Smartphone-Based Recognition of Human Activities and Postural Transitions Data Set

M.Sc Data Science and Engineering

Matemathics in Machine learning

Politecnico di Torino

Nicola Scarano

May 04, 2022

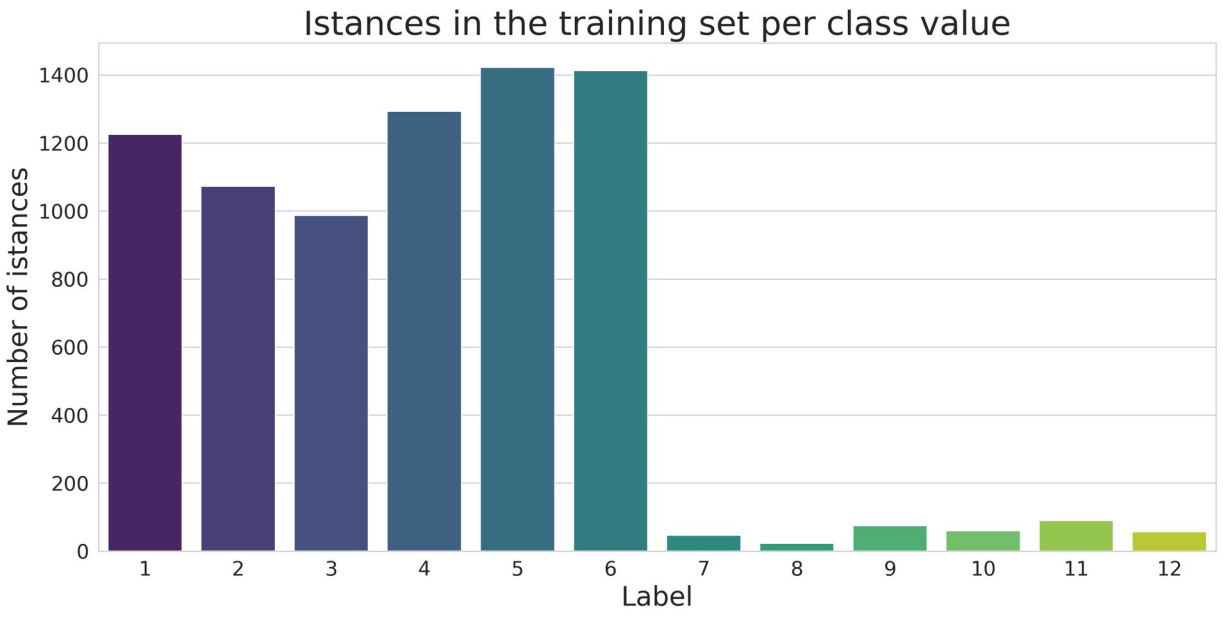
Dataset overview

- Dataset has features extracted from row accelerometer and gyroscope signals
- Numerical continuous 561 features
- Training set size : 7767
- Test set size: 3162

Classification problem: Predict the correct activity label

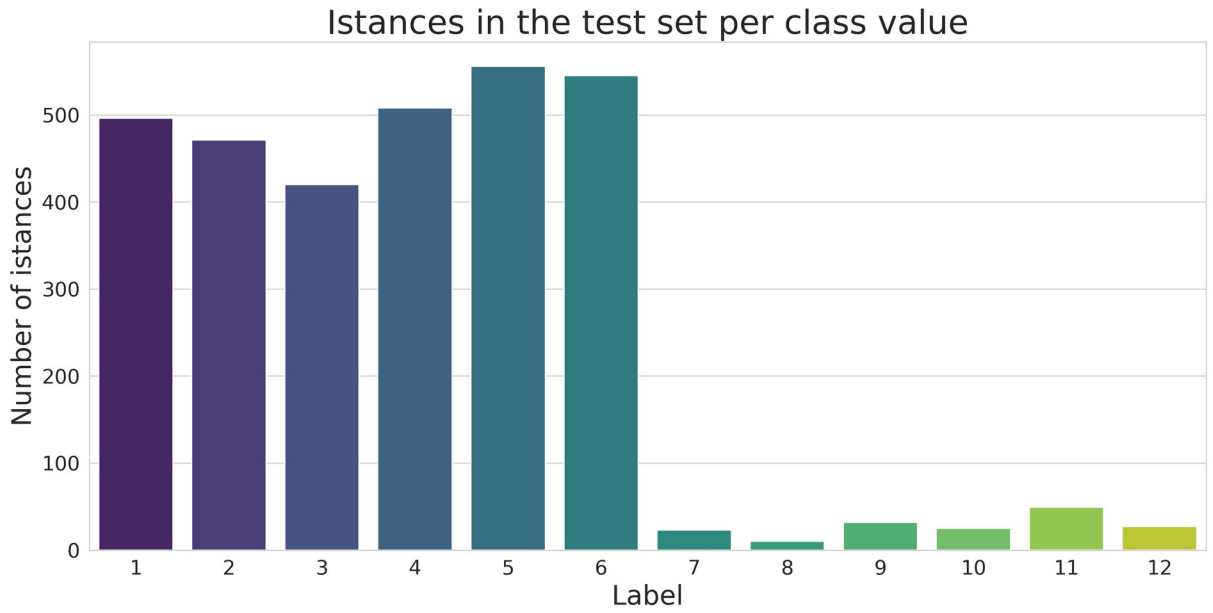
12 labels associated to activity or postural transiction

Target distribution



1: WALKING
2: WALKING UPSTAIRS
3: WALKING DOWNSTAIRS
4: SITTING
5: STANDING
6: LAYING
7: STAND TO SIT
8: SIT TO STAND
9 SIT TO LIE
10 LIE TO SIT
11 STAND TO LIE
12 LIE TO STAND

