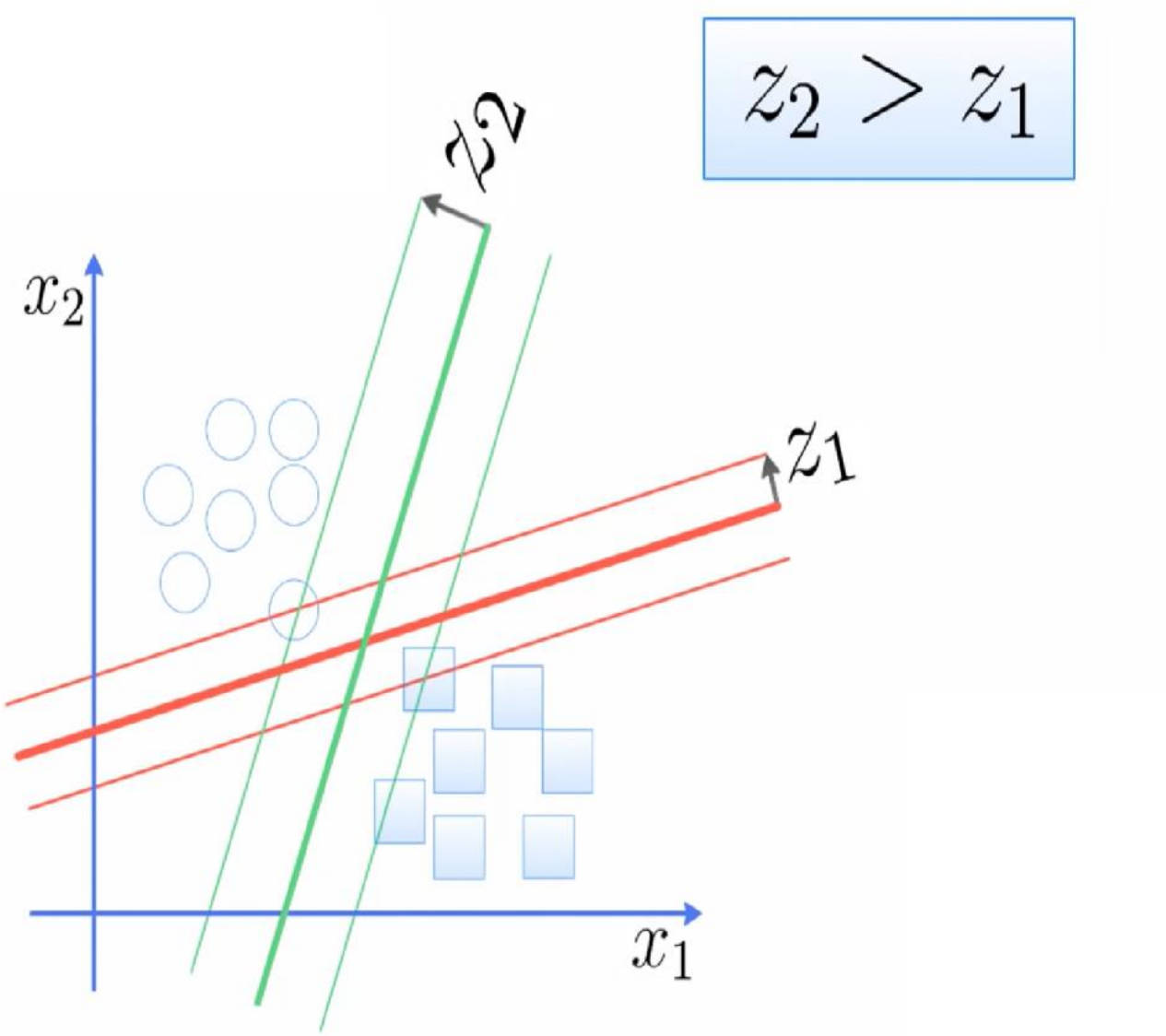
Σκοπός της διπλωματικής

- Σύγκριση των αλγορίθμων Κατηγοριοποίησης σε διάφορα case study
  - Δέντρα Απόφασης - Classification And Regression Trees (C.A.R.T)
  - K – Nearest Neighbors
  - Naïve Bayes
  - Neural Networks (Multilayer Perceptron)
  - Random Forests
  - Support Vector Machines
  - Adaboost
  - XGBoost (Gradient Boosting Framework)
- Εκμάθηση των διαθέσιμων βιβλιοθηκών για Data Mining (Scikit, Xgboost σε Python)
- Εξαγωγή συμπερασμάτων

Support Vector Machines



Σκοπός μας είναι να βρούμε το διανυσματικό υποχώρο το οποίο διαχωρίζει καλύτερα τα δείγματα μας.

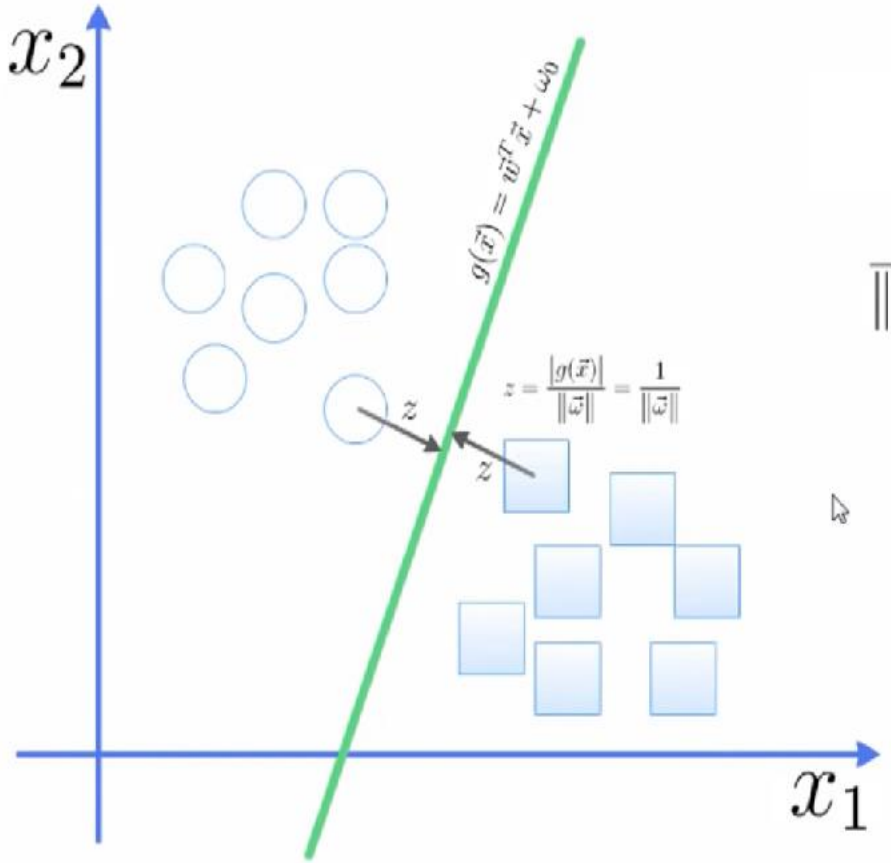
Υπάρχουν άπειρες ευθείες που να διαχωρίζουν τα δείγματα.

Πώς θα επιλέξουμε την καλύτερη?

Θα διαλέξουμε το διανυσματικό υποχώρο ο οποίος αφήνει το μεγαλύτερο κενό ανάμεσα στα σημεία των δύο κλάσεων.

Support Vector Machines

$$g(\vec{x}) \geqslant 1, \quad \forall \vec{x} \in \text{class } 1$$
$$g(\vec{x}) \leqslant -1, \quad \forall \vec{x} \in \text{class } 2$$



$$\frac{1}{\|\vec{\omega}\|} + \frac{1}{\|\vec{\omega}\|} = \frac{2}{\|\vec{\omega}\|}$$

g(x) : συνάρτηση απόφασης

$z = \frac{1}{||w||}$: η απόσταση του κοντινότερου σημείου κάθε κλάσης (support vector) στην ευθεία.

Για να πετύχουμε την μεγαλύτερη διαχωρισιμότητα πρέπει να ελαχιστοποιήσουμε το ||w||.

Support Vector Machines

Η ελαχιστοποίηση του ||w|| είναι πρόβλημα του μη-γραμμικού προγραμματισμού.

Η λύση του επιτυγχάνεται με τη βοήθεια του Θεωρήματος των Karush – Kuhn – Tucker μετασχηματίζοντας το πρόβλημα με την βοήθεια των πολλαπλασιαστών Lagrange (λ_ι)

$$|w|| = \sum_{i=0}^{N} \lambda_\iota y_\iota \chi_\iota$$

Στη βιβλιοθήκη Scikit-Learn δεν κάνουν αυτό!

Λύνουν το δυικό πρόβλημα :

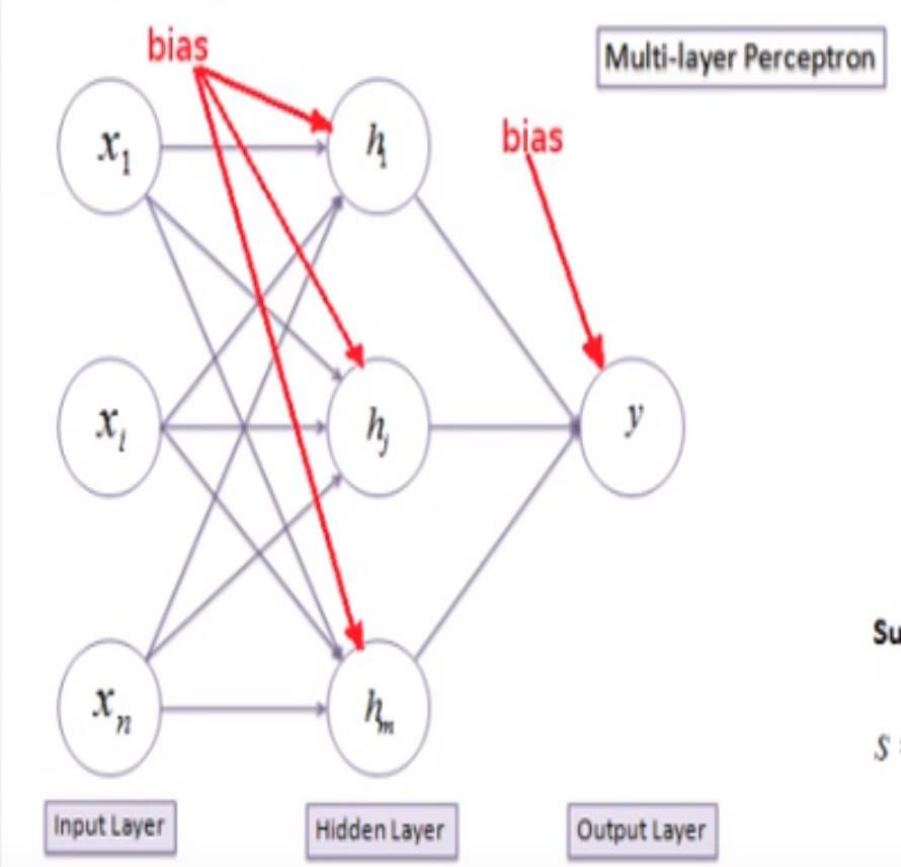
$$maxL_D(\lambda_j) = \sum_{i=1}^{l} \lambda_i - \frac{1}{2} \sum_{i=1}^{l} \lambda_i \lambda_j y_i y_j (x_i \cdot x_j)$$

