetic data and OPNMF transformed genetic data (3 right bar groups).*

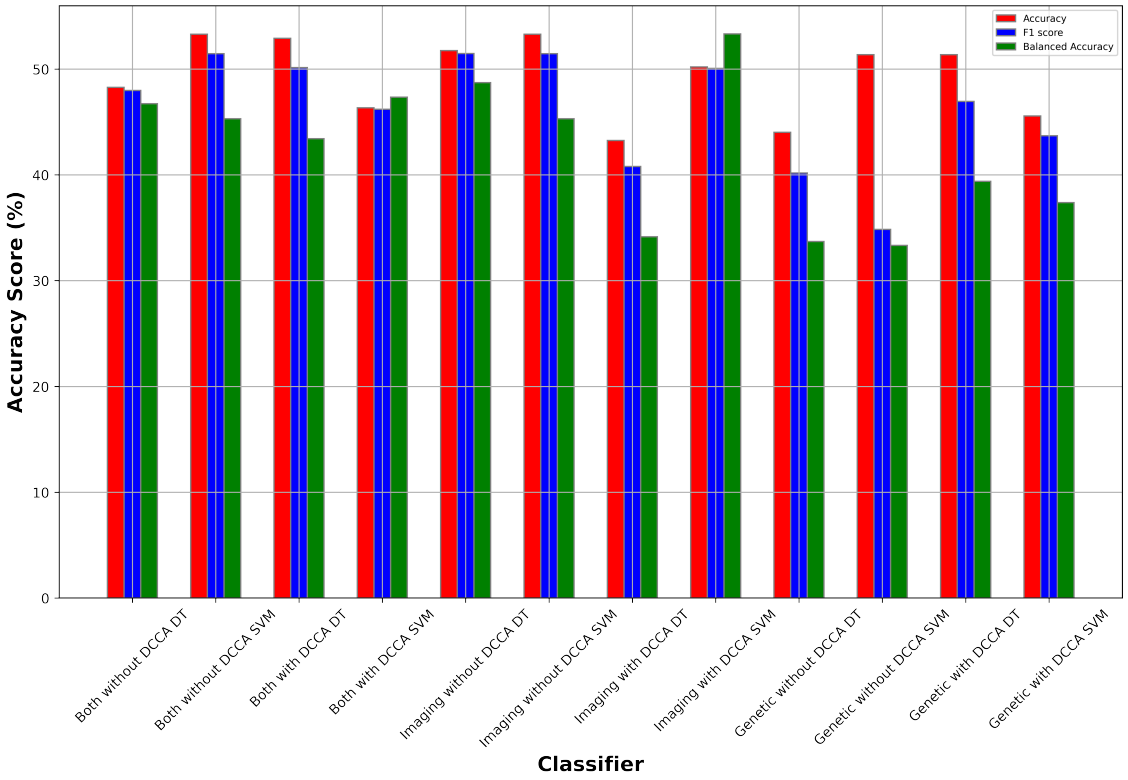


Σχήμα 5.36: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using OPNMF transformed imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw genetic data and OPNMF transformed genetic data (3 right bar groups).*



Σχήμα 5.37: *Classification metric using Bagging on the OPNMF transformed imaging and genetic data.*

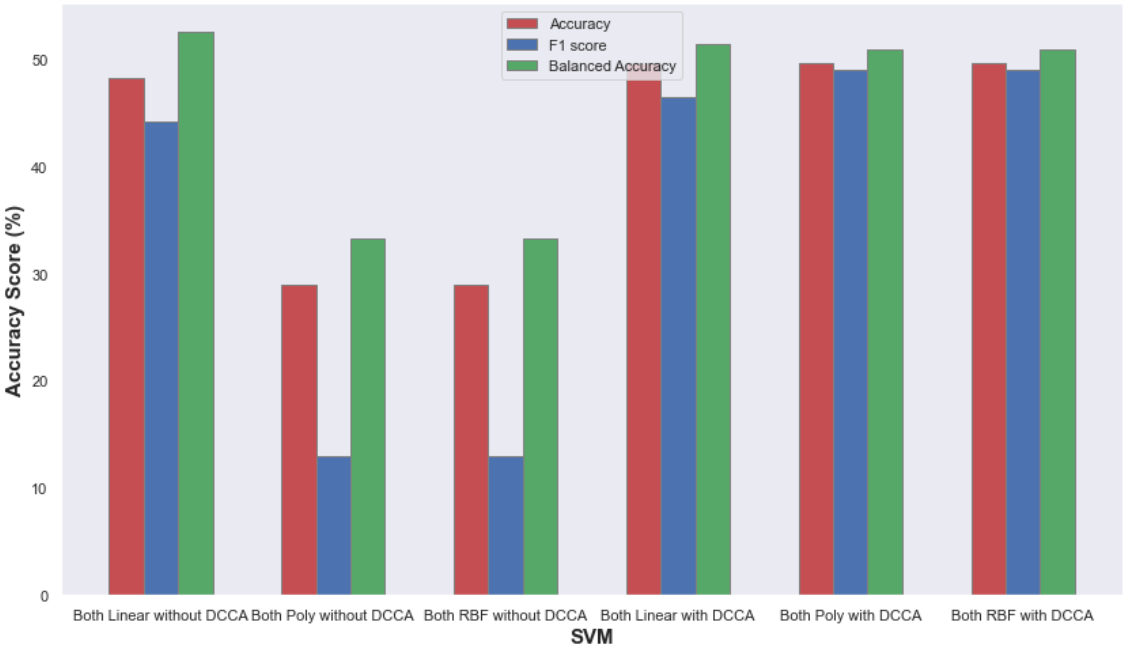
Σχήμα 5.38: *The Confusion Matrices for each class, with Bagging, for the OPNMF transformed imaging and genetic data.*



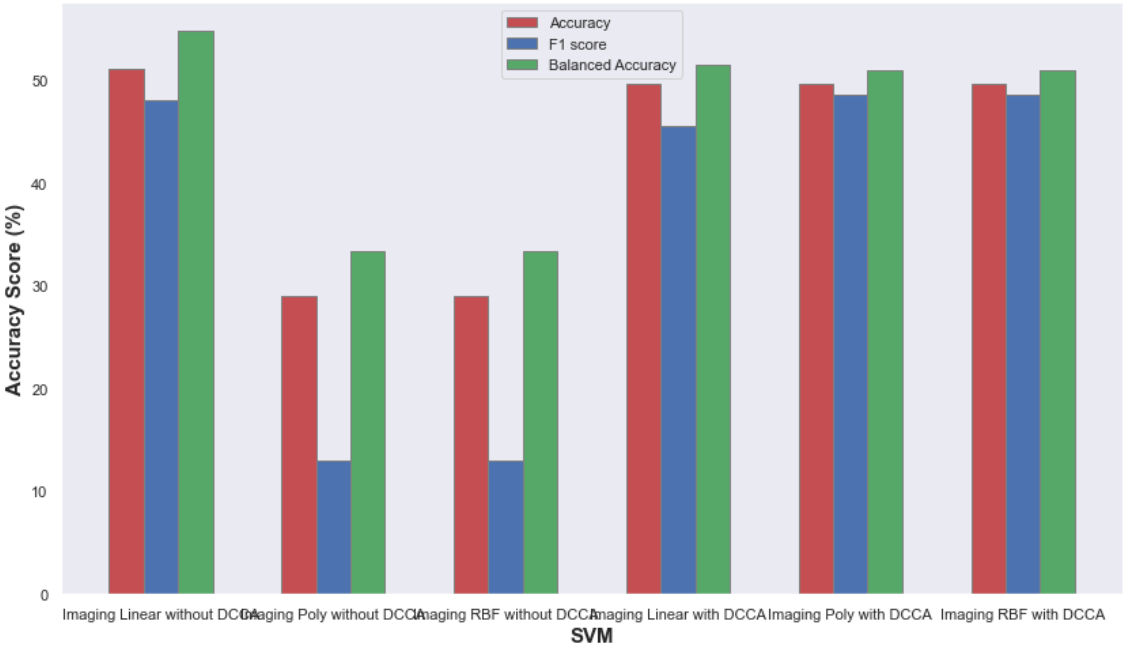
Σχήμα 5.39: *Classification metric using AdaBoost on the OPNMF transformed imaging and genetic data.*

Σχήμα 5.40: *The Confusion Matrices for each class, with AdaBoost, for the OPNMF transformed imaging and genetic data.*

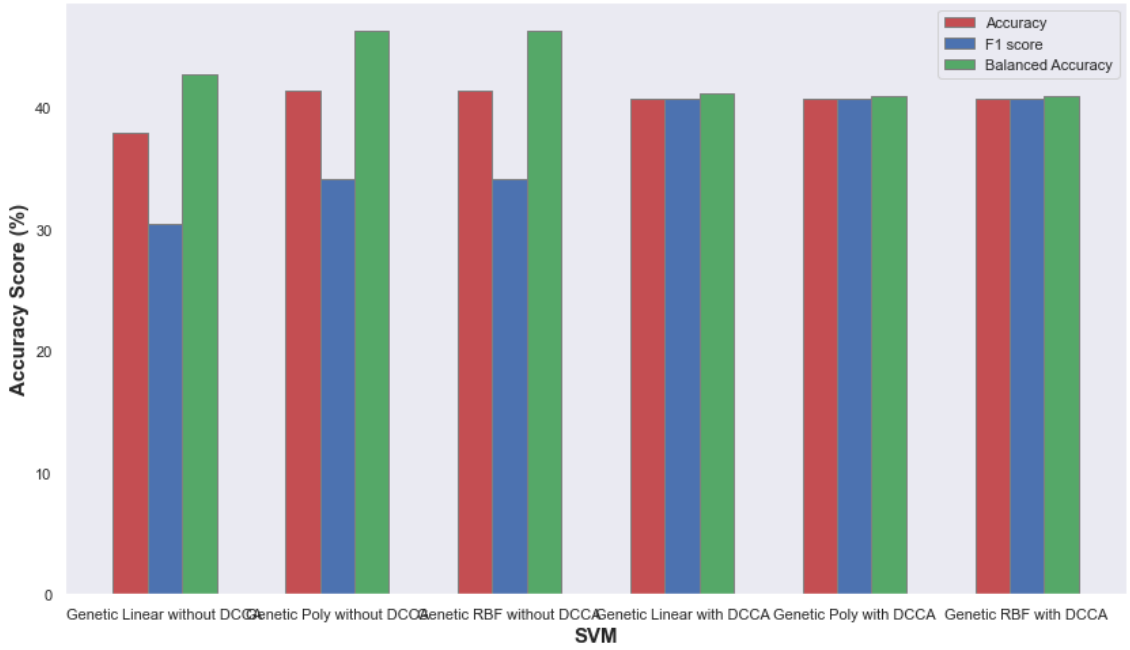
5.3.2 With scaling and balancing:



Σχήμα 5.41: *Classification metric scores using Both views (Imaging and Genetic), on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw genetic and OPNMF transformed imaging data (3 left bar groups) vs using DCCA transformed raw genetic and OPNMF transformed imaging data (3 right bar groups)*