Target distribution



- Unbalanced target distribution
- Same distribution in test and training set

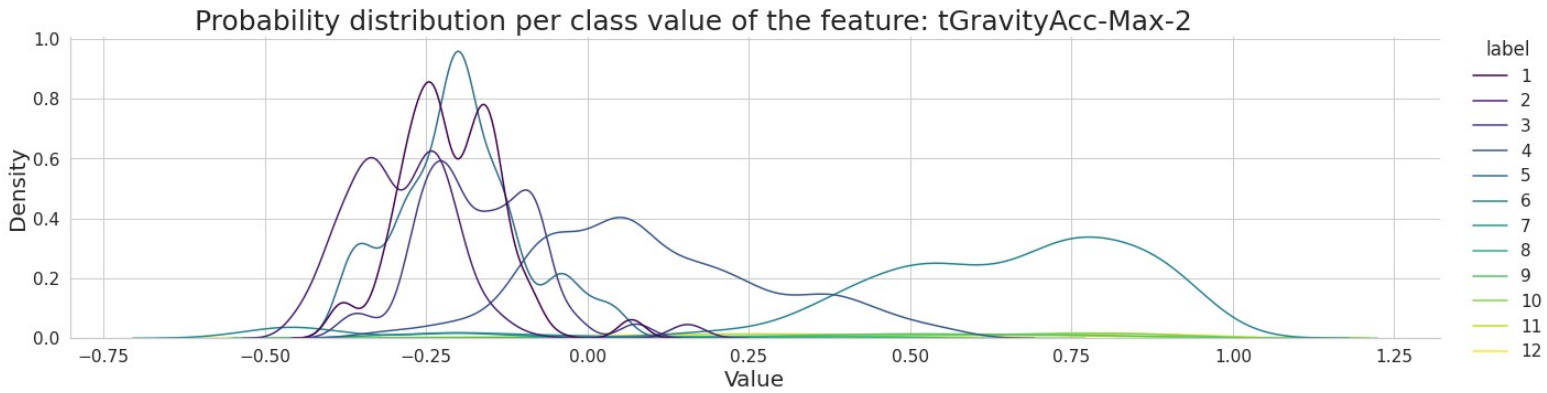
Features distribution

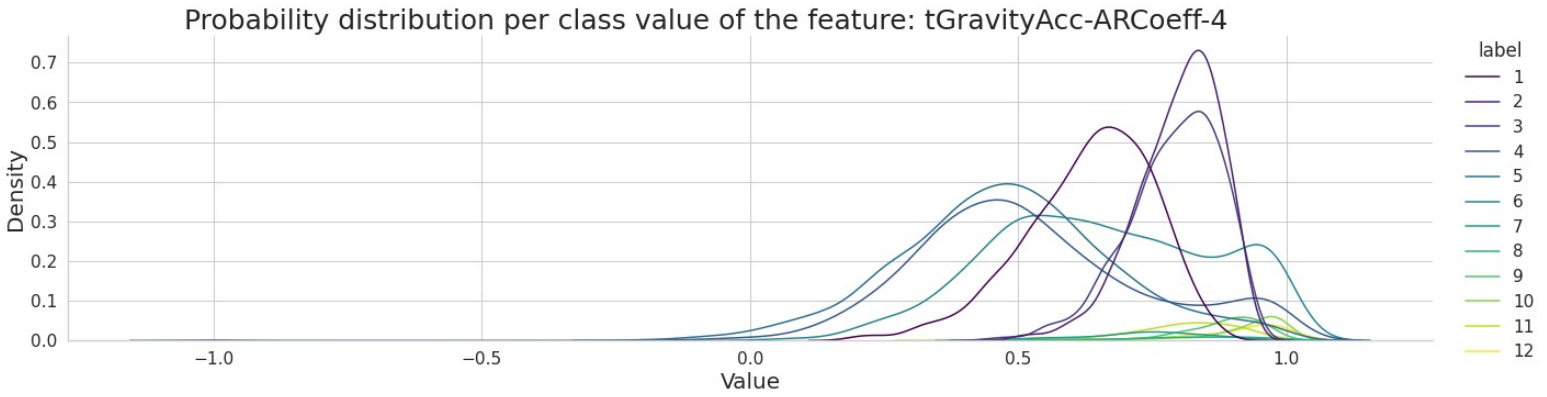
561 continuous features already normalized and bounded [-1,1]

We select the following features to be visualized:

- 50th feature: tGravityAcc-Max-2
- 68th feature: tGravityAcc-ARCoeff-4
- 120th feature: tBodyGyro-Mean-1
- 389th feature: fBodyAccJerk-BandsEnergyOld-9
- 555th feature: tBodyAccJerk-AngleWRTGravity-1

