Tesina on Drug Consumption dataset

Mathematics in Machine Learning

Candidate: Luca Benfenati (s286582)
Professors: Gasparini Mauro, Vaccarino Francesco

November 8th, 2022

Politecnico di Torino

Data Science and Engineering
Dipartimento di Automatica e Informatica
Politecnico di Torino

Topics

1. Introduction
2. Problem Definition
3. Exploratory Data Analysis (EDA)
4. Data Preparation
5. Methods
6. Results
7. Conclusions

Introduction

Drug Consumption Dataset

The problem of evaluating an individual's risk of drug consumption and misuse is highly important. An online survey methodology was employed to collect the following data:

Feature	Description	Class of Feature
Age	Age of the subject	Demographic information
Gender	Gender of the subject	
Education	Level of Education of the subject	
Country	Country of Residence of the subject	
Ethnicity	Ethnicity of the subject	
Nscore	Neuroticism score	Personality measurements (including Five Personality Traits → NEO-FFI-R)
Escore	Extraversion score	
Oscore	Openness to experience	
Ascore	Agreeableness	
Cscore	Conscientiousness	
Impulsive	Inclination to Impulsivity (measure by the Barratt Impulsiveness Scale)	Inclination information
SS	Inclination to Sensation Seeking (measured by ImpSS Scale)	

Problem Defintion | Classification Problem

Problem

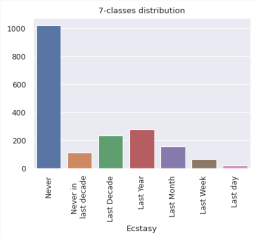
Database contains 18 classification problems, one for each drug. Possible problems to solve:

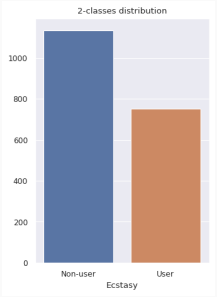
1. 7-classes classification: from Never Used (**CL0**) to Used in Last Day (**CL6**).
2. binary classification: **User** vs **Non-User**;

Binary Classification Problem

We distinguish between drug user and non-user. Thus:

1. Non-User class includes CL0 and CL1.
2. User class includes from CL2 to CL6;



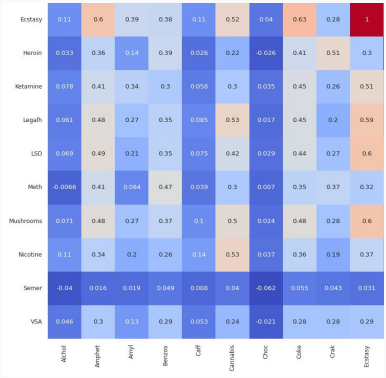


Problem Definition | Drug selection

We select one drug to classify. We introduce the Spearman's correlation coefficients:

$$\rho = 1 - \frac{6 \sum_i d_i^2}{n(n-1)}$$

whre $d_i = x_i - y_i$ is the difference between the ranks of the two features of the instance i.



Rationale: usage of some drugs are significantly correlated between each other. We then select the drug with the highest mean correlation coefficients with respect to others (**Ecstasy**).

Feature Scaling

When data have different scales, feature scaling standardizes the data. Scaling may improve the convergence of gradient-based estimators and is useful when visualizing data on vastly different scales.

- Standard Scaler (Z-score normalization):

$$z = \frac{x - \mu}{\rho} \tag{1}$$

where $\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i)$ and $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$, and N number of samples

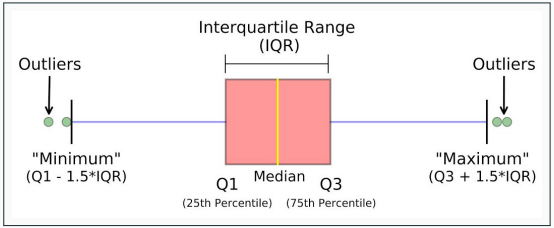
- MinMax Scaler:

$$x_{sc} = \frac{x - \min_i(x_i)}{\max_i(x_i) - \min_i(x_i)} \tag{2}$$

Outlier Detection and Management

Outliers are data that largely deviate from the other observations. There are many methods that are able to detect outliers in different ways.

