In this section there will be an exploration of practice the basics of Machine Learning Classification by using k-means clustering and PCA on Human Activity Recognition Set and Code.

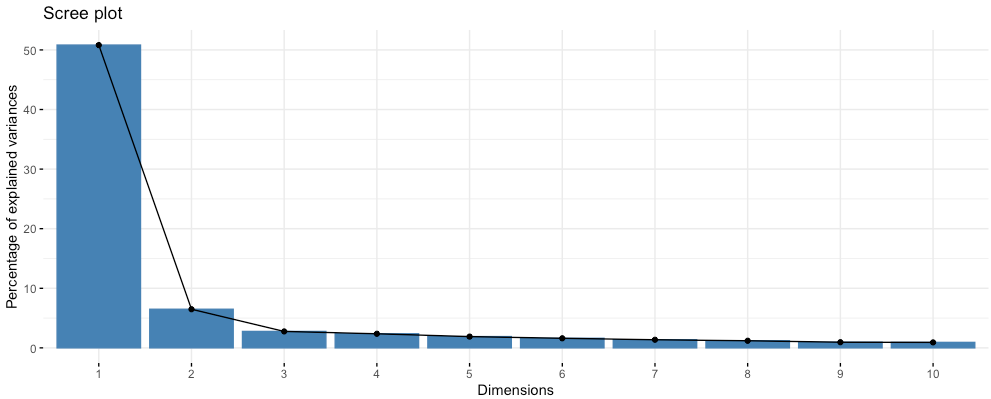
Human Activity Recognition database built from the recordings of 30 subjects performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors. The experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (*WALKING, WALKING\_UPSTAIRS, WALKING\_DOWNSTAIRS, SITTING, STANDING, LAYING*) wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually [1].

The data was standardize using the StandardScaler from sklearn.preprocessing library.

Eg. StandardScaler().fit\_transform(X)

Principle Component Analysis, PCA, was in R (code on GitHub) to find the ratio of each PCA contribution as well each feature contribution to highest contribution PCA.

Below shows the PC breakdown across 10 PCs, it can be seen PC 1 is ~ 50% of total contribution.



Below is a snipped showing the contribution of few of the elements to the first 3 PCs.

It can be seen that tBodyAcc.std.X , tBodyAcc.std.Y and tBodyAcc.std.Z have the highest show contribution to PC1.