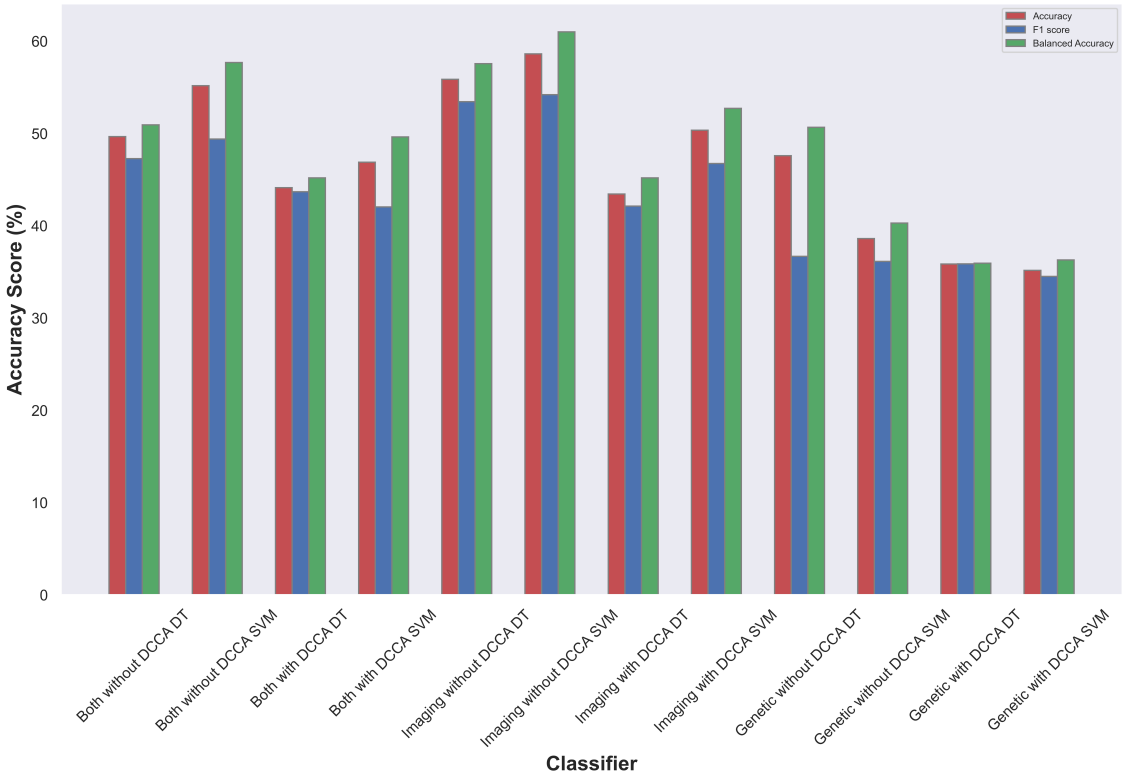


Σχήμα 5.42: *Classification metric scores using only the Imaging view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using OPNMF transformed imaging data (3 left bar groups) vs using the DCCA transformed imaging data, trained on raw genetic data and OPNMF transformed genetic data (3 right bar groups).*

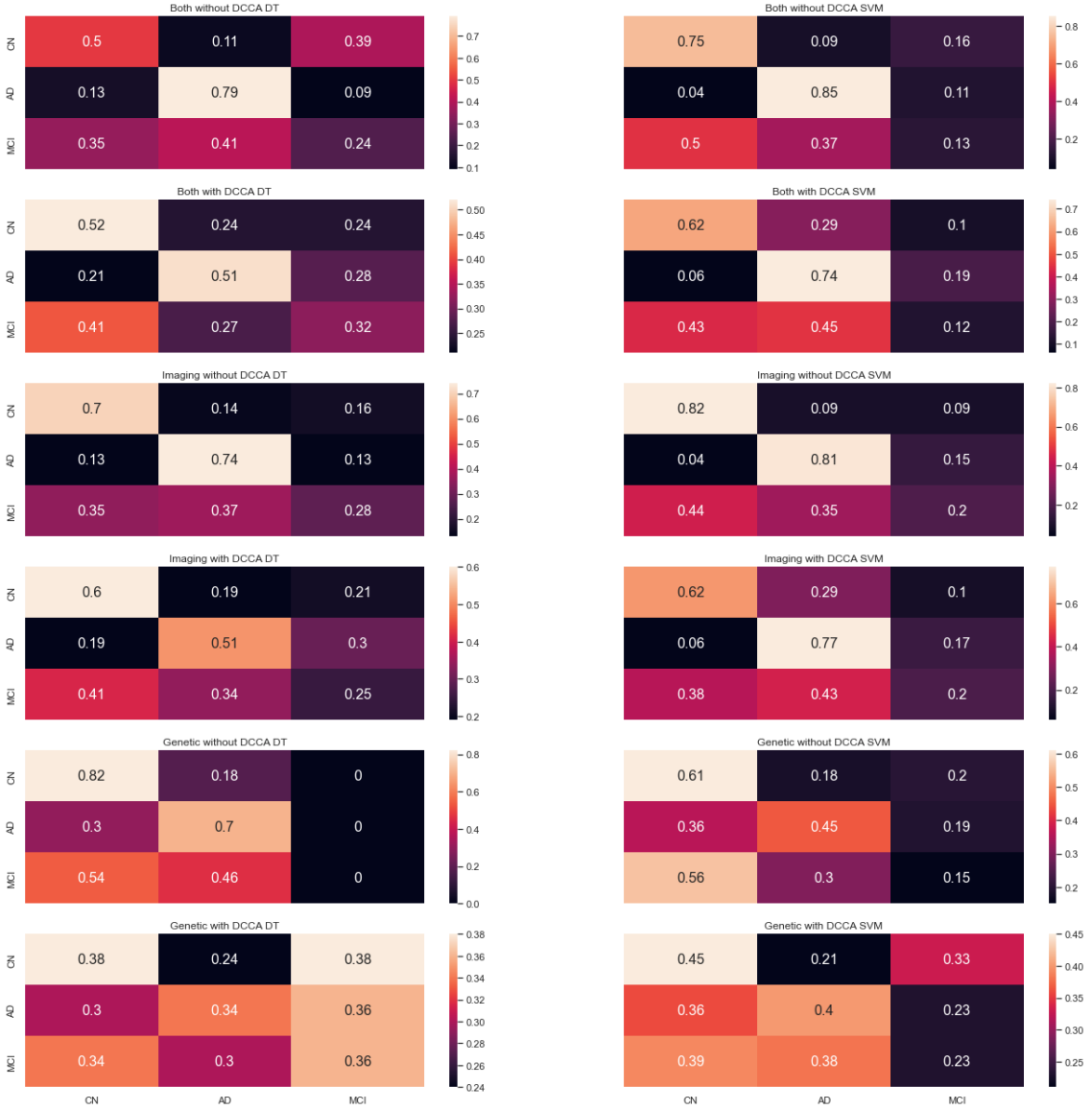


Σχήμα 5.43: *Classification metric scores using only the genetic view, on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using raw genetic data (3 left bar groups) vs using the DCCA transformed genetic data, trained on raw genetic data and OPNMF transformed imaging data (3 right bar groups).*

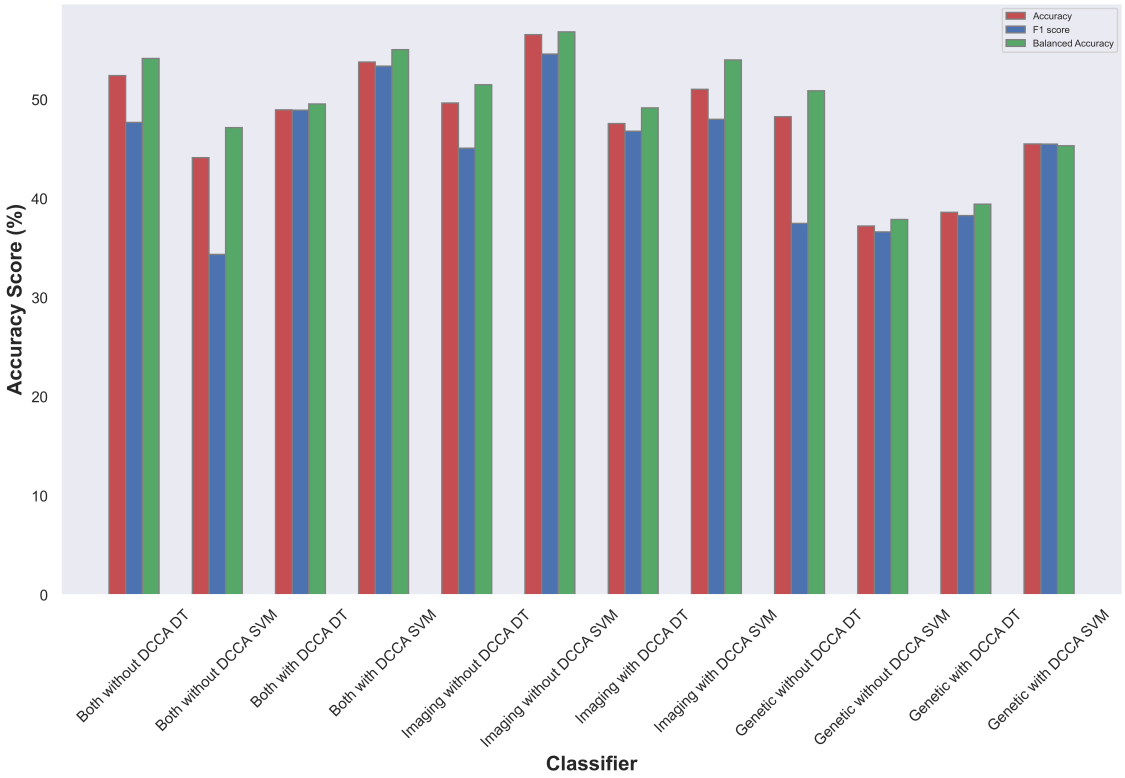
Σχήμα 5.44: *The Confusion Matrices for each class, per model, using both views (top row), only the imaging view (middle row), and only the genetic view (bottom row). The three left columns represent the CM of the raw genetic and OPNMF transformed imaging data classification, while the three right columns represent the CM of the DCCA transformed data classification.*



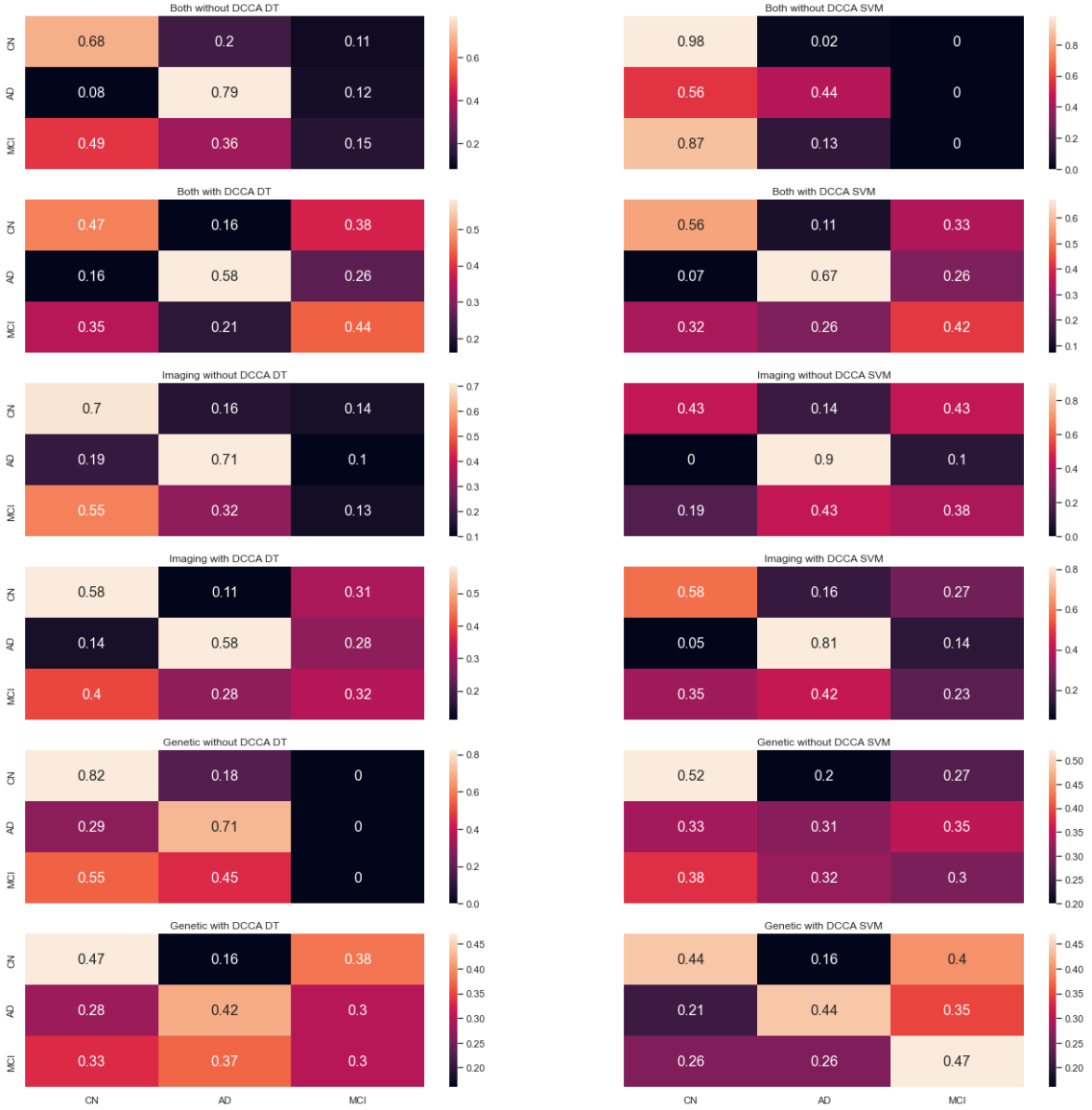
Σχήμα 5.45: *Classification metric using Bagging on the OPNMF transformed imaging and genetic data.*



