Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Πανεπιστήμιο Πατρών

Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης

Εργαστηριακή Άσκηση
Εαρινό Εξάμηνο 2024

Μακρυγιάννης Παντελεήμων 1067526 up1067526@ac.webmail.upatras.gr

Εισαγωγή

Η εργασία πραγματοποιήθηκε στο πλαίσιο του μαθήματος «Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης». Στόχος είναι η ανάλυση του συνόλου δεδομένων HARTH (https://archive.ics.uci.edu/dataset/779/harth), η εκπαίδευση τριών ταξινομητών, η σύγκριση τους και ο διαχωρισμός των χρηστών σε συστάδες μέσω 2 διαφορετικών αλγόριθμων συσταδοποίησης, όπου στο τέλος ζητείται και η σύγκριση τους.

Πιο συγκεκριμένα στο πείραμα συμμετείχαν 22 άτομα, από τα οποία το καθένα έφερε 2 αισθητήρες. Ο πρώτος αισθητήρας βρισκόταν στο κάτω μέρος της πλάτης, ενώ ο δεύτερος στο πίσω μέρος του μηρού. Για κάθε ένα από τα 22 άτομα, δημιουργήθηκε και ένα dataset με τις αντίστοιχες μετρήσεις. Το κάθε dataset περιλαμβάνει τη στήλη *timestamp* όπου πρόκειται για την ημερομηνία και ώρα της μέτρησης, τις στήλες *back_x, back_y, back_z* και *thigh_x, thigh_y, thigh_z* με τις τιμές του αισθητήρα ανά διάσταση και τη στήλη *label* η οποία προσδιορίζει τη δραστηριότητα του συμμετέχων.

Η στήλη *label* μπορεί να πάρει τις εξής τιμές:

1: walking
2: running
3: shuffling
4: stairs (ascending)
5: stairs (descending)
6: standing
7: sitting
8: lying
13: cycling (sit)
14: cycling (stand)
130: cycling (sit, inactive)
140: cycling (stand, inactive)

Υλοποίηση

Περιβάλλον και Βιβλιοθήκες

Ως γλώσσα υλοποίησης ορίζεται η Python. Η υλοποίηση του 1^{ου} και 2^{ου} ερωτήματος έγινε στο περιβάλλον του Jupyter σε Notebook. Η υλοποίηση του 3^{ου} ερωτήματος λόγω της ανάγκης των αυξημένων υπολογιστικών πόρων, πραγματοποιήθηκε σε Notebook του Google Colab.
Οι βιβλιοθήκες που χρησιμοποιήθηκαν είναι οι εξής:

- **os**: δίνεται πρόσβαση στα αρχεία του υπολογιστικού συστήματος.

- **pandas**: χρησιμοποιείται για τη δημιουργία και επεξεργασία των DataFrame (δομές δεδομένων).
- **matplotlib.pyplot**: απαραίτητη βιβλιοθήκη για την αναπαράσταση διαγραμμάτων, μέσω του module pyplot.
- **seaborn**: απαραίτητη βιβλιοθήκη για την αναπαράσταση boxplot, heatmap και histplot
- **numpy**:
- **tqdm**: εισάχθηκε για λόγους παρακολούθησης της προόδου σε κάποια σημεία κώδικα που απαιτούσαν αρκετή ώρα για την ολοκλήρωση τους.
- **sklearn**: βασική βιβλιοθήκη για την εκτέλεση του 2^{ου} και του 3^{ου} ερωτήματος. Πιο συγκεκριμένα:
    1. Από την sklearn.metrics χρησιμοποιείται η συνάρτηση classification_report για την εξαγωγή των μετρικών Precision, Recall και F1 score.
    2. Από την sklearn.model_selection χρησιμοποιείται η συνάρτηση train_test_split για τον διαχωρισμό των δεδομένων σε train/test.
    3. Από την sklearn.preprocessing χρησιμοποιείται η συνάρτηση StandardScaler για την κανονικοποίηση των δεδομένων.
    4. Από την sklearn.neural_network επιλέγεται ο MLPClassifier, έτσι ώστε να εκπαιδευτεί το νευρωνικό δίκτυο.
    5. Από την sklearn.ensemble επιλέγεται ο RandomForestClassifier, έτσι ώστε να εκπαιδευτεί ο Random Forest.
    6. Από την sklearn.naive_bayes επίλέγεται ο GaussianNB για να εκπαιδευτεί ο ταξινομητής που είναι βασισμένος σε Bayesian Networks.
    7. Από την sklearn.cluster επιλέγεται η KMeans για την συσταδοποίηση των δεδομένων.
    8. Από την sklearn.cluster επιλέγεται η DBSCAN για την συσταδοποίηση των δεδομένων.
    9. Από την sklearn.cluster επιλέγεται η AgglomerativeClustering για την συσταδοποίηση των δεδομένων.
    10. Από την sklearn.decomposition επιλέγεται η PCA για την μείωση των διαστάσεων, έτσι ώστε να είναι εφικτή η αναπαράσταση των παραπάνω τεχνικών συσταδοποίησης.
    11. Από την sklearn.metrics επιλέγεται η silhouette_score, για την αξιολόγηση των αποτελεσμάτων συσταδοποίησης.

- Από την google.colab επιλέγεται το drive, για την διαχείριση δεδομένων μέσω google drive.

Ανάλυση

Αρχικά εισάγονται όλα τα dataset στο περιβάλλον υλοποίησης και εκτυπώνονται οι στήλες του κάθε dataset, έτσι ώστε να εξεταστεί ότι περιέχουν τον ίδιο αριθμό στηλών, για τη συνέχεια της επεξεργασίας. Παρακάτω παρουσιάζονται κάποιες ενδεικτικές φωτογραφίες:

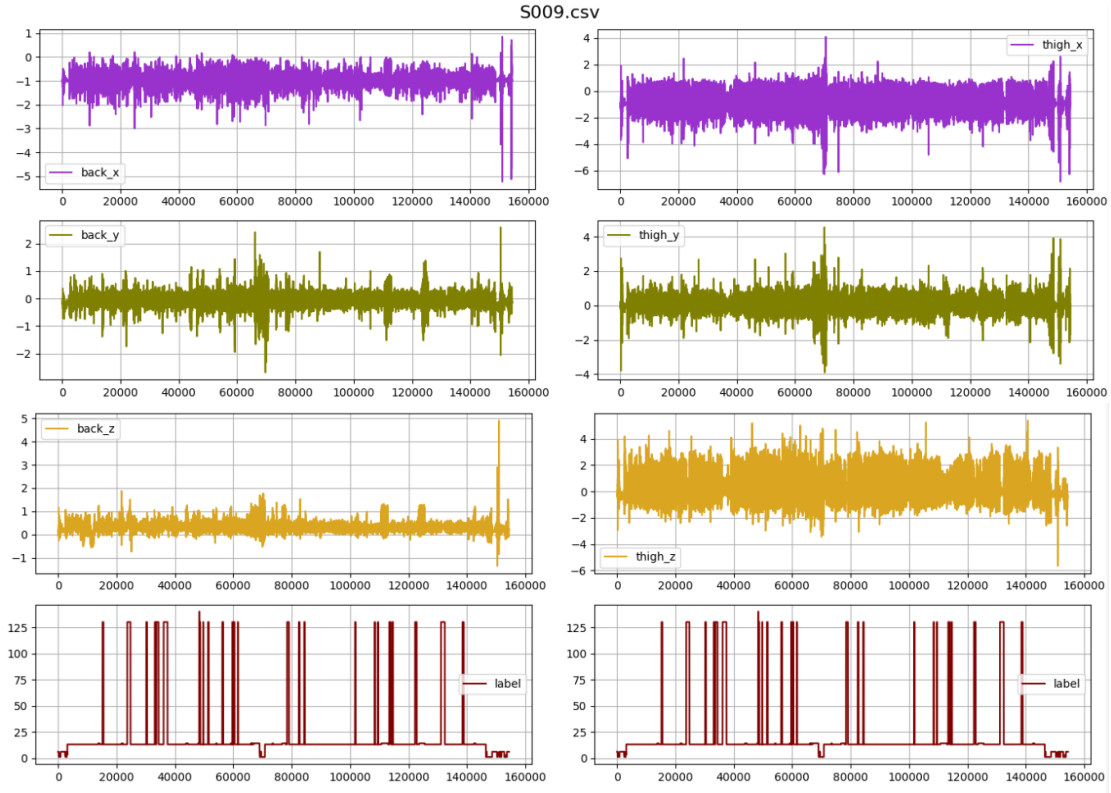
```
File: S006.csv
Columns: ['timestamp', 'back_x', 'back_y', 'back_z', 'thigh_x', 'thigh_y', 'thigh_z', 'label']
File: S015.csv
Columns: ['timestamp', 'index', 'back_x', 'back_y', 'back_z', 'thigh_x', 'thigh_y', 'thigh_z', 'label']
File: S023.csv
Columns: ['Unnamed: 0', 'timestamp', 'back_x', 'back_y', 'back_z', 'thigh_x', 'thigh_y', 'thigh_z', 'label']
```

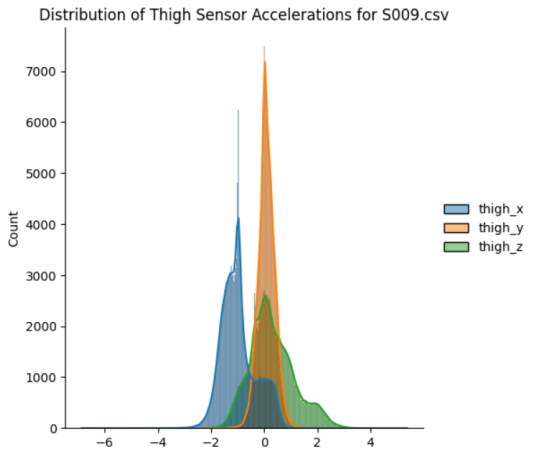
Παρατηρείται ότι σε κάποια datasets υπάρχουν κάποιες στήλες, οι οποίες δεν περιέχουν κάποια σημαντική πληροφορία για την ανάλυση. Έτσι αυτές αφαιρούνται, και πιο συγκεκριμένα από τα dataset S015, S021, S023. Στην συνέχεια γίνεται έλεγχος για μηδενικές τιμές που ενδεχομένως περιλαμβάνονται στα δεδομένα. Μετά από έλεγχο, προκύπτει ότι κανένα dataset δεν περιέχει μηδενικές τιμές. Ενδεικτικά:

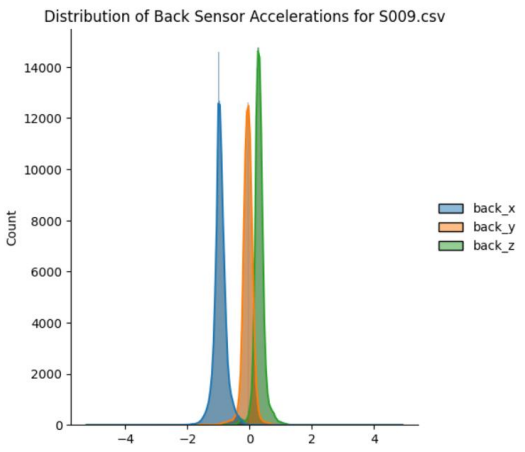
```
Info about: S006
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 408709 entries, 0 to 408708
Data columns (total 8 columns):
 #   Column     Non-Null Count   Dtype
---  ------     --------------   -----
 0   timestamp  408709 non-null  object
 1   back_x     408709 non-null  float64
 2   back_y     408709 non-null  float64
 3   back_z     408709 non-null  float64
 4   thigh_x    408709 non-null  float64
 5   thigh_y    408709 non-null  float64
 6   thigh_z    408709 non-null  float64
 7   label      408709 non-null  int64
dtypes: float64(6), int64(1), object(1)
memory usage: 24.9+ MB
```

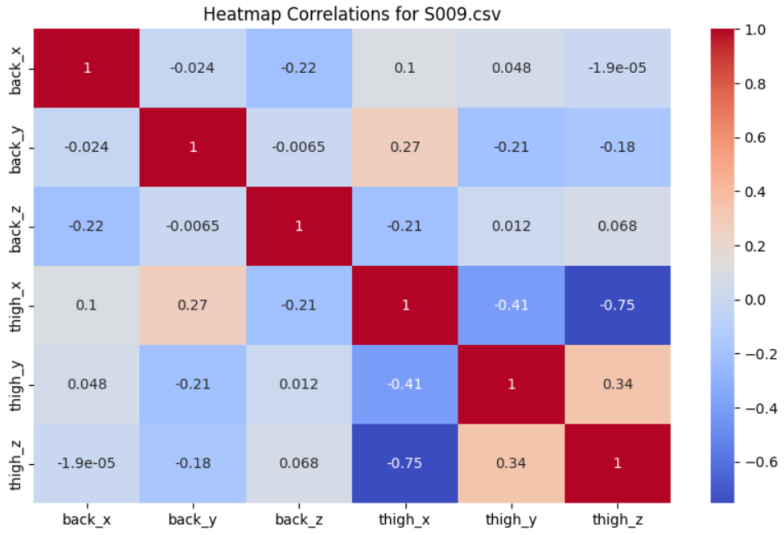
Ανάλυση ανά συμμετέχων

Μετά από αυτούς τους αρχικούς ελέγχους γίνεται μια ανάλυση σε κάθε dataset, δηλαδή για κάθε συμμετέχων του πειράματος. Παρακάτω παρουσιάζονται subplots για κάθε αισθητήρα, όπου φαίνονται οι διακυμάνσεις που λαμβάνει κάθε χρονική στιγμή. Αντίστοιχα subplots παρουσιάζονται και για το label, δηλαδή τη δραστηριότητα του συμμετέχων. Επίσης παρουσιάζεται ένα correlation matrix, το οποίο ελέγχει αν υπάρχουν συσχετισμοί ανάμεσα στους αισθητήρες. Τέλος παρουσιάζονται 2 displots για τους 2 αισθητήρες αντίστοιχα. Ενδεικτικά για τους S009 και S027:

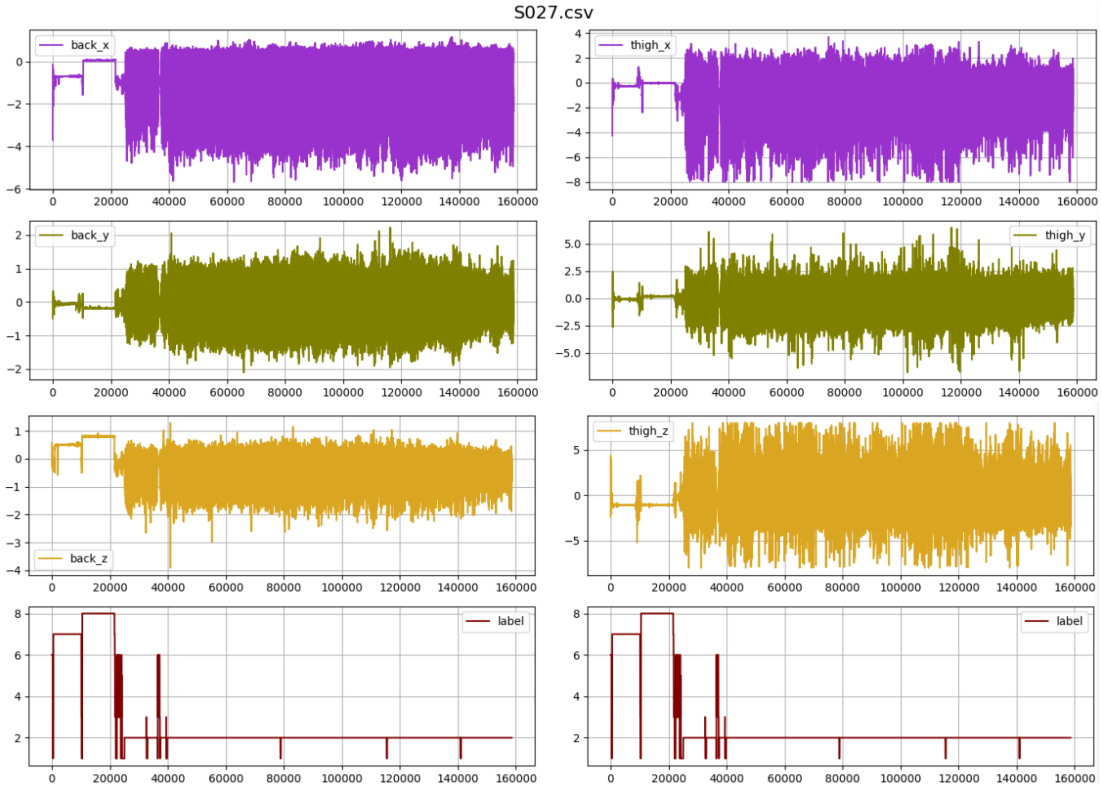


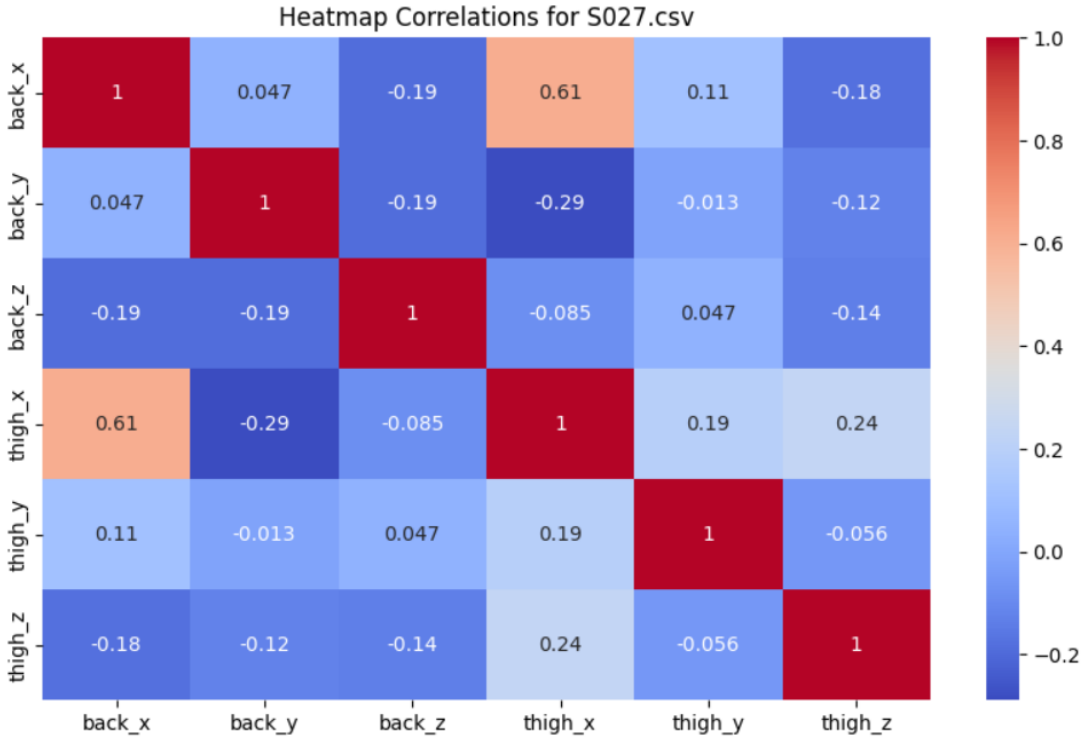






Ο συμμετέχων S009 βρίσκεται κυρίως στις καταστάσεις cycling (sit) (13) και cycling (sit, inactive) (130). Παρατηρείται πως οι κινήσεις που καταγράφονται από τον αισθητήρα του μηρού, είναι πιο έντονες από αυτές της πλάτης. Επίσης φαίνεται να υπάρχει μια σχετικά ισχυρή συσχέτιση μεταξύ των thigh_z και thigh_x.

