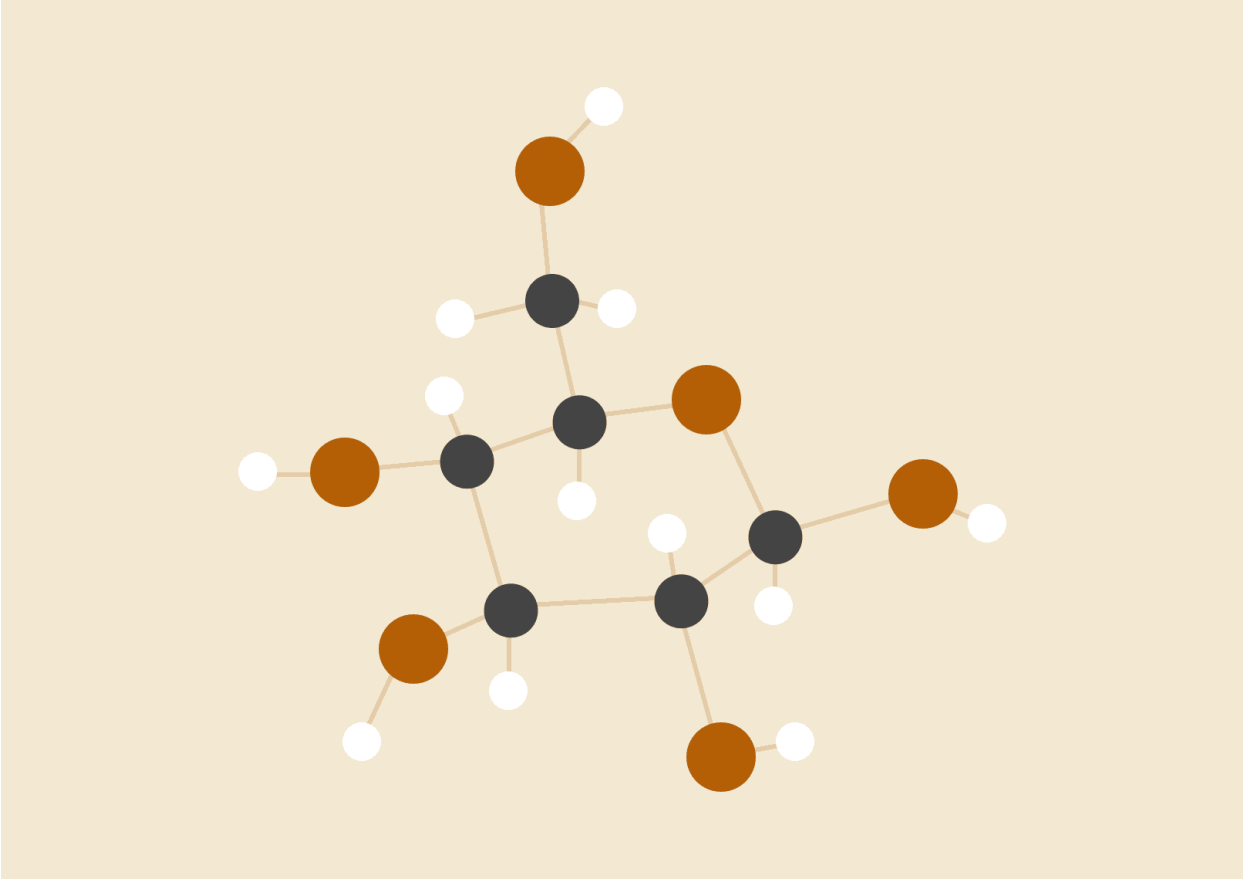
UNIVERSITÀ DELLA CALABRIA DIPARTIMENTO DI MATEMATICA E INFORMATICA

Course of Data Analytics (Machine Learning)

Machine Learning Project COMPAS Scores

A Machine Learning Model that predicts if a defendant becomes a recidivist



Team: *404 name not found*

- Canonaco Martina [231874]
- Gabriele Giada [235799]
- Gena Davide [231873]
- Morello Michele [223953]

a.y. 2021/2022

Instructor: *Prof. Rullo Pasquale*

Teaching Assistant: *Ph.D. Liguori Angelica*

ABSTRACT 4

INTRODUCTION 5

BUSINESS UNDERSTANDING 7
- Business Objectives 7
  - Background 7
  - Business Objectives 7
  - Business Success Criteria 7
- Situation Assessment 8
  - Inventory of resources 8
- Data Mining Goals 9
  - Goals 9
  - Success Criteria 9
- Project Plan 9

DATA UNDERSTANDING 11
- Initial Data Collection 11
- Data Description 12
- Data Exploration 14
  - Numerical attributes 14
  - Non numerical attributes 18
  - Histograms, bar plots and scatter plots according to the class label 22
  - Attributes regarding dates 24
- Data Quality 26
  - Missing values 26
  - Duplicated attributes 27
  - Incorrect values 27
  - Other considerations 28

DATA PREPARATION 29
- Data Set Description 29
- Data Selection 29
- Data Cleaning 29
- Data Construction 30
- Data Integration 30
- Reformatted Data 30

MODELING 31
- Modeling Technique Selection 31
  - Naïve Bayes Classifier 31

Decision Tree Classifier 31
K-Nearest-Neighbors Classifier 32
Random Forest Classifier 32
AdaBoost Classifier 32
Test Design 32
Build Model 32
Model Assessment 33
EVALUATION 34
Results Evaluation 34
Process Review 39
Possibile Actions Decision 39
DEPLOYMENT 39
CONCLUSION 39
RESOURCES 40

ABSTRACT

Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) is a case management and decision support tool developed and owned by Northpointe (now Equivant) used by U.S. courts to assess the likelihood of a defendant becoming a recidivist. The Northpointe Suite is an **integrated web-based assessment** and **case management system** for criminal justice practitioners.

COMPAS was first developed in **1998** and has been revised over the years as the knowledge base of criminology has grown and correctional practice has evolved. COMPAS has been used by the U.S. states of New York, Wisconsin, California, Florida's Broward County, and other jurisdictions. The dataset analyzed in this documentation is `compas-scores.csv` available here. The dataset collects over 11k records registered from January 2013 to December 2014.

The report is structured following the step of the **CRISP-DM Methodology**: the first chapter contains a brief introduction and description of the dataset and the CRISP-DM Methodology.

Then, the second chapter explains the phase of Business Understanding and project plan. The third section shows a deeper description of the data set and the step of Data Understanding. The fourth chapter is about the Data Preparation, in which there are the phases of data quality and data cleaning. The fifth section contains the model used to fit and analyze the dataset and then, the sixth section shows the evaluations of the various models.