Features distribution

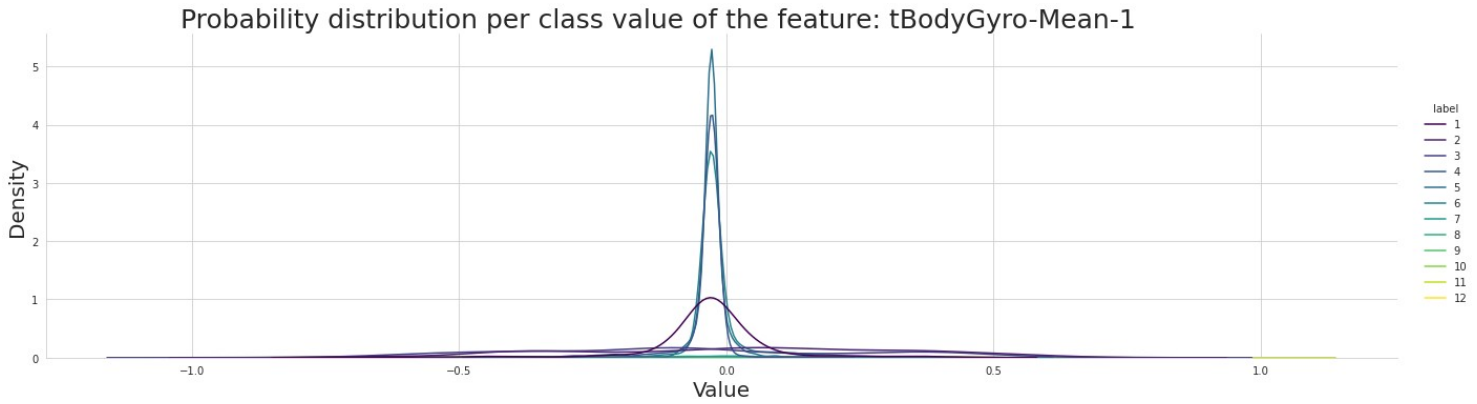


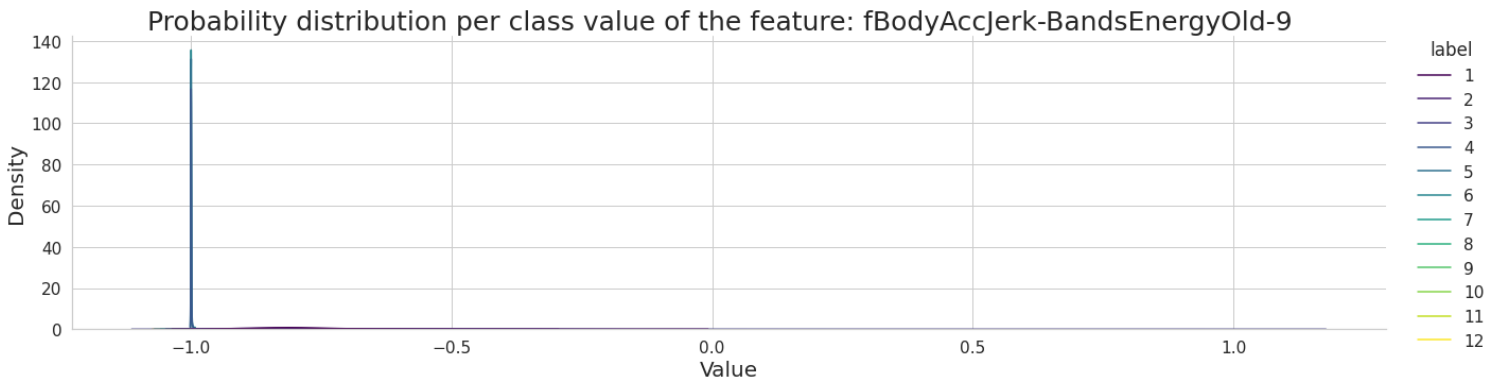


Unbalance problem

Non Gaussian distribution

Features distribution

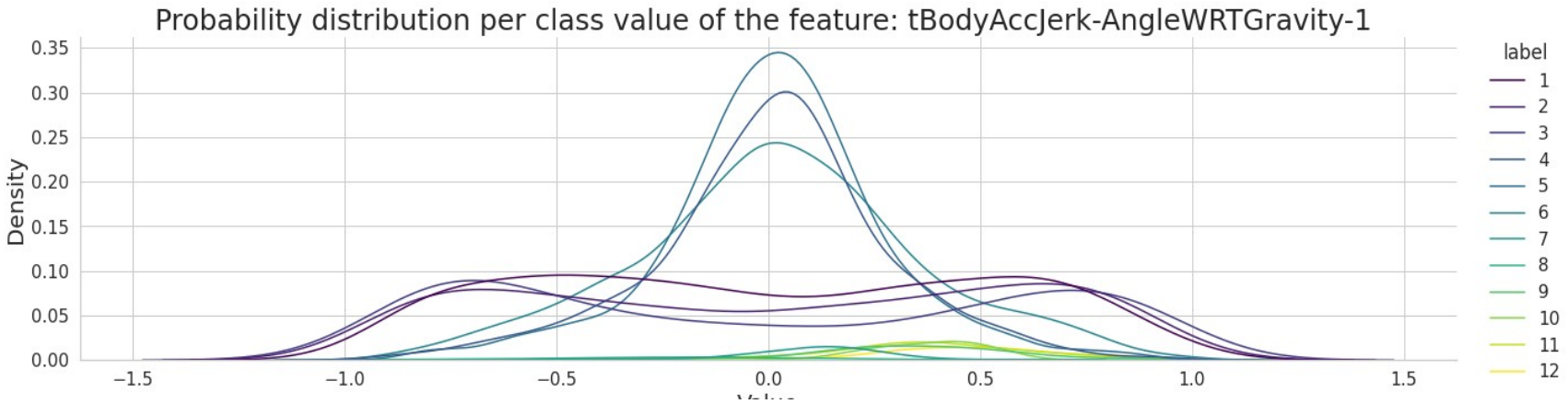