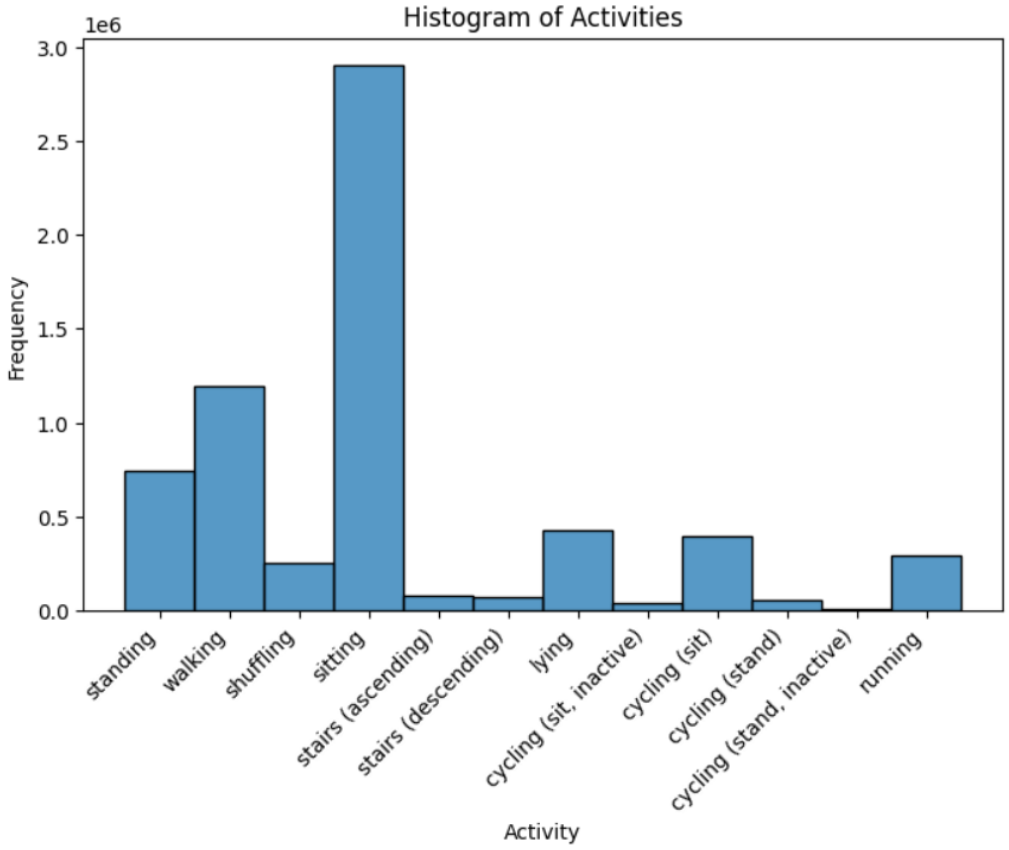




Ο συμμετέχων S027 βρίσκεται κυρίως στην κατάσταση running (2). Αυτός είναι και ο λόγος που οι τιμές που καταγράφει ο αισθητήρας του μηρού είναι μεγαλύτερες από αυτές της πλάτης. Επίσης διακρίνεται μια συσχέτιση μεταξύ των thigh_x και back_x.

Ανάλυση ενιαίου Dataset

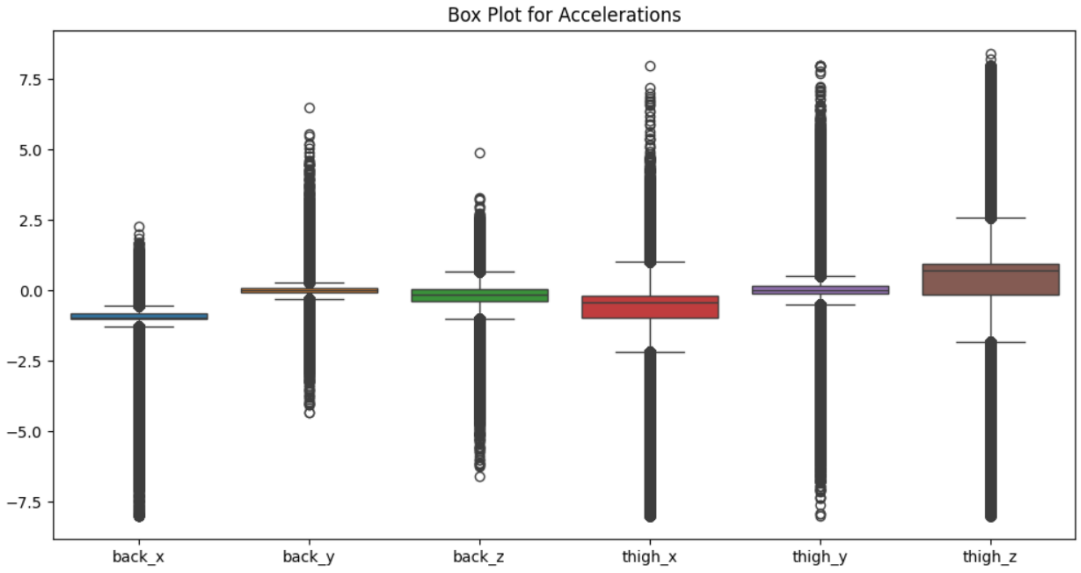
Στην συνέχεια μετά από συνένωση όλων των δεδομένων σε ένα ενιαίο dataframe, παρουσιάζεται παρακάτω ένα ιστόγραμμα όπου φαίνεται η κατανομή των δεδομένων ανά δραστηριότητα.



