# Εργασία Εξαμήνου: «Ανάλυση και Διαχείριση Μεγάλων και Πολυδιάστατων Δεδομένων 2023-2024»

Ιωάννης Τσάμπρας ΣΜΗΝ 1066584

Δομή Εργασίας:

1. Προεπεξεργασία Clinical
   1. Αριθμιτικοποίηση
   2. Καθαρισμός
   3. Μέθοδοι Συμπλήρωσης Κενών
2. Κατηγοριοποίηση Clinical
   1. Μέθδοι
   2. Αποτελέσματα
3. Προεπεξεργασία Beacons
   1. Κανονικοποίηση
   2. Εξαγωγή Χαρακτηριστηκών
4. Συνδιασμός Clinical και Beacons Features
5. Συσταδοποίηση τελικού Dataset
   1. Συσταδοποίηση
   2. PCA
   3. Αποτελέσματα
6. Προεπεξεργασία Clinical

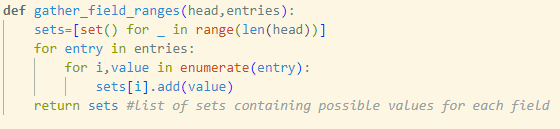
## Αριθμιτικοποίηση

Αρχικά φορτώνουμε το .csv αρχείο στη μνήμη ως δισιδιάστατό array. Παρατήρησα πως το αρχείο που μας δώθηκε χρησιμοποιούσε encoding “utf-8-sig». Ύστερα πραγματοποιούμε έλεγχο δομής (αν όλες οι γραμμές έχουν ίδιο αριθμό στηλών) και ξεχωρίζουμε header από body.

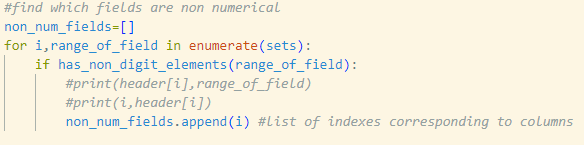


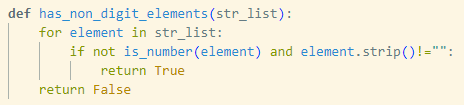
Η πρώτη μας σημαντική δοκιμασία είναι η αριθμιτικοποίηση των δεδομένων. Για αυτό τον σκοπό δημιουργούμε μια δομή πίνακα από «set» , ένα για κάθε στήλη, και ελέγχοντας κάθε γραμμή προσθέτουμε τις τιμές του κάθε κελιού στο set της αντίστοιχης στήλης.

Αφού προσπελάσουμε όλο το dataset έχουμε για κάθε πεδίο το σχετικό σύνολο τιμών.

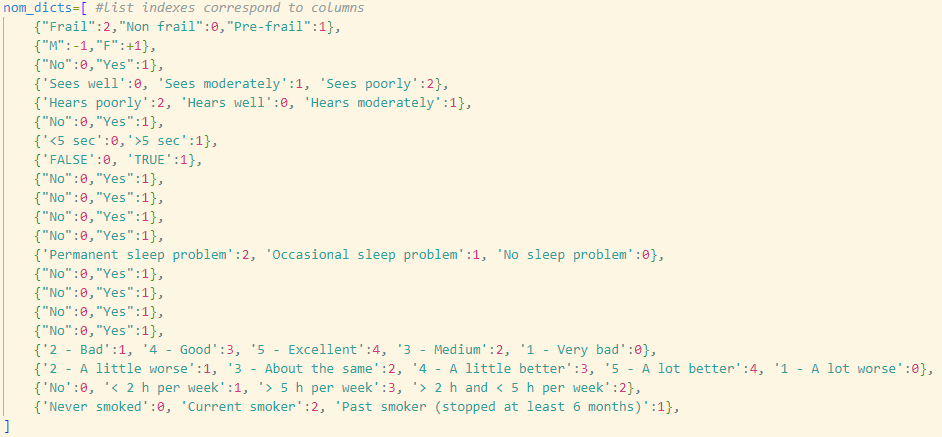


Έχοντας τα σύνολα τιμών διερευνούμε ποιά πεδία είναι αριθμητικά και ποιά όχι καθώς απαιτούν διαφορετική προεπεξεργασία. Εδώ είναι σημαντικό να πούμε πως αγνοούμε κενούς χαρακτήρες, οπότε αριθμητικές στήλες με κενά δεν θεωρούνται μη αριθμητικές.