Features have a very different distribution

Frequency feature

Features distribution

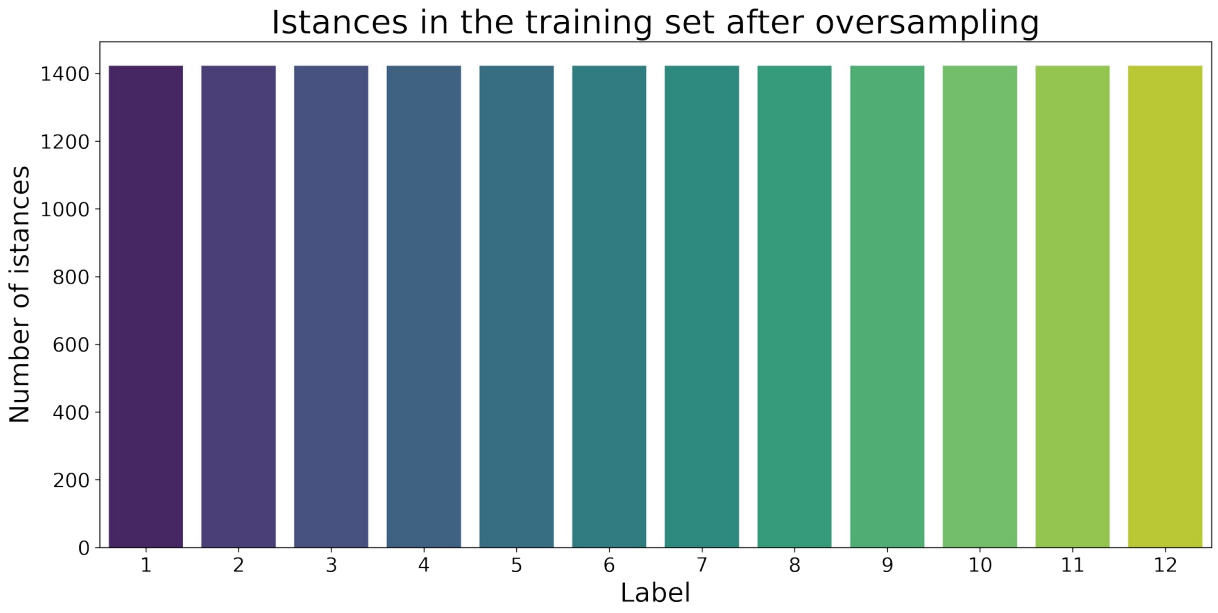


Oversampling

We tested different Oversampling techniques:

- ADASYN
- BorderlineSMOTE
- KmeansSMOTE
- SVMSMOTE

We choose the best using K-fold Cross Validation

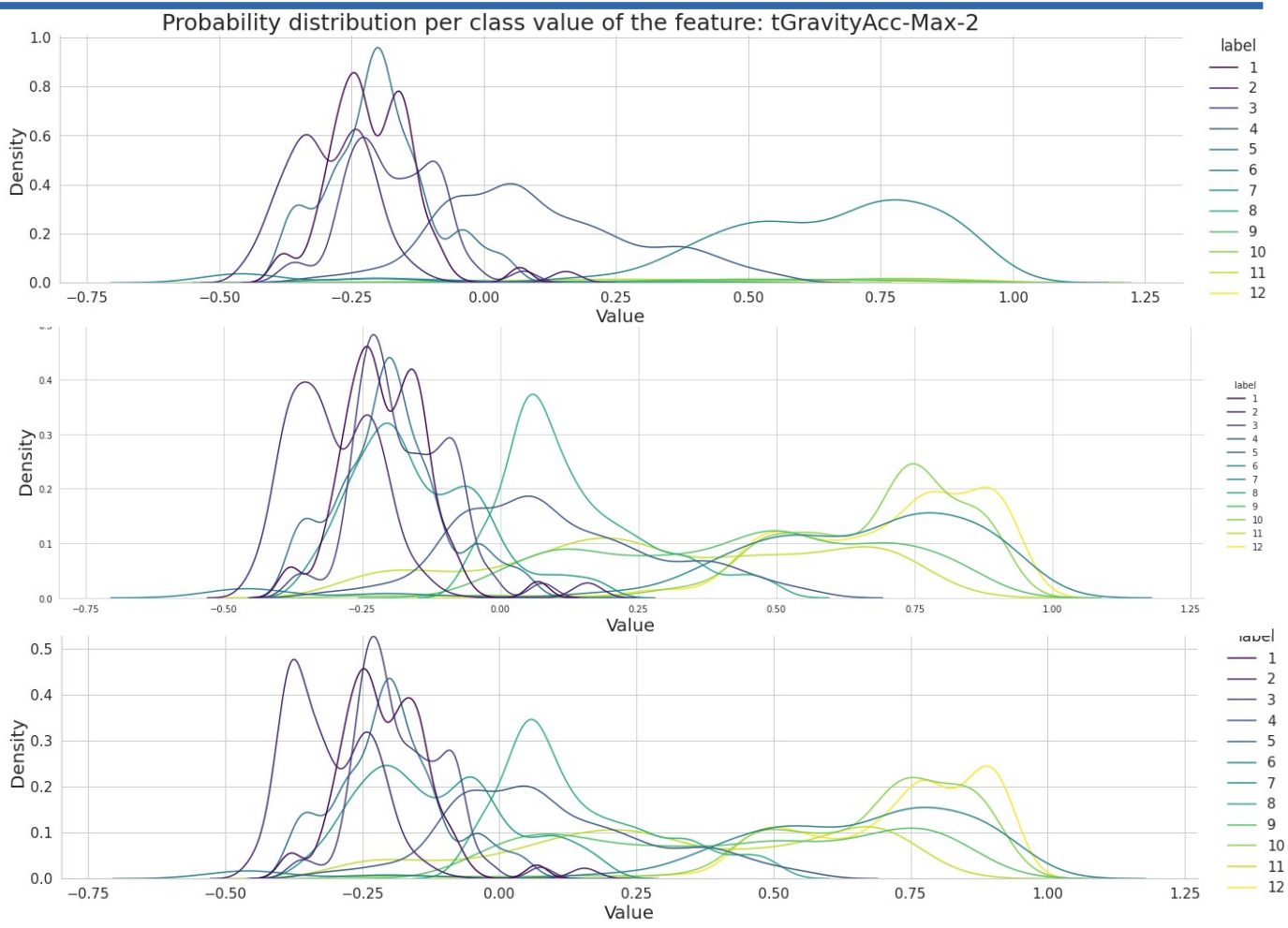


Oversampling

No oversampling

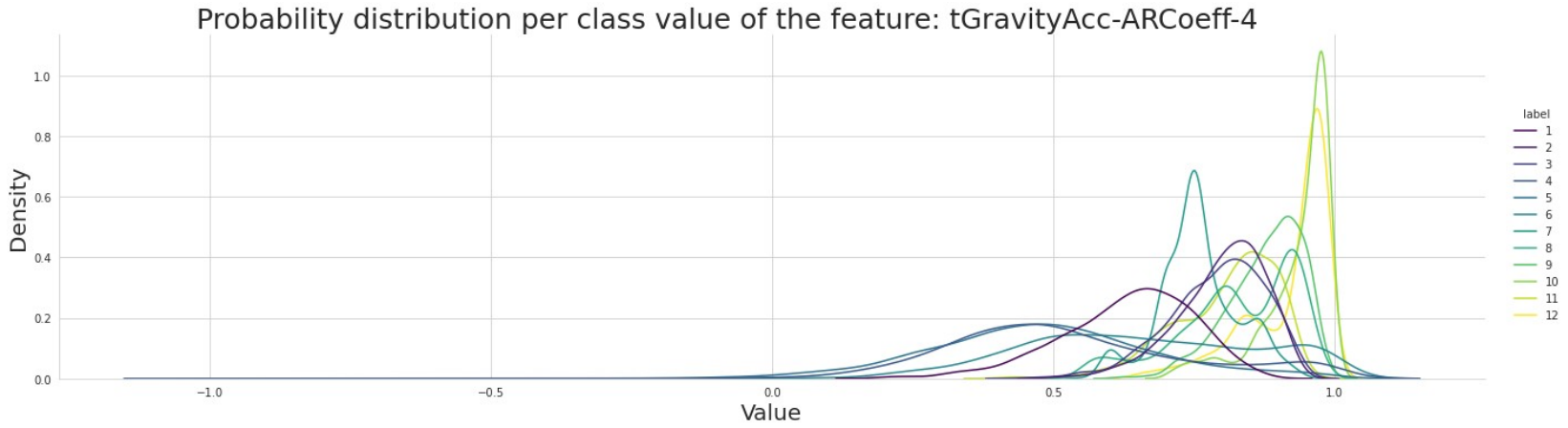
ADASYN oversampling

Borderline SMOTE

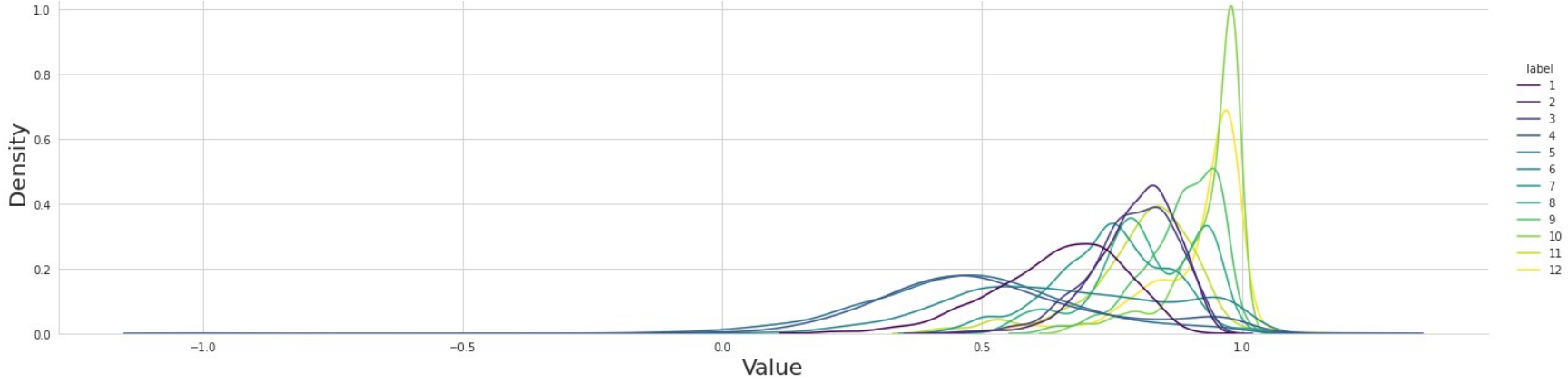


Oversampling

Kmeans SMOTE



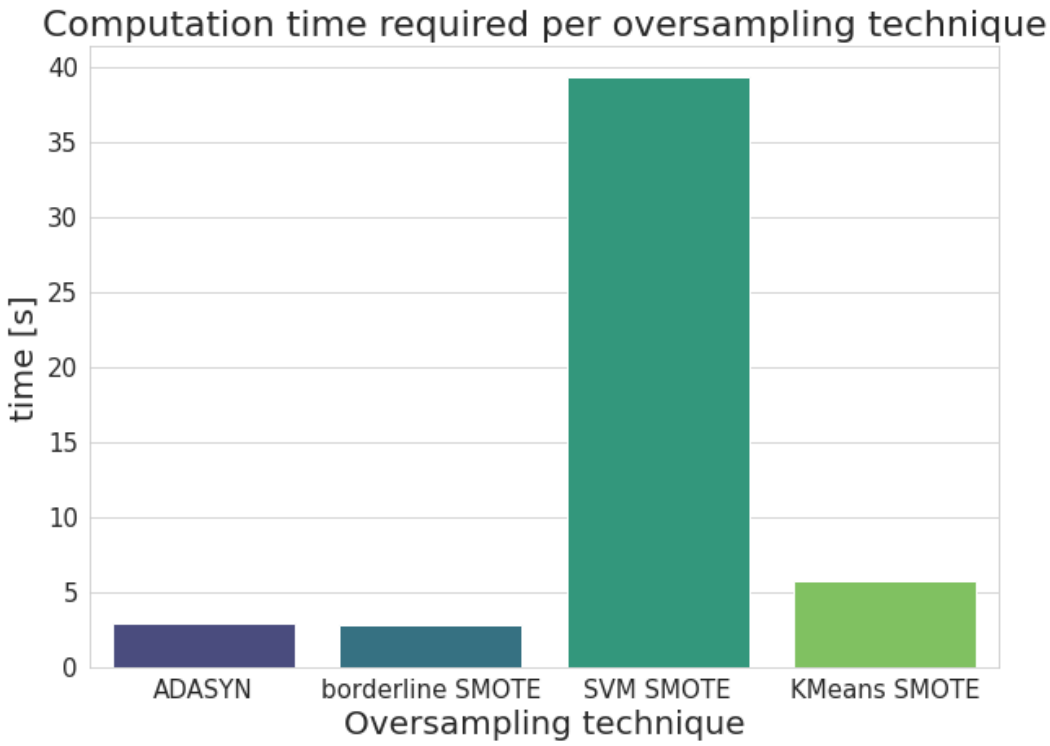
SVMSMOTE