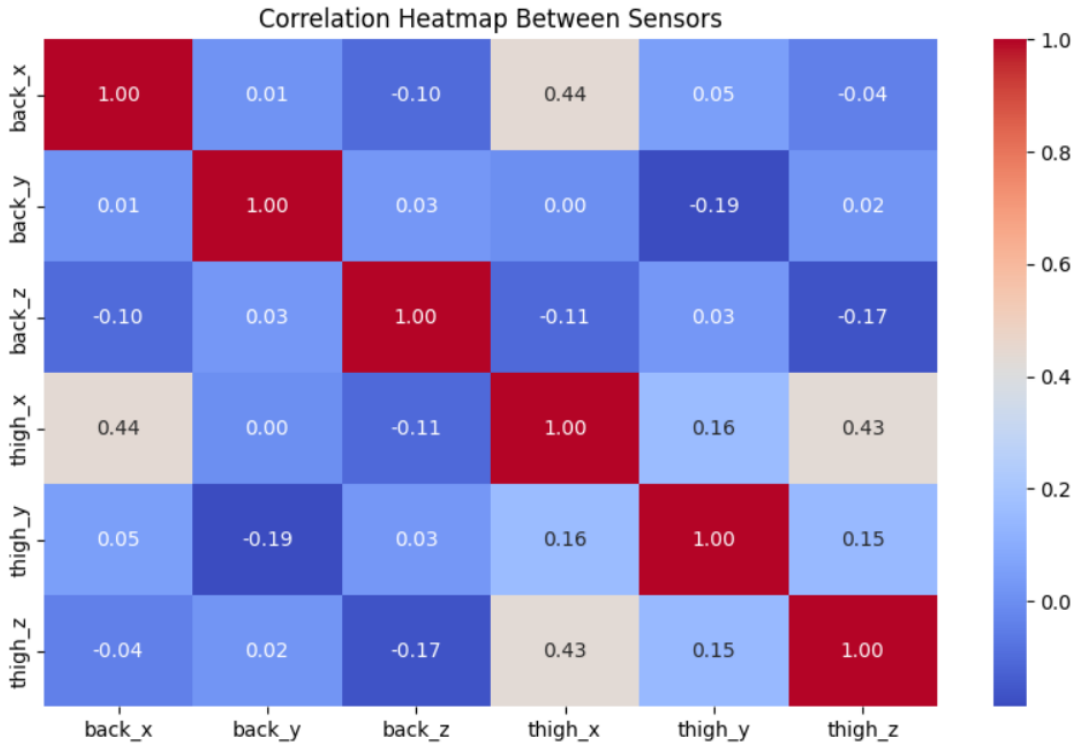
Επίσης παρουσιάζονται και κάποια βασικά στατιστικά στοιχεία του ενιαίου dataframe, όπως και ένα plotbox και ένα correlation matrix:

	back_x	back_y	back_z	thigh_x	thigh_y	thigh_z
count	6461328.000000	6461328.000000	6461328.000000	6461328.000000	6461328.000000	6461328.000000
mean	-0.884957	-0.013261	-0.169378	-0.594888	0.020877	0.374916
std	0.377592	0.231171	0.364738	0.626347	0.388451	0.736098
min	-8.000000	-4.307617	-6.574463	-8.000000	-7.997314	-8.000000
25%	-1.002393	-0.083129	-0.372070	-0.974211	-0.100087	-0.155714
50%	-0.974900	0.002594	-0.137451	-0.421731	0.032629	0.700439
75%	-0.812303	0.072510	0.046473	-0.167876	0.154951	0.948675
max	2.291708	6.491943	4.909483	7.999756	7.999756	8.406235

Παρατηρείται ότι η ελάχιστη τιμή των δεδομένων είναι -8 και η μέγιστη 8.4. Ο μέσος όρος για κάθε κατεύθυνση των αισθητήρων, είναι κοντά στο μηδέν και η τυπική απόκλιση τους μικρή. Αυτό σημαίνει ότι τα δεδομένα ακολουθούν μια κανονική κατανομή.



Από το boxplot επιβεβαιώνονται τα παραπάνω στατιστικά και με γραφικό τρόπο. Επιπλέον φαίνεται πως υπάρχουν αρκετά outliers, παρόλα αυτά φαίνεται πως τα δεδομένα κατανέμονται συμμετρικά γύρω από το μηδέν.



Στο παραπάνω heatmap αποτυπώνονται οι συσχετίσεις που έχουν οι αισθητήρες μεταξύ τους. Παρατηρείται ότι υπάρχει μια μικρή συσχέτιση ανάμεσα στα thigh_x και back_x και στα thigh_z και thigh_x.

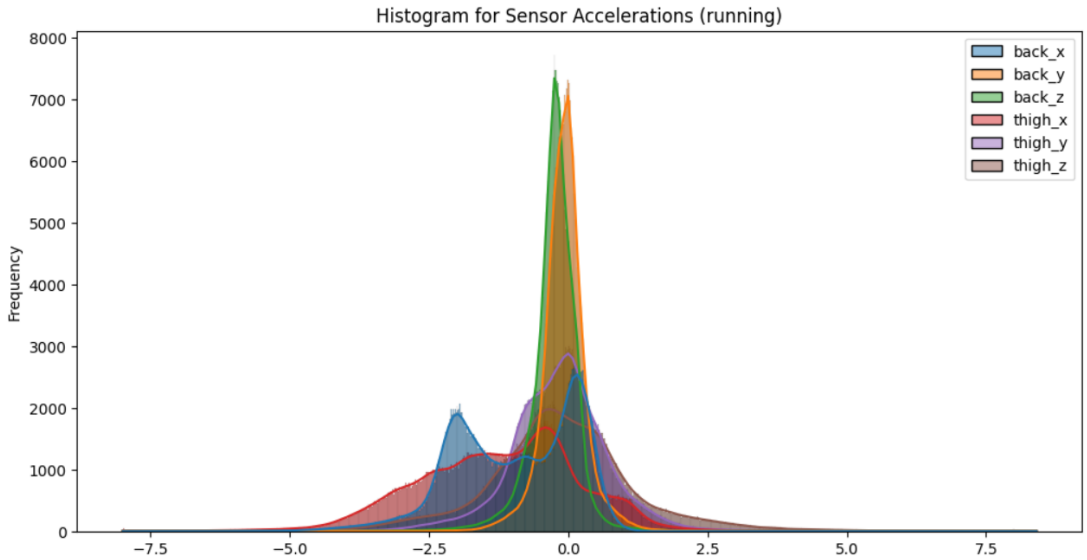
Ανάλυση ανά συμμετέχων

Ο τρίτος τρόπος που προσεγγίζεται το σύνολο δεδομένων είναι ανά δραστηριότητα. Σε αυτό το σημείο δημιουργούνται 12 νέα dataframes, ένα για κάθε δραστηριότητα. Για αυτά τα νέα dataframes, υπολογίζονται τα στατιστικά τους, ένα ιστόγραμμα όπου δείχνει τις κατανομές των τιμών που λαμβάνουν οι αισθητήρες και ένα correlation matrix. Ενδεικτικά για τη δραστηριότητα running:

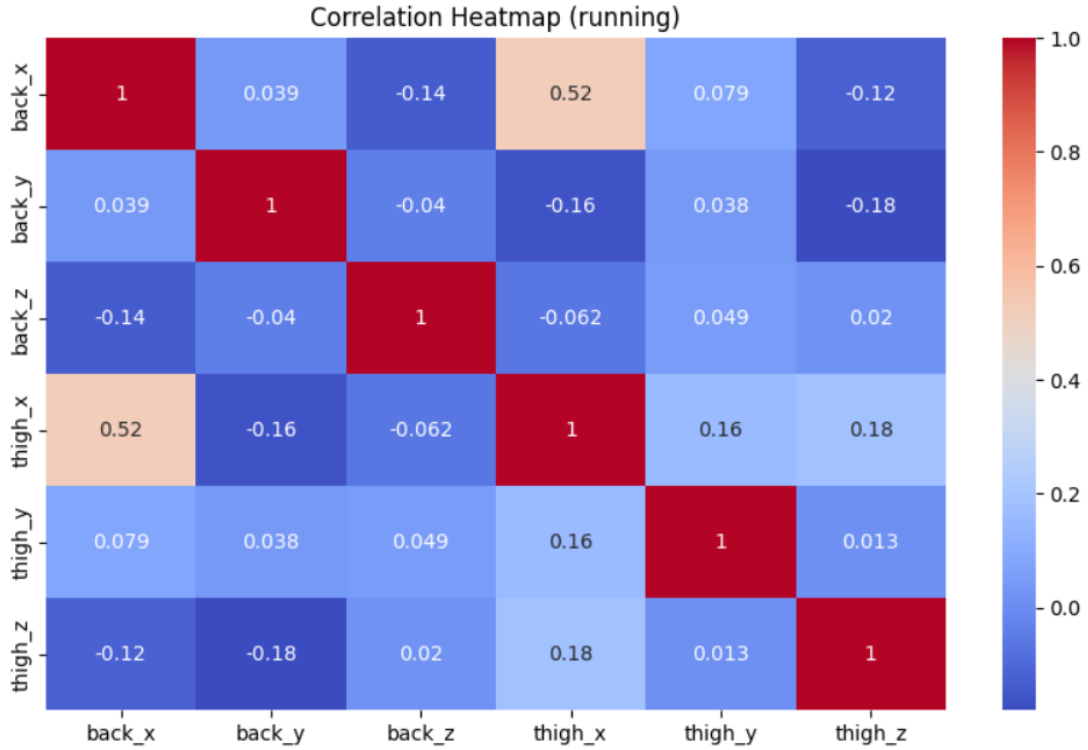
	back_x	back_y	back_z	thigh_x
count	291356.000000	291356.000000	291356.000000	291356.000000
mean	-0.965280	-0.076626	-0.259829	-1.246811
std	1.113858	0.407701	0.451772	1.438550
min	-8.000000	-4.307617	-6.574463	-8.000000
25%	-1.871094	-0.278076	-0.430176	-2.219727
50%	-0.858643	-0.070312	-0.225586	-1.141688
75%	0.040771	0.121338	-0.022217	-0.265869
max	1.698069	6.491943	3.306308	7.999756

	thigh_y	thigh_z	label
count	291356.000000	291356.000000	291356.0
mean	-0.164790	-0.140530	2.0
std	0.898353	1.381836	0.0
min	-7.929199	-8.000000	2.0
25%	-0.680930	-0.871582	2.0
50%	-0.142822	-0.169678	2.0
75%	0.338391	0.560547	2.0
max	7.999756	8.406235	2.0

Παρατηρείται ότι το εύρος τιμών που λαμβάνουν οι αισθητήρες είναι μεγάλο, αφού παραπάνω περιλαμβάνονται και οι min και max τιμές του ενιαίου dataset, που είναι -8 και 8.4 αντίστοιχα. Αυτό είναι λογικό, αφού πρόκειται για την δραστηριότητα του τρεξίματος, η οποία αποτελεί έντονη δραστηριότητα. Επίσης παρατηρείται ότι η τυπική απόκλιση σε κάποιες περιπτώσεις είναι μεγαλύτερη από 1, πιο συγκεκριμένα στις back_x, thigh_x και thigh_z. Αυτό μπορεί να σημαίνει ότι οι τιμές των δεδομένων για τις συγκεκριμένες στήλες θα είναι πιο διασκορπισμένες στον άξονα. Το ιστόγραμμα επιβεβαιώνει τα παραπάνω.



Στο παρακάτω heatmap, φαίνεται να υπάρχει μια συσχέτιση ανάμεσα στις thigh_x και back_x.

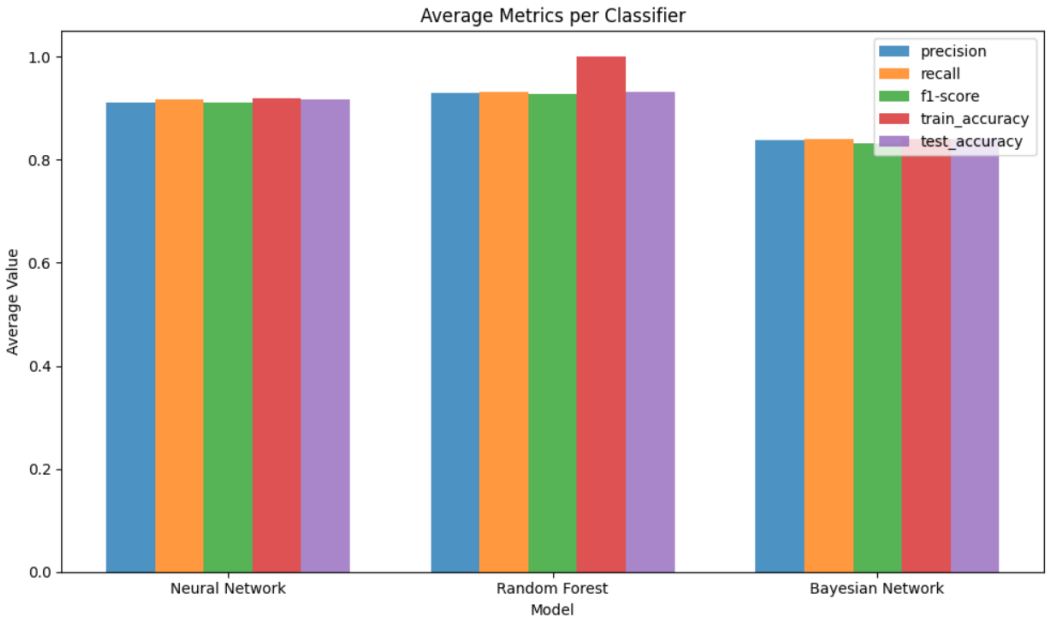


Ταξινομητές

Σε αυτό το κομμάτι εκπαιδεύτηκαν 3 classifiers, πιο συγκεκριμένα ένας βασισμένος σε Neural Networks, ένας σε Random Forest και ένας σε Bayesian Networks. Αρχικά κάθε dataset χωρίζεται σε X_train, X_test, y_train και y_test με test_size=0.2, αφού πρώτα αφαιρεθεί η στήλη timestamp. Έπειτα γίνεται κανονικοποίηση των δεδομένων με τη χρήση Standard Scaler.