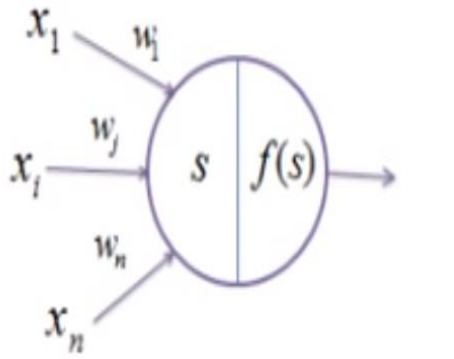
υπό τους περιορισμούς :

$$\sum_{i=1}^{l} \lambda_i y_i = 0 \; and \; \lambda_i \geq 0$$

Η τελική απόφαση λαμβάνεται από το πρόσημο της συνάρτησης απόφασης : $f(x) = \left(\sum_{i=1}^{l} a_i y_i x_i \cdot u\right) + b$

Multilayer Perceptron





Summation

$$s = \sum w \cdot x$$

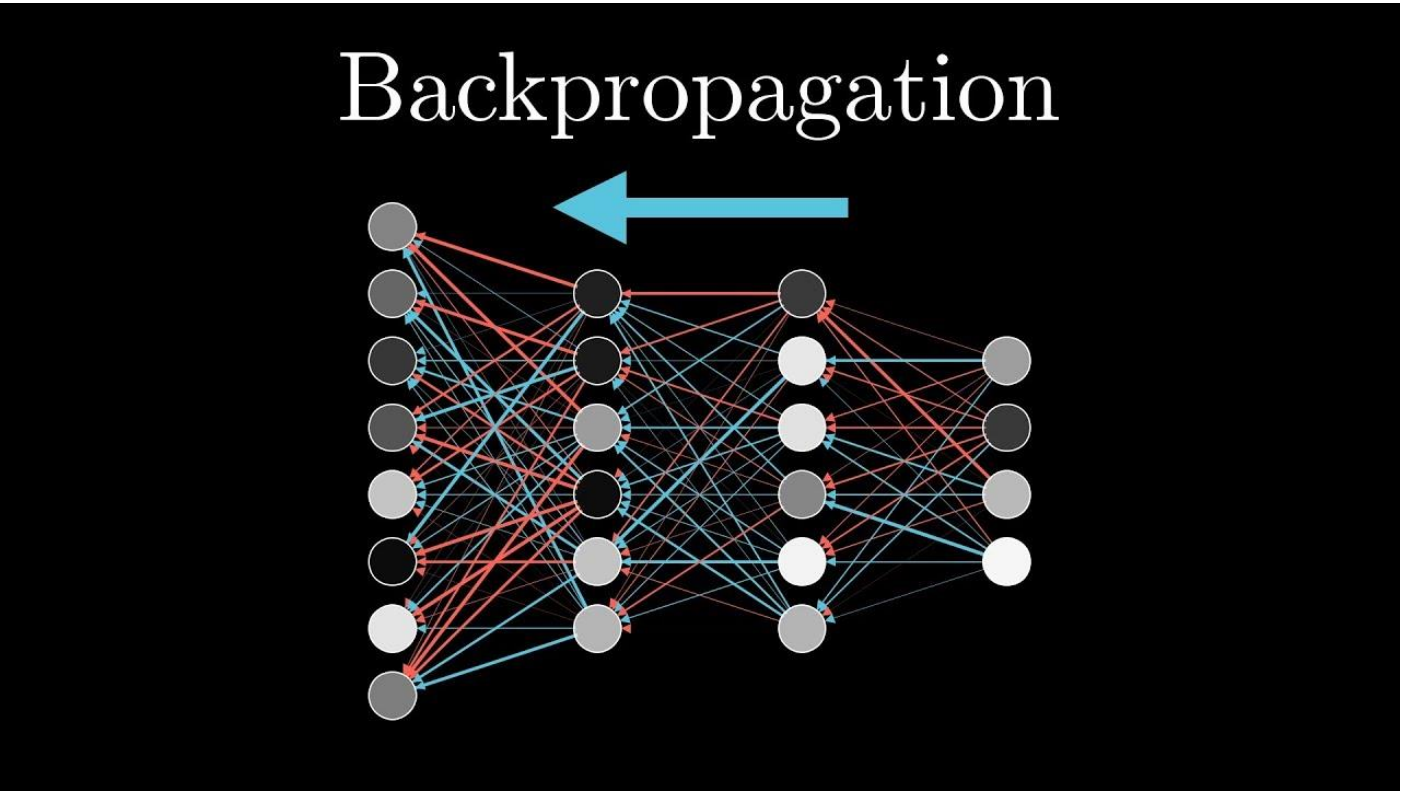
Transformation

$$f(s) = \frac{1}{1+e^{-s}}$$

- Δίκτυο από «νευρώνες»
- Αποτελείται από τριών ειδών επίπεδα (input layer, hidden layer , output layer)
- Input layer : κάθε κόμβος αντιστοιχεί σε ένα χαρακτηριστικό των δεδομένων
- Hidden layer : κάθε κόμβος είναι το σταθμισμένο άθροισμα (weighted sum) των συνδεδεμένων κόμβων επαυξημένο κατά μια σταθερά (bias).
- Η τιμή που εκπέμπει κάθε κόμβος περνάει από activation function.

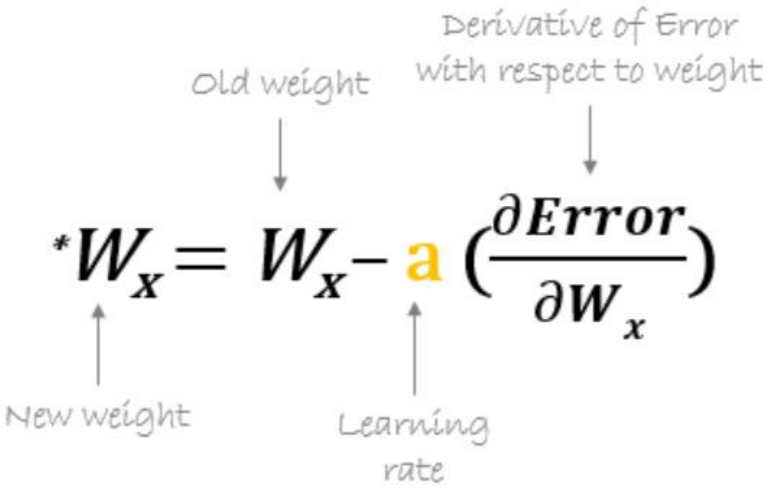
Multilayer Perceptron

Πως ‘μαθαίνει’ ο αλγόριθμος ?

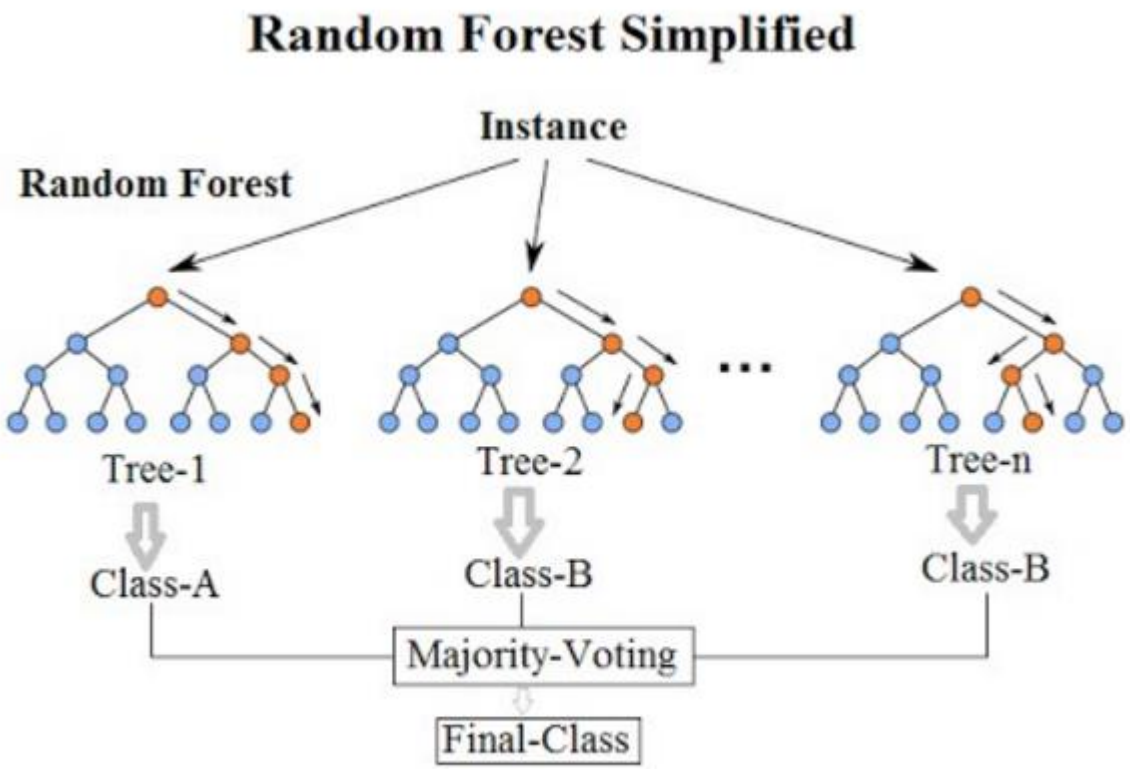


Όταν μια καταγραφή έχει κατηγοριοποιηθεί λάθος, υπολογίζεται το σφάλμα.

Αναδρομικά τα βάρη όλων των κόμβων που οδήγησαν σε αυτή την απόφαση ανανεώνονται.



Random Forests



Forest : Συλλογή από δέντρα (Δέντρα Απόφασης)

Τα δέντρα εκπαιδεύονται χρησιμοποιώντας τον αλγόριθμο Bagging.

Κάθε νέο δέντρο εκπαιδεύεται σε ένα τυχαίο υποσύνολο των δεδομένων (δειγματοληψία με εναπόθεση).