Method	Description
Isolation Forest	Isolate anomalies by creating decision trees over random attributes: when randomly partitioning the domain space, the anomaly will be detected in smaller number of partitions than a normal point
Local Outlier Factor	Unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbors
DBSCAN	Density-based and unsupervised machine learning algorithm that groups "densely grouped" data points into a single cluster, identifying clusters in large spatial datasets based on local density of the data points

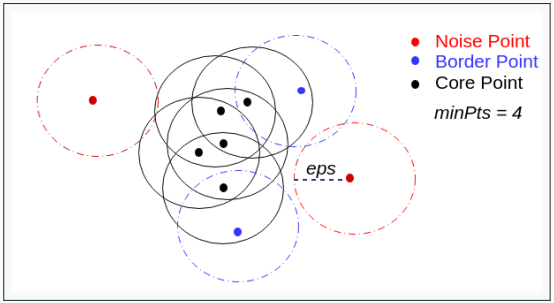


Data Preparation | Outliers - DBSCAN

DBSCAN relies on the definition of:

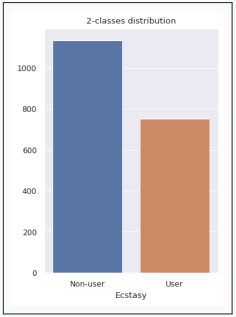
- **Core**: a point that has at least *minPts* points within distance *eps* from itself.
- **Border**: a point that has at least one Core point at a distance *eps*.
- **Noise**: a point that is neither a Core nor a Border.

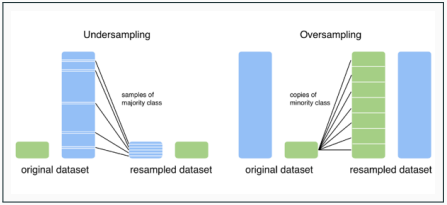
All the points labeled as Noise are considered outliers and they are removed from out dataset.



Sampling techniques

Our dataset is slightly unbalanced because the number of Ecstasy Non-User instances is higher than the User ones. This can affect the performances of the classifier increasing the mis-classification of the less represented class.

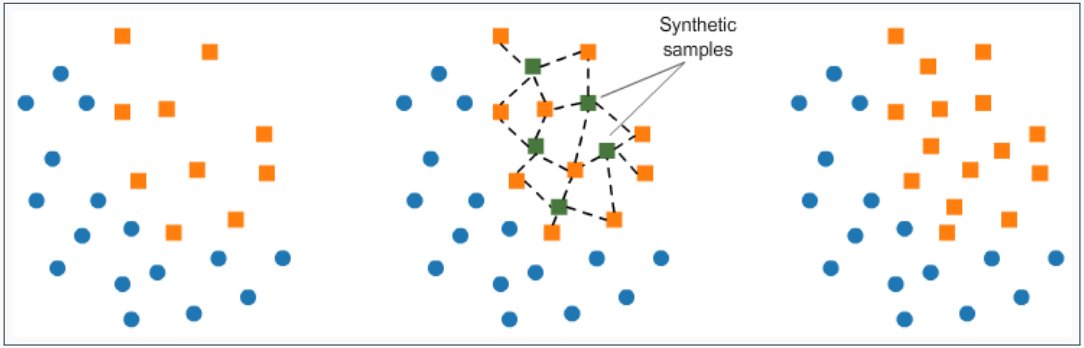




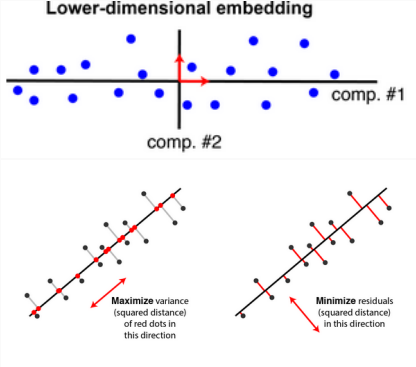
Type	Technique
Undersampling	Random Clustered Centroids
Oversampling	Borderline-SMOTE SVM-SMOTE ADASYN