A screenshot of a cell phone

Description automatically generated

Below is the list of the top features that attributed to PC1, they were chosen

tBodyAccJerkMag.iqr tBodyAccJerkMag.max

0.3296044 0.3300089

tBodyAcc.sma fBodyAccJerk.mean.X

0.3306103 0.3308718

tBodyAccJerk.mad.X tBodyAccJerk.std.X

0.3313329 0.3314136

fBodyBodyAccJerkMag.mad tBodyGyroJerkMag.mean

0.3317676 0.3326879

tBodyGyroJerkMag.sma fBodyAccMag.mean

0.3326879 0.3326945

fBodyAccMag.sma tBodyGyroJerk.sma

0.3326945 0.3346974

tBodyAccJerkMag.std tBodyAccJerkMag.mad

0.3355263 0.3355557

fBodyBodyAccJerkMag.mean fBodyBodyAccJerkMag.sma

0.3373821 0.3373821

tBodyAccJerkMag.mean tBodyAccJerkMag.sma

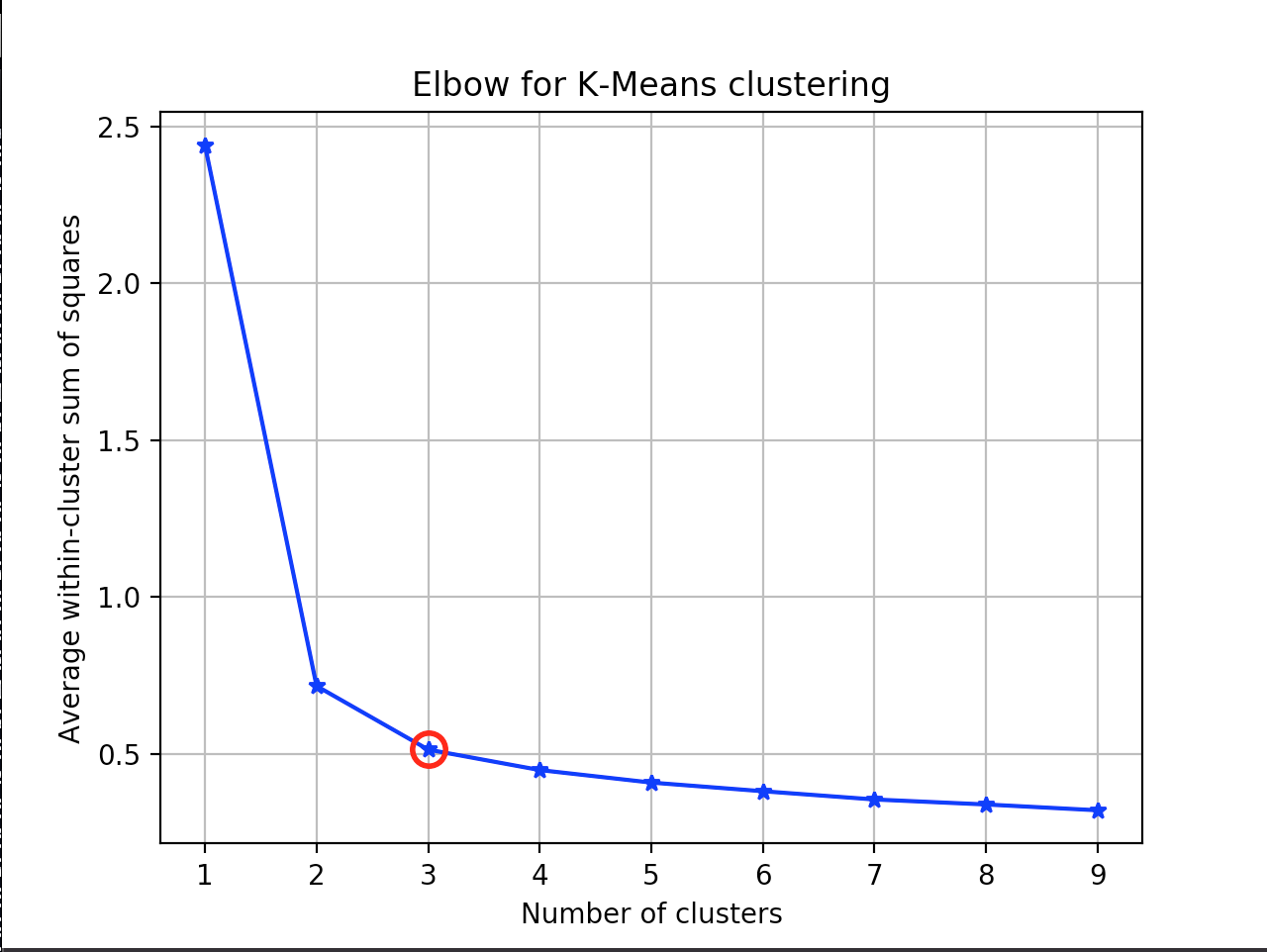
0.3414347 0.3414347

tBodyAccJerk.sma fBodyGyro.sma

0.3420842 0.3425234

fBodyAccJerk.sma fBodyAcc.sma

0.3427119 0.3427775



So we see that a good K to use in our model would be 3. After which that is used to find the clusters of data.

**k**-**means** algorithm is known to have a **time complexity** of O(n 2 )

between kNN and RF was large when the training sample size increased from sub-dataset-1 to sub-dataset-4 for both imbalanced and balanced cases; however, the difference between various training sample sizes of the SVM classifier was insignificant. For all three classifiers, when the training sample was large enough (greater than 750 pixels/class), with both imbalanced and balanced datasets (iset\_5/bset\_5, iset\_6/bset\_6, and iset\_7/bset\_7), the OA was approximately similar and high (over 93.85%). Furthermore, it is recommended that in land cover classification using remote sensing images and machine learning algorithms, the training sample size should represent approximately 0.25% of the total study area.

a comparison of Random forests and kNN for the MNIST dataset

A close up of a piece of paper

Description automatically generated

A close up of text on a white background

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A close up of a piece of paper

Description automatically generated

A close up of a piece of paper

Description automatically generated

A close up of text on a white background

Description automatically generated

A close up of text on a white background

Description automatically generatedA close up of a piece of paper

Description automatically generated

A close up of text on a white background

Description automatically generated

1 <https://www.kaggle.com/ruslankl/k-means-clustering-pca>