Αφού εκτυπώσουμε τα μη αριθμητικά πεδία και τα σύνολα τιμών τους κατασκευάζουμε με το χέρι το παρακάτω λεξικό που αντιστοιχεί κάθε πιθανή σωστή είσοδο στην αντίστοιχη αριθμιτική τιμή της.



Ύστερα προσπελνούμε το dataset εφαρμόζοντας το λεξικό για την αριθμητικοποίηση, όσες τιμές δεν συναύδουν με το πεδίο ορισμού του λεξικού (π.χ. κενά κελιά) αντικαθίστονται με “None”.

A screenshot of a computer

Description automatically generated

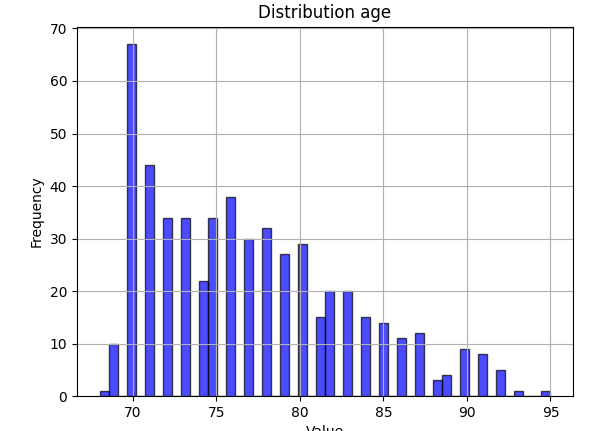
## Καθαρισμός

Για τις αριθμητικές στήλες πρέπει να προχωρήσουμε σε καθαρισμό λανθασμένων τιμών, τις σκέτες κενές εισόδους τις αντικαθιστούμε με None. Κατά τη γνώμη μου είναι λάθος να διαγράψουμε εντελώς όλες τις «999» εγγραφές καθώς σε πολλά πεδία το «999» μπορεί να χρησημοποιείται από τους ερευνητές για να δηλώσουν «άπειρο» χρόνο.

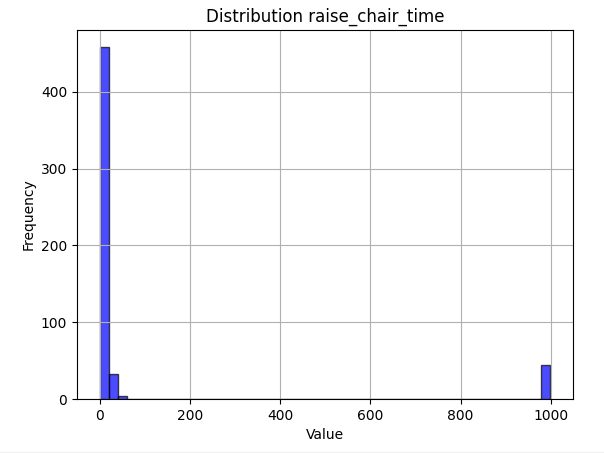
Π.χ. στην μέτρηση χρόνου σηκώματος, το 999 χρησημοποιείται αρκετά συχνά για να είναι απλό λάθος και μάλλον υποδεικνύει αδυναμία ολοκλήρωσης του τέστ (ο συμμετέχον μπορεί να μην μπορεί να σηκωθεί από το κρεββάτι). Αυτή η πληροφορία δεν πρέπει να χαθεί οπότε αντί της πλήρης διαγραφής της τιμής την αντικαθιστώ με ένα ανώτατο όριο που προκύπτει από τις υπόλοιπες μη «ακραίες» εγγραφές.