Data Preparation | Sampling - SMOTE

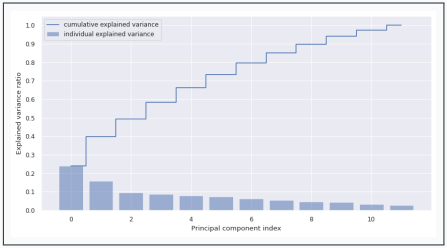
SMOTE (Synthetic Minority Oversampling TEchnique) consists of synthesizing elements for the minority class, based on those that already exist. It works randomly picking a point from the minority class and computing the k-nearest neighbors for this point. The synthetic points are added between the chosen point and its neighbors.



Principal component analysis is an unsupervised dimensionality-reduction method that, compute eigenvectors and eigenvalues of the covariance matrix, identify the principal components (i.e. directions that explain a **maximal amount of variance**)

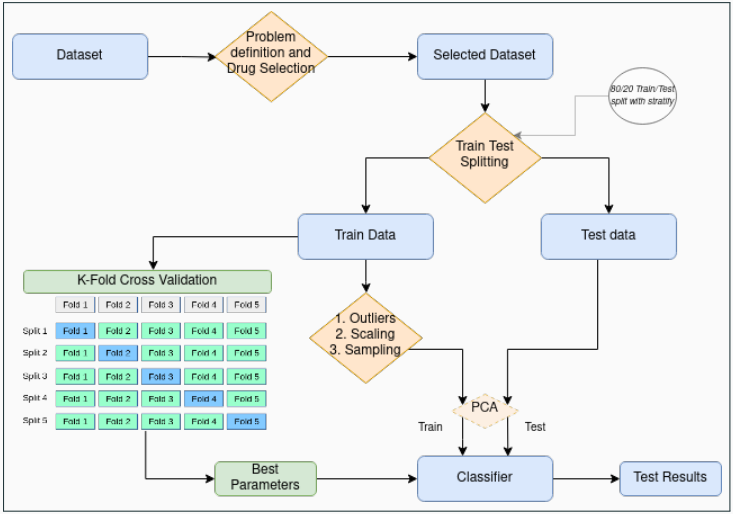


Example with 2 principal components



Principal components and associated explained variance: an acceptable value of explained varianced is included between 80% and 90%

Methods | Pipeline



Analysis pipeline

Our flow of work considers different:

- outlier removal methods
- sampling methods
- scaling methods
- binary classifiers

Best combination has been found and tested with **K-Fold Cross Validation**.

Metrics

- **Accuracy**, the most com mon and intuitive metric of evaluation:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total number of Predictions Made}}$$

- **Confusion Matrix**, to summarize the performance of a classification algorithm:

		Predicted: P	Predicted: N
Actual	P	TP	FN
	N	FP	TN

- **F1 Score**, useful for unbalanced data:

$$\text{F1 Score} = 2 * \frac{Precision \cdot Recall}{Precision + Recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

- **ROC curve**, comparing:

$$TPR = \frac{TP}{FN + TP} \qquad FPR = \frac{FP}{TN + FP}$$

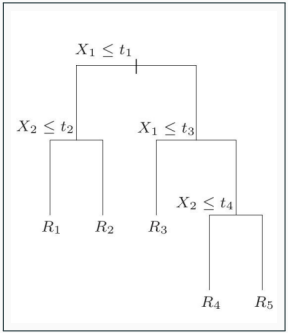
Methods | Algorithms: Decision Tree

Decision tree is a simple surpervised algorithm that models a set of sequential and hierarchical decision rules: it divides the predictor space in non-overlapping regions that are high dimensional rectangles.
Algorithm overview:

- assign to each leaf a label according to a **majority vote**
- among all possible splits, choose the one that **minimizes impurity**
- different metrics to compute impurity: GINI and entropy

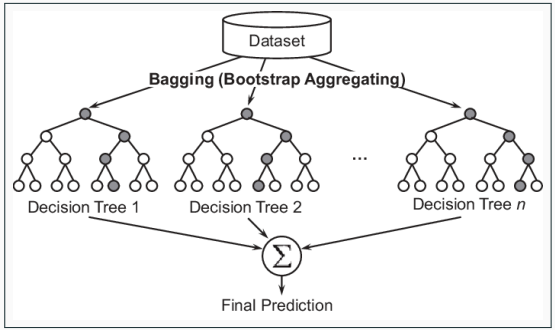
$$Gini(t) = 1 - \sum_j p(j|t)^2$$

$$Entropy(t) = -\sum_j p(j|t)\log_2 p(j|t)$$



Random Forest is a supervised algorithm that is based on an ensemble of trees.

- **Bagging**: creating N training set starting from the original one with bootstrap. On each of these datasets a decision tree is trained using only a subset of the feature;
- **Feature randomness**: each individual tree can pick only from a random subset of features, thus forcing more variation among the trees and lower correlation.



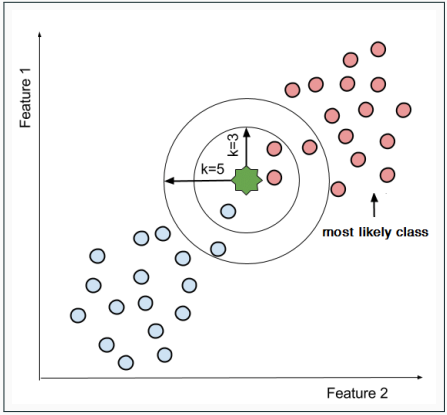
The idea behind **K-Nearest Neighbours** is to memorize the training set and then to predict the label of any new instance on the basis of the labels of its closest neighbors in the training set.

- Given Minkowski distance:

$$d_p = \left(\sum_{i=1}^{D} |a_i - b_i|^p \right)^{\frac{1}{p}}$$

- identify K-Neighbourhood N_0
- assign x_0 to class with highest estimated probability

$$P(Y = j | X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

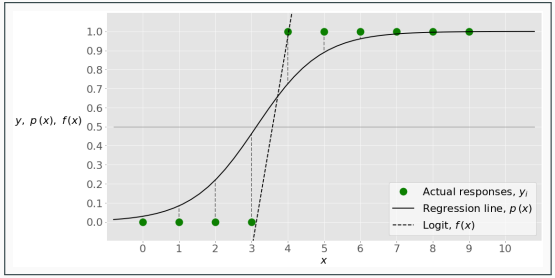


Logistic regression can be used for classification tasks, interpreting $p(x)$ as the probability of x of being 1. The hypothesis class associated with logistic regression is the composition of a sigmoid function $\sigma : \mathbb{R} \to [0, 1]$ over the class of linear functions.

- Sigmoid function: $\sigma(t) = \frac{1}{1+e^{-t}}$
- $t = \beta_0 + \beta_1 x$
- then the general logistic function $p : \mathbb{R} \to (0, 1)$:

$$p(x) = \sigma(t) = \frac{1}{1 + e^{-(\beta_0+\beta_1 x)}}$$

- The parameters of a logistic regression are most commonly estimated by maximum-likelihood estimation (**MLE**).



Support Vector Machine is a supervised learning algorithm, whose main goal is to find an hyperplane that divides the training data belonging from different classes.