Κάθε νέο δέντρο εκπαιδεύεται με ένα τυχαίο υποσύνολο των χαρακτηριστικών.

Η τελική απόφαση προκύπτει από τη πλειοψηφία των αποφάσεων όλων των δέντρων του δάσους.

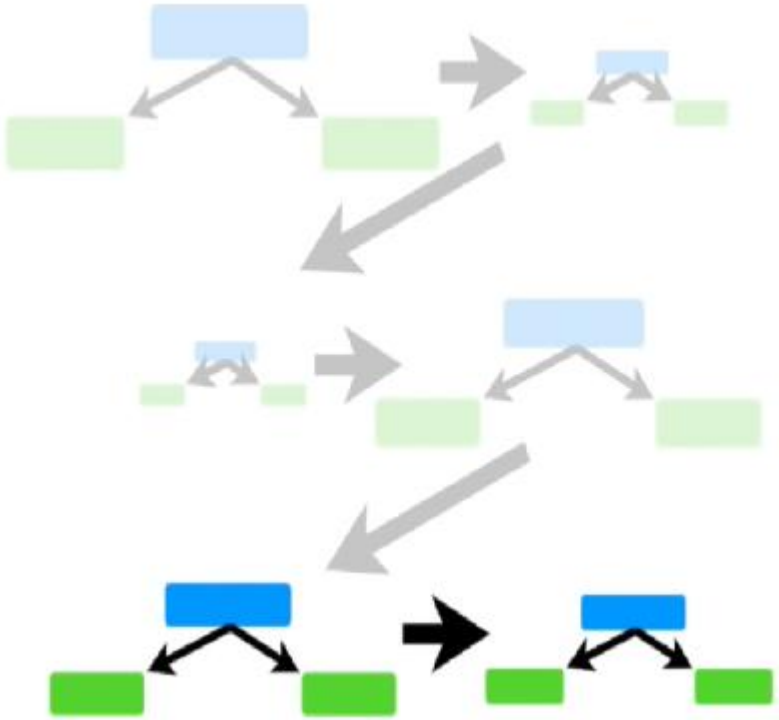
Adaboost

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	1/8
No	Yes	180	Yes	1/8
Yes	No	210	Yes	1/8
Yes	Yes	167	Yes	1/8
No	Yes	156	No	1/8
No	Yes	125	No	1/8
Yes	No	168	No	1/8
Yes	Yes	172	No	1/8

Stump : 1-depth Decision Tree

Adaboost : Δημιουργεί μια σειρά από δέντρα, το καθένα με διαφορετικό βάρος.

Τα λάθη κάθε δέντρου διορθώνονται από το επόμενο.



Αρχικοποίηση : όλα τα δεδομένα παίρνουν το ίδιο βάρος

Δημιουργούμε το πρώτο stump με βάση το Gini index.

$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

Adaboost

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	New Weight	Norm. Weight
Yes	Yes	205	Yes	1/8	0.05	0.07
No	Yes	180	Yes	1/8	0.05	0.07
Yes	No	210	Yes	1/8	0.05	0.07
Yes	Yes	167	Yes	1/8	0.33	0.49
No	Yes	156	No	1/8	0.05	0.07
No	Yes	125	No	1/8	0.05	0.07
Yes	No	168	No	1/8	0.05	0.07
Yes	Yes	172	No	1/8	0.05	0.07

Υπολογίζουμε το νέο βάρος για τα δεδομένα και τα κανονικοποιούμε

$$\text{New Sample Weight} = \text{sample weight} \times e^{\text{-amount of say}}$$

Adaboost

Τέλος φτιάχνουμε ένα νέο dataset ίδιων διαστάσεων, και επιλέγουμε κάθε φορά έναν τυχαίο αριθμό ανάμεσα στο 0 και στο 1.

Θεωρώντας τον τυχαίο αριθμό ως αθροιστική πιθανότητα, εισάγουμε στο νέο dataset τη γραμμή στο οποίο αθροίζεται η πιθανότητα αυτή.

Για παράδειγμα αν ο τυχαίος αριθμός είναι το 0.14, η 3η γραμμή θα έμπαινε στο νέο dataset.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	0.07
No	Yes	180	Yes	0.07
Yes	No	210	Yes	0.07
Yes	Yes	167	Yes	0.49
No	Yes	156	No	0.07
No	Yes	125	No	0.07
Yes	No	168	No	0.07
Yes	Yes	172	No	0.07

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease

XGBoost

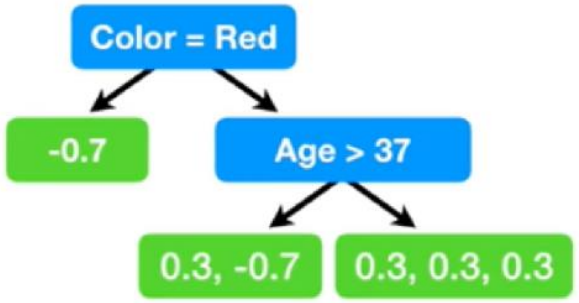
Αρχικά κάνουμε μια πρόβλεψη για κάθε δεδομένο να ανήκει στην κλάση που θέλουμε (log of odds – log(4/2)) και υπολογίζουμε το σφάλμα (observed – predicted).

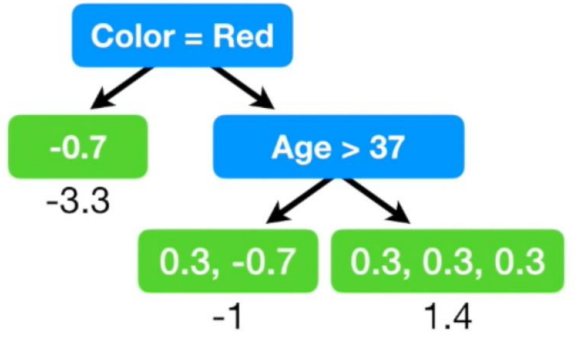
Στη συνέχεια χτίζουμε ένα δέντρο απόφασης για να προβλέψουμε τα σφάλματα.

Υπολογίζουμε τα output values για κάθε φύλλο του δέντρου.

Κάνουμε update το log of odds, τα μετατρέπουμε σε πιθανότητες μέσο της λογιστικής συνάρτησης και υπολογίζουμε τα νέα τα residuals

Likes Popcorn	Age	Favorite Color	Loves Troll 2	Residual
Yes	12	Blue	Yes	0.3
Yes	87	Green	Yes	0.3
No	44	Blue	No	-0.7
Yes	19	Red	No	-0.7
No	32	Green	Yes	0.3
No	14	Blue	Yes	0.3





$$\frac{\sum \text{Residual}_i}{\sum \left[\text{Previous Probability}_i \times (1 - \text{Previous Probability}_i)\right]}$$

Learning rate = 0.8

$$\text{Probability} = \frac{e^{-0.1}}{1 + e^{-0.1}}$$

log(odds) Prediction = 0.7 + (0.8 × -1) = -0.1

Immunotherapy & Cryotherapy Datasets

Immunotherapy Dataset : πληροφορίες 90 ασθενών στους οποίους εφαρμόστηκε ως μέθοδος θεραπείας των κονδυλωμάτων η ανοσοθεραπεία.

Χαρακτηριστικά του δείγματος (8) :

Φύλο(1 ή 0), Ηλικία, Χρόνος πριν από την θεραπεία, Πλήθος κονδυλωμάτων,

Είδος κονδυλώματος(1,2 ή 3), Εμβαδό επιφάνειας, Διάμετρος,

Αποτέλεσμα θεραπείας (1 ή 0)

Cryotherapy Dataset : πληροφορίες 90 ασθενών στους οποίους εφαρμόστηκε ως μέθοδος θεραπείας των κονδυλωμάτων η κρυοθεραπεία.

Χαρακτηριστικά του δείγματος (7) :

Φύλο, Ηλικία, Χρόνος πριν από την θεραπεία, Πλήθος κονδυλωμάτων,

Είδος κονδυλώματος, Εμβαδό επιφάνειας, Αποτέλεσμα θεραπείας

	sex	age	Time	Number_of_Warts	Type	Area	induration_diameter	Result_of_Treatment
0	1	22	2.25	14	3	51	50	1
1	1	15	3.00	2	3	900	70	1
2	1	16	10.50	2	1	100	25	1
3	1	27	4.50	9	3	80	30	1
4	1	20	8.00	6	1	45	8	1

	sex	age	Time	Number_of_Warts	Type	Area	Result_of_Treatment
0	1	35	12.00	5	1	100	0
1	1	29	7.00	5	1	96	1
2	1	50	8.00	1	3	132	0
3	1	32	11.75	7	3	750	0
4	1	67	9.25	1	1	42	0

Immunotherapy & Cryotherapy Datasets – Preprocessing

Βασική ιδέα : ένωση των δύο dataset, δημιουργία ενιαίου μοντέλου

(δεν έχει ξαναγίνει στη βιβλιογραφία)

1ό Πρόβλημα : το Immunotherapy Dataset έχει ένα χαρακτηριστικό επιπλέον (Induration diameter)

Υπάρχουν δύο στρατηγικές :

- Αφαίρεση της επιπλέον στήλης
- Συμπλήρωση των missing values για το Cryotherapy

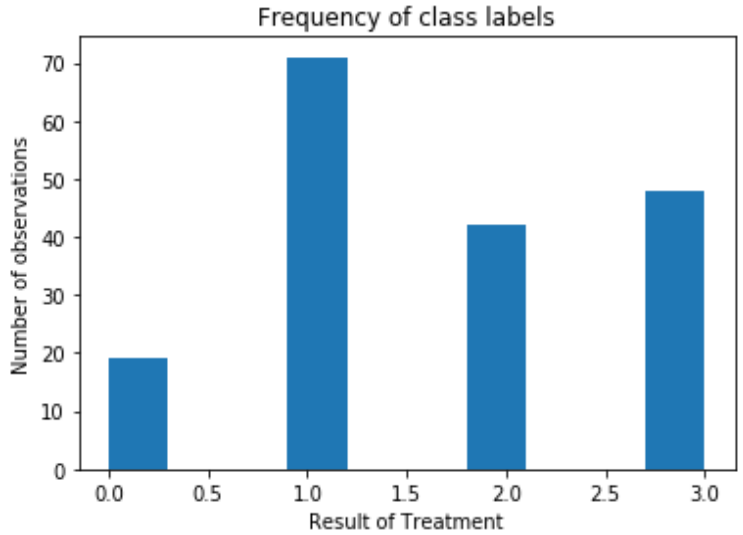
Γιατί είναι κακή ιδέα να ακολουθήσουμε την εύκολη λύση?