At the end, there is a short section about the Deployment plan, our conclusions and the list of the resources used to develop and implement the whole project.

INTRODUCTION

Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) is a case management and decision support tool developed and owned by Northpointe (now Equivant) used by U.S. courts to assess the likelihood of a defendant becoming a recidivist. The Northpointe Suite is an **integrated web-based assessment** and **case management system** for criminal justice practitioners. The Northpointe Suite has modules designed for pretrial, jail, probation, prison, parole and community corrections applications.

COMPAS was first developed in **1998** and has been revised over the years as the knowledge base of criminology has grown and correctional practice has evolved. In many ways changes in the field have followed new developments in risk assessment. We continue to make improvements to COMPAS based on results from norm studies and recidivism studies conducted in jails, probation agencies, and prisons. COMPAS is **periodically updated** to keep pace with emerging best practices and technological advances.

The dataset analyzed in this documentation is `compas-scores.csv` available here. The dataset collects over 11k records registered from January 2013 to December 2014. Each record represents a criminal with a name (first and last), date of birth, age, gender, ethnicity and other information about his/her arrest and screening date, type of offense and the recidivism.

To develop this academic project, we followed the **CRISP-DM Methodology**. The **CRoss Industry Standard Process for Data Mining** (CRISP-DM) is a process model that serves as the base for a data science process.

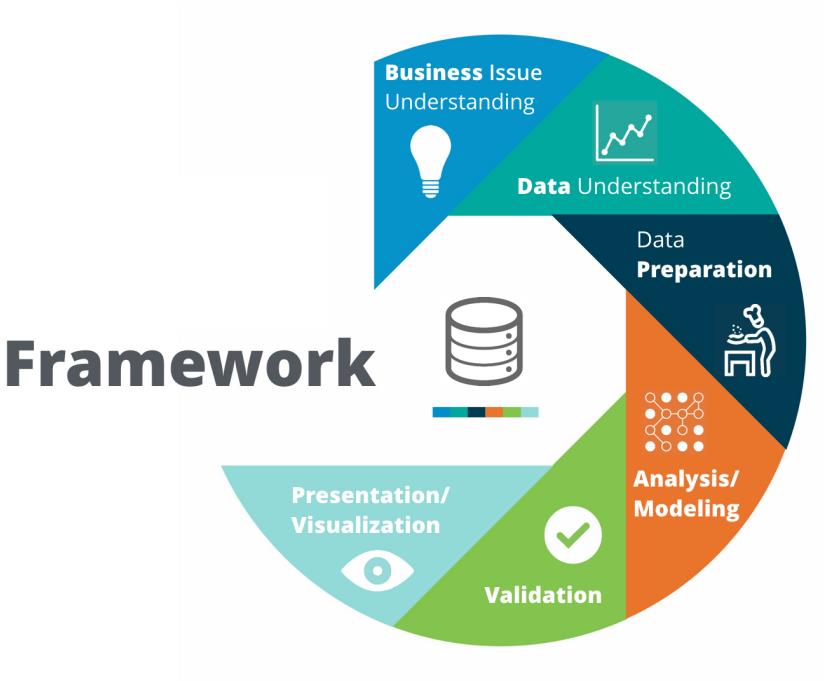


Figure 1. *CRISP-DM Methodology Schema*

It has **six sequential phases**:

1. **Business understanding** – What does the business need?
2. **Data understanding** – What data do we have / need? Is it clean?
3. **Data preparation** – How do we organize the data for modeling?
4. **Modeling** – What modeling techniques should we apply?
5. **Evaluation** – Which model best meets the business objectives?
6. **Deployment** – How do stakeholders access the results?

Published in **1999** to standardize data mining processes across industries, it has since become the most common methodology for data mining, analytics, and data science projects. Data science teams that combine a loose implementation of CRISP-DM with overarching **team-based agile project management** approaches will likely see the best results.

BUSINESS UNDERSTANDING

Business Objectives

Background

The project is about COMPAS Scores dataset that collects over 11k records and 47 attributes. COMPAS is a case management and decision support tool developed and owned by Northpointe (now Equivant) used by U.S. courts. The Northpointe Suite is an integrated web-based assessment and case management system for criminal justice practitioners.

Business Objectives

The primary objective of the project is to determine and predict if a defendant becomes a recidivist. It will be useful for community corrections applications.
As a secondary objective, we want to show the correlation between a “normal” recidivist and a violent recidivist. So, we will first work on the whole dataset to reach the main objective and then we want to focus on which of the criminals signed as a recidivist that is also a violent recidivist.

Moreover we want to set two sub objectives: the first one is focused on how many days passes from the first crime to the second one (the one in which a criminal becomes recidivist, so only the rows with `is_recid = 1`), the second one has the same idea but is focused on the violent crime (`is_violent_recid = 1`).

Business Success Criteria

To reach the business objective, it is important to follow some success criteria such as:

- Achieve a good prediction level, in order to improve the re-education plan for the defendants
- Less are the recidivists, less will be the cost to maintain detention institutions full of criminals
- To invest in readmission in the society will improve the wellness of the whole community

Situation Assessment

Inventory of resources

To develop the project we used the following technologies, tools and libraries:

Name	Logo	Description	Online Reference
Python 3.8		**Python** is a **high-level**, interpreted, general-purpose **programming language**. Its design philosophy emphasizes code readability with the use of significant indentation. Python consistently ranks as one of **the most popular** programming languages.	Python.org
Jupyter Lab		**JupyterLab** is the latest **web-based interactive development environment** for notebooks, code, and data. Its **flexible interface** allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and **machine learning**.	Jupyter Notebook
Pandas		**Pandas** is an **open source data analysis** and manipulation tool, built on top of the Python programming language. We used this library in order to **manipulate datasets quickly** thanks to its wide amount of features provided for this purpose.	Pandas
Seaborn		**Seaborn** is a Python **data visualization library** based on matplotlib. It provides a **high-level interface** for drawing attractive and informative statistical graphics.	Seaborn
Visual Studio Code		**Visual Studio Code** is a **source code editor** developed by Microsoft for Windows, Linux and macOS.	Visual Studio Code
GitHub		**GitHub** is a provider of Internet hosting for **software development**. It offers the distributed version control and source code management (SCM) functionality of Git, plus its own features. It provides **access control** and several **collaboration** features.	GitHub
Kaggle		**Kaggle** is a Machine Learning and Data Science Online **Community**	Kaggle

Table 1. *List of technologies, tools and libraries used.*

Data Mining Goals

Goals

The project is about a **binary classification problem** in which the prediction task is to determine if a defendant is a recidivist or not, according to the following discretization:

- If the attribute `is_recid` is equal to `0` → the Criminal is not a recidivist.
- If the attribute `is_recid` is equal to `1` → the Criminal is a recidivist.
- If the attribute `is_recid` is equal to `-1` → the information about Criminal's recidivism is unknown.