Σχήμα 5.46: *The Confusion Matrices for each class, with Bagging, for the OPNMF transformed imaging and genetic data.*



Σχήμα 5.47: *Classification metric using AdaBoost on the OPNMF transformed imaging and genetic data.*

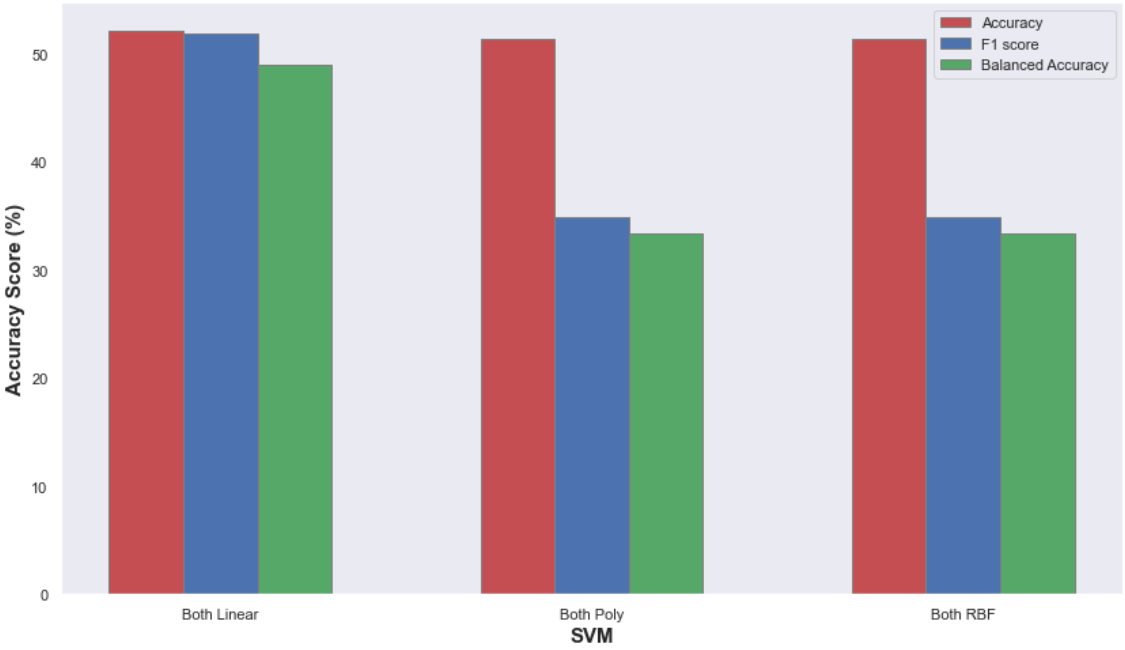


Σχήμα 5.48: *The Confusion Matrices for each class, with AdaBoost, for the OPNMF transformed imaging and genetic data.*

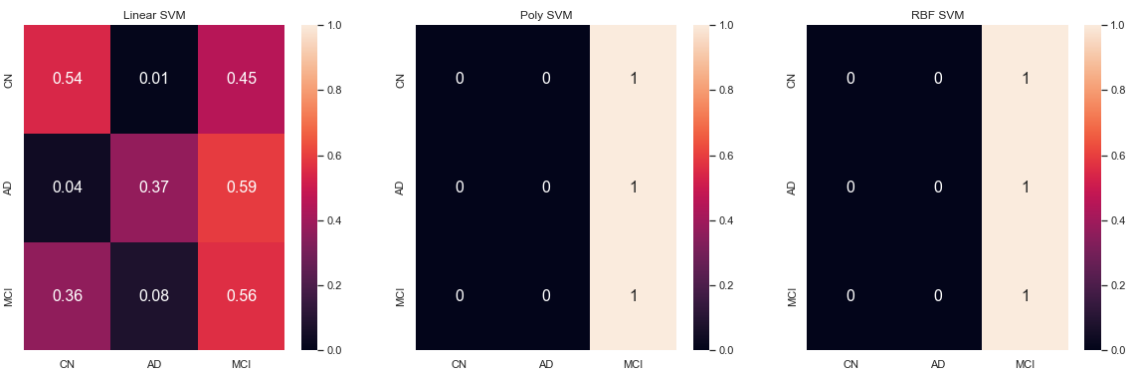
5.4 MCA OPNMF

Combining the two previous techniques, we now apply MCA to the genetic data, and OPNMF to the imaging data, and combine the two transformed views for the classification task.

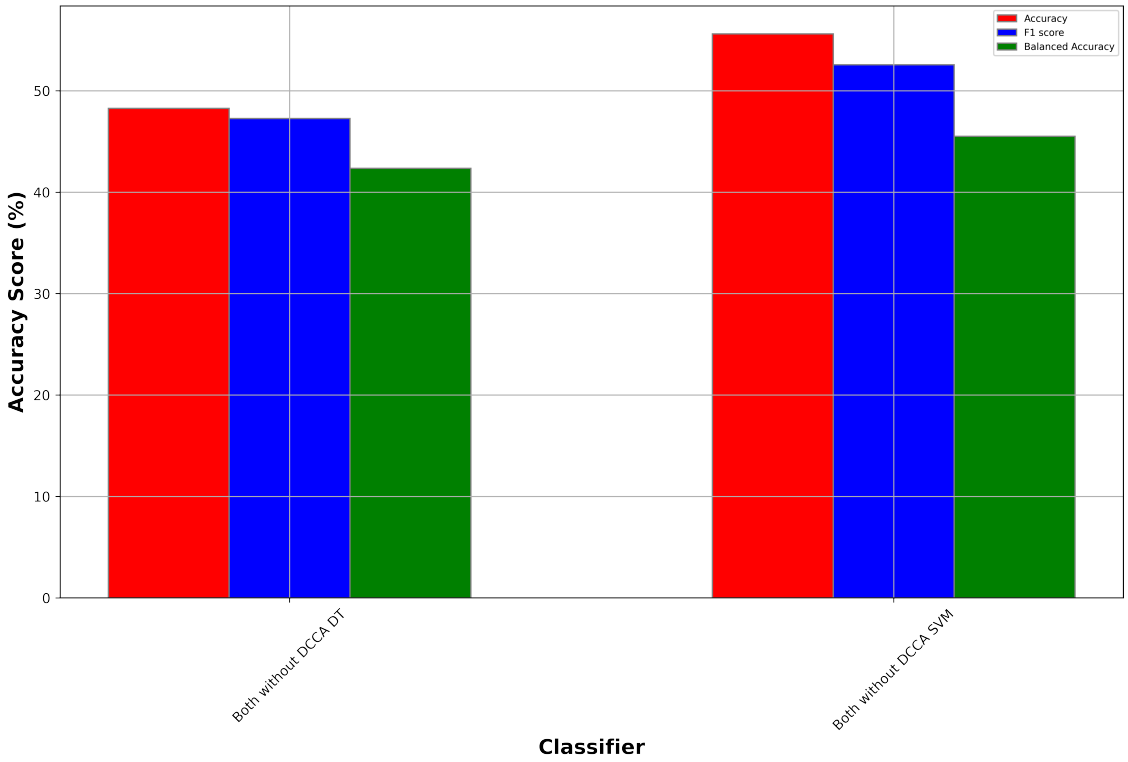
5.4.1 Without scaling or balancing:



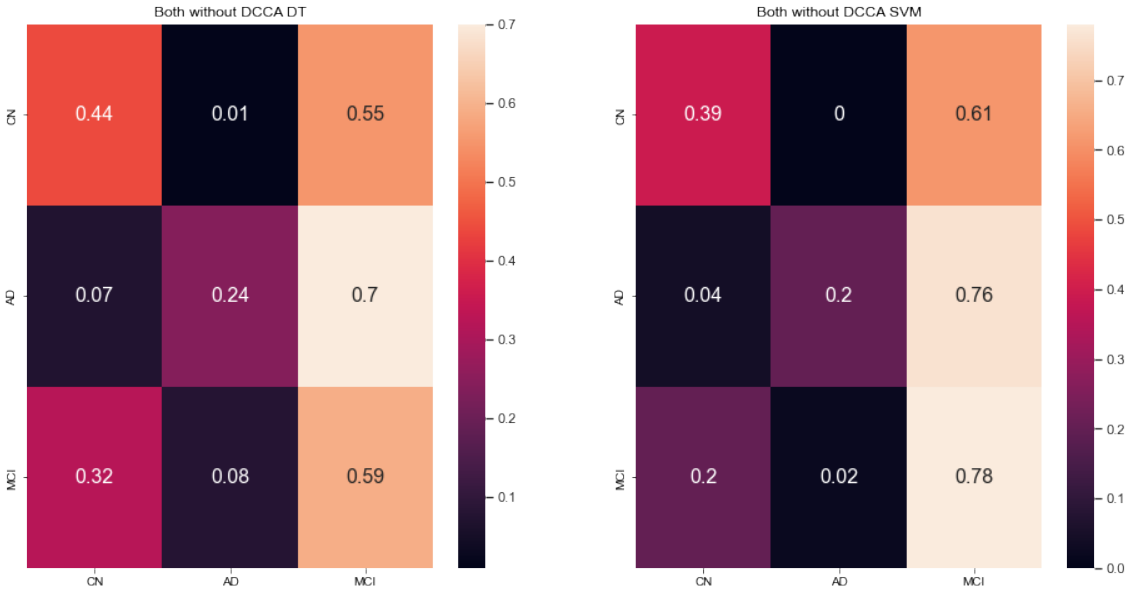
Σχήμα 5.49: *Classification metric using both views (imaging and genetic) on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using the MCA transformed genetic and OPNMF transformed imaging data.*



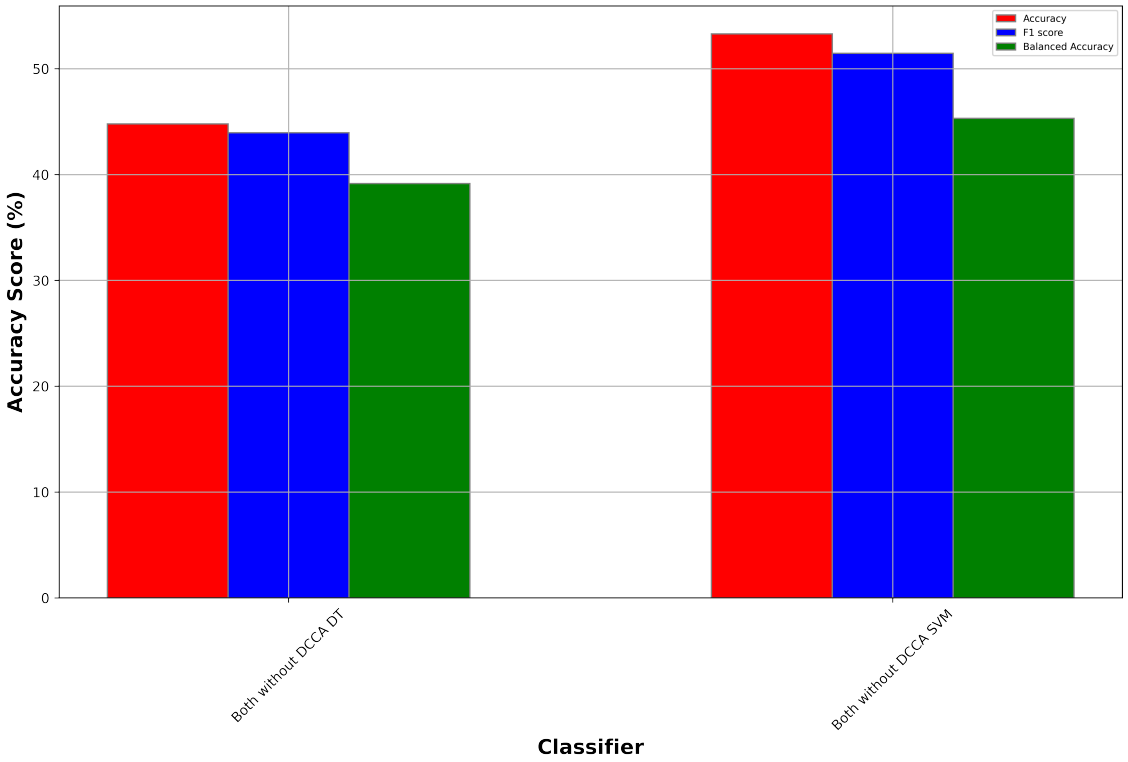
Σχήμα 5.50: *The Confusion Matrices for each class, per SVM kernel, for the MCA transformed genetic and OPNMF transformed imaging data.*



