

Machine Learning - SS 2021

Exercise 1: Classification

Federico Ambrogi, e1449911@student.tuwien.ac.at

Adam Höfler, e11847620@student.tuwien.ac.at

Matteo Panzeri 12039996@student.tuwien.ac.at

TU Wien

April 24, 2021

Contents

1	Introduction	1
1.1	Data Sets Description	1
1.2	Framework and Implementation	3
2	Data Exploration and Pre-processing	3
2.1	Asteroids Data Set	3
3	Performance Tests	3
3.1	Model Training Parameters	3
3.2	Confusion Matrix	4
3.3	Holdout and Cross Validation	4
3.4	Class Balance	4
4	Comparison	4

1 Introduction

In this document we describe the results of the implementation of three algorithms to solve classification problems on four different dataset.

1.1 Data Sets Description

Here we briefly introduce our four datasets.

Drug consumption data set This is the data set taken from exercise 0, so here we only provide basic details. The primary task of our implementation is to classify the likeliness of the frequency of usage of a certain type of drug (from never used to current usage), given personal aspects such as gender, ethnicity, education background etc. and psychological traits. In detail:

- **attributes:** 'age', 'gender', 'education', 'ethnicity', 'Nscore', 'Escore', 'Oscore', 'Ascore', 'Cscore', 'Impulsive', 'SS'
- **targets:** 'alcohol', 'amphetamines', 'amylNitrite', 'benzodiazepine', 'caffeine', 'cannabis', 'chocolate', 'cocaine', 'crack', 'ecstasy', 'heroin', 'ketamine', 'legal', 'LSD', 'methadone', 'mushrooms', 'nicotine', 'volatileSubstance' divided in **7 classes:** "Never", "<10 Years Ago", "Last Decade", "Last Year", "Last Month", "Last Week", "Last Day"

Asteroids data set The asteroids data set is designed to perform a binary classification task. It consists of a total of astrophysical data, such as the distance from the Earth, the size of the major axis of the orbits, their mass etc. as well metadata regarding their name. The target feature for classification is a string variable that maps to boolean values "True" in the case the asteroid represents a concrete hazard for the Earth i.e. with high risk of impact, and "False" otherwise.

In detail:

- **attributes:** 'Absolute Magnitude', 'Est Dia in KM(min)', 'Est Dia in KM(max)', 'Relative Velocity km per sec', 'Miss Dist.(kilometers)', 'Minimum Orbit Intersection', 'Jupiter Tisserand Invariant', 'Eccentricity', 'Semi Major Axis', 'Inclination', 'Asc Node Longitude', 'Orbital Period', 'Perihelion Distance', 'Perihelion Arg', 'Aphelion Dist', 'Perihelion Time', 'Mean Anomaly', 'Mean Motion'
- **target:** 'Hazardous' divided in **classes:** "True" and "False"

Breast Cancer (Kaggle Data Set) ccc

- **attributes:** 'radiusMean', 'textureMean', 'perimeterMean', 'areaMean', 'smoothnessMean', 'compactnessMean', 'concavityMean', 'concavePointsMean', 'symmetryMean', 'fractalDimensionMean', 'radiusStdErr', 'textureStdErr', 'perimeterStdErr', 'areaStdErr', 'smoothnessStdErr', 'compactnessStdErr', 'concavityStdErr', 'concavePointsStdErr', 'symmetryStdErr', 'fractalDimensionStdErr', 'radiusWorst', 'textureWorst', 'perimeterWorst', 'areaWorst', 'smoothnessWorst', 'compactnessWorst', 'concavityWorst', 'concavePointsWorst', 'symmetryWorst', 'fractalDimensionWorst'
- **target:** 'class' divided in **classes:** "True" and "False"

Advertising Bidding (Kaggle Data Set) ccc

In detail:

- **attributes:** 'Region', 'City', 'AdExchange', 'Adslotwidth', 'Adslotheight', 'Adslotfloorprice', 'CreativeID', 'Biddingprice', 'AdvertiserID', 'interest_news', 'education', 'automobile', 'interest_realestate', 'IT', 'electronicgame', 'interest_fashion', 'entertainment', 'luxury', 'homeandlifestyle', 'interest_health', 'food', 'interest_divine', 'interest_motherhood_parenting', 'sports', 'interest_travel_outdoors', 'interest_social', 'Inmarket_3cproduct', 'Inmarket_appliances', 'Inmarket_clothing_shoes_bags', 'Inmarket_Beauty_PersonalCare', 'Inmarket_infant_momproducts', 'Inmarket_sportsitem', 'Inmarket_outdoor', 'Inmarket_healthcareproducts', 'Inmarket_luxury', 'Inmarket_realestate', 'Inmarket_automobile', 'Inmarket_finance', 'Inmarket_travel', 'Inmarket_education', 'Inmarket_service', 'art_photography_design', 'onlineliterature', 'Inmarket_electronicgame', '3c', 'Inmarket_book', 'Inmarket_medicine', 'Inmarket_fooddrink', 'culture', 'sex', 'Demographic_gender_male', 'Demographic_gender_female', 'Inmarket_homeimprovement', 'Payingprice', 'imp', 'click', 'Browser', 'Adslotvisibility', 'Adslotformat'
- **target:** 'conv' divided in **classes:** "True" and "False"

In detail:

- **attributes:** 'Absolute Magnitude', 'Est Dia in KM(min)', 'Est Dia in KM(max)', 'Relative Velocity km per sec', 'Miss Dist.(kilometers)', 'Minimum Orbit Intersection', 'Jupiter Tisserand Invariant', 'Eccentricity', 'Semi Major Axis', 'Inclination', 'Asc Node Longitude', 'Orbital Period', 'Perihelion Distance', 'Perihelion Arg', 'Aphelion Dist', 'Perihelion Time', 'Mean Anomaly', 'Mean Motion'
- **target:** 'Hazardous' divided in **classes:** "True" and "False"

1.2 Framework and Implementation

ccc

Algorithms We chose three different algorithms for our study.

Decsion Three a

kNN - k Nearest Neighbours b

naive Bayes c

2 Data Exploration and Pre-processing

In this section we describe the necessary preliminary steps to import and prepare the data to make them suitable for the learning algorithms. The steps are handled by the script *data_preparation.py* which includes dedicated function to pre-process each data set.

2.1 Asteroids Data Set

The original data set contained 40 columns of different variable types (numeric and categorical). However, some columns were unnecessary, not usable or redundant for classification purposes, or contained constant values.

3 Performance Tests

3.1 Model Training Parameters

We remind here the definition of the metric parameters we will used to quantify the performance of our classifiers i.e. precision (P), recall (R) and accuracy (A):

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad A = \frac{TP + TN}{\#all} \quad (1)$$

It is straightforward to calculate these parameters for binary classification tasks out of the confusion matrix. In case of multiple labels, we need to calculate these parameter for each class, given that:

- TP s are the values in the diagonal;
- FN s for a certain class are the sum of values in the corresponding row excluding the TP ;
- FP s for a certain class are the sum of values in the corresponding column excluding the TP ;
- TN s for a certain class are the sum of all rows and columns, excluding the class's column and row.

Another convenient metric, particularly because it is calculated directly from the proper *scikit-learn* function, is called *f1 - score*, which is defined as the harmonic mean of the precision and recall:

$$f1 - score = 2 \times \frac{P\dot{R}}{P + R} \quad (2)$$

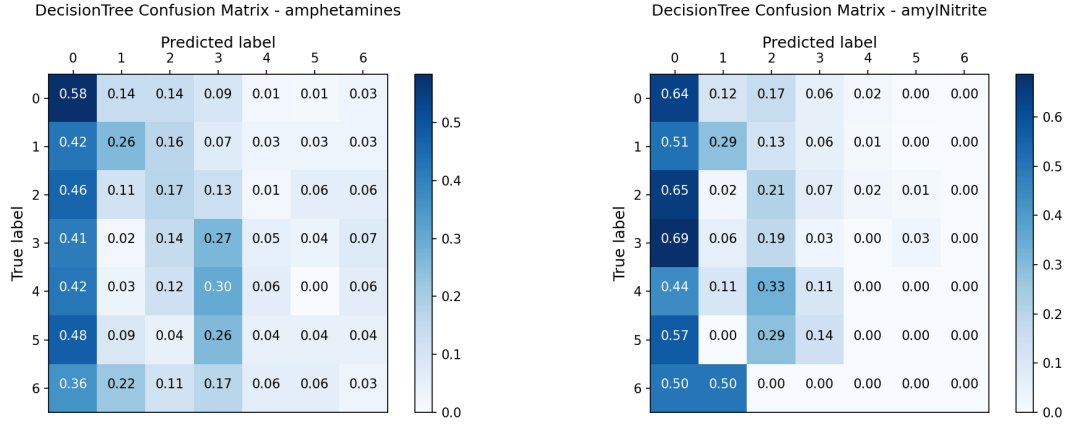


Figure 1: Example confusion matrices for REDO, just an example

3.2 Confusion Matrix

For each classifier, we produce a confusion matrix where each entry i, j corresponds to the number of observations in group i , but predicted to be in group j . We chose to normalize the entries according to the sum of each row. In case of binary classification, the matrix reduces to the number of true negatives (TN), false positives (FP), false negatives (FN) and true positives (TP).

Examples can be seen in Fig. 1.

Describe micro averaging and macro averaging, will we report both ?

3.3 Holdout and Cross Validation

Here we describe briefly the techniques of "holdout" and "cross validation" that are used to evaluate a model. The **holdout** method is essentially based on the splitting on the input data set into two subsets, one used for training the model, and one used for testing the model, for example in 80%–20% proportion, although there is not a fixed recipe for this split. Once the model is trained, the evaluation can be performed on the test data set, and this is therefore possible to check if the prediction of the model matches the data. One big issue of this model is that the training strongly depends on the splitting of the initial data set, for example if the characteristics of the training data set are not representative of the whole data set.

A more powerful method is the **cross-validation** or "k-fold cross validation". From the full dataset, a test is held out for final evaluation, but the validation set is no longer needed. For this, the training set is split into k sets, so that the model is trained using $k - 1$ folds as training data, and the remaining one is used for the validation as a test set (to compute the interesting metric of the model). Once we obtain such k number of metrics, the final result is the average of these parameters, obtained for each iteration of the cross-validation on each distinct fold.

3.4 Class Balance

In this section we discuss the problem of imbalanced data. Whenever classes are not represented equally (imbalanced data), the result of the training and testing of the model might be biased. In particular, a metric such as the accuracy might appear very satisfying while, in fact, it is only representing the underlying class distribution. The model is in fact favoured to learn the class with the majority of instances.

Fig. 2 shows the distribution of the various class attributes for our datasets. Starting with the binary classification problems we see that the "breast cancer" dataset is slightly imbalanced (37,5%- 62,5%) recurrence events vs non recurrence, while the imbalance is more pronounced for the other dataset, in particular we see (17%-83%) hazardous vs non hazardous, and (2%-98%) buy vs non buy classes.

In the case of the drug consumption dataset, the situation is more complicated to analyze since it varies largely depending on the considered drug.