In addition, we want to understand if there is a correlation between defendants that are recidivists and violent recidivists. This will be our secondary goal.

Success Criteria

Our final goal is to achieve a good level of accuracy and prediction of recidivism based on significant attributes such as the age, the gender and the ethnicity.

Moreover, we want to compare recidivists and violent recidivists of a violent crime. In addition we decided to add the sub goal of predicting the difference (in days) between the date of the first crime anche the date of the recidivist or the violent recidivist offense.

Project Plan

The project implementation is divided in 5 main steps, following the CRISP-DM Methodology. We assumed that the Business Understanding was already done. Also, for this academic project, the Deployment part will be a presentation for the final exam, so it will not be present in this documentation. Each step covers a certain number of days of development. The whole project requires 5 weeks to be completed.

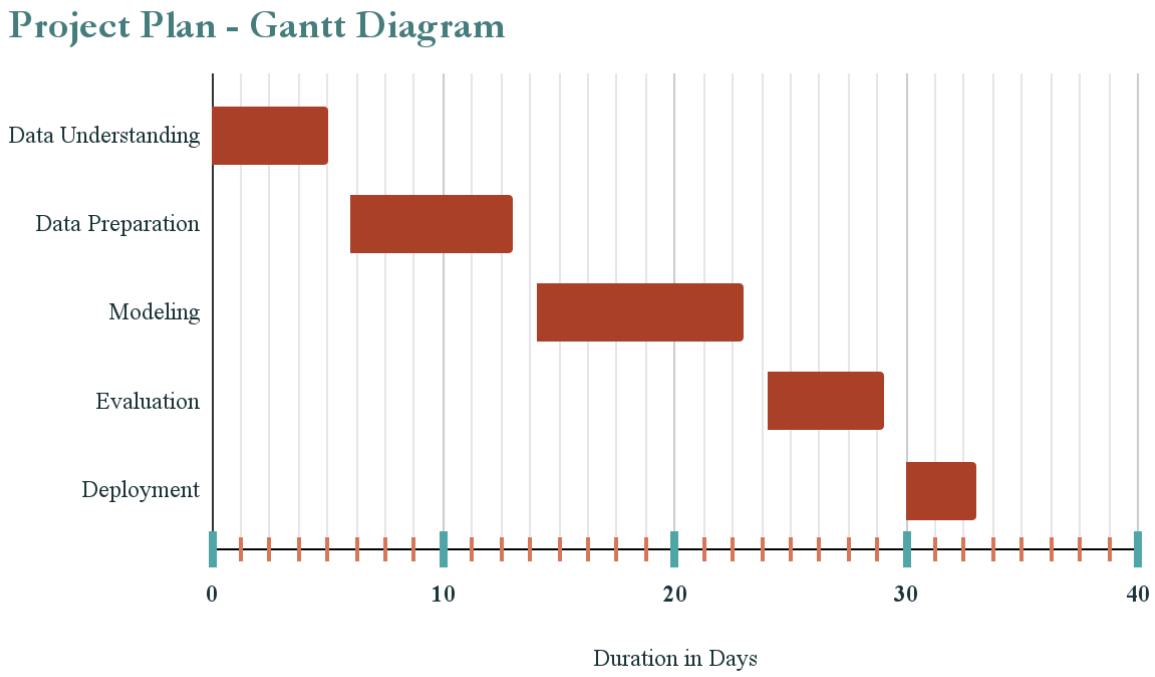
The following table shows each phase, its duration, the starting and the ending date, and its predecessor(s):

Project Step	Duration	Status	Start	End	Pred.
01 Data Understanding	5 days	Completed	10/05/2022	15/05/2022	//
02 Data Preparation	7 days	Completed	16/05/2022	23/05/2022	01
03 Modeling	10 days	In progress	24/05/2022	02/06/2022	02

Project Step	Duration	Status	Start	End	Pred.
04 Evaluation	5 days	Not started	03/06/2022	08/06/2022	03
05 Deployment	4 days	Not started	09/06/2022	12/06/2022	04

Table 2. *Project Status*

The following Gantt Diagram presents the distribution of the phases in relation with the time expressed in number of days:



Plot 1 *Project Plan - Gantt Diagram*

DATA UNDERSTANDING

Initial Data Collection

The dataset COMPAS score has been obtained from the analysis made by the Northpointe tool, called COMPAS, in 2016 and updated until 2017. The data collected by the COMPAS algorithm represent one of the most popular scores used nationwide and this algorithm is increasingly being used in pretrial and sentencing. Specifically, the data collected concern the county of Broward because it is a large jurisdiction using the COMPAS tool in pretrial release decisions and Florida has strong open-records laws. Through a public records request, ProPublica obtained two years worth of COMPAS scores from the Broward County Sheriff's Office in Florida. The dataset was uploaded in 2016 but collects data for all **18,610 people** who were scored in **2013** and **2014**.

Because Broward County primarily uses the score to determine whether to release or detain a defendant before his trial, scores that have been assessed in probation or at other stages of the criminal justice system have been discarded in the dataset. This left us with **11,757 people** who were evaluated at the preliminary stage and who correspond to the tuple number of the final dataset.
Each pretrial defendant received at least three COMPAS scores:

- "Risk of Recidivism"
- "Risk of Violence"
- "Risk of Failure to Appear"

Moreover, COMPAS scores for each defendant ranged from 1 to 10:

- Scores 1 to 4 were labeled as "**Low**";
- 5 to 7 were labeled "**Medium**";
- 8 to 10 were labeled "**High**."

So, the goal of our analysis is to obtain a model able to decide a status of "Recidivism" for new defendant entries.

But what do we mean by "Recidivism"?
According to Northpointe definition: "a finger-printable arrest involving a charge and a filing for any uniform crime reporting (UCR) code." We interpreted that to mean a criminal offense that resulted in a jail booking and took place after the crime for which the person was COMPAS scored. In general, we defined recidivism as a new arrest within two years. We based this decision on Northpointe's practitioners guide, which says that

its recidivism score is meant to predict “a new misdemeanor or felony offense within two years of the COMPAS administration date.”