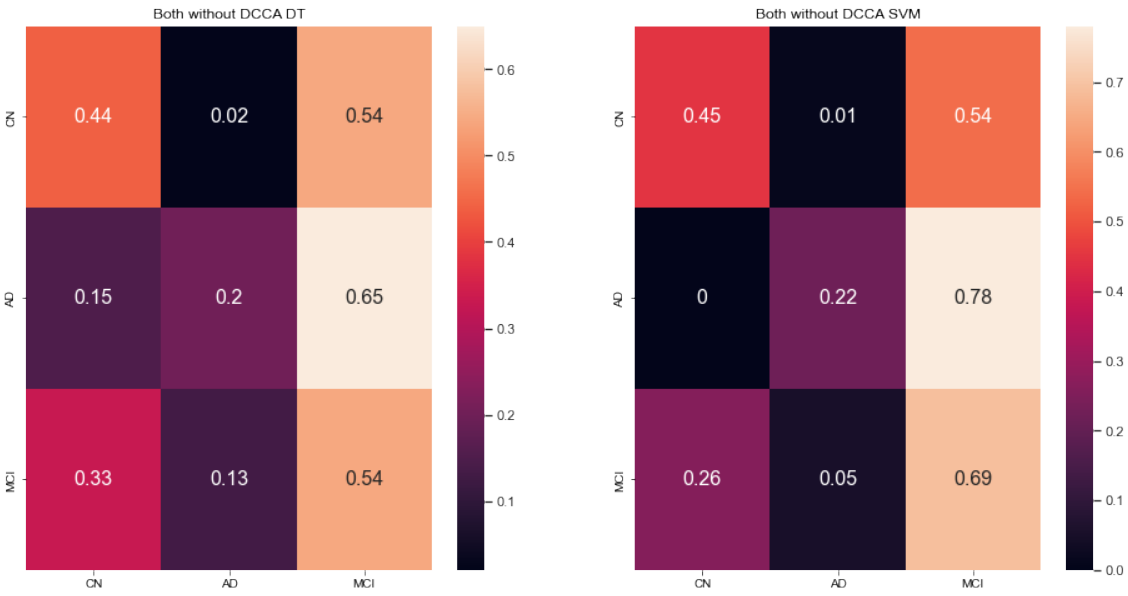
Σχήμα 5.51: *Classification metric using Bagging on the MCA and OPNMF transformed imaging and genetic data.*



Σχήμα 5.52: *The Confusion Matrices for each class, with Bagging, for the MCA and OPNMF transformed imaging and genetic data.*

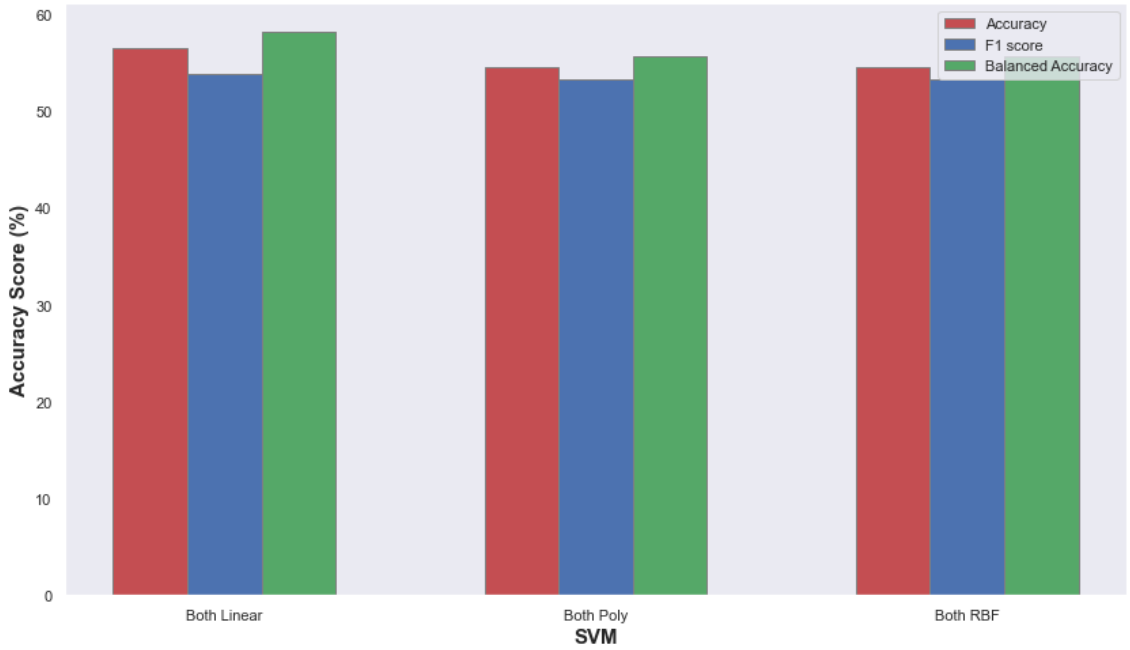


Σχήμα 5.53: *Classification metric using AdaBoost on the MCA and OPNMF transformed imaging and genetic data.*

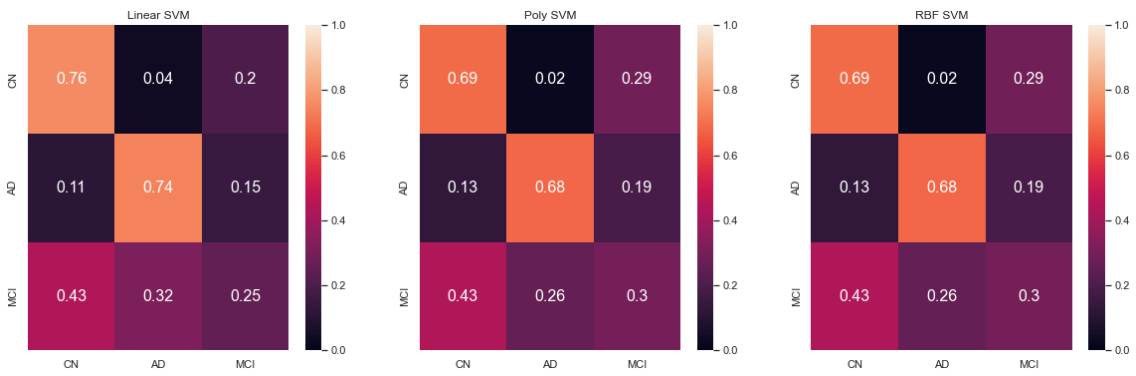


Σχήμα 5.54: *The Confusion Matrices for each class, with AdaBoost, for the MCA and OPNMF transformed imaging and genetic data.*

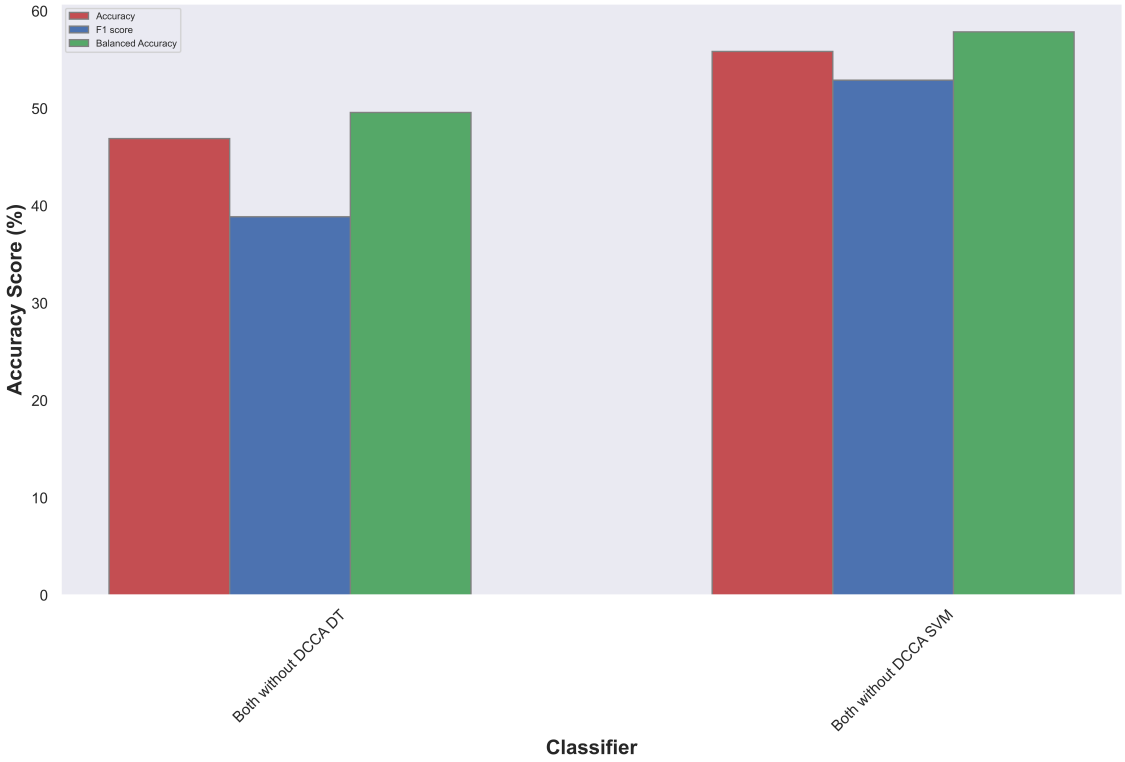
5.4.2 With scaling and balancing:



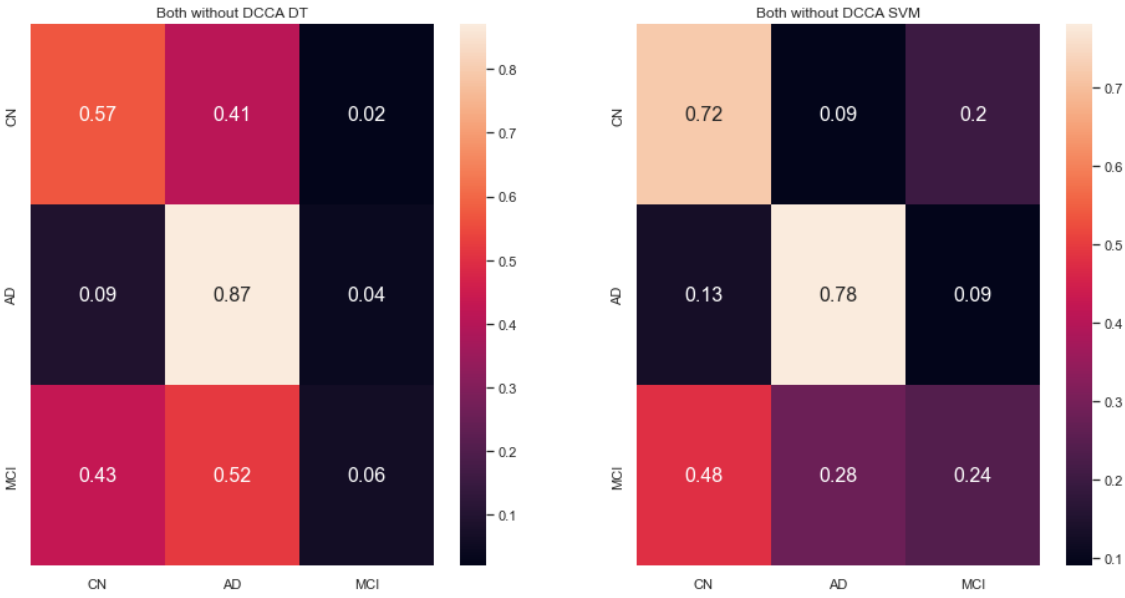
Σχήμα 5.55: *Classification metric using both views (imaging and genetic) on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using the MCA transformed genetic and OPNMF transformed imaging data.*