Η εκπαίδευση γίνεται σε κάθε dataset ξεχωριστά και οι classifiers δεν χρησιμοποιούν γνώση από προηγούμενα datasets. Με κάθε επανάληψη δηλαδή, ο classifier επανεκπαιδεύεται από την αρχή. Όλες οι μετρικές αποθηκεύονται και στο τέλος της εκπαίδευσης βγαίνει ένας μέσος όρος για κάθε μετρική του εκάστοτε classifier. Στον παρακάτω πίνακα παρατίθενται τα αποτελέσματα:

Classifier	Precision	Recall	F1-score	Train Accuracy	Test Accuracy
Neural Network	0.9111	0.9179	0.9108	0.9195	0.9179
Random Forest	0.9290	0.9321	0.9274	1.0000	0.9321
Bayesian Network	0.8381	0.8405	0.8319	0.8407	0.8405



Αρχικά ο Random Forest φαίνεται να έχει τη καλύτερη απόδοση, αφού όλες οι μετρικές είναι υψηλότερες σε σχέση με των υπόλοιπων μοντέλων. Παρόλα αυτά παρουσιάζει train accuracy 100%, δηλαδή αποδίδει τέλεια στα train δεδομένα, το οποίο αποτελεί ένδειξη overfitting. Η διαφορά μεταξύ των train accuracy και test accuracy, είναι ικανοποιητικά μεγάλη και κατά πάσα πιθανότητα το μοντέλο δεν γενικεύει αρκετά καλά.

Σε αντίθεση, ο classifier που βασίζεται σε Bayesian Networks έχει αρκετά χαμηλότερη απόδοση σε σχέση με αυτή του Random Forest. Αυτό είναι αναμενόμενο, αφού λειτουργεί καλύτερα σε μικρότερα datasets.

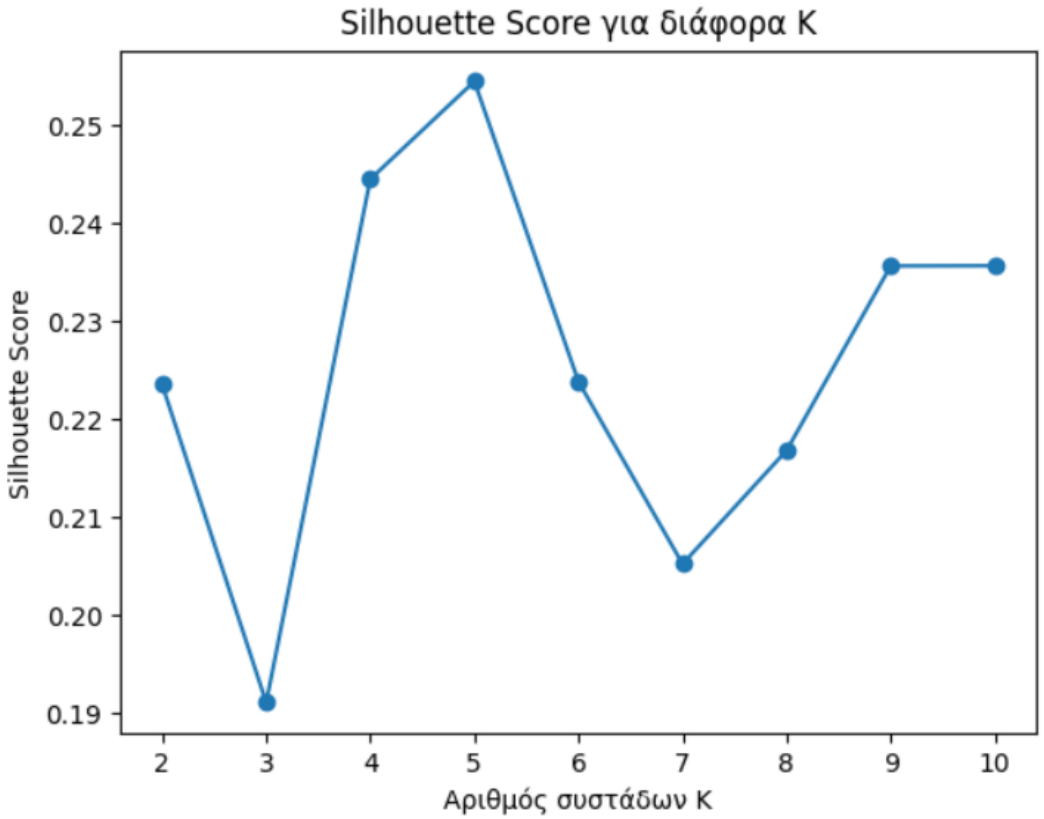
Τέλος, τα Neural Networks, δείχνουν να είναι η καλύτερη επιλογή για το συγκεκριμένο πρόβλημα, αφού έχουν την δεύτερη υψηλότερη απόδοση.