Για να αποκτήσουμε intuition για τα δεδομένα μας πριν προχωρήσουμε σε αλλαγές εκτελούμε γραφικές αναπαραστάσεις των κατανομών κάθε αριθμητικού πεδίου και ελέγχουμε για ακραίες τιμές (όχι μόνο τις «999») ενώ ταυτόχρονα με βάση τον αριθμό εμφάνισης εγγραφών «999» και την κατανομή των δεδομένων αποφασίζουμε τη διαγραφή ή εφαρμογή ανώτατου ορίου.

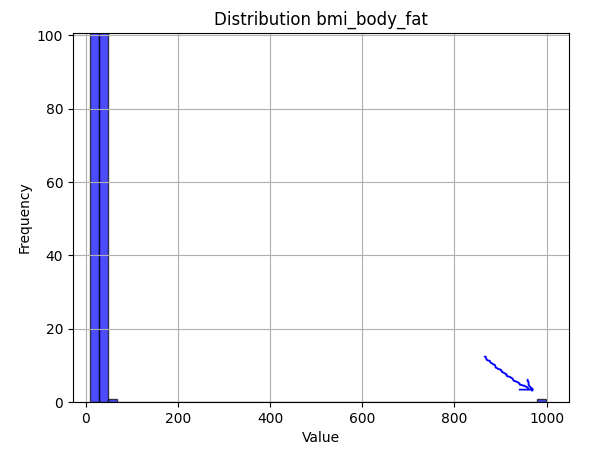
Μερικές από τις κατανομές φαίνονται παρακάτω:



Παράδειγμα Καλής Κατανομής



Παράδειγμα Προβληματικής Κατανομής με Σημασιολογικής Βαρύτητας Λάθη



Παράδειγμα Προβληματικής Κατανομής με Καθαρά Λάθη

Για τον καθαρισμό ή την εφαρμογή ορίου στις αριθμητικές κατανομές κατασκευάστηκε η παρακάτω λίστα κανόνων όπου :

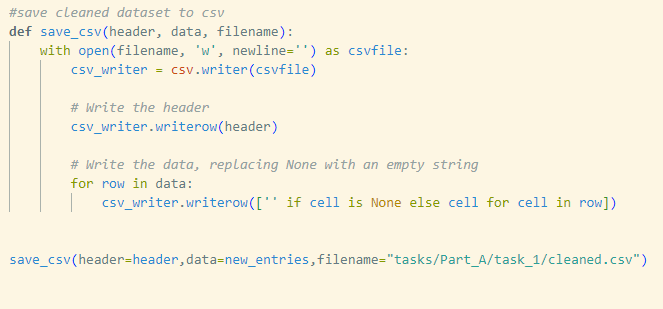
* None -> Καμία Αλλαγή
* ‘<T’ -> Διαγραφή όσων κελιών είναι μεγαλύτερα του Τ
* ‘>T’ -> Διαγραφή όσων κελιών είναι μικρότερα του Τ
* ‘x<T’ -> Αλλαγή όσων κελιών είναι μεγαλύτερα του Τ σε Τ
* ‘x>T’ -> Αλλαγή όσων κελιών είναι μικρότερα του Τ σε Τ



Ύστερα προχωρούμε σε εφαρμογή των κανόνων μας.



Και τέλος σώζουμε το καθαρισμένο και αριθμητικοποιημένο μας dataset σε νέο csv.



## Μέθοδοι Συμπλήρωσης Κενών

Στα πλαίσια της εργασίας αξιοποιήθηκαν **δύο μέθοδοι για την συμπλήρωση** των κενών οι οποίοι συγκρίνονται και κατά την κατηγοριοποίηση.