Immunotherapy & Cryotherapy Datasets - Preprocessing

Αφού προστέθηκε η επιπλέον στήλη στο Cryotherapy Dataset, μπορούμε να δουλέψουμε στα ενοποιημένα δεδομένα. Πρώτα όμως πρέπει να αλλάξουμε την κλάση Result of Treatment.

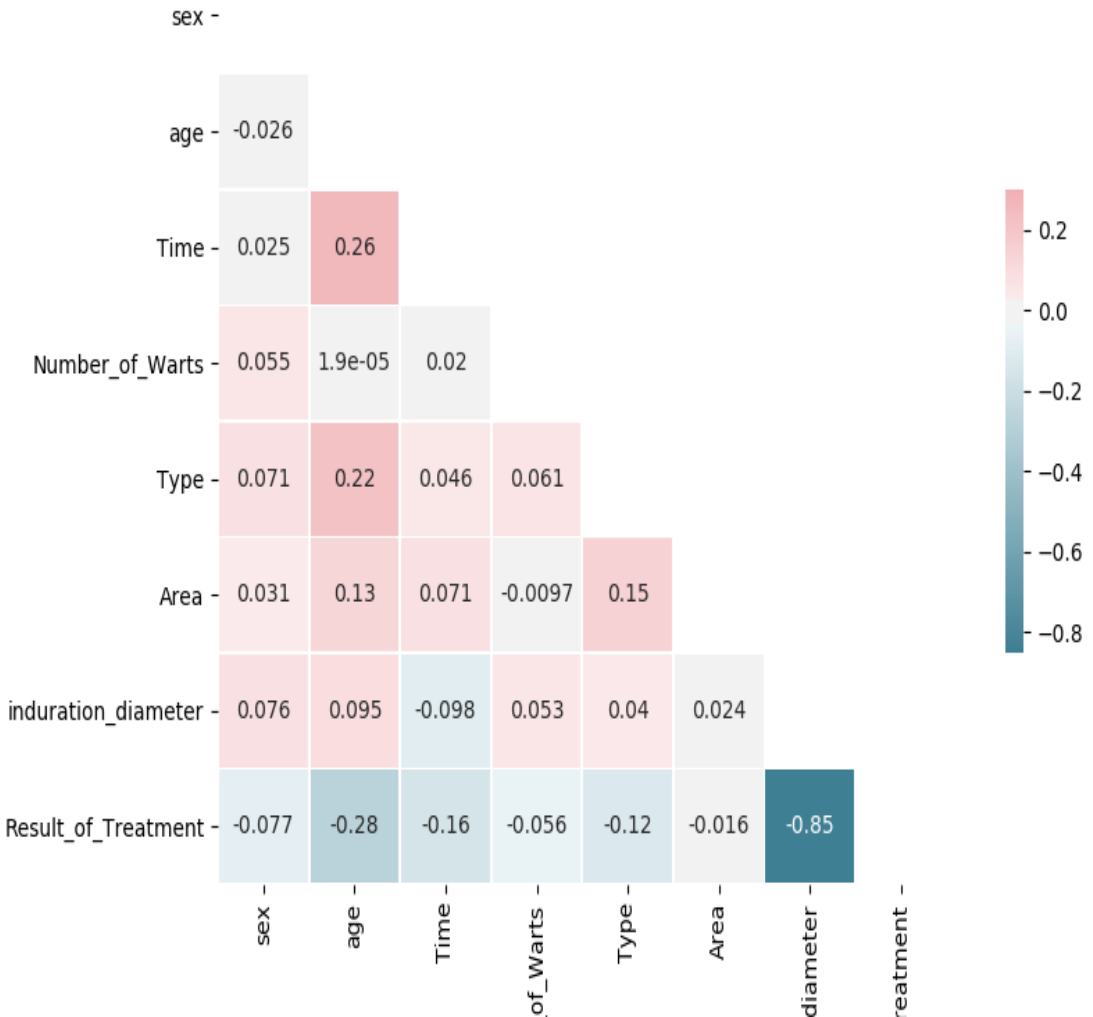
- 0 : η ανοσοθεραπεία αποτυγχάνει
- 1 : η ανοσοθεραπεία επιτυγχάνει
- 2 : η κρυοθεραπεία αποτυγχάνει
- 3 : η κρυοθεραπεία επιτυγχάνει

Quick check : Μήπως τα δεδομένα κάποιας κλάσης υπερτερούν καθοδηγώντας τα αποτελέσματα?



Υπάρχει μια μικρή απόκλιση για την κλάση 0 αλλά δεν είναι αισθητή.

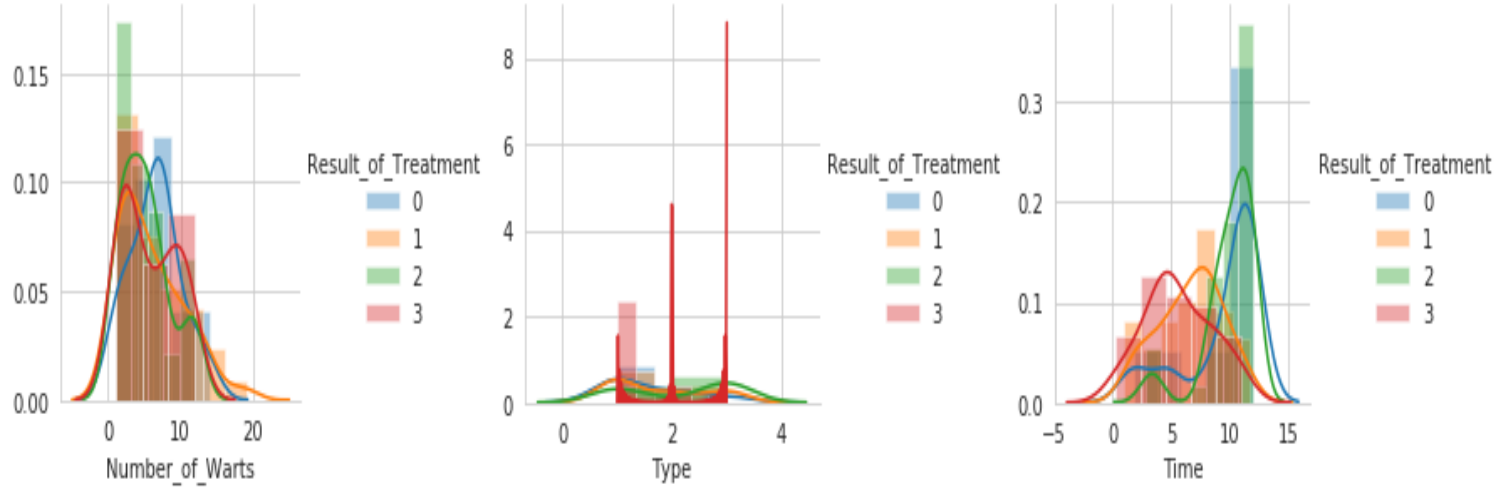
Immunotherapy & Cryotherapy Datasets - Preprocessing

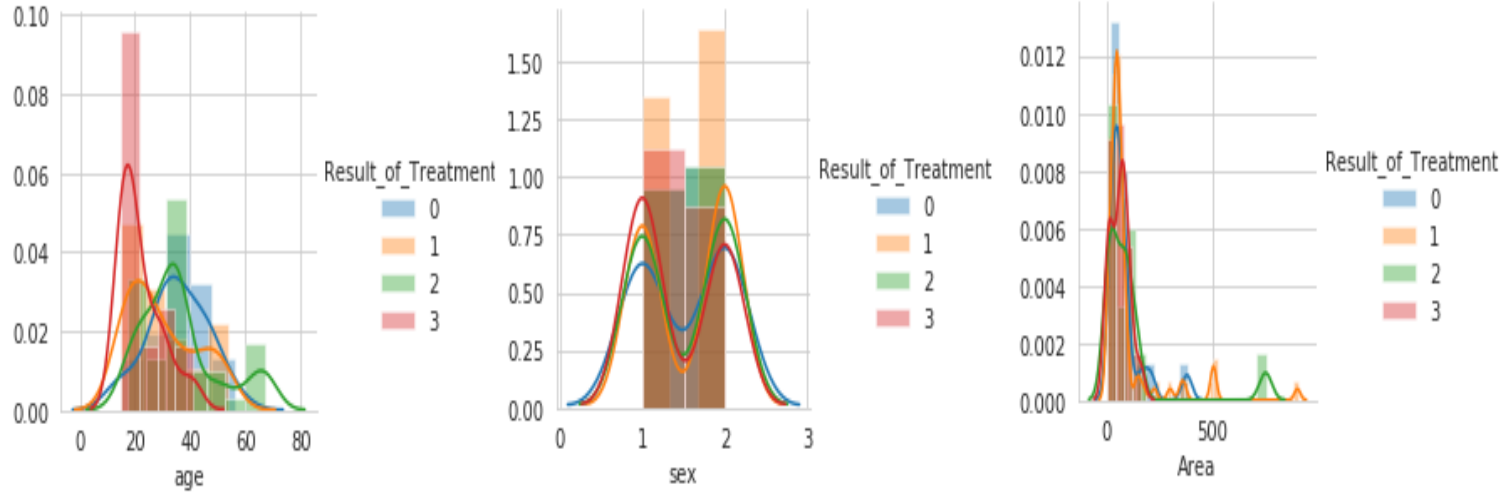


Από τον πίνακα συσχέτισης του Spearman βλέπουμε ότι το χαρακτηριστικό αυτό παίζει τον σημαντικότερο ρόλο στην πρόβλεψη της κλάσης που μας ενδιαφέρει (Result of Treatment).
Η διαγραφή της στήλης θα οδηγούσε στην απώλεια σημαντικής πληροφορίας.