Oversampling

Time needed to perform the oversampling on the training set

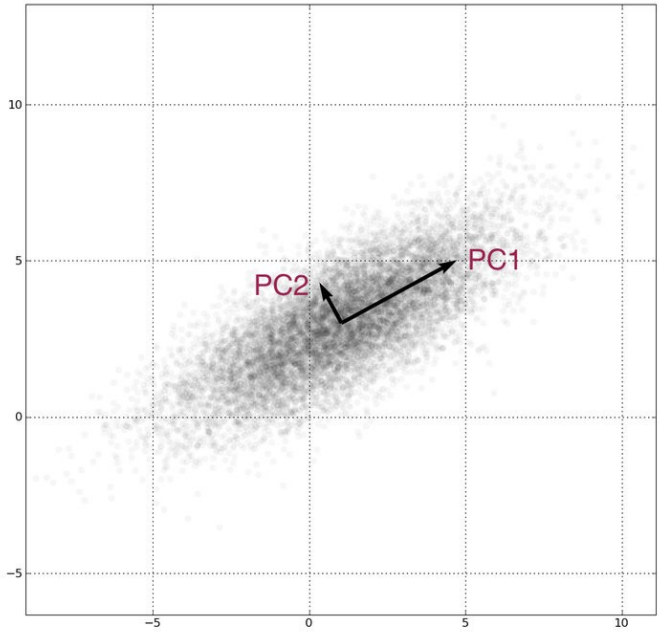
Times comparable for all the methods except for SVM SMOTE



Principal Component Analysis

Why dimensionality reduction on our dataset?

- Reduction of the time for the training
- Interpretability of the data, finding meaningful structure in data, illustration