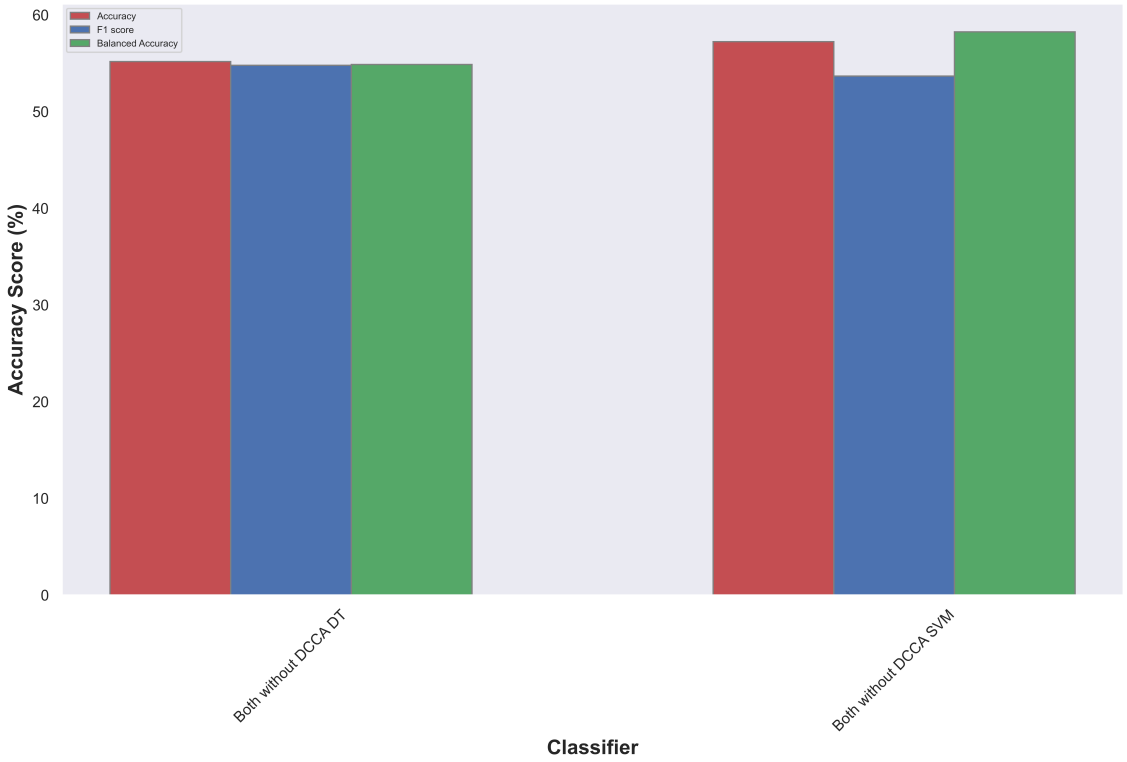
Σχήμα 5.56: *The Confusion Matrices for each class, per SVM kernel, for the MCA transformed genetic and OPNMF transformed imaging data.*



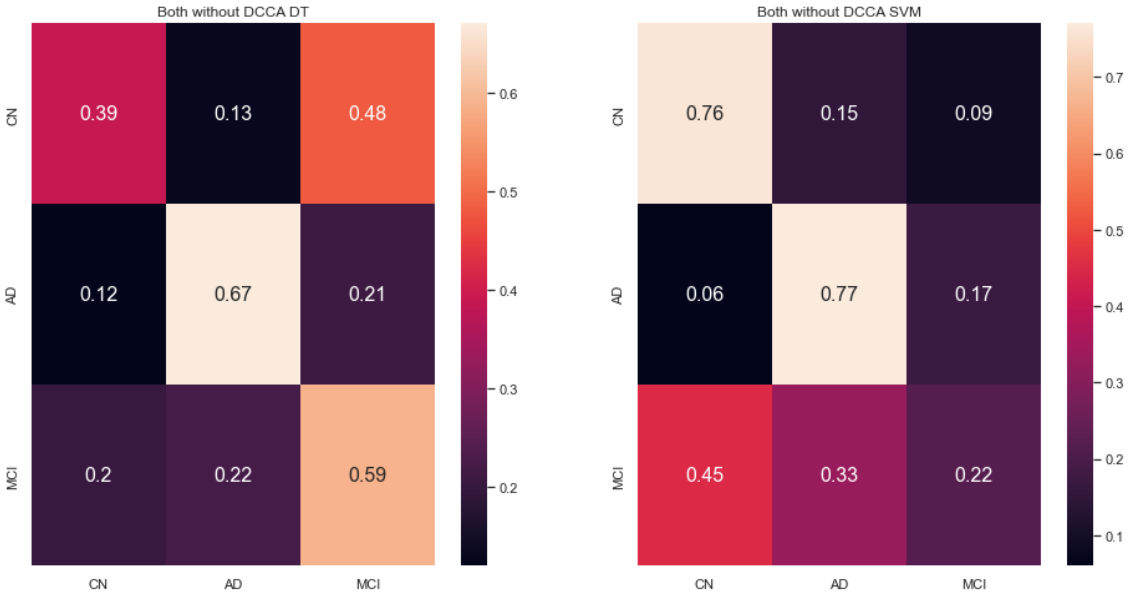
Σχήμα 5.57: *Classification metric using Bagging on the MCA and OPNMF transformed imaging and genetic data.*



Σχήμα 5.58: *The Confusion Matrices for each class, with Bagging, for the MCA and OPNMF transformed imaging and genetic data.*



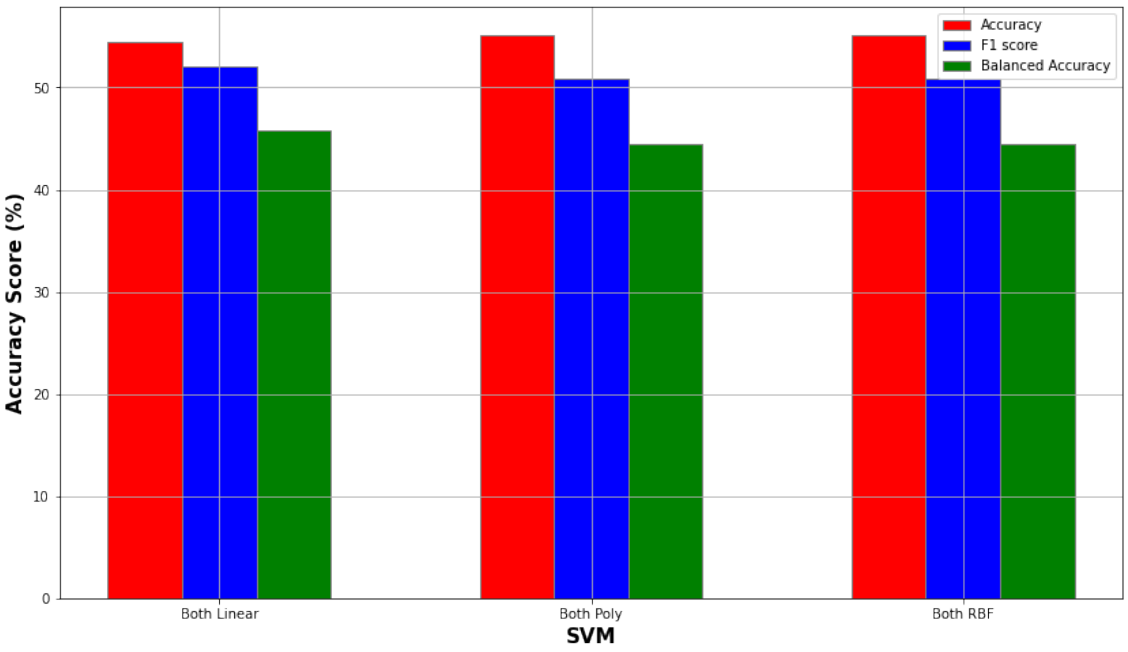
Σχήμα 5.59: *Classification metric using AdaBoost on the MCA and OPNMF transformed imaging and genetic data.*



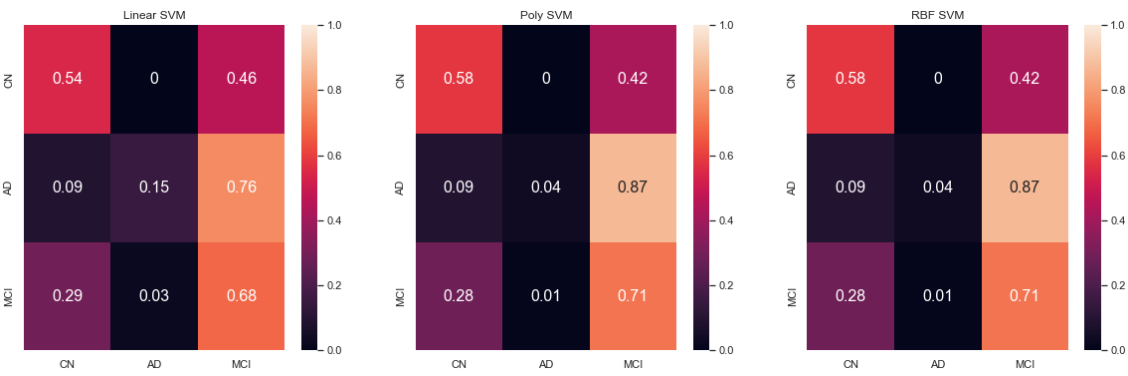
Σχήμα 5.60: *The Confusion Matrices for each class, with AdaBoost, for the MCA and OPNMF transformed imaging and genetic data.*

5.5 FAMD

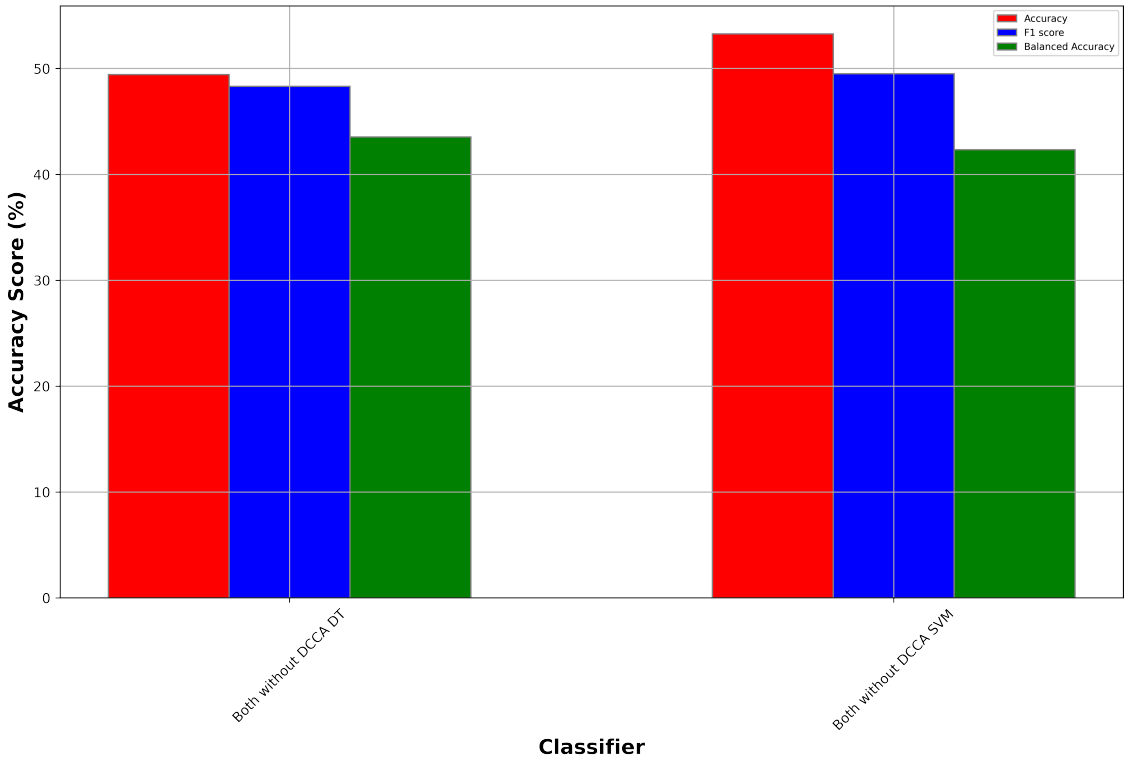
5.5.1 Without scaling or balancing:



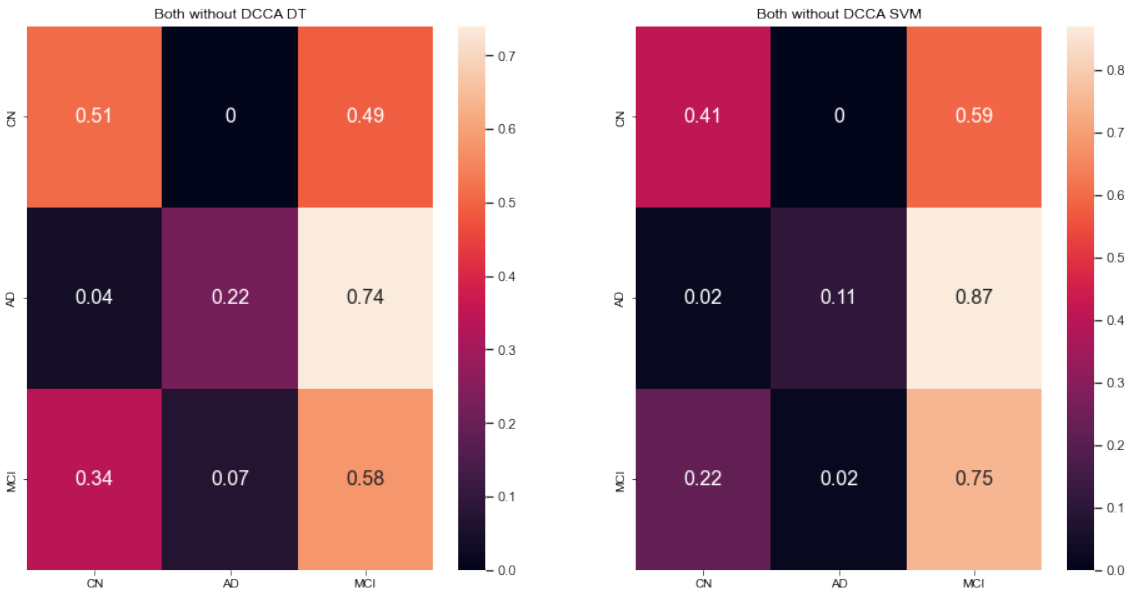
Σχήμα 5.61: *Classification metric using both views (imaging and genetic) on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using the FAMD transformed imaging and genetic data.*



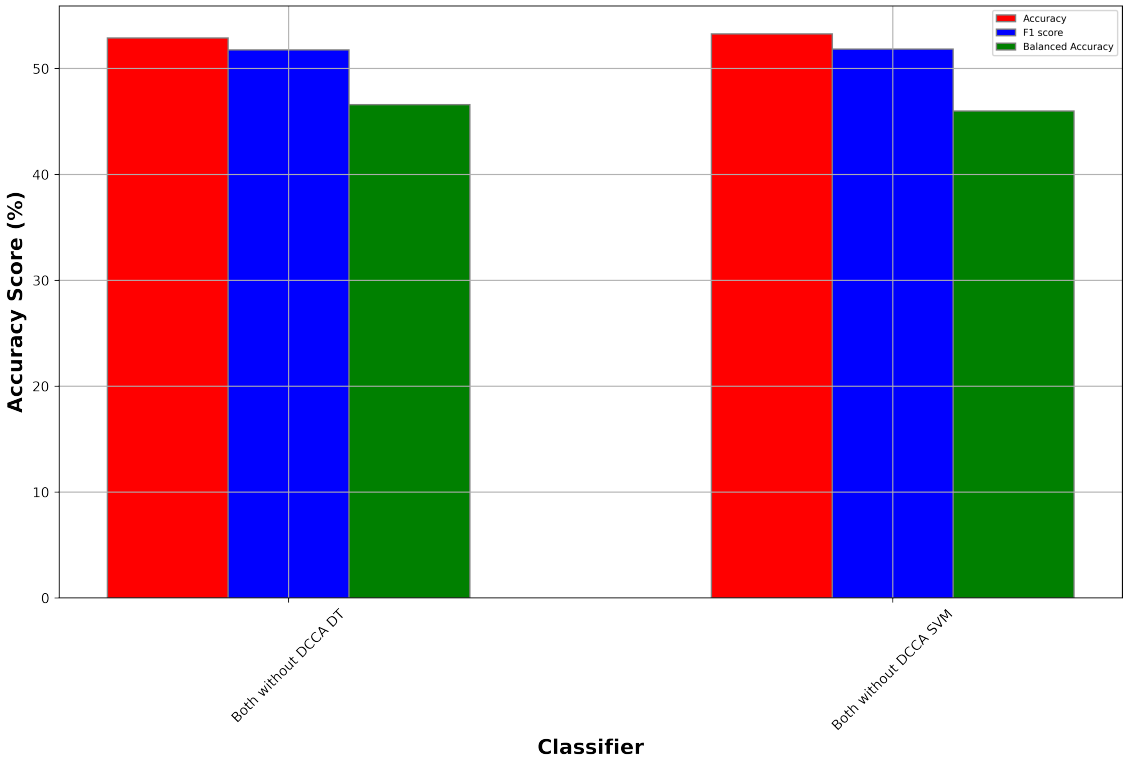
Σχήμα 5.62: *The Confusion Matrices for each class, per SVM kernel, for the FAMD transformed imaging and genetic data.*



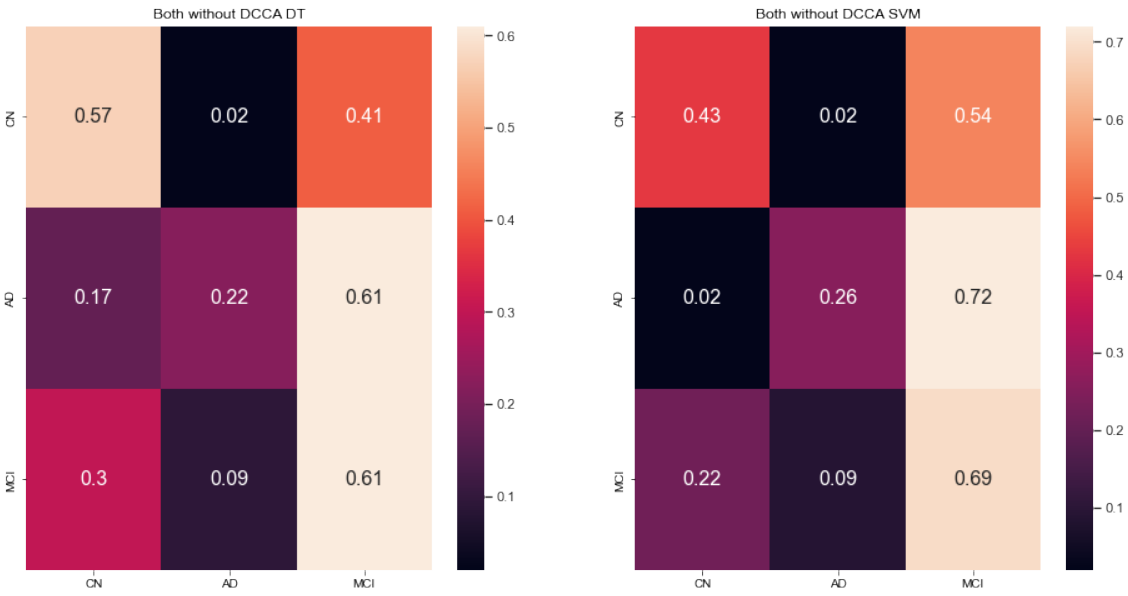
Σχήμα 5.63: *Classification metric using Bagging on the FAMD transformed imaging and genetic data.*



Σχήμα 5.64: *The Confusion Matrices for each class, with Bagging, for the FAMD transformed imaging and genetic data.*

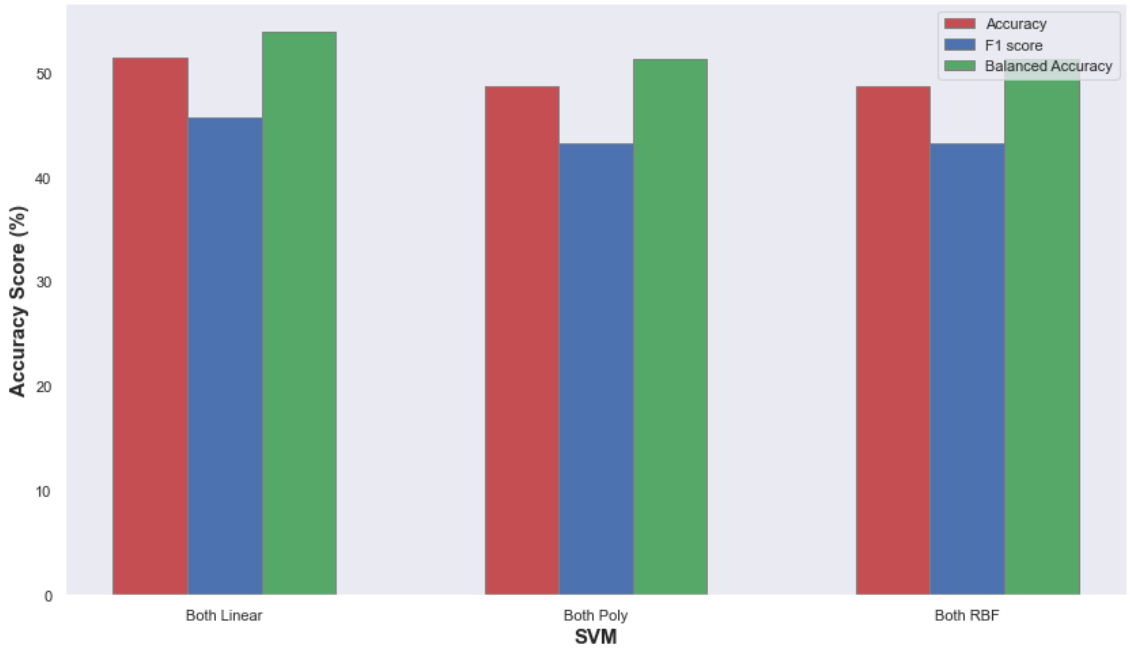


Σχήμα 5.65: *Classification metric using AdaBoost on the FAMD transformed imaging and genetic data.*

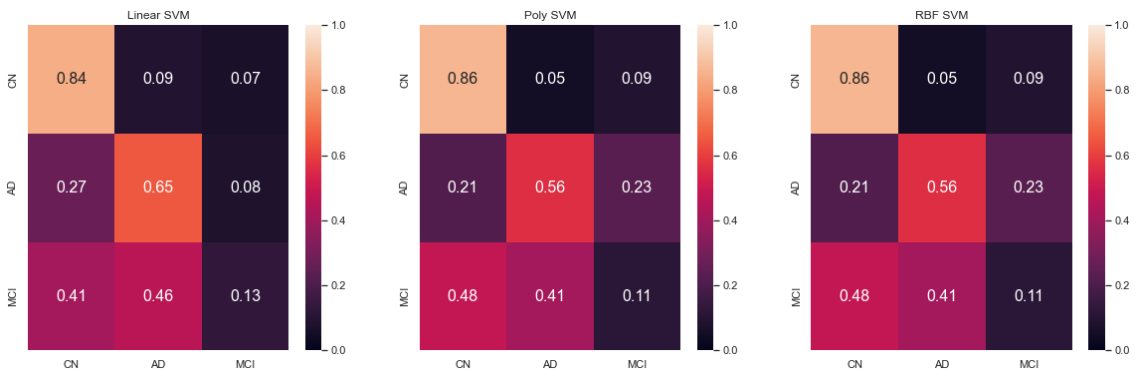


Σχήμα 5.66: *The Confusion Matrices for each class, with AdaBoost, for the FAMD transformed imaging and genetic data.*

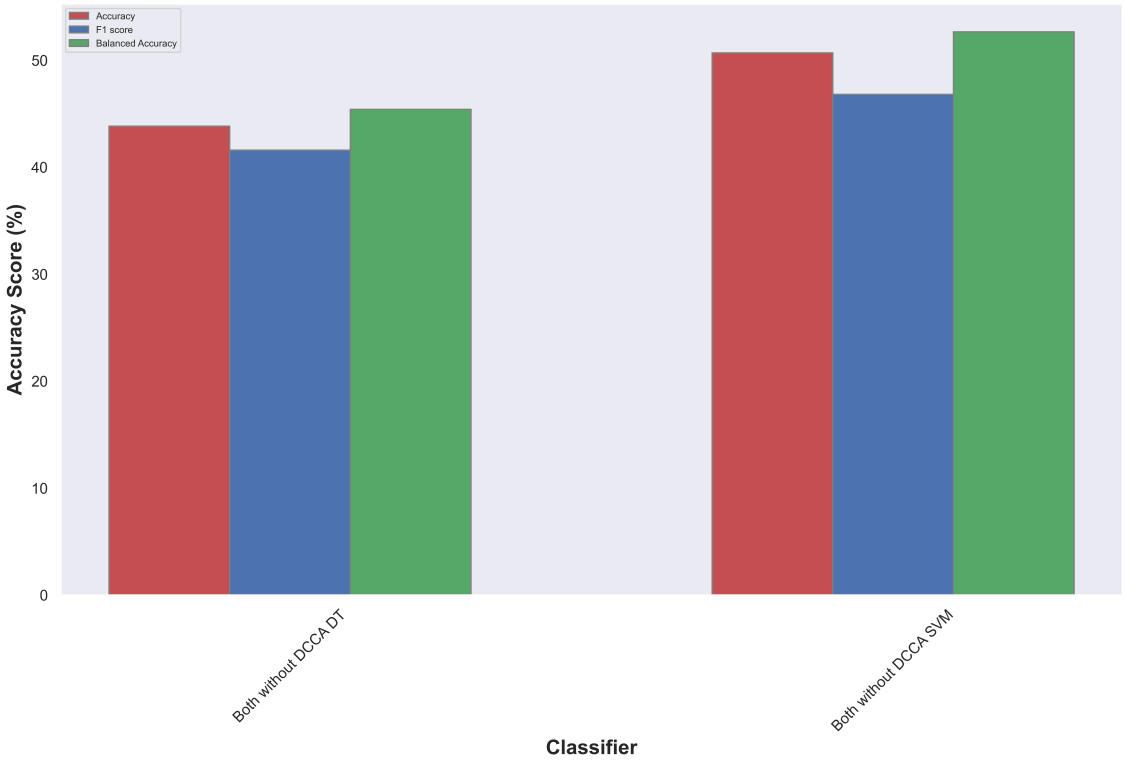
5.5.2 With scaling and balancing:



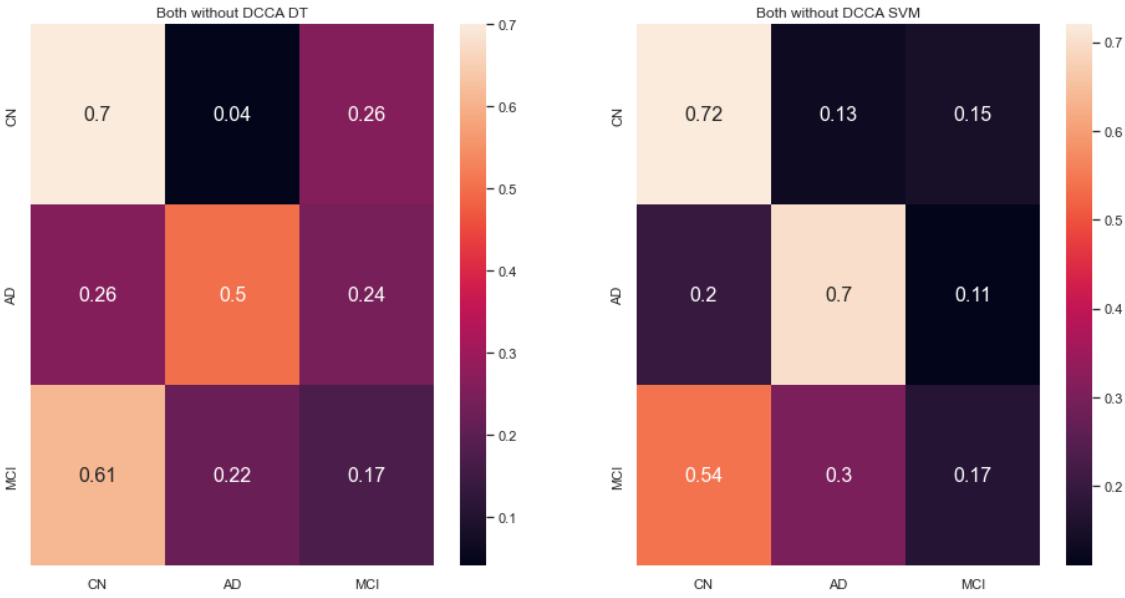
Σχήμα 5.67: *Classification metric using both views (imaging and genetic) on the SVM kernels previously mentioned (Linear, Polynomial, RBF), using the FAMD transformed imaging and genetic data.*



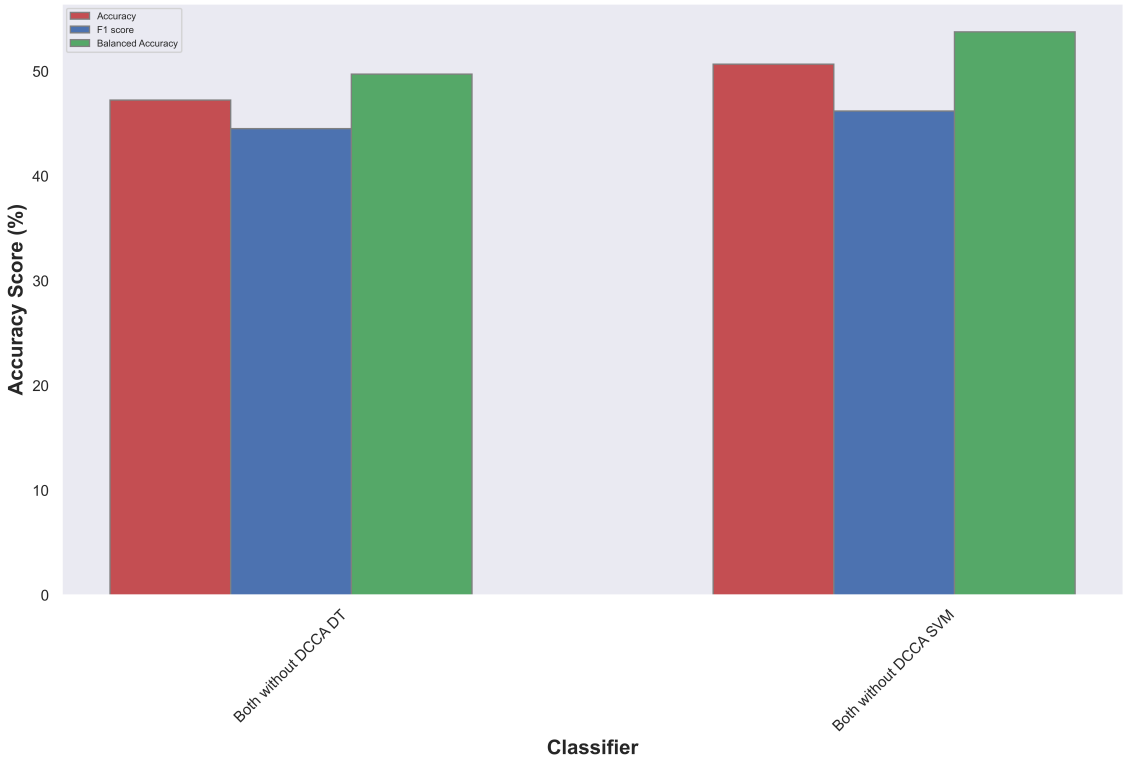
Σχήμα 5.68: *The Confusion Matrices for each class, per SVM kernel, for the FAMD transformed imaging and genetic data.*



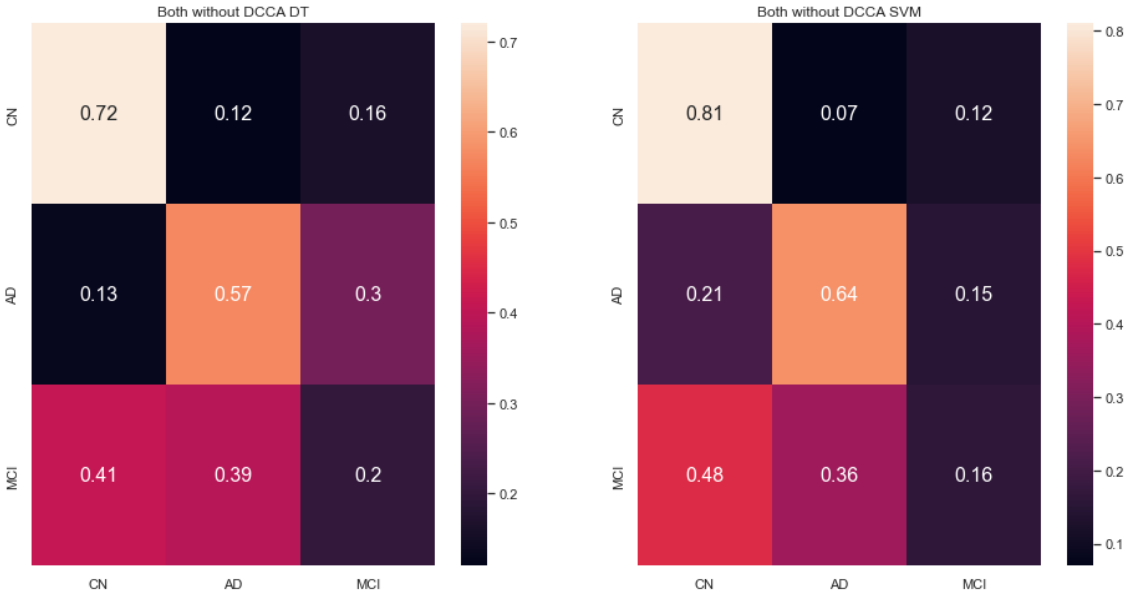
Σχήμα 5.69: *Classification metric using Bagging on the FAMD transformed imaging and genetic data.*



Σχήμα 5.70: *The Confusion Matrices for each class, with Bagging, for the FAMD transformed imaging and genetic data.*



Σχήμα 5.71: *Classification metric using AdaBoost on the FAMD transformed imaging and genetic data.*



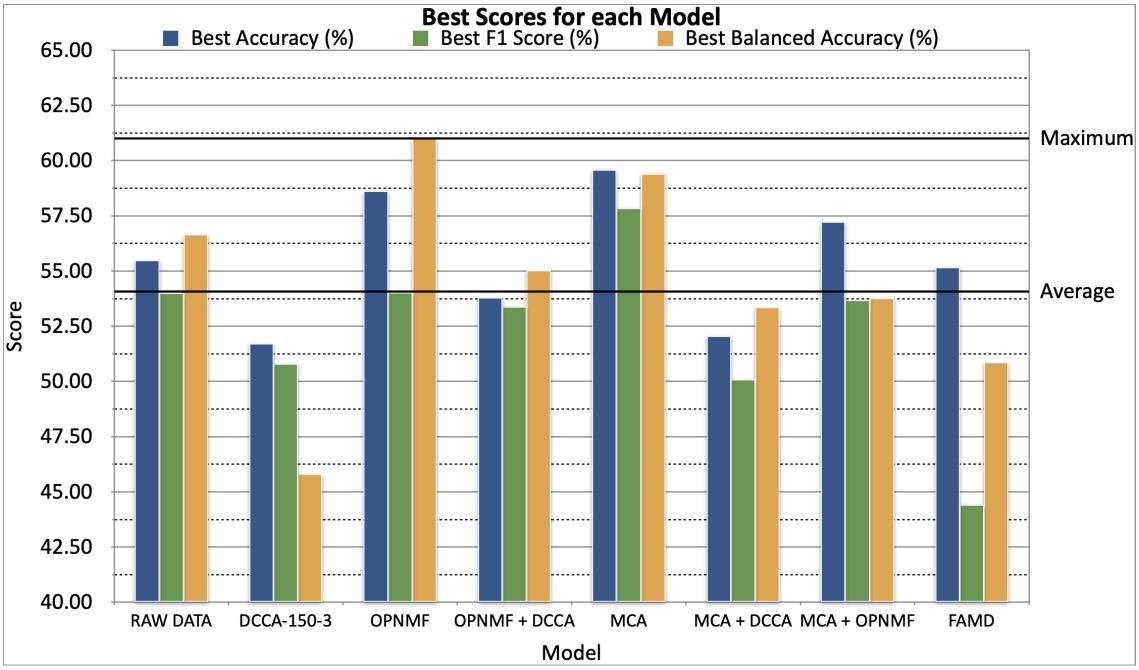
Σχήμα 5.72: *The Confusion Matrices for each class, with AdaBoost, for the FAMD transformed imaging and genetic data.*

5.6 Results Summary

To summarize the results, we present a table with the best Accuracy, F1 Score and Balanced Accuracy scores for each model, along with some notes. The diagram below visualizes the best results of each model, in order for them to be compared more easily.