In order to balance the dataset, we implemented an algorithm that performs the following steps. First, it extracts the number of entries in the under-represented class (which by chance is always class "True" for the three binary classification tasks). Then we randomly select a number of rows from the

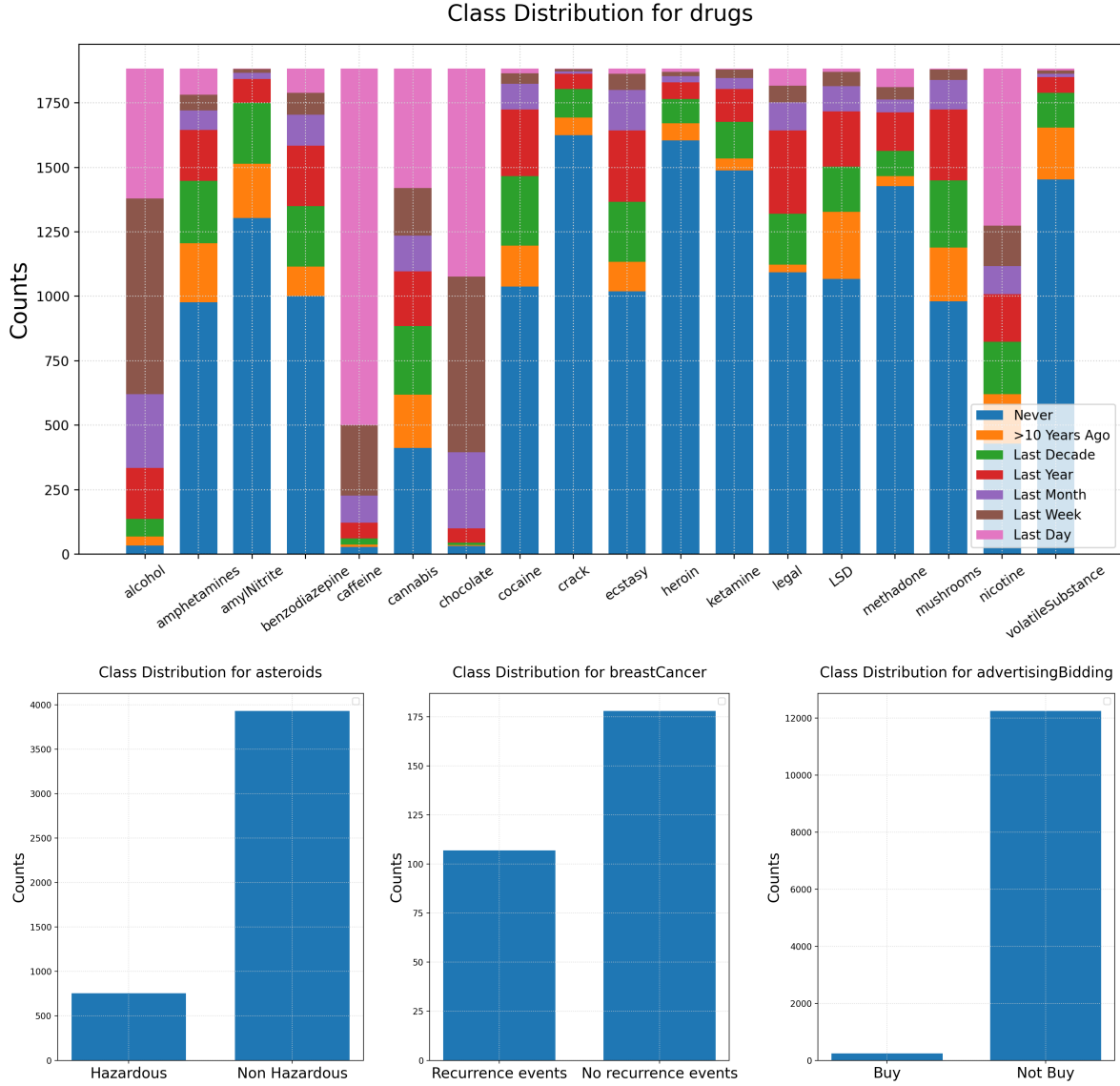


Figure 2: Distribution of classes for the target attributes for the multiclass classification (top panel, drug dataset) and binary classification (bottom panel, asteroids, breast cancer and advertising bidding datasets).

remaining data of where class is equal to "False", adding an extra 10% to this total. Then, we create a final dataset combining the rows with the class "True" and the randomly selected "False". This step is done at a preliminary phase before any splitting into the training and testing subsets.

4 Comparison