Hint – Όταν τα χαρακτηριστικά ενός δείγματος ακολουθούν διαφορετικές κατανομές ο πίνακας συσχέτισης του Spearman δίνει ποιο έγκυρα αποτελέσματα από τη μέθοδο Pearson

Immunotherapy & Cryotherapy Datasets – Exploratory Data Analysis – Univariate Analysis



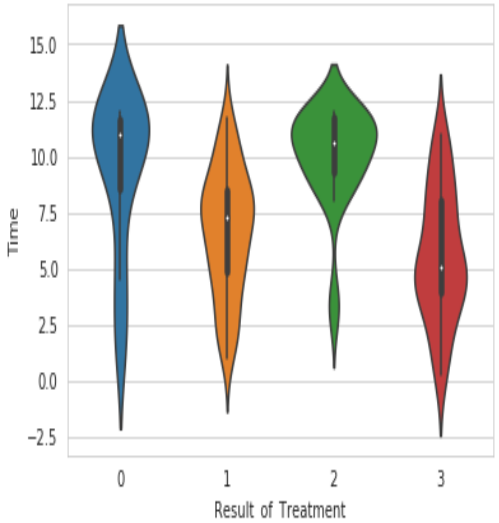


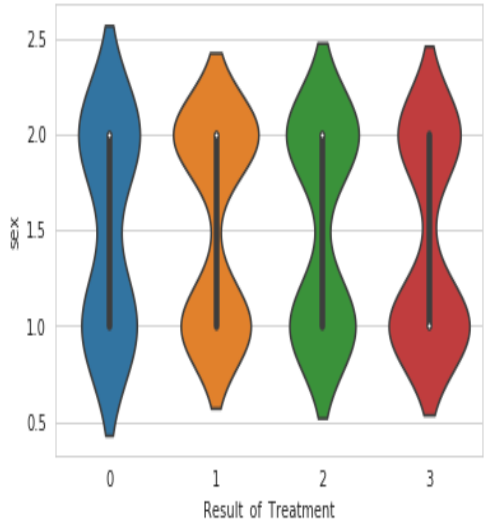
Univariate Analysis : Μελέτη ενός χαρακτηριστικού του δείγματος

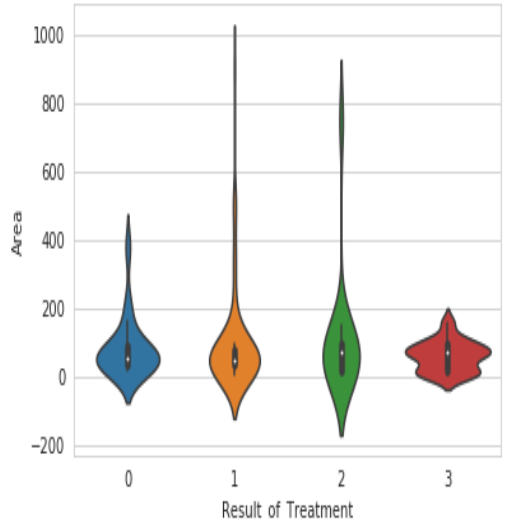
Μελετώντας την κατανομή κάθε χαρακτηριστικού για κάθε κλάση μπορούμε να εντοπίσουμε ποιο χαρακτηριστικό έχει τη μεγαλύτερη διαχωρισιμότητα.

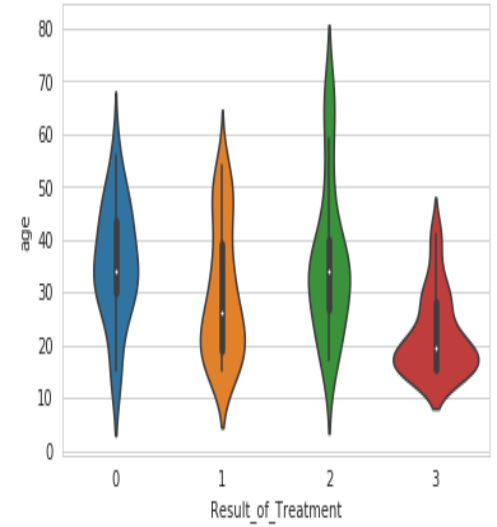
Τί θέλουμε να δούμε : Ξεχωριστές, μη επικαλυπτόμενες κατανομές (Time,age)
Τα χαρακτηριστικά με επικαλυπτόμενες κατανομές δεν έχουν μεγάλη προβλεπτική ικανότητα (sex, Area, Number of Warts)

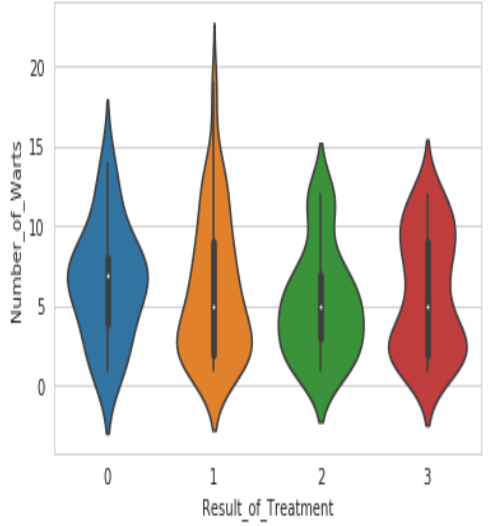
Immunotherapy & Cryotherapy Datasets – Exploratory Data Analysis – ‘Violinplot’ Analysis

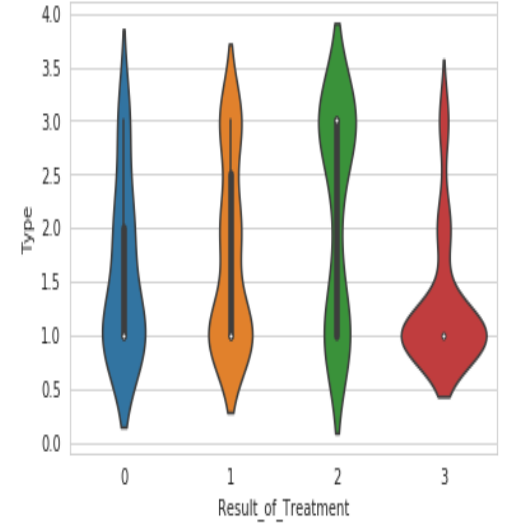












Boxplot Analysis : Βασικό εργαλείο της Στατιστικής ανάλυσης.
Μπορούμε να εντοπίσουμε εύκολα χαρακτηριστικά με ακραίες τιμές (Area).
Οι πλευρές του ‘βιολιού’ είναι απεικόνιση της κατανομής.

Χαρακτηριστικά στα οποία τα boxplot έχουν σημαντική υψομετρική διαφορά, έχουν μεγάλη διαχωριστικότητα επομένως και μεγαλύτερη προβλεπτική δύναμη από τα υπόλοιπα.

Immunotherapy & Cryotherapy Datasets – Κατασκευή μοντέλων - Αποτελέσματα

Για τη δημιουργία των μοντέλων θα χρησιμοποιήσουμε τα χαρακτηριστικά Time, induration diameter και Age.
Πού θα κατασκευάσουμε τα μοντέλα? : 10 – fold cross validation (fixed seed)
Τι metric θα χρησιμοποιήσουμε για να αξιολογήσουμε τα μοντέλα? : balanced class -> Accuracy

Algorithm\Fold	1	2	3	4	5	6	7	8	9	10	Mean	Sd	Time
XGBoost	83.3%	94.4%	94.4%	83.3%	88.8%	94.4%	100%	94.4%	88.8%	100%	92.2%	5.6%	0.17
Random Forest	83.3%	100%	83.3%	83.3%	88.8%	94.4%	100%	88.8%	88.8%	77.7%	88.8%	7%	1.54
C.A.R.T	77.7%	83.3%	77.7%	83.3%	77.7%	88.8%	100%	88.8%	83.3%	88.8%	85%	6.5%	0.07
Adaboost	77.7%	88.8%	88.8%	83.3%	88.8%	88.8%	94.4%	83.3%	88.8%	77.7%	86.1%	5.1%	3.75
Neural Network	72.2%	100%	88.8%	77.7%	88.8%	77.7%	100%	83.3%	83.3%	72.2%	83.8%	8.7%	2
SVM	72.2%	94.4%	88.8%	77.7%	88.8%	88.8%	88.8%	72.2%	83.3%	83.3%	83.8%	7.2%	0.06
Naïve Bayes	61.1%	94.4%	88.8%	72.2%	83.3%	83.3%	83.3%	77.7%	83.3%	88.8%	81.6%	8.9%	0.07
KNN	88.8%	88.8%	77.7%	66.6%	88.8%	88.8%	100%	77.7%	77.7%	88.8%	84.4%	8.8%	0.07

APS Failure at Scania Trucks Dataset

Industrial Challenge 2016 at The 15th International Symposium on Intelligent Data Analysis (IDA)

Περιέχει δεδομένα που σχετίζονται με το σύστημα πεπιεσμένου αέρα των φορτηγών της Scania.

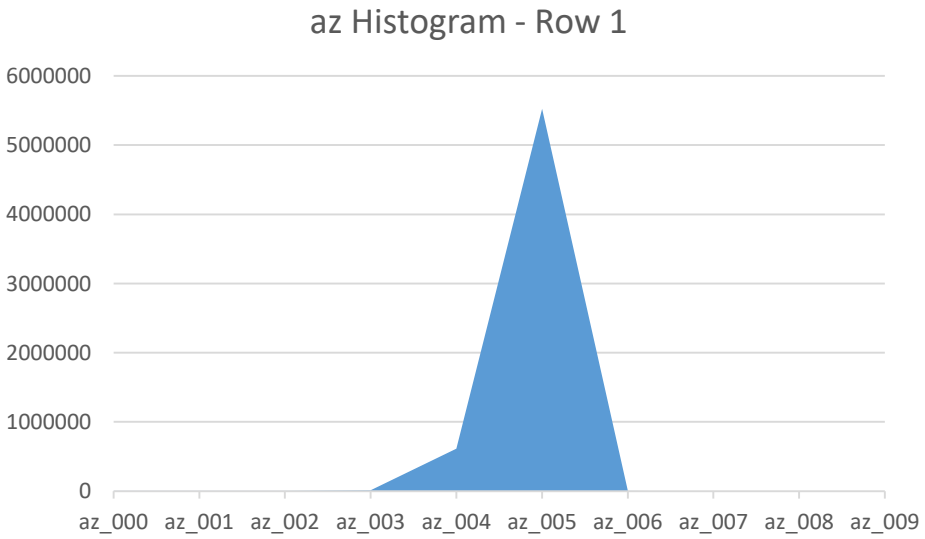
Είναι χωρισμένο σε training και testing set με το training set να αποτελείται από 60.000 εγγραφές και το testing από 16.000 εγγραφες.

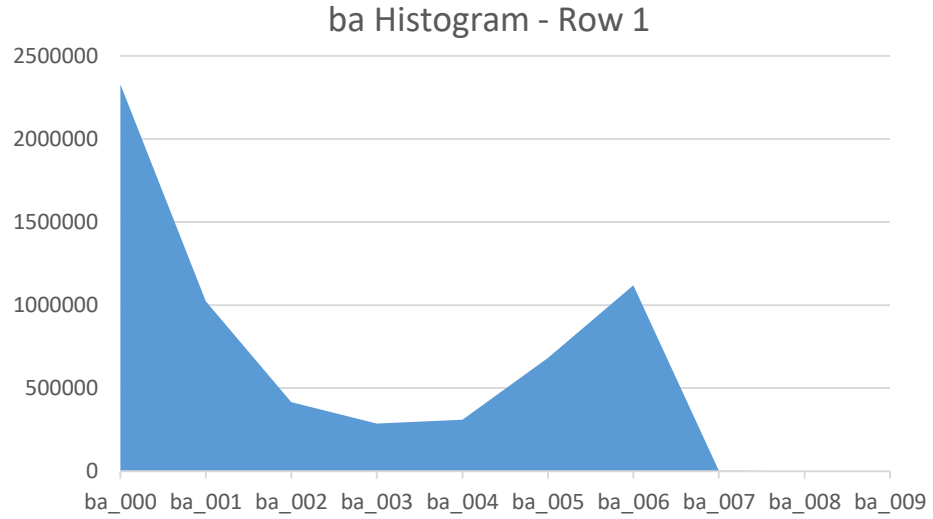
Έχει 170 στήλες + 1 class (0 εάν η βλάβη του φορτηγού δεν σχετίζεται με το APS και 1 εάν σχετίζεται) οι οποίες έχουν ανωνυμοποιηθεί.