Ο πρώτος είναι η πιο naive προσσέγγιση όπου γεμίζουμε κάθε κελί με την **μέση τιμή** της στήλης. Τα αποτελέσματα αυτής της μεθόδου βρίσκονται στο “cleanedANDfilled2.csv”.

Ο δεύτερος τρόπος **είναι η εκπαίδευση N feedforward νευρωνικών δικτύων**, όπου N το πλήθως των στηλών με έστω και μια κενή εγγραφή, για την **πρόβλεψη των κενών τιμών** με βάση τα περιεχόμενα της γραμμής.

Η εκπαίεδευση των νευρωνικών δικτύων γίνεται πάνω στις γραμμές που δεν έχουν κανένα κενό εξαρχής (περίπου οι μισές εγγραφές) και αν μια εγγραφή έχει πάνω από μία κενή τιμή τότε οι κενές τιμές της συμπληρώνονται πρώτα με βάση την μέθοδο μέσων τιμών και μετά εφαρμόζονται τα νευρωνικά για την διόρθωση των τιμών της.

Η εκπαίδευση για κάθε νευρωνικό γίνεται μία φορά και αποθηκεύεται στη μνήμη σε περίπτωση που προκύψει πάλι κενό ίδιας στήλης.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

Οι υπερπαράμετροι του νευρωνικού δικτύου και της εκπαίδευσής του αποκτήθηκαν με βάση πειραμματισμό. Οι στήλες με ακέραιες τιμές στρογγυλοπούνται στην κοντινότερη, της πρόβλεψης, ακέραια τιμή.

Όπως φαίνεται στην παρακάτω φωτογραφία το πρόγραμμα κάνει cache τα μοντέλα για επαναχρησιμοποίηση σε μετεγενέστερες εγγραφές.

A screenshot of a computer program

Description automatically generated

Μέρος της εκτέλσης του complete.py

Τέλος το «γεμισμένο» dataset σώζεται στο αρχείο “cleanedANDfilled.csv”.

# Κατηγοριοποίση Clinical

## Μέθοδοι

Για την κατηγοριοποίηση των ασθενών αξιοποιήθηκαν δύο μέθοδοι:

* KNN
* Feed Forward NN

Για το validation accuracy του νευρωνικού δικτύου εκτελέστηκαν 16 προπονήσεις και κρατήσαμε τον μέσο όρο. Επιπλέον τα αποτελέσματα συγκρίνονται και με τα δύο dataset “cleanedANDfilled.csv” και “cleanedANDfilled2.csv” που παρήχθησαν από τις δύο μεθόδους συμπλήρωσης κενών.

Λόγω της αδυναμίας μου να αξιοποιήσω GPU κατά την εκπαίδευση εφάρμοσα παράλληλη εκτέλση των πολλαπλών προπονήσεων (για την απόκτηση μέσου όρου για τα validation accuracy) σε CPU.

A screenshot of a computer code

Description automatically generated

Parallel Training

A screenshot of a computer program

Description automatically generated

Neural Network

Για την εκπαίδευση αφαιρέθηκαν οι στήλες «0,1, 9, 10, 16, 17, 18» του dataset.