And what do we mean for “Violent Recidivism”?
For violent recidivism, we used the FBI’s definition of violent crime, a category that includes murder, manslaughter, forcible rape, robbery and aggravated assault.

Data Description

The final dataset is composed of 47 **attributes** and **11,757 instances**. A brief description of each attribute is collected in the following table:

Attribute name	Type	Description
id	int64	refers to the dataset’s identifier of the defendant
name	object	refers to the defendant’s complete name (first and last name)
first	object	refers to the defendant’s first name
last	object	refers to the defendant’s last name
compas_screening_date	object	refers to the date on which the assessment was made
sex	object	refers to the defendant’s sex, expressed in a binary way (M or F)
dob	object	refers to the date of birth of the defendant
age	int64	refers to the defendant’s age
age_cat	object	refers to the the age of the defendants expressed in three ranges (less than 25, 25-45, greater than 45)
race	object	refers to the ethnicity of the defendant
juv_fel_count	int64	number of grave crimes committed at an early age
decile_score	int64	is a number, from 1 to 10, that indicates the risk of recurrence in general (the higher the risk, the higher the number)
juv_misd_count	int64	number of minor grave crimes committed at an early age
juv_other_count	int64	number of other crimes committed at an early age
priors_count	int64	refers to the defendant’s number of priors
days_b_screening_arrest	float64	number of days between the COMPAStest and the arrest day
c_jail_in	object	refers to the date on which the criminal enters prison
c_jail_out	object	refers to the date on which the criminal is released from prison

c_case_number	object	refers to the identifier of the criminal's case
c_offense_date	object	refers to the date on which the criminal committed the crime
c_arrest_date	object	refers to the date on which the criminal was arrested
c_days_from_compas	float64	number of days between the COMPAS test and the crime
c_charge_degree	object	severity of the crime (felony (F), misdemeanor (M), other (O))
c_charge_desc	object	refers to the description of the criminal’s charge
is_recid	int64	indication of whether the person is a repeat offender
num_r_cases	float64	it is the number of recidivism cases
r_case_number	object	refers to the identifier of the recidivist's case
r_charge_degree	object	severity of the recid crime (felony (F), misdemeanor (M), other (O))
r_days_from_arrest	float64	refers to the number of days since the arrest of the recidivist
r_offense_date	object	refers to the date on which the recidivist was arrested
r_charge_desc	object	refers to the description of the recidivist’s charge
r_jail_in	object	refers to the date on which the recidivist enters prison
r_jail_out	object	refers to the date on which the recidivist is released from prison
is_violent_recid	int64	indication of whether the person is a recidivist of a violent crime
num_vr_cases	float64	it is the number of violent recidivism cases
vr_case_number	object	refers to the identifier of the recidivist of a violent crime's case
vr_charge_degree	object	severity of the violent crime (felony (F), misdemeanor (M), other (O))
vr_offense_date	object	refers to the date on which the recidivist of a violent crime was arrested
vr_charge_desc	object	refers to the description of the recidivist of a violent crime’s charge
v_type_of_assessment	object	refers to the valuation given to the violent defendant
v_decile_score	int64	is a number from 1 to 10, that indicates the risk of recidivism in violent crimes. When evaluating a case in COMPAS, the two scores are generated (among other things)
v_score_text	object	it is the description referring to the given score
v_screening_date	object	refers to the date on which the assessment was made

type_of_assessment	object	refers to the valuation given to the defendant
decile_score.1	int64	the same as “decile_score”
score_text	object	it is the description referring to the given score
screening_date	object	refers to the date on which the assessment was made

Table 3. *Attributes Description and Types*

We also reported on the attached Jupyter Notebook the statistics description for the numerical attributes.

Data Exploration

In this phase we used `Pandas` and a couple of `Python` libraries usually used for data mining and for printing plots, to make more in-depth analysis.
Since there are 47 attributes, we decided to analyze them by dividing in three main groups:

- Numerical Attributes
- Categorical Attributes
- Dates Attributes

Numerical attributes

First of all, we can take a look to the numerical attributes, their ranges and their statistics description:

	count	mean	std	min	25%	50%	75%	max
id	11757.0	5879.000000	3394.097892	1.0	2940.0	5879.0	8818.0	11757.0
age	11757.0	35.143319	12.022894	18.0	25.0	32.0	43.0	96.0
juv_fel_count	11757.0	0.061580	0.445328	0.0	0.0	0.0	0.0	20.0
decile_score	11757.0	4.371268	2.877598	-1.0	2.0	4.0	7.0	10.0
juv_misd_count	11757.0	0.076040	0.449757	0.0	0.0	0.0	0.0	13.0
juv_other_count	11757.0	0.093561	0.472003	0.0	0.0	0.0	0.0	17.0
priors_count	11757.0	3.082164	4.687410	0.0	0.0	1.0	4.0	43.0
days_b_screening_arrest	10577.0	-0.878037	72.889298	-597.0	-1.0	-1.0	-1.0	1057.0
c_days_from_compas	11015.0	63.587653	341.899711	0.0	1.0	1.0	2.0	9485.0
is_recid	11757.0	0.253806	0.558324	-1.0	0.0	0.0	1.0	1.0
num_r_cases	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
r_days_from_arrest	2460.0	20.410569	74.354840	-1.0	0.0	0.0	1.0	993.0
is_violent_recid	11757.0	0.075019	0.263433	0.0	0.0	0.0	0.0	1.0
num_vr_cases	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
v_decile_score	11757.0	3.571489	2.500479	-1.0	1.0	3.0	5.0	10.0
decile_score.1	11757.0	4.371268	2.877598	-1.0	2.0	4.0	7.0	10.0