Συσταδοποίηση

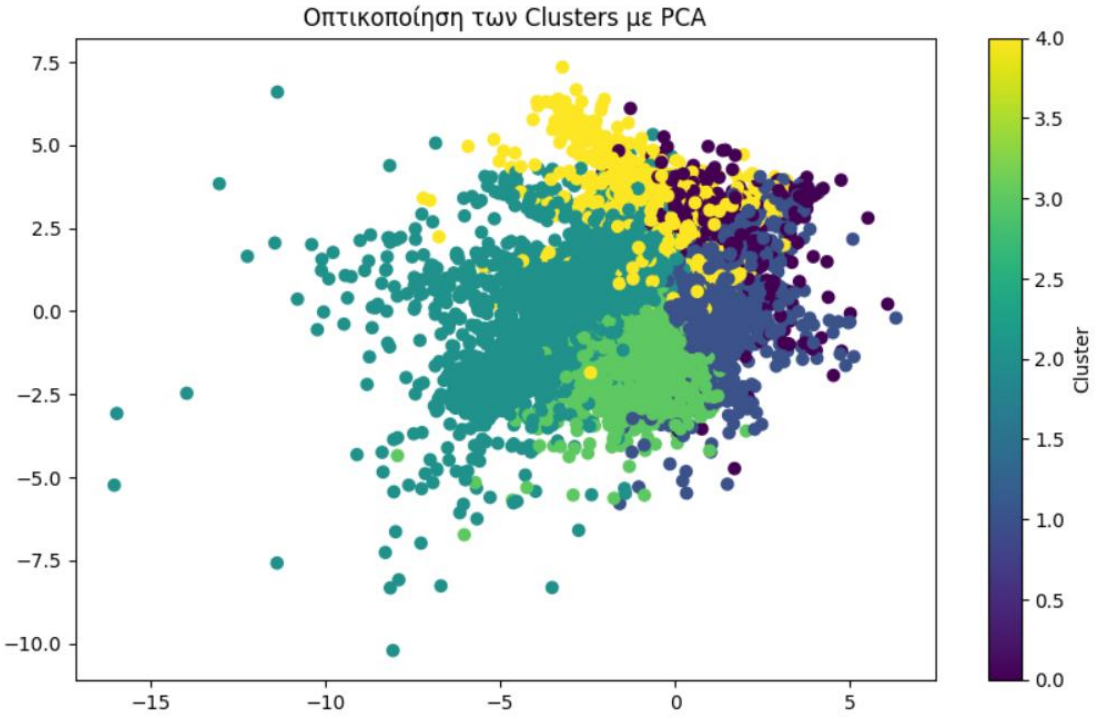
K-Means

Αυτό το ερώτημα εκτελέστηκε εξ ολοκλήρου στο περιβάλλον του Google Colab, λόγω υψηλής υπολογιστικής ισχύς που απαιτούσε. Ως πρώτος αλγόριθμος συσταδοποίησης, επιλέχθηκε ο K-Means. Τα δεδομένα φορτώθηκαν στο περιβάλλον του Google Colab, όπου έγινε sampling και επιλέχθηκε το 1% των δεδομένων, λόγω του ότι το dataset είναι πολύ μεγάλο και οι παραπάνω τεχνικές φάνηκαν να μην μπορούν να ανταπεξέλθουν σε τόσο μεγάλο όγκο δεδομένων. Αφού έγινε το sampling μετά με τη χρήση Standard Scaler έγινε και η κανονικοποίηση των δεδομένων. Έπειτα μέσω της Silhouette Score και για τιμές από 2 έως 11 για το K, έγινε προσπάθεια για την βέλτιστη επιλογή του αριθμού των συστάδων. Οι 2 καλύτερες επιλογές φαίνονται παρακάτω:

```
Silhouette Score for 4 clusters: 0.24449648141687394
Silhouette Score for 5 clusters: 0.25447469837144143
```



Ως βέλτιστη επιλογή για το K, φαίνεται να είναι το 5. Για την αναπαράσταση των δεομένων θα χρειαστεί να γίνει μείωση των διαστάσεων. Αυτό γίνεται με τη χρήση του PCA. Παρακάτω φαίνονται τα αποτελέσματα των 5 clusters:



Cluster label	0	1	2	3	4
1	29	768	6070	4911	129
2	286	334	1452	148	644
3	2	38	604	1831	6
4	0	39	312	410	14
5	3	40	283	273	15
6	1	76	2042	5324	3
7	12	28494	127	201	424
8	1323	851	0	2	2091
13	1	211	1639	2049	62
14	3	13	316	184	25
130	0	7	183	222	1
140	2	0	71	12	0

Στον πρώτο cluster, παρόλο που τοποθετούνται κάποια δείγματα, δεν παράγεται κάποια καθαρή εικόνα έτσι ώστε να εξαχθεί κάποιο συμπέρασμα. Φαίνεται όμως η δραστηριότητα 8 (lying), να είναι μοιρασμένη στον πρώτο και τον πέμπτο cluster. Σε αντίθεση με τον δεύτερο, όπου είναι φανερό ότι περιλαμβάνει σχεδόν όλα τα δείγματα της δραστηριότητας 7 (sitting). Στον τρίτο cluster, τοποθετούνται κυρίως τα δείγματα της δραστηριότητας 2 (running). Οι δραστηριότητες 1 (walking), 4 stairs (ascending), 5 stairs (descending), 13 cycling (sit), 14 cycling (stand), 130 cycling (sit, inactive), 140 cycling (stand, inactive), φαίνεται να μοιράζονται μεταξύ του τρίτου και του τέταρτου cluster. Στο τέταρτο cluster εντοπίζονται κυρίως οι δραστηριότητες 3 (shuffling) και 6 (standing). Παρατηρείται λοιπόν ότι δεν μπορεί μπορεί να να εξαχθεί κάποιο συμπέρασμα με σιγουριά. Αυτό συμβαίνει κατά πάσα πιθανότητα στην δειγματοληψία που έχει γίνει.

Παρακάτω εμφανίζονται τα αποτελέσματα για K=4, όπου προέκυψε το δεύτερο υψηλότερο silhouette score. Φαίνεται να μην υπάρχει κάποια σημαντική διαφορά σε σχέση με το K=5:

Cluster_4	0	1	2	3
label				
1	39	827	5883	5158
2	352	454	1851	207
3	2	51	555	1873
4	1	41	299	434
5	3	43	273	295
6	0	115	1793	5538
7	13	28415	138	692
8	1324	1754	0	1189
13	1	243	1639	2079
14	4	15	315	207
130	0	8	183	222
140	2	0	71	12

Agglomerative Clustering

Ως δεύτερη μέθοδος επιλέχθηκε το Agglomerative Clustering, αλλά σε ακόμη μικρότερο ποσοστό του dataset, μόλις 0,5%. Αρχικά επιλέχθηκε να υλοποιηθεί με 5 clusters, όπου το silhouette score και τα αποτελέσματα παρουσιάζονται παρακάτω:

Silhouette Score: 0.2360180097480331

Cluster label	0	1	2	3	4
1	3573	433	24	1962	0
2	741	183	403	121	0
3	389	21	1	797	0
4	212	17	1	151	0
5	200	10	4	110	0
6	1362	34	0	2335	0
7	68	14285	204	30	0
8	0	454	1144	2	555
13	667	279	2	997	0
14	145	37	2	95	0
130	95	6	0	113	0
140	35	3	0	5	0

Με μια πρώτη ματιά φαίνεται πως στον πέμπτο cluster, υπάρχουν μόνο δείγματα της δραστηριότητας 8 (lying). Παρόλα αυτά τα περισσότερα δείγματα αυτής της δραστηριότητας κατατάσσονται στον τρίτο cluster. Οι δραστηριότητες 1 (walking), 2 (running), 4 stairs (ascending), 5 stairs (descending), 14 cycling (stand), 140 cycling (stand, inactive), φαίνεται να συγκεντρώνονται στον πρώτο cluster, χωρίς να μπορούν να εξαχθούν συμπεράσματα αφού πολλά δείγματα των παραπάνω δραστηριοτήτων κατατάσσονται και σε άλλα clusters. Το ίδιο ισχύει και για τις δραστηριότητες 3 (shuffling), 6 (standing), 13 cycling (sit), 130 cycling (sit, inactive), αφού τα περισσότερα δείγματα συγκεντρώνονται στον τέταρτο cluster, παρόλο που πολλά από αυτά περιέχονται και στον πρώτο cluster.

Με τη προσθήκη ενός ακόμη cluster, δεν φαίνεται να λύνεται το παραπάνω πρόβλημα, παρόλο που παρουσιάζεται να έχει καλύτερο silhouette score:

Silhouette Score: 0.2466924502298852

Cluster label	0	1	2	3	4	5
1	433	3282	24	1962	0	291
2	183	299	403	121	0	442
3	21	385	1	797	0	4
4	17	205	1	151	0	7
5	10	192	4	110	0	8
6	34	1360	0	2335	0	2
7	14285	68	204	30	0	0
8	454	0	1144	2	555	0
13	279	661	2	997	0	6
14	37	128	2	95	0	17
130	6	95	0	113	0	0
140	3	35	0	5	0	0