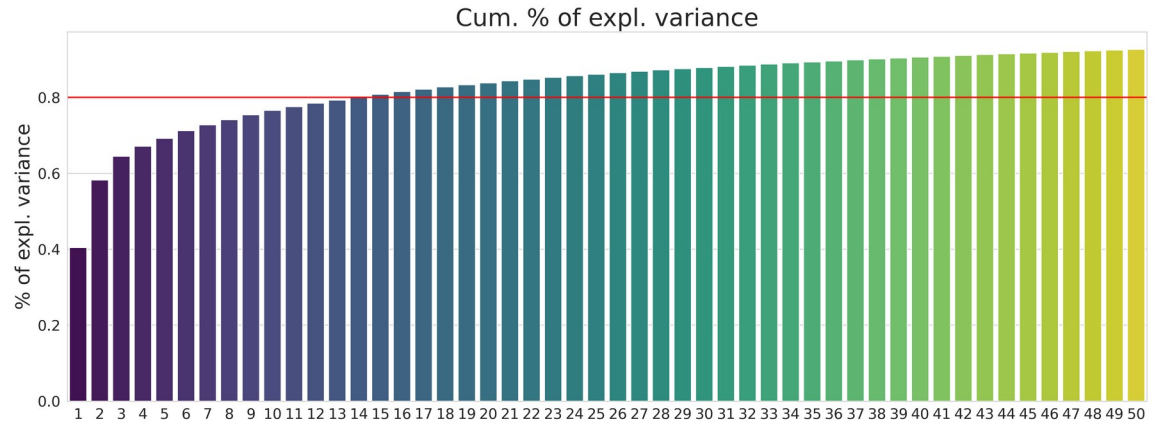


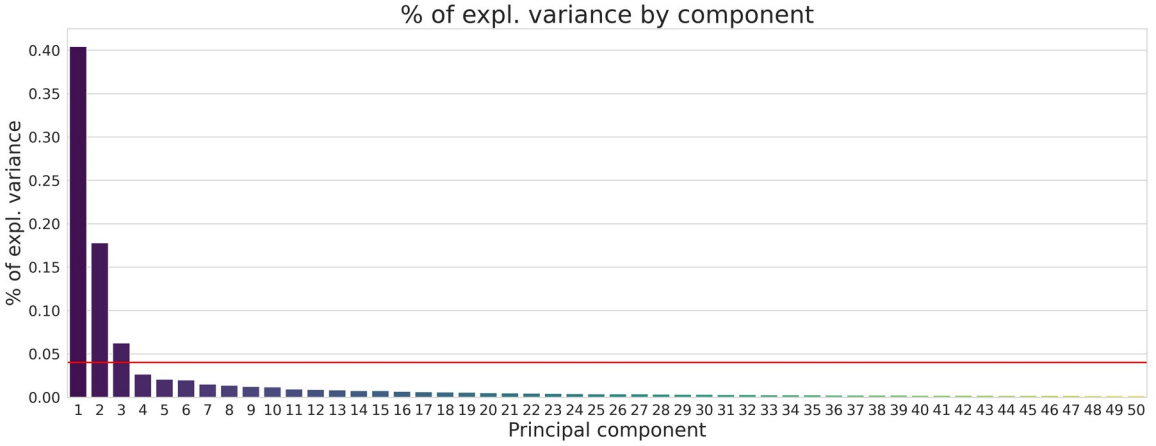
Principa Component Analysis

First three PCs have % VE much higher than others: “elbow” at fourth PC

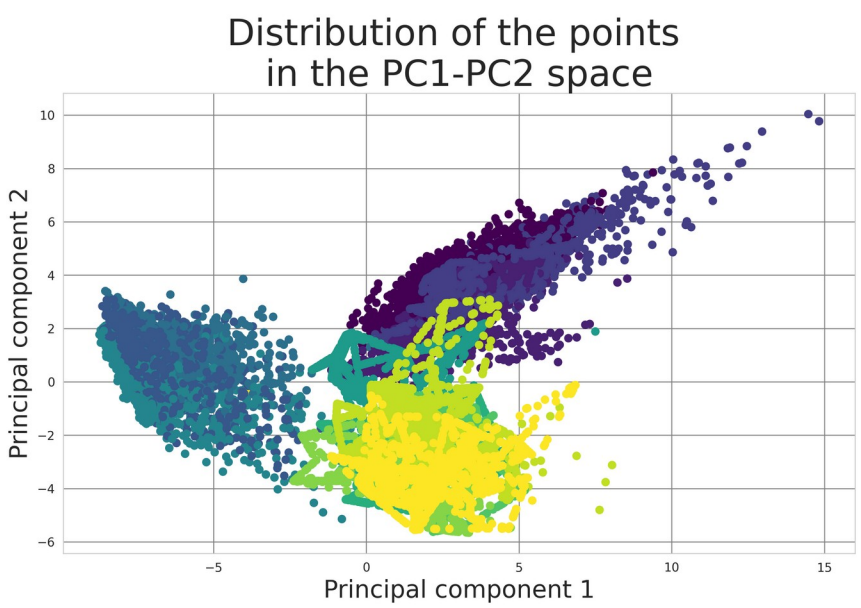
From the fourth PC the gain in adding a new PC decrease

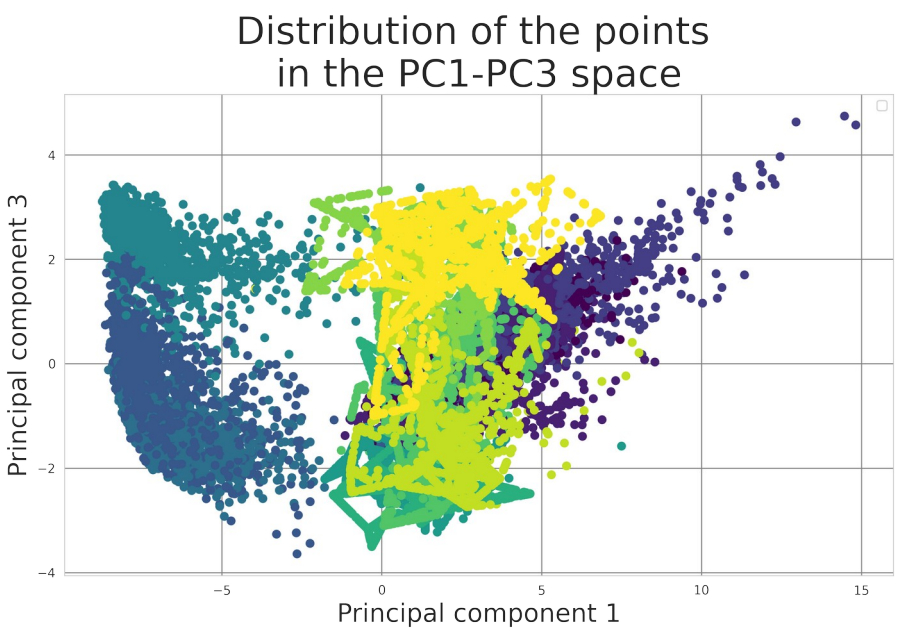
80% of the total variance is explained by the first 15 PCs