Τα χαρακτηριστικά μας χαρακτηρίζονται σε 2 κατηγορίες : standalone και histogram.

- Standalone : η στήλη είναι ένα ξεχωριστό χαρακτηριστικό του dataset (πχ aa_000,ab_000)
- Histogram : μια 10-αδα από στήλες αποτελούν ένα ιστόγραμμα (πχ ag_000 – ag_009)

Συνολικά το dataset μας αποτελείται από 100 μοναδικά χαρακτηριστικά και 7 ιστογράμματα



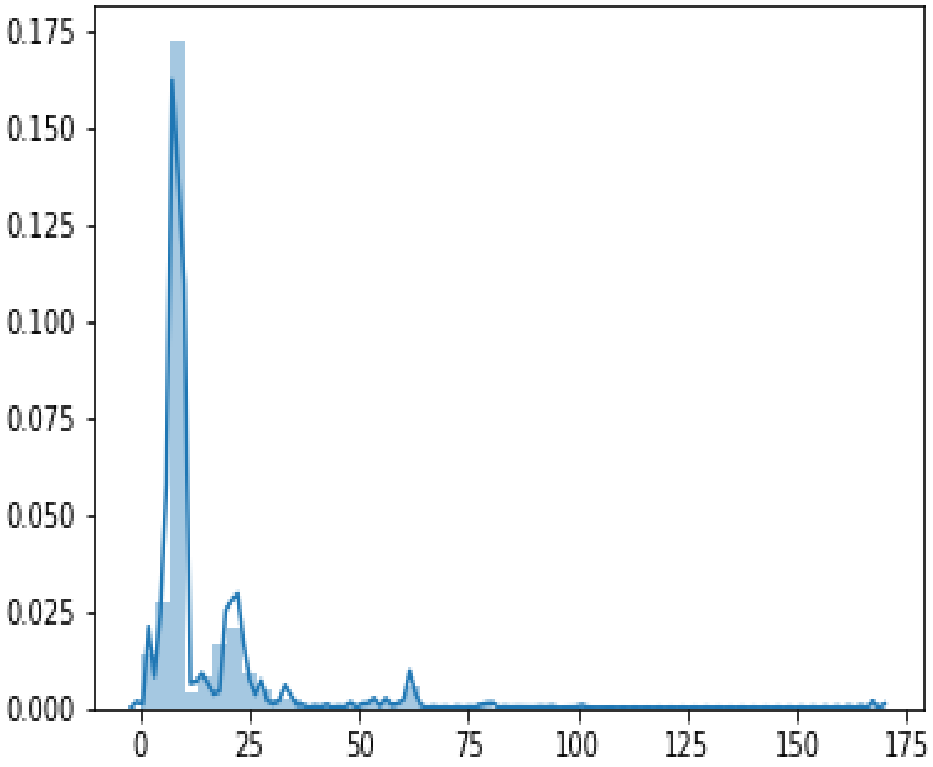


APS Failure at Scania Trucks Dataset - Preprocessing

Προβλήματα που πρέπει να αντιμετωπιστούν στο preprocessing

- High Dimensionality : 170 features , 60000 γραμμές
- Missing Values : Υπάρχουν 850.015 NA (το 17% των γραμμών περιέχει 20 NA)
- Highly Imbalanced : 59.000 καταγραφές της κλάσης 0 και μόνο 1000 της κλάσης 1

Κατανομή του πλήθους των NA



Για να μειωθεί ο αριθμός των γραμμών αλλά και το πλήθος των missing values αφαιρέθηκαν οι γραμμές που είχαν πάνω από 60 NA (μικρή έως μηδενική απώλεια πληροφορίας)

Η κλάση που θέλουμε να προβλέψουμε περιέχει πολύ λίγες περιπτώσεις της θετικής κλάσης (1). Πρέπει να προσέξουμε να μην διώξουμε πολλές.
Μετά της αφαίρεση των γραμμών έμειναν 57.079 περιπτώσεις της κλάσεις 0 και 947 της κλάσεις 1 (1000 αρχικά)

APS Failure at Scania Trucks Dataset - Preprocessing

Για τις υπόλοιπες τιμές που λείπουν έγινε αντικατάσταση τους λαμβάνοντας υπόψιν τους 50 κοντινότερους γείτονες τις εκάστοτε γραμμής. Η τιμή που δόθηκε ήταν η μέση τιμή των χαρακτηριστικών των γειτόνων.

Η μετρική που χρησιμοποιήθηκε για την εύρεση των γειτόνων ήταν:

$$D_i(x_i, y_i) = \begin{cases} 1 \\ d_i(x_i, y_i) \end{cases}$$

Η απόσταση δύο χαρακτηριστικών ήταν η διαφορά τους όταν και, τα δύο δεν είναι NA αλλιώς 1

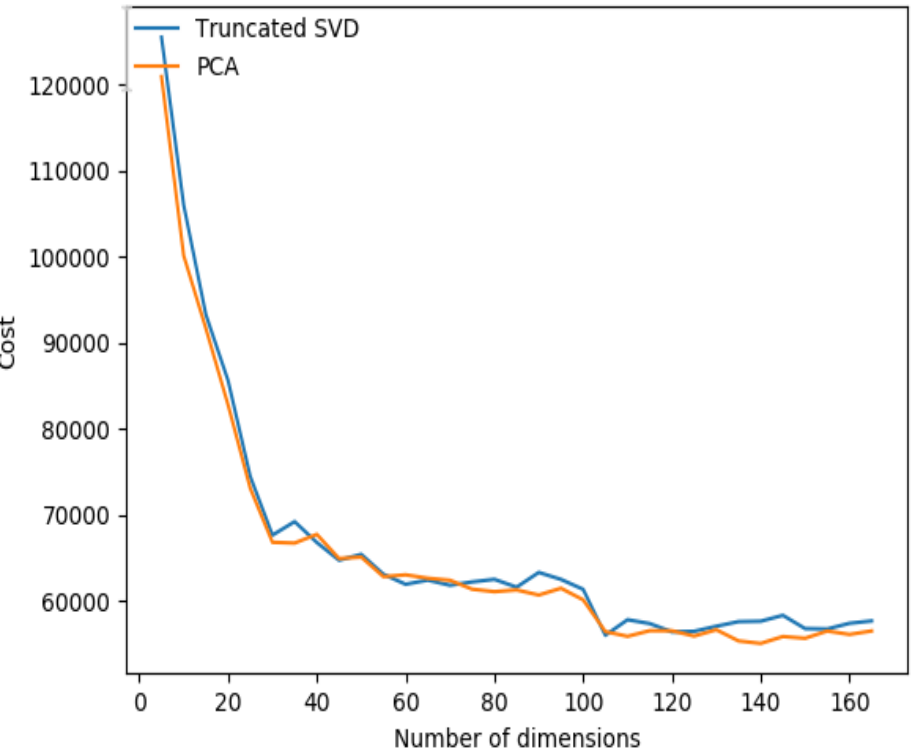
$$D(x, y) = \sum D_i(x_i, y_i)$$

Η απόσταση δύο γραμμών είναι το άθροισμα των αποστάσεων των χαρακτηριστικών τους

APS Failure at Scania Trucks Dataset - Preprocessing

Dimensionality Reduction : Προβολή των αρχικών δεδομένων σε διανυσματικών υποχώρο μικρότερης διάστασης

- Principal Component Analysis
- Singular Value Decomposition



Μεθοδολογία που ακολούθησα : εκπαίδευση των δεδομένων στο training set με τον XGBoost, πρόβλεψη του testing set, καταγραφή του κόστους (μετρική που χρησιμοποιήθηκε στον διαγωνισμό IDA)

Cost = 500 * FN + 10 * FP

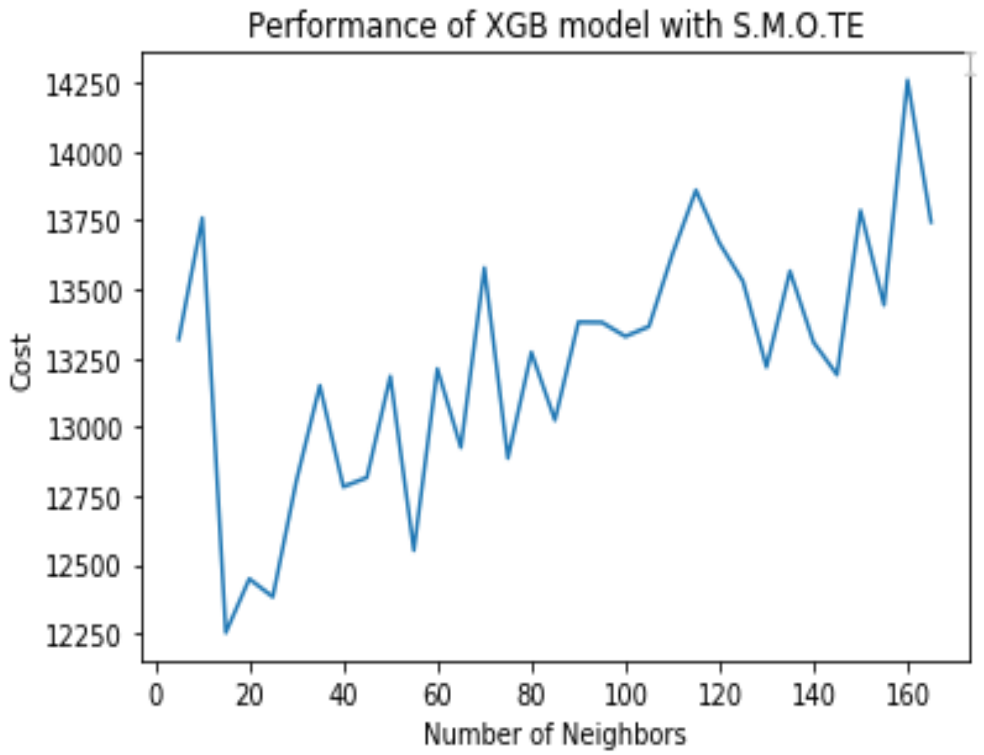
Αξίζει να προσέξουμε ότι σε πολύ μικρές διαστάσεις το κόστος εκτινάσσεται. Οφείλεται στο γεγονός ότι στα αρχικά δεδομένα η κλάση 1 είχε λίγους αντιπρόσωπους και ‘χάνεται’ η πληροφορία τους όσο μειώνεται η διάσταση