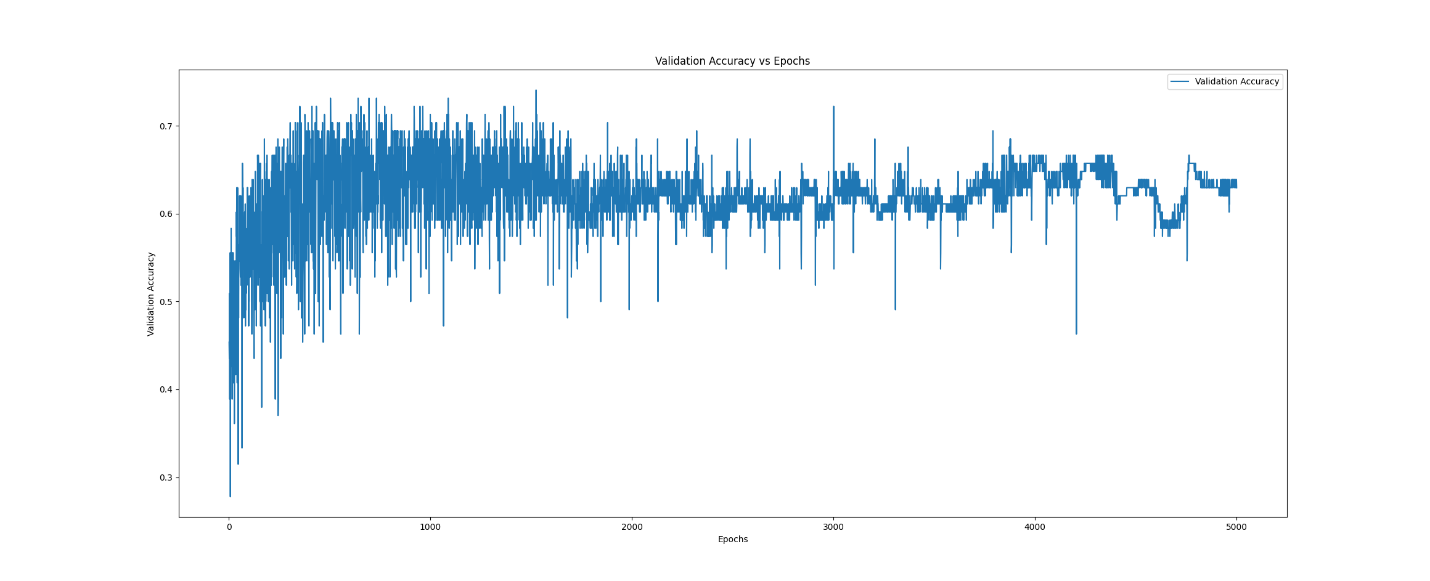
A screenshot of a computer program

Description automatically generated

KNN

Για τους παράμετρους του νευρωνικού έγιναν πολλαπλές δοκιμές και κατέληξα σε :

* 3 layers (1024,128,3)
* Learning rate 0.0001
* 2000 epochs
* Batchsize 64

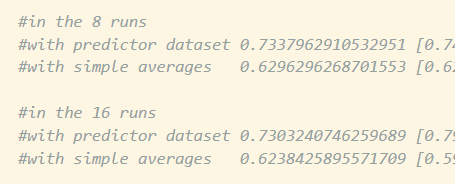


Validation Accuracy through Epochs

## Αποτελέσματα

Για τον ΚΝΝ τα αποτελέσμτα δεν αλλάζουν, με βάση το dataset, με μέγιστο accuracy 0.66 και μέσο 0.59 για τους πρώτους 40 γείτονες.

Για το νευρωνικό δίκτυο πραγματοποιήθηκαν αρχικά 8 και ύστερα επιπλέον 16 trainings με κάθε dataset και από αυτά προέκυψαν οι παρακάτω validation accuracies:



Όπου φαίνεται η **αύξηση της ακρίβειας (κατά 11%) με τη χρήση του dataset όπου οι τιμές συμπληρώθηκαν με τη χρήση νευρωνικού δικτύου** έναντι απλών μέσω τιμών.

# Προεπεξεργασία Beacons

## Κανονικοποίηση

Αρχικά καθαρίζουμε τις εγγραφές από μη προβλεπόμενα IDs και ύστερα συλλέγουμε όλες τις διαφορετικές ονομασίες δωματίων.