Figure 2. *Attributes Statistics Description*

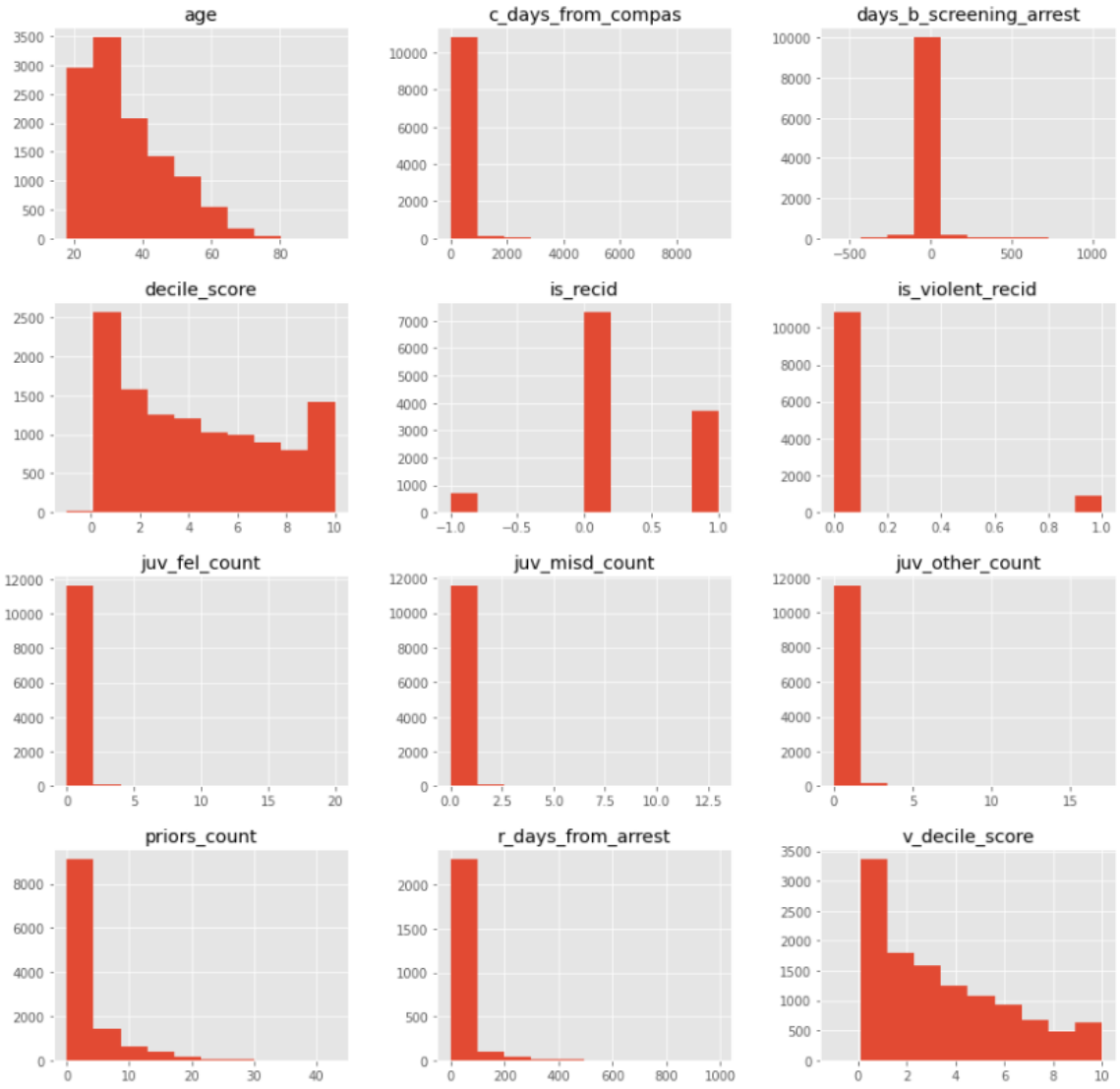


Figure 3. *Histograms of the Numerical Attributes*

We also reported the box plot of the numerical attributes, to show better some linear values and few mainly characterized by outliers.

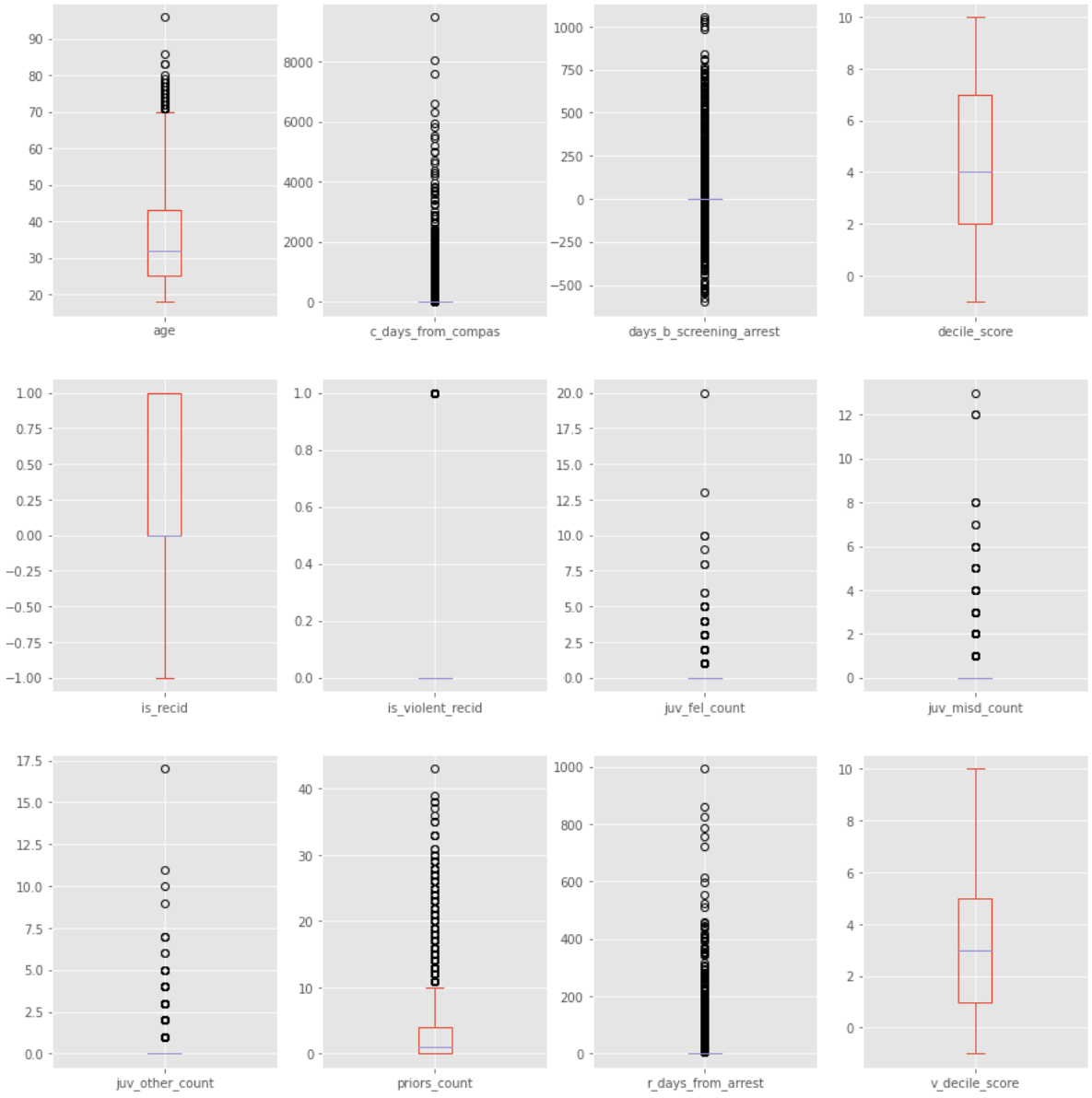


Figure 4. *Box plots of the numerical attributes*

As you can see, some of the numerical attributes are missing. This is because it is not useful to show the histograms of null values, we will present this problem better in the *Data Quality* phase.