- **Hard-Margin SVM** (linearly separable)

$$\text{argmax}_{(w,b):||w||=1} \left[\min_{i \in [m]} || < w, x_i > +b| | \right]$$

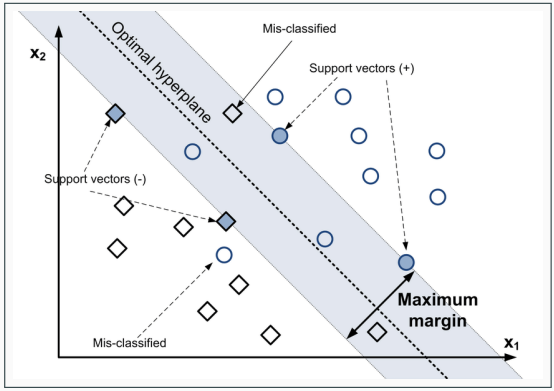
$$\text{s.t.} \quad \forall i, y_i(< w, x_i > +b) > 0$$

- **Soft-Margin SVM** (not linearly separable)

$$\text{argmin}_{w,b,\xi} \left(\lambda ||w||^2 + \frac{1}{m} \sum_{i=1}^{m} \xi_i \right)$$

$$\text{s.t.} \quad | < w, x_i > +b| \geq 1 - \xi_i, \quad \sum_{i=1}^{m} \xi_i \leq C$$

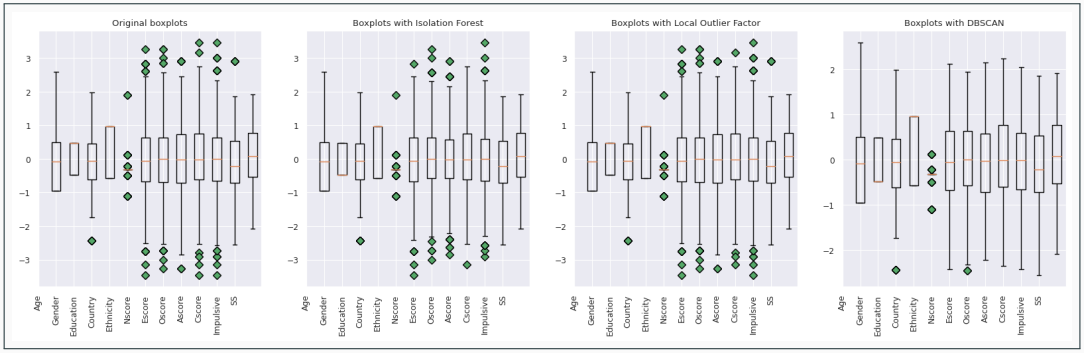
- **Kernel trick**: non-linearly separable, $\psi : X \to F$ s.t. $K(x, x) =< \psi(x), \psi(x) >$



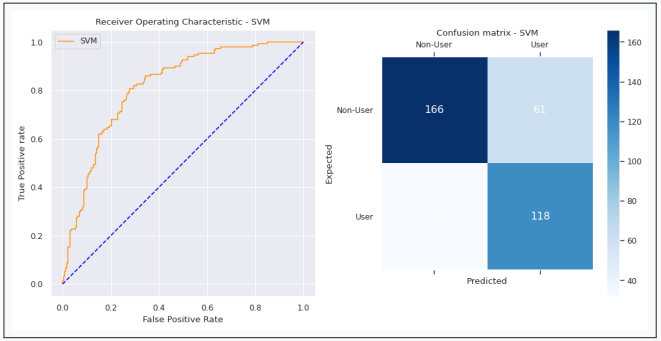
Results | Outlier removal method choice

The best suited outliers detection method is chosen comparing the three methods in terms of number of outliers removed, both numerically and graphically.

We look at how the original boxplots are impacted by the different outliers removal methods.