Το χαμηλότερο κόστος επιτεύχθηκε με τον PCA στις 140 διαστάσεις

APS Failure at Scania Trucks Dataset - Preprocessing

Imbalanced Class : Προβολή των αρχικών δεδομένων σε διανυσματικών υποχώρο μικρότερης διάστασης

- Under sampling (επιλεκτική δειγματοληψία)
- Oversampling (δημιουργία τεχνητών δεδομένων)



Hint : σε imbalanced dataset η ακρίβεια (accuracy) δεν είναι αποτελεσματικό metric (Accuracy Paradox).
Recall : Μέτρο πληρότητας
Precision : Μέτρο ακρίβειας
F-score : Σταθμισμένος μέσος του Recall και Precision

Η στρατηγική που ακολούθησα ήταν oversampling με την βοήθεια του αλγορίθμου S.M.O.TE (Synthetic Minority Oversampling Technique)

Το χαμηλότερο κόστος επιτεύχθηκε με την χρήση των 15 κοντινότερων γειτόνων

APS Failure at Scania Trucks Dataset – Κατασκευή μοντέλων - Αποτελέσματα

Για τη δημιουργία των μοντέλων θα χρησιμοποιήσουμε τα χαρακτηριστικά 140 χαρακτηριστικά του PCA.
Πού θα κατασκευάσουμε τα μοντέλα? : 10 – fold cross validation (fixed seed)
Τι metric θα χρησιμοποιήσουμε για να αξιολογήσουμε τα μοντέλα? : Cost (Requested by Scania)

Algorithm\fold	1	2	3	4	5	6	7	8	9	10	Mean	StDev	Time
XGBoost	12010	12370	11950	12110	11640	12120	12590	12650	12460	12630	12253	321,654	2255.5
CART	20000	22480	29200	23240	23210	33130	21140	20640	33330	25410	25178	4738,55	336.9
Random Forest	12660	12330	15940	12410	12430	14960	12690	13360	13570	11310	13166	1298,22	113.1
Naïve Bayes	16660	16300	16700	16710	16690	16640	16640	16660	16660	16660	16632	112,942	5.7
Adaboost	15600	14520	16950	15910	14310	15220	15880	14240	14030	16970	15363	1029,64	1130.3
SVM	29830	26410	49750	29000	29190	62380	38670	36530	27130	49230	37812	11570,1	9064.4
Neural Network	13040	16580	19620	16720	13630	13370	13510	12940	13860	14830	14810	2064,12	794.7
KNN	18920	20150	18940	18930	18900	18910	19910	18950	18920	19910	19244	492,467	641.6

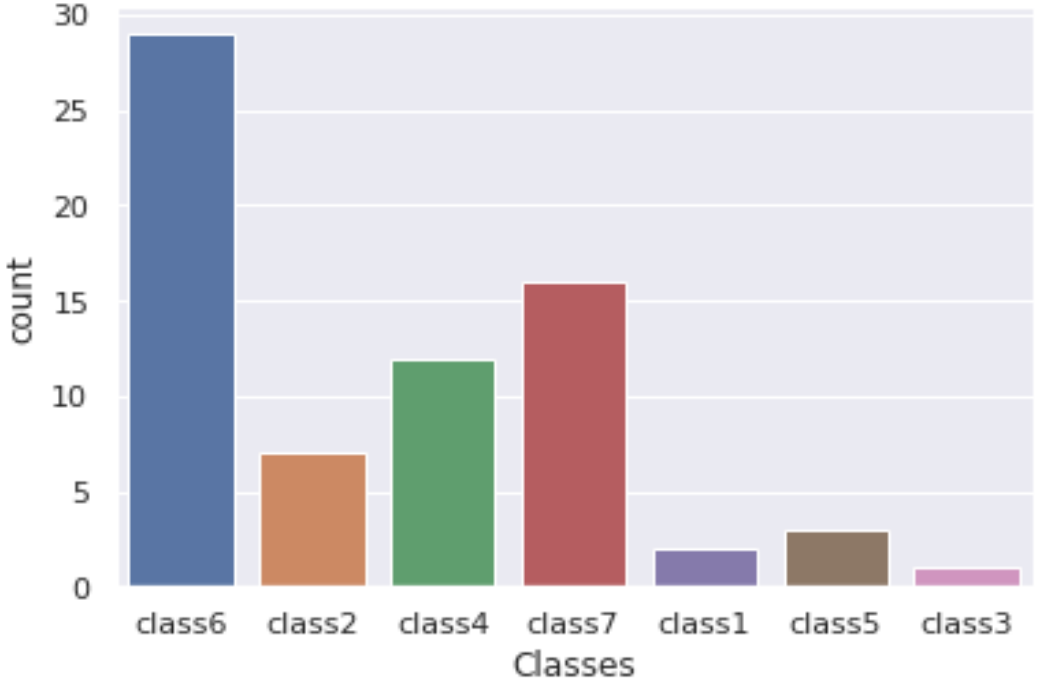
SCADI Dataset

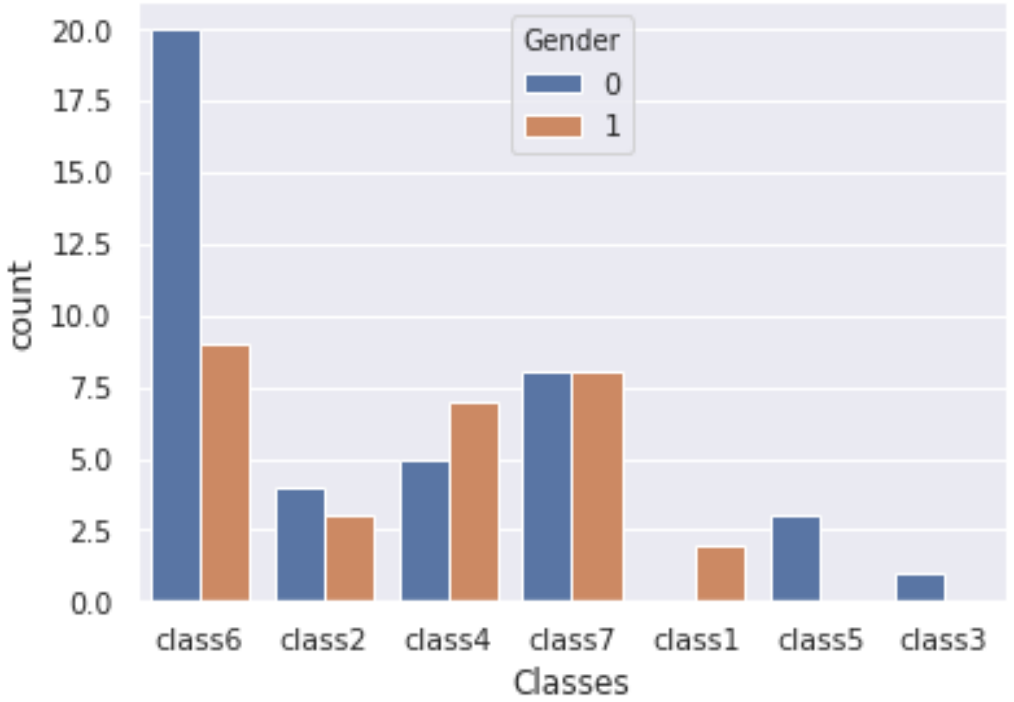
Περιέχει πληροφορίες από 70 παιδιά με σωματικές και κινητικές δυσκολίες.
Για την ακρίβεια περιέχει την ηλικία, το φύλο, και 203 χαρακτηριστικά των οποίων τα ονόματα είναι κωδικοποιημένα με βάση το ICF – CY (International Classification of Functioning – Children and Youth).
Για παράδειγμα το χαρακτηριστικό d 5400 αφορά την ικανότητα του παιδιού να ντυθεί μόνο του.
Τα χαρακτηριστικά αυτά είναι λογικές μεταβλητές (0 ή 1) ανάλογα με αν το παιδί έχει ή όχι την συγκεκριμένη δυσκολία.
Η κλάση που θέλουμε να προβλέψουμε είναι στο είδος τις δυσκολίας που έχει το κάθε παιδί και έχει 7 κατηγορίες οι οποίες έχουν καθοριστεί από εργοθεραπευτές.
Οι κατηγορίες είναι :

- class 1 : Caring for body parts problem
- class 2 : Toileting problem
- class 3 : Dressing problem
- class 4 : Washing oneself and Caring for body parts and Dressing problem
- class 5 : Washing oneself, Caring for body parts, Toileting and Dressing problem
- class 6 : Eating, Drinking, Washing oneself, Caring for body parts, Toileting, Dressing, Looking after one’s health and Looking after one’s safety problem
- class 7 : No problem