We also analyzed the attribute distributions respect to the main class label `is_recid` and the secondary label `is_violent_recid`:

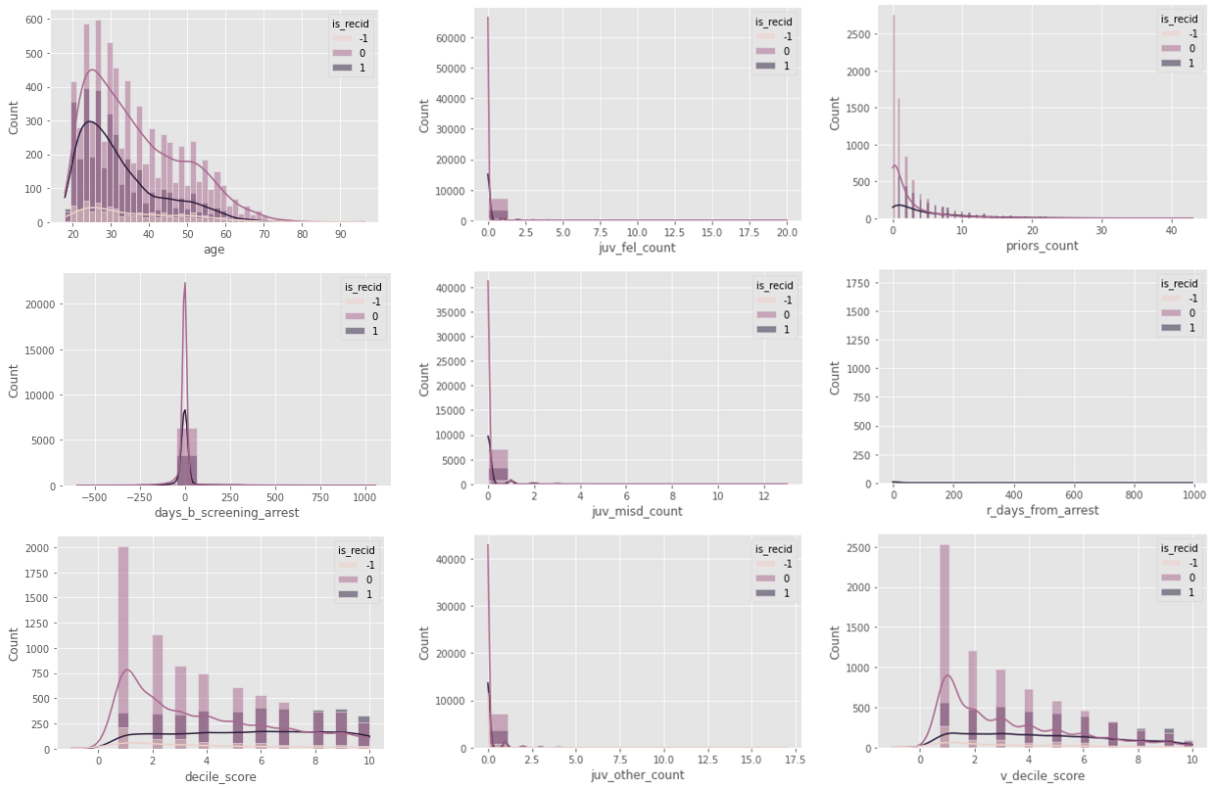


Figure 5. *Numerical Attribute Distributions Respect to* `is_recid`

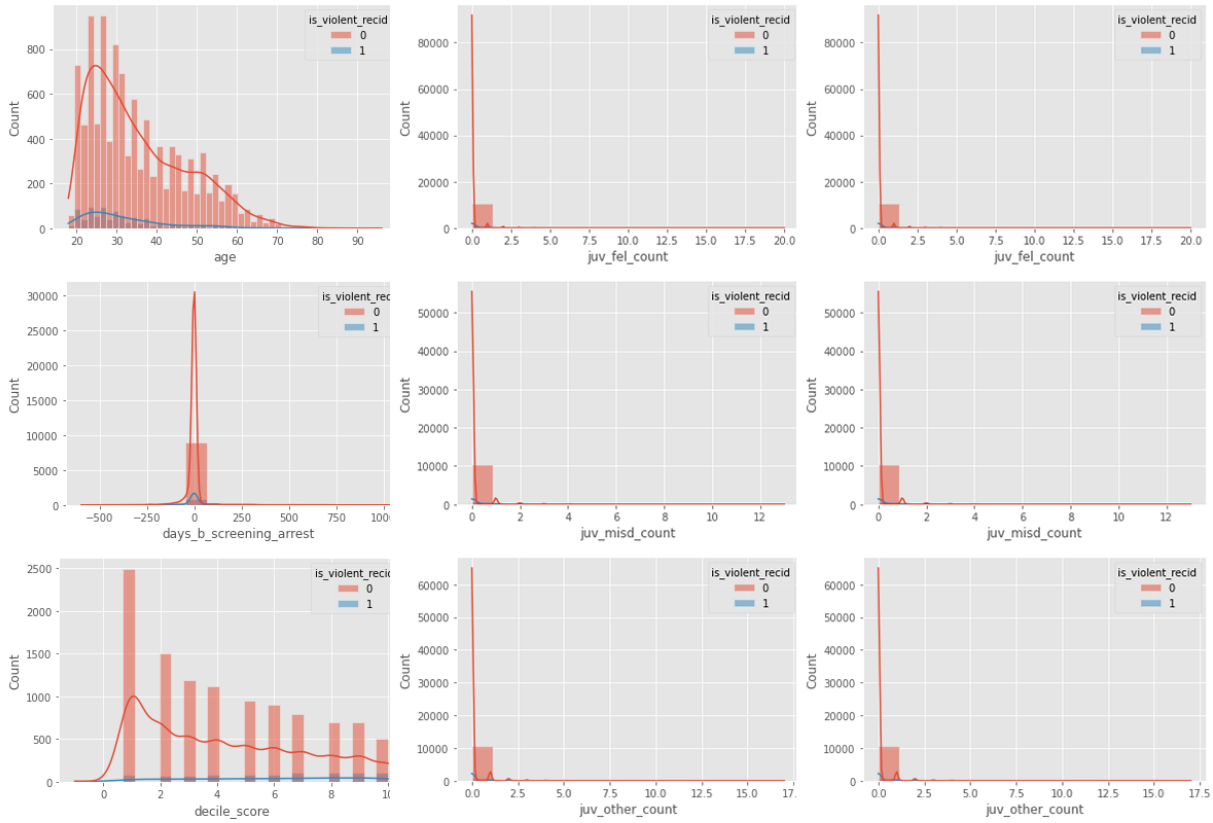


Figure 6. *Numerical Attribute Distributions Respect to* `is_violent_recid`

Non numerical attributes

About the other attributes, we have 31 non numerical attributes. Most of them are text descriptions, like `name`, `first`, `last`, `c_charge_desc`, `r_charge_desc` so they are useless for our analysis. Other attributes are identification codes (such as `id`, `c_charge_degree`, `vr_charge_degree`) and some others have a unique value (like `type_of_assessment` and `v_type_of_assessment`) . The majority of the other non numerical attributes are dates, such as `dob`, `compas_screening_date`, `c_jail_in`, `c_jail_out` and more.

The most relative attributes are `sex`, `age_cat` and `race` (that we rename in `ethnicity` for moral reasons). Regarding the `decile_score` we decided to maintain the attributes `score_text` and `v_score_text` in order to increase readability. We transformed these attributes with the function `astype('category')` and we analyzed their distribution in the following plots and histograms (for completeness we plotted also

the class labels attributes):

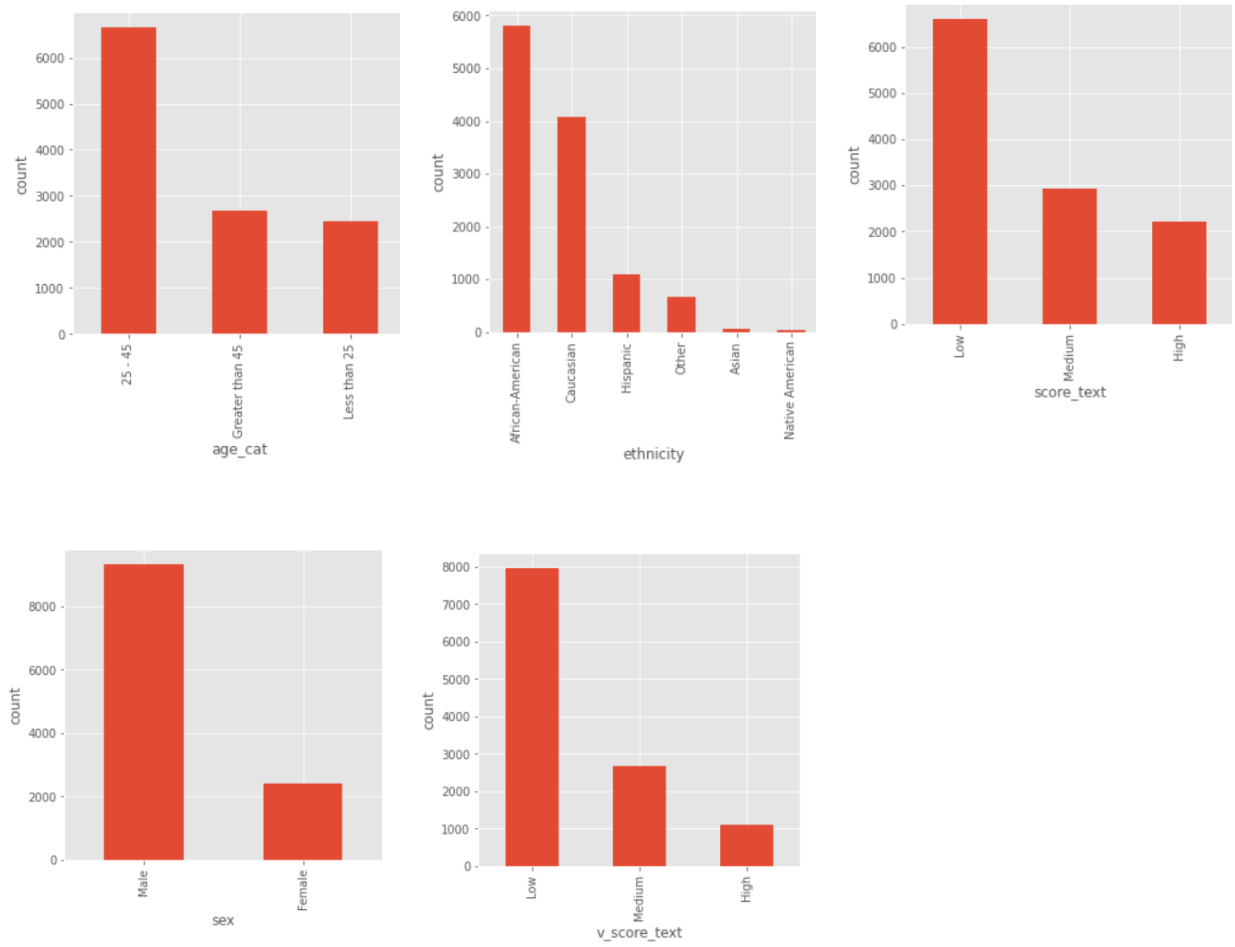
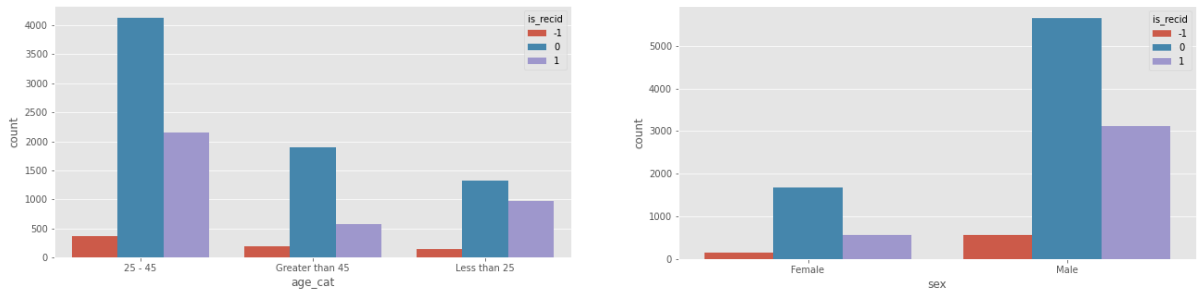