Results | Best performance

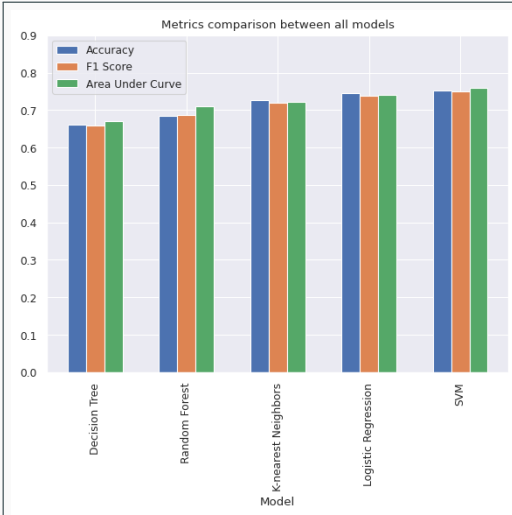


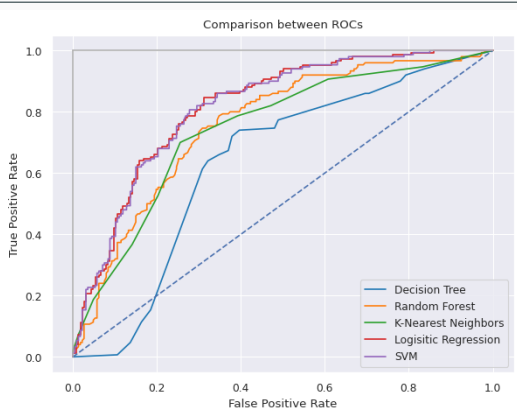
We show the results in terms of ROC curve and Confusion Matrix of the best performing model, which have been found considering:

- DBSCAN as outliers removal method
- MinMax Scaler as scaler method
- ADASYN as oversampling method
- SVM classification model with $(C, \gamma, kernel) = (100, 0.001, rbf)$

In the next slide, we compare the performance of all the classification methods, considering all metrics (accuracy, f1 score, area under curve and roc curve).

Results | Comparison among models

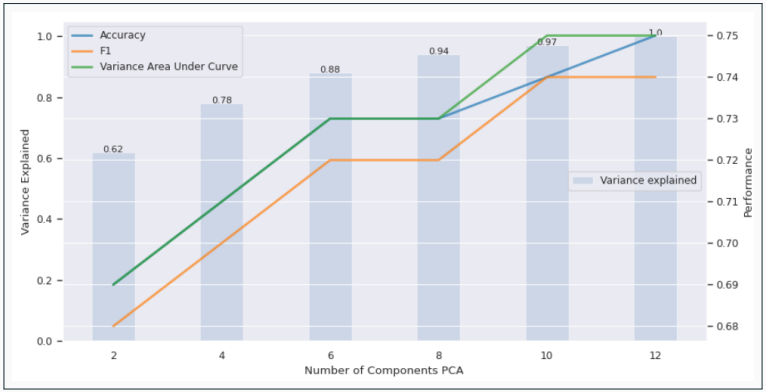




Results | Impact of PCA

Objective

The objective is to obtain a model which is less complex and lighter, without compromising too much its performance. In order to do that, we consider the model with the best performance in the and we see how it is impacted, in terms of performance, by dimensionality reduction.



Example:

- # components = 4
- 78% of explained variance
- metrics $\geq$ 70%.

5% drop in performance may be justified by the huge simplification of the model.

Conclusions

Model	Hyperparameter		Best Results		
	Param	Value	Accuracy	F1-score	AUC
Decision Tree	criterion max_depth	[gini, entropy, **log_loss**] [0, 2, **4**, 5, 6, 10, 15]	0.660	0.658	0.671
Random Forest	n_estimators criterion	[10, 50, 100, 250, 500, **750**, 1000] [gini, **entropy**, log_loss]	0.684	0.660	0.675
K-nearest Neighbors	n_neighbors algorithm	1, 5, 7, 11, 13, 15, **20**] [auto, ball_tree, kd_tree, **brute**]	0.727	0.719	0.722
Logistic Regression	penalty	[l2, **None**]	0.745	0.738	0.741
SVM	C γ kernel	[0.001, 0.1, 1, 10, **100**, 1000, 10'000] [0.0001, **0.001**, 0.01, 0.1, 1] [linear, poly, **rbf**, sigmoid]	**0.753**	**0.749**	**0.760**

Performance and Parameters of Classification models with ADASYN as over-sampling method and DBSCAN for outlier removal method (best combination)

Thank you for the attention
Luca Benfenati