Principal Component Analysis



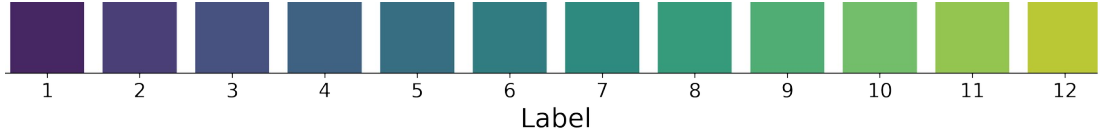


Principal Component Analysis

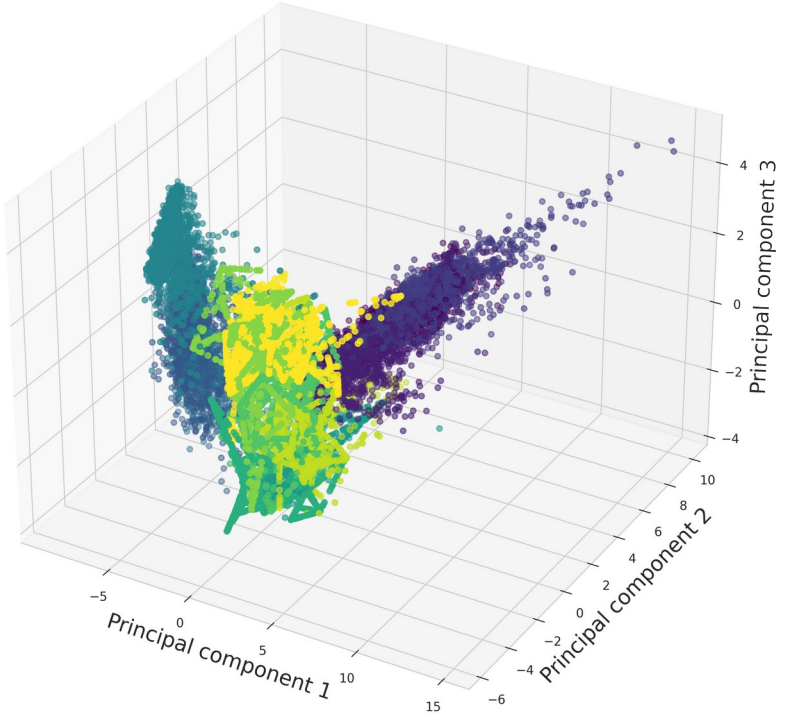
Not simple to discriminate among all the 12 classes

Looking carefully we can see three main clusters of points

1: WALKING
2: WALKING UPSTAIRS
3: WALKING DOWNSTAIRS
4: SITTING
5: STANDING
6: LAYING
7: STAND TO SIT
8: SIT TO STAND
9 SIT TO LIE
10 LIE TO SIT
11 STAND TO LIE
12 LIE TO STAND

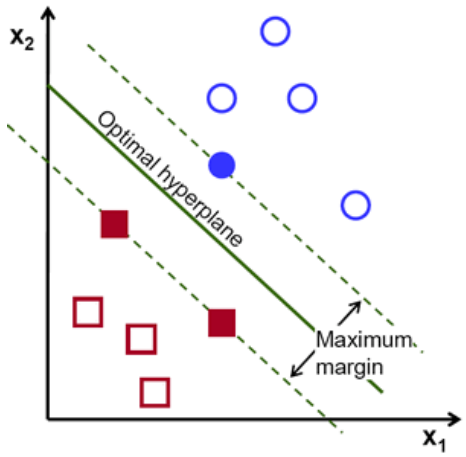


Distribution of the points
in the PC1-PC2-PC3 space



Model Selection

SVM



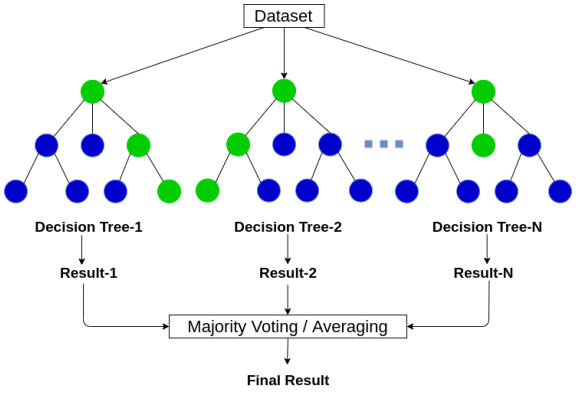
SVM advantages:

- Interpretability of the results
- We do not need a lot of data

Random forest advantages:

- Generalization, reduce overfitting
- Robust to noise and outliers

Random Forest



Results

Training strategy:

- PCA and oversampling techniques tuned like an hyperparameter
- SVM and Random forest use default sklearn hyperparameter vaues
- Cross Validation with K= 5 with metrix = f1-micro

Hyperparameter

SMV:
- OvR strategy
- ‘rbf’ kernel
- regularization parameter C = 1

Random Forest:
- number of trees = 100
- criterion = “Gini“

Results

	SVM				Random Forest			
	ADASYN	bSMOTE	KMSMOTE	SVMSMOTE	ADASYN	bSMOTE	KMSMOTE	SVMSMOTE
3 Principal Component	0.65	0.65	0.75	0.65	0.74	0.82	0.87	0.80
15 Principal Component	0.87	0.87	0.87	0.92	0.91	0.93	0.94	0.88
50 Principal Component	0.96	0.96	0.96	0.97	0.91	0.95	0.95	0.96
Full dataset	0.97	0.98	0.97	0.98	0.96	0.97	0.97	0.98

- Some of the **information is lost** using 3 and 15 PCs
- **50 PCs** performance ≈ full dataset
- Full dataset CV **time** with SVM > 1 min while RF > 2 min
- **Oversampling** techniques behave differently especially when the dataset have few features
- **SMOTE variants** perform better than ADASYN
- SVM perform better than RF with more features

Test performance (f1-micro):

- 50 PCs, SVMSMOTE, SVM classifier: 0.92
- 3PCs, KmeansSMOTE, RF classifier: 0.78

Thank you for the attention