Model	Best Accuracy (%)	Best F1 Score (%)	Best Balanced Accuracy (%)	Notes
RAW DATA	55.48	54.00	56.65	145 ROIs (Scaled) and 54 SNPs (Balanced). Both AdaBoost DT.
DCCA-150-3	51.72	50.80	45.82	Output Dimension 150, 3 Hidden Layers, no scaling or balancing. Imaging Linear.
OPNMF	58.62	54.02	61.01	30 Imaging Components (After OPNMF) Balanced only. Imaging Bagging SVM.
OPNMF + DCCA-150-3	53.79	53.38	55.03	30 Imaging Components (After OPNMF) and 54 SNPs, then DCCA, then scaled and balanced. Both AdaBoost SVM.
MCA	59.59	57.85	59.41	145 ROIs (Scaled), 10 Genetic components, Balanced only. Both Poly SVM.
MCA + DCCA-150-3	52.05	50.10	53.37	145 ROIs and 10 Genetic components (After MCA), then DCCA, then scaled and balanced. Imaging Bagging SVM.
MCA + OPNMF	57.24	53.68	53.77	30 Imaging Components (After OPNMF) and 10 Genetic components (After MCA), Balanced only. Both AdaBoost SVM.
FAMD	55.17	44.42	50.87	10 Components, no scaling, no balancing. Both Poly / RBF SVM.

Σχήμα 5.73: *For each model, the best Accuracy, F1 Score, Balanced Accuracy achieved are shown, along with notes explaining the combination of data and parameters used. Highlighted in green are the best and second best (lighter green) results achieved among all models.*



Σχήμα 5.74: *For each model, the best Accuracy, F1 Score and Balanced Accuracy are plotted, along with the maximum score achieved and the average score achieved.*

Κεφάλαιο 6

Discussion

Looking at the previous table and the last figure (5.73 ,5.74), one can clearly see that the models that achieve superior results are the OPNMF and MCA models. The first model, employed the OPNMF technique in order to transform the 145 ROI values for the imaging view of the dataset into 30 imaging components. After that, without the genetic view, balancing on the imaging view was performed, and the classification was performed after training a Bagging Ensemble classifier, using as a base model a linear SVM, achieving the highest Balanced Accuracy score, of 61.01%. The second model, utilized the MCA technique on the genetic view of the dataset, reducing the 54 (categorical) values of the SNPs into 10 genetic components (continuous). After that, along with the imaging view, balancing was performed, and the classification task was carried out using the model of Support Vector Machine (single classifier). The polynomial kernel as well as the radial basis function were both used and both achieved identical results, which are the best in the metrics of raw Accuracy and F1 Score, 59.59% and 57.85% respectively.

It is immediately obvious as well that all models are achieving scores that are close to each other, with their best variations being around the 50% mark and above. That however is not the case for every variation of the models' parameters, as is easily observable through the many figures of the fifth chapter. However there is a clear benefit in using the imaging view, since not only many genetic-only models are quite far from the best models (achieving metric scores in the mid 30s) but also no genetic-only model has been able to achieve better results than another model that included the imaging view.

Nevertheless, the same cannot be said for the opposite; that is there is no clear benefit in including the genetic view in imaging-only models, since models that have both views (imaging and genetic) are often outperformed by imaging-only models, yet they trade places depending on the model. For example the two previously mentioned models (that achieve the best metric scores) are imaging-only (the case of the OPNMF model), and both views (the case of the MCA model).

Looking at the results of the previous chapter, there is no clear conclusion to be drawn about the benefit of the techniques of balancing and scaling, since there is no clear trend. In the cases of (a) the data having no data analysis technique applied to them before classification, i.e. raw, (b) MCA-transformed genetic view models, (c) MCA-transformed genetic view and OPNMF-transformed imaging view models, balancing the dataset and applying scaling if needed seems to help, while in the case of the rest of the models it seems

to worsen the results.

As for the Deep Canonical Correlation Analysis, one can clearly observe that it produces objectively worse results than equivalent no-DCCA methods. Even in the case of the raw data, the DCCA transformed data produce results that are across the board worse in every metric. This is compounded by the fact that the DCCA networks not only took a considerable amount to train and optimize, but also the transformed outputs were clearly more computationally expensive and therefore the fitting of the classifiers to the DCCA-transformed dataset took more time than that of their no-DCCA counterpart methods. The aforementioned facts can only lead us to conclude that the use of the method cannot be recommended for this type of problem, with the reservation that the model was not adequately trained or had not a nearly enough number of parameters. This however could very well go beyond the scope of this work, and is discussed in the next chapter in more lengths. A hypothesis that can be made is that the DCCA method is not suitable for the great dissimilarity of the two views (the imaging being continuous and the genetic view being categorical). We attempted to remedy that with the MCA and OPNMF methods, which definitely improved the results of the models that utilized the DCCA method, however none of the resulting methods could achieve the results that the no-DCCA equivalent methods could achieve. The author believes that the model while having the potential to non-linearly transform data in such a way that they are maximally correlated, this case is not a suitable case, as Neural Networks often require a much greater volume of data, while having this many dissimilar features only worsened the likelihood of this method succeeding.

An interesting result is that of the method of Factor Analysis of Mixed Data, where both views are transformed into a reduced number of common components. This method, as previously mentioned, is similar to performing Principal Component Analysis on the continuous imaging view while performing Multiple Correspondence Analysis on the categorical genetic view, but combined. In our case the number of the resulting components was 10, which proved to yield worse results than many of the other methods. Due to the lack of published works concerning the method, there was little to no guidance as to how properly configure the method in order to best make use of the methods' potential. However, the duration of the training was at least as much as the training of the MCA model, while performing much worse both on average and on best case scenarios. It is entirely possible that the number of chosen components might not be enough to capture all of the information that the two views provide, and only achieve a substantial dimensionality reduction.

As can be seen by the extensive figures of the previous chapters, or more conveniently from the table's 5.73 notes, the Ensemble classifiers were in general more successful in predicting the class of the patient. This aligns well with the empirical knowledge that ensemble methods of even simple (as Decision Trees and linear Support Vector Machines are) can improve upon the single method performance, even if the single method is more complex. The performance of the ensemble classifiers achieved not only better best-case results, but on average was better overall, with even the worst performing parameter combinations beating out the worst performing parameter combinations of the single classifier

models of SVM. This was intuitively hypothesized, but also empirically proven as well.

Moreover, comparing the different Ensemble methods, it is not clear as to which method has the clear edge. The two methods seem to be trading blows, with Bagging seemingly being in more models marginally better than AdaBoost. However, the differences in the various metrics for nearly all of the model combinations are so little as to attribute this to margin of error. Furthermore, comparing the two methods for the base model classifier, which are the Decision Tree model and the Linear kernel Support Vector Machine, we can conclude that in nearly every case, the Linear SVM base model for the classifiers is better, for both kinds of Ensembles. While the difference between the performances of the two base models is not substantial, there is a clear trend. It is worth noting that both of those base models were chosen for their relatively simplistic nature, because of computational time limitations and as to create a basis for comparing single classifier models to Ensemble classifiers.

On the other hand, concerning the single classifier model of the different Support Vector Machine Kernels, we can see that in most cases the polynomial and radial basis function kernels outperform the linear kernel. However, all three are handily beaten by the Ensemble Classifier models. Additionally, the polynomial and RBF kernels seem to outperform in the best case the linear kernel, however the linear kernel seems to be more robust and on the worst case scenario, it clearly handles the classification much better. That can be seen in many Confusion Matrices, where the polynomial and RBF kernels classify the dataset as if they were dummy classifiers. There is however, a very interesting note to be added, concerning the models that had DCCA applied to the dataset before the classification task. In those cases, the linear kernel is performing better both on the best case scenario, but also on average and worst case than the polynomial and RBF kernels. This might indicate that indeed the DCCA method achieves the goal of linearly correlating the two views, however that might be at the expense of information loss, as those models are beaten by simpler methods without having DCCA applied to the dataset. This alone might be indicative of the need for not only more computational time and power devoted to the training of the DCCA parallel neural networks, but also the need for bigger and more complex variations of the network explored.

Expanding on the previous analysis, it is important to mention that this work utilized the DCCA method as it was implemented (see 2.2.2, 3.3). Therefore, since the task was substantially different than that of the DCCA methods' original paper, it is expected to perform not as expected. The author experimented with different activation functions (such as Sigmoid functions or ReLU), normalization techniques and optimization algorithms(such as Root Mean Squared Propagation and Limited-memory BFGS), but there was no clear trend in the initial testing phase, so there was little effort towards hyperparameter tuning.

Another important aspect that can be observed is the success of the models in the classification of a specific class. Across the board, we can see that most models (except from the models with distinctively poor overall performance) could reliably classify the patients that were suffering from Alzheimer's Disease from the Cognitively Normal people, while struggling to accurately decide for the case of the Mild Cognitive Impairment class. The Confusion Matrices in the figures of the previous chapter highlight exactly that, with

most models having a high percentage of the upper left portion of their Confusion Matrix well-defined, while the outer right and bottom part (which is the part of the MCI class) being quite confounded. As an example, one can observe the Confusion Matrices for the OPNMF method (no-DCCA), with the classifier being an Ensemble Classifier especially on using both or only the imaging view (5.46 Bagging, 5.48 Adaboost), but also the Confusion Matrices of the MCA method on both views, with scaling or balancing, using a single classifier model (5.28 SVMs). This is indicative of the strong performance of the model, as well as the difficulty of the problem, as the MCI class is apparently blurring the lines of the other classes. Since the methods and the models were optimized and trained for the problem of multi-class classification on the classes of Cognitive Normal, Mild Cognitive Impaired and Alzheimer's Disease, aforementioned one-vs-one comparisons should not be taken as a fact, but rather as an indication of performance.

The utilization of the Grid Search and Cross Validation methods for the evaluation of the different model parameters' performance while producing extensive and very useful results, was limiting the number of models that could possibly be explored. Both the optimizations of the DCCA networks, as well as the classification methods training, took for every model quantities of time in the order of hours. As a result, only relatively simple methods such as Support Vector Machines and Decision Trees were selected to be explored, since training Multilayer Perceptrons and similar methdos for each classification task and then optimizing them not only with every model and every view, but also with their respective grids of parameters was out of the question. Another limiting factor was the insufficient amount of data points that are inherently available to studies like these, which is due to the nature of the problem, since biomedical data and especially neuroimaging data are not only hard to collect, but pose storing, processing and visualization problems as well. Finally, the time frame for this study was not unbounded, and thus there had to be a selective process as to the direction of experimentation.

Finally, as to what the future directions might be for further research on the lessons learned from this work, there are many possible and valid steps. One might be to explore, as mentioned previously, different optimization algorithms, activation functions and generally different architectures for the parallel neural networks of the DCCA method. Another might direction might be data augmentation, or creation of synthetic data, as to enrich the dataset and unlock the potential of the DCCA method. Concerning the data analysis techniques, the OPNMF method is one that could benefit from experimenting with different number of components, something that wasn't done on this study due to computational power limitations. Furthermore, more complex classifiers can be used, such as MLPs and K-Nearest Neighbors classifiers. That can also be extended to the ensemble methods, not only for the base classifier models, but also for the ensemble methods, with other boosting methods and methods such as stacking being obvious candidates. Finally, to address the issue of the different in nature and type views, one could explore the use of deep autoencoders in order to alleviate the problem of handling categorical data.

Κεφάλαιο 7

Conclusions

This study was intended to be an extensive comparison and application of Data Analysis methods, as well as Machine and Deep Learning methods, applied to the problem of CN / MCI / AD classification. From the work that was performed, there are some clear takeaways. First of all, data analysis techniques such as Multiple Correspondence Analysis and Orthonormal Projective Non-Negative Matrix Factorization are essential to achieve better results. Furthermore, the method of Deep Canonical Correlation Analysis as stated in the original paper, is not beneficial to this problem, at least not with further tuning. Ensemble Classifier methods are superior to the simplistic single classifier methods such as Support Vector Machines. The use of the methods of dataset balancing and data scaling have no clear positive or negative effect on the classification problem. Finally, it is clear that using only genetic data is not sufficient to yield higher-quality results. The combinations that achieved the best results used Imaging or Imaging along with Genetic data, either OPNMF on Imaging data or MCA on the Genetic data, accompanied with either Bagging ensembles of SVMs or a polynomial kernel SVM as classifiers.

Βιβλιογραφία

Συντομογραφίες - Αρκτικόλεξα - Ακρωνύμια

βλπ	βλέπε
κ.λπ.	και λοιπά
κ.ο.κ	και ούτω καθεξής
ΤΕΙ	Τεχνολογικό Εκπαιδευτικό Ίδρυμα
BPF	Band Pass Filter

Απόδοση ξενόγλωσσων όρων

Απόδοση	Ξενόγλωσσος όρος
αδερφός	sibling
αμεταβλητότητα	idempotency
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface
διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union
αδερφός	sibling
αμεταβλητότητα	idempotency
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface
διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union

αδερφός	sibling
αμεταβλητότητα	idempotency
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface
διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union
αδερφός	sibling
αμεταβλητότητα	idempotency
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface
διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union
αδερφός	sibling
αμεταβλητότητα	idempotency
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface

διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union