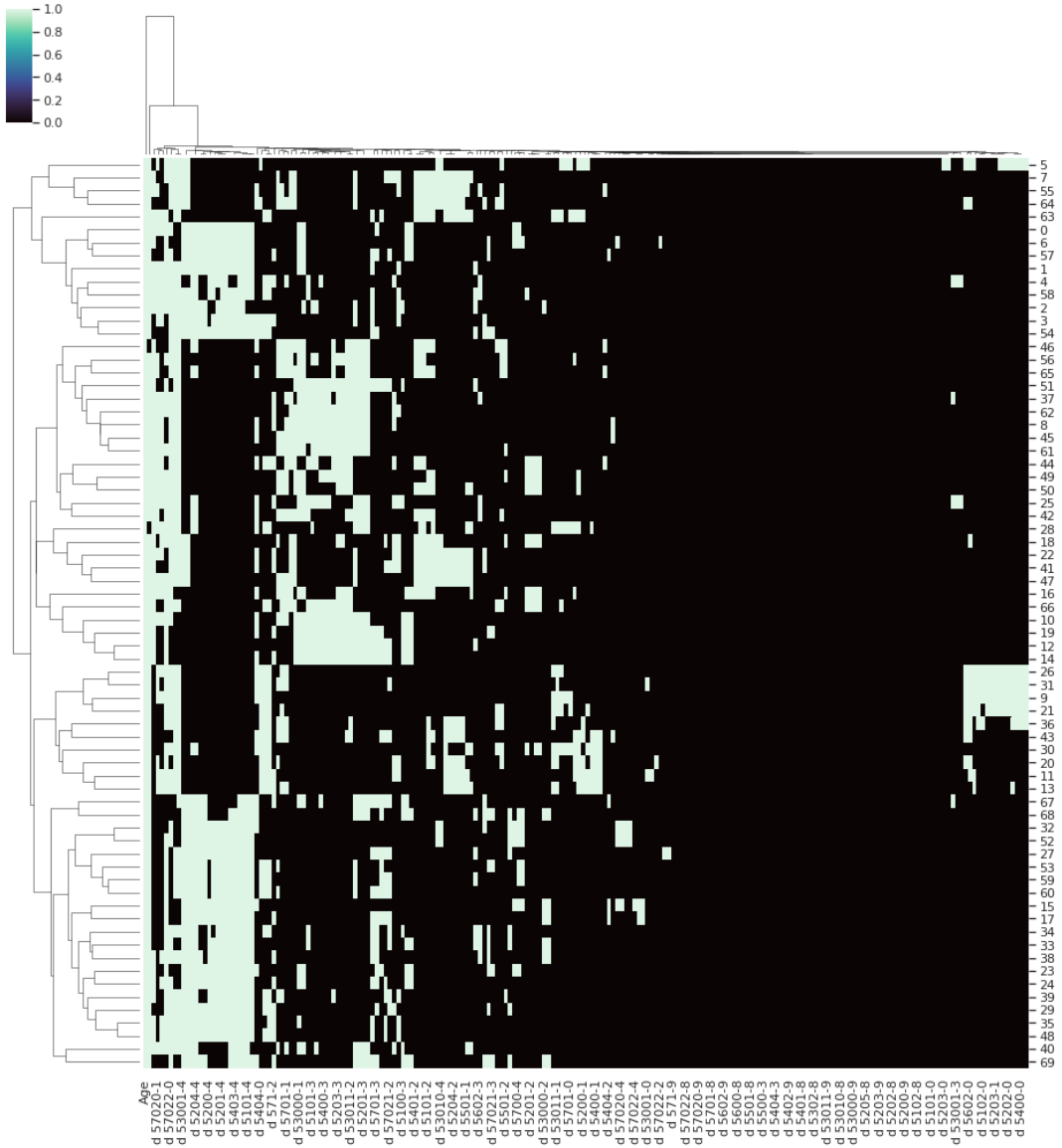
SCADI Dataset - EDA

Υπάρχει μεγάλο πρόβλημα στην κατανομή των κλάσεων το οποίο οφείλεται στο μικρό πλήθος παρατηρήσεων.





SCADI Dataset - EDA



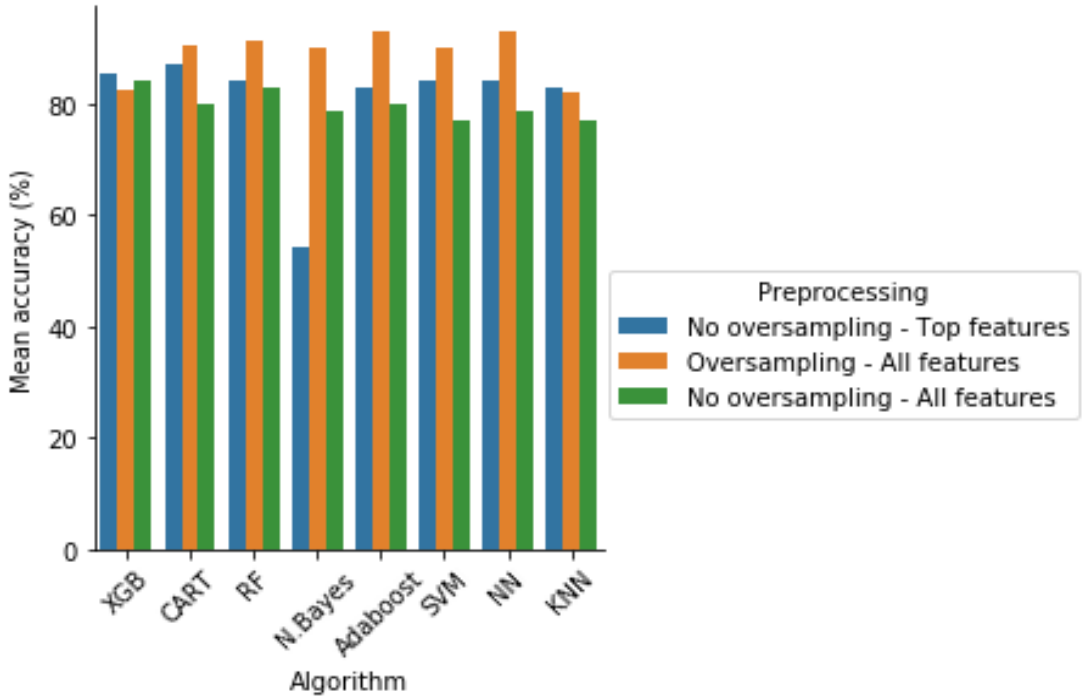
Seaborn's Clustermap : Μας επιτρέπει να βρούμε cluster χρησιμοποιώντας τον ιεραρχικό αλγόριθμο.
Οι κάθετες μαύρες γραμμές δείχνουν ότι σε εκείνα τα χαρακτηριστικά όλες οι παρατηρήσεις είχαν την ίδια τιμή.
Υπήρχαν 62 στήλες στις οποίες τα χαρακτηριστικά είχαν όλα την ίδια τιμή.
Αυτές οι στήλες δεν παρείχαν καμμιά πληροφορία και αφαιρέθηκαν μειώνοντας αισθητά τις διαστάσεις του προβλήματος.

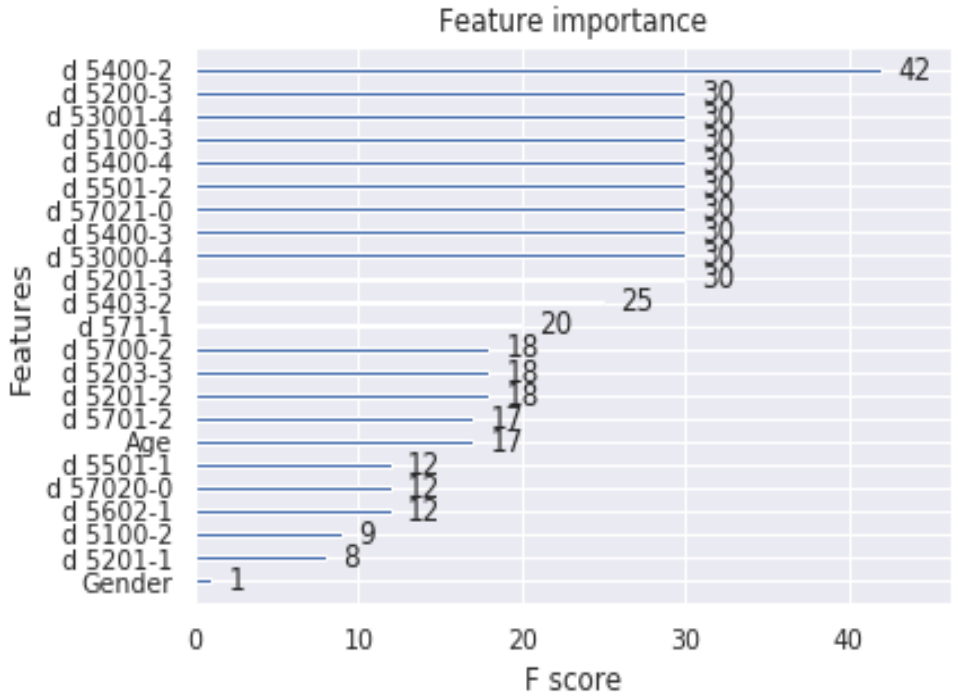
SCADI Dataset - EDA

Παρόλο που τα χαρακτηριστικά συνεχίζουν να είναι πολλά ακόμα και μετά τη μείωση τους η εφαρμογή των PCA και SVD δεν βοήθησαν στην προβλεπτική ικανότητα των μοντέλων.
Έγιναν δοκιμές με τρεις διαφορετικές εκδοχές για το Preprocessing:

- Χωρίς oversampling και χρήση μόνο των καλύτερων χαρακτηριστικών (F-score > 30)
- Με oversampling (S.M.O.TE) και χρήση όλων των χαρακτηριστικών
- Χωρίς oversampling και χρήση όλων των χαρακτηριστικών

Η χρήση του oversampling με όλα τα χαρακτηριστικά είχε το καλύτερο αποτέλεσμα.
Την σημαντικότητα των χαρακτηριστικών την βρίσκουμε εκπαιδεύοντας τον XGBoost στα αρχικά δεδομένα





SCADI Dataset - Αποτελέσματα

Για τη δημιουργία των μοντέλων θα χρησιμοποιήσουμε και τα 205 χαρακτηριστικά του dataset.

Πού θα κατασκευάσουμε τα μοντέλα? : 10 – fold cross validation (fixed seed)

Τι metric θα χρησιμοποιήσουμε για να αξιολογήσουμε τα μοντέλα? : Accuracy

Algorithm\fold	1	2	3	4	5	6	7	8	9	10	Mean	StDev	Time
XGBoost	81,81	54,54	100	80	70	100	90	70	100	80	82,64	14,42932933	13,15
CART	90,9	81,81	90,9	80	70	100	100	90	100	100	90,36	9,835613301	0,05
Random Forest	100	72,72	100	80	80	100	100	90	90	100	91,27	9,911702982	1,59
Naïve Bayes	100	90,9	100	90	80	100	90	80	90	80	90,09	7,750670939	0,05
Adaboost	100	100	100	90	70	100	90	90	90	100	93	9	3,38
SVM	100	72,72	100	80	80	100	100	90	90	90	90,27	9,475539879	0,11
Neural Network	100	81,81	100	90	80	100	100	90	90	100	93,18	7,522556015	2,2
KNN	100	81,81	100	80	80	100	100	90	90	100	92,18	8,482384629	0,03

Συμπεράσματα

Ο XGBoost ήταν ο αλγόριθμος με την καλύτερη απόδοση στα πρώτα δυο dataset, αλλά δεν μπόρεσε να κερδίσει τα Νευρωνικά δίκτυα στο τρίτο.

- Δεν υπάρχει « μαγικός » αλγόριθμος που να έχει την καλύτερη απόδοση σε κάθε σετ δεδομένων
- Η απόδοση των αλγορίθμων εξαρτάται από το preprocessing αλλά και την φύση των δεδομένων
- Κάθε αναλυτής θα πρέπει να έχει γνώση των διαθέσιμων αλγορίθμων για να επιλέγει τον καλύτερο ανάλογα με τα δεδομένα που επεξεργάζεται
- Οι αλγόριθμοι boosting έχουν μεγαλύτερη μέση απόδοση από τους αλγορίθμους bagging αλλά είναι επιρρεπής σε overfitting σε αντίθεση με τους bagging