Figure 7. *Plots of the categorical attributes*

The following plots are important for the Data Understanding phase. The first two show the correlation between the categorical attributes and the two class labels:



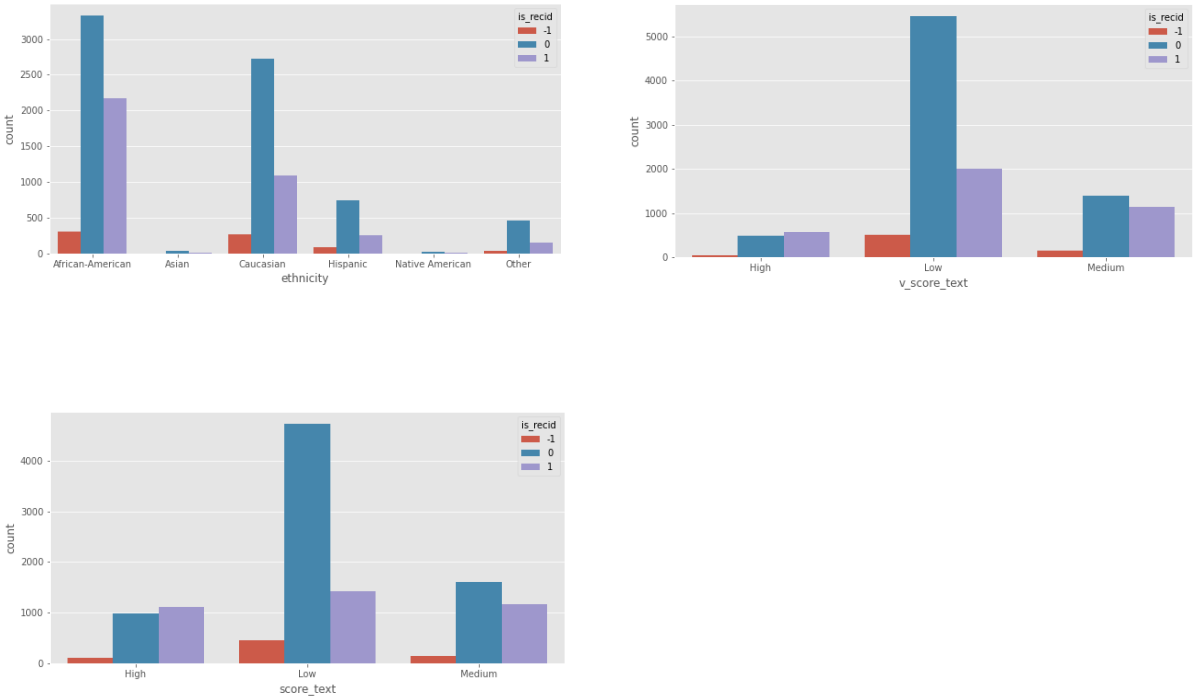
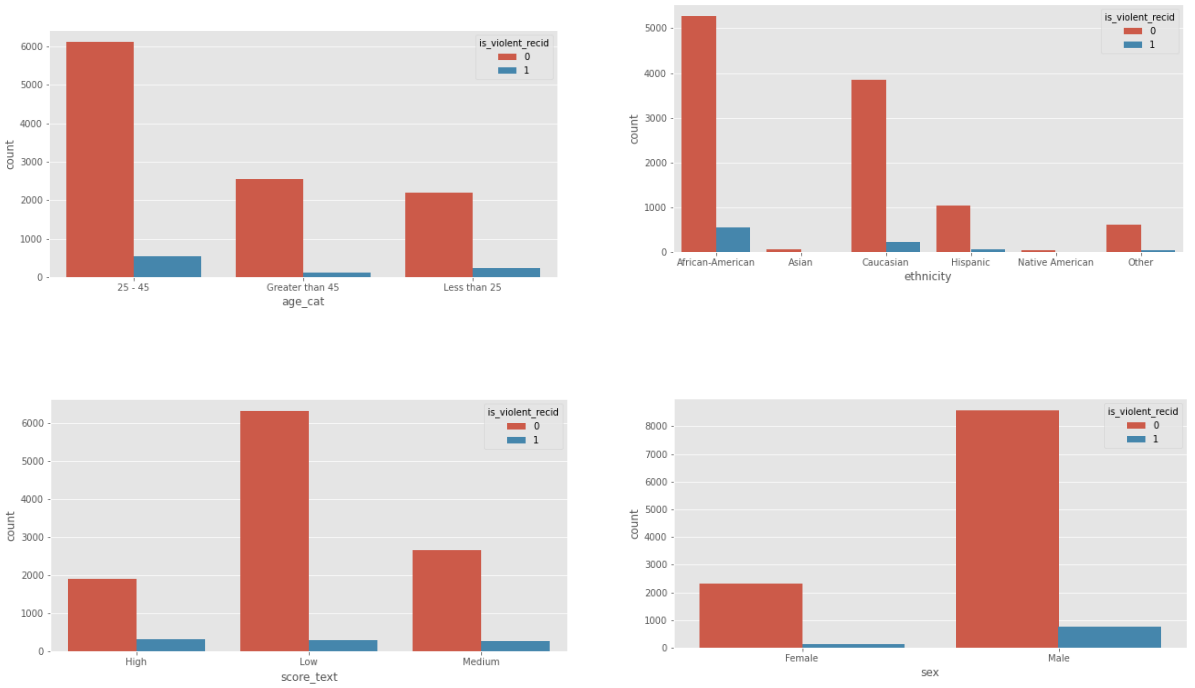


Figure 8. *Correlation between categorical attributes and* `is_recid`



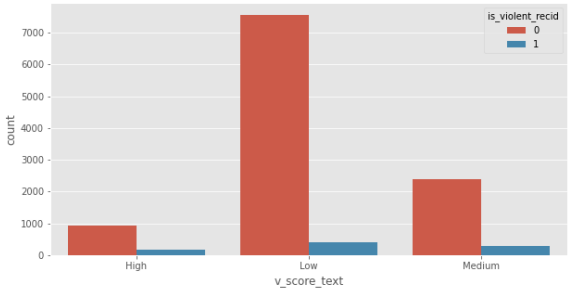