A screen shot of a computer code

Description automatically generated

Ύστερα χειροκίνητα κατασκευάζουμε ένα λεξικό αντιστοίχησης όλων των διαφορετικών ονομασιών με τις πραγματικές ονομασίες που επιλέξαμε οι οποίες είναι 8:

* Bathroom
* Bedroom
* Diningroom
* Hall
* Kitchen
* Livingroom
* Office
* Outdoor

Το λεξικό φαίνεται στην παρακάτω εικόνα:

|  |  |
| --- | --- |
| A screenshot of a computer  Description automatically generated | A screenshot of a computer  Description automatically generated |
|  |  |

Ύστερα αλλάζουμε με τη χρήση του λεξικού τις εγγραφές με τα σωστά ονόματα δωματίων ενώ ταυτόχρονα συμπτύσουμε τις στήλες ημερομηνίας και ώρας και τις μετατρέπουμε σε UNIX time για να σιευκολύνουμε την επεξεργασία τους αργότερα.

A screenshot of a computer program

Description automatically generated

Έπειτα κάνουμε sort με βάση το datetime και το participant ID και αποθηκεύουμε το αποτέλεσμα.

A screenshot of a computer code

Description automatically generated

## Εξαγωγή Χαρακτηριστηκών

Για τον υπολογισμό των χρόνων διαμονής σε κάθε δωμάτιο ορίζουμε τις 3 παρακάτω συναρτήσεις γαι να μας διευκολύνουν στις περιπτώσης αλλαγής ημέρας μεταξύ δύο εγγραφών σύμφωνα με τις οδηγίες που μας δώθηκαν.

A screenshot of a computer code

Description automatically generated

Έπειτα βρίσκουμε τις ξεχωριστές διαμονές των συμμετεχόντων

A computer code on a white background

Description automatically generated

Και τέλος υπολογίζουμε τους επιμέρους χρόνους και ποσοστά διαμονής πριν τα αποθηκεύσουμε στο νέο dataset.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

New Dataset Format

1. Συνδιασμός Clinical και Beacons Feature

Ακόμη συνδιάζουμε το Clinical με το New\_Dataset (που προέκυψε από την εξαγωγή χαρακτηριστηκών από το Beacons) σε ένα νέο.

A screenshot of a computer code

Description automatically generated

A close up of a screen

Description automatically generated

Merged Dataset Format

1. Συσταδοποίηση τελικού Dataset
2. Συσταδοποίηση

Τα δεδομένα μας αρχικά διαιρούνται κατά στήλες με τον μέσο της στήλης για να εξισώσουμε τη συνεισφορά κάθε στείλεις στην γεωμτερική απεικόνιση των εγγραφών.

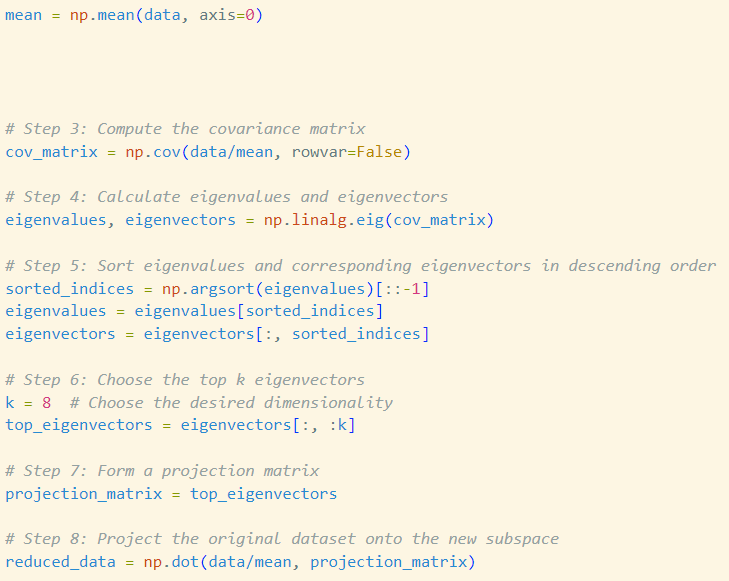
Από τα δεδομένα μας αφαιρούμε τις στήλες των IDs και το Fried-Index.

Για την συσταδοποίηση των δεδομένων αξιοποιούμε k-means και δοκιμάζουμε πολλαπλά πλήθη κεντροειδών. Αξιοποιούμε σε κάθε πιθανό πλήθος την μετρική της «αδράνειας» και το sil index και τα τοποθετούμε σε γραφική παράσταση για να βρούμε τον αγκώνα του γραφήματος και το σχετικό sil index.



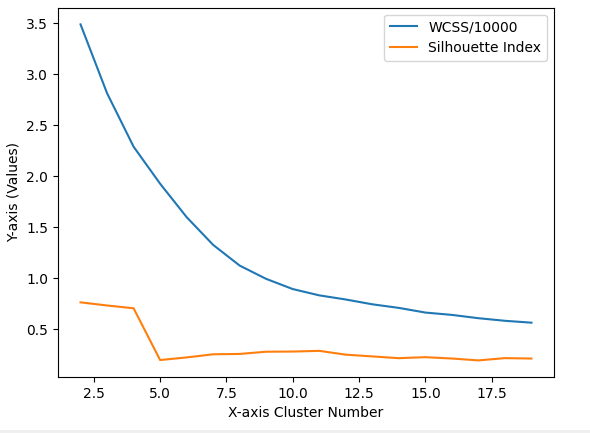
## PCA

Επιπλέον παράγουμε ένα δεύτερο set δεδομένων με χρήση PCA για να μικρύνουμε τη διαστατικότητα. Για το πλήθως των components “k” δοκιμάστηκαν διάφορες επιλογές και τα αποτελέσματα συζητούνται παρακάτω.



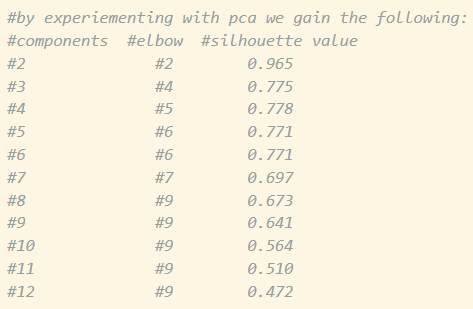
## Αποτελέσματα

Για το απλό clustering παίρνουμε το παρακάτω διάγραμμα.

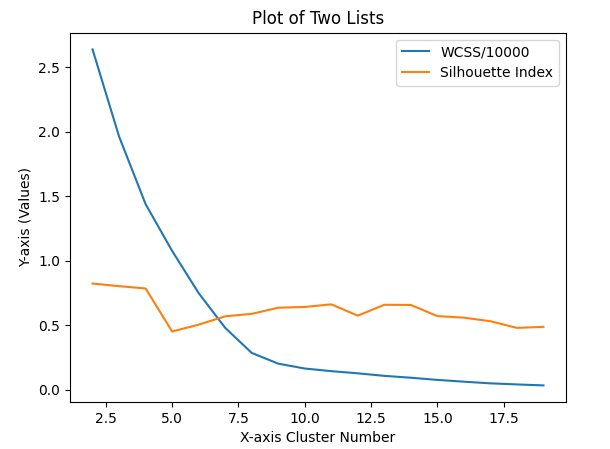


Οπτικά εντοπίζουμε τον αγκώνα του γραφήματος στα 9 κεντροειδή και το αντίστοιχο sil index είναι 0.291, σχετικά ασθενές.

Με PCA αρχικά δοκιμάζουμε πιθανά πλήθη για τα components και συναντάμε τα 9 κεντροειδή στα 8 components (όσα περισσότερα components και αν προσθέσουμε τα κεντροειδή θα παραμείνουν 9 αφού τόσα βρήκαμε και πριν το dimensionality reduction).



Για 8 components παίρνουμε το παρακάτω γράφημα:



Με 9 κεντροειδή και sil index 0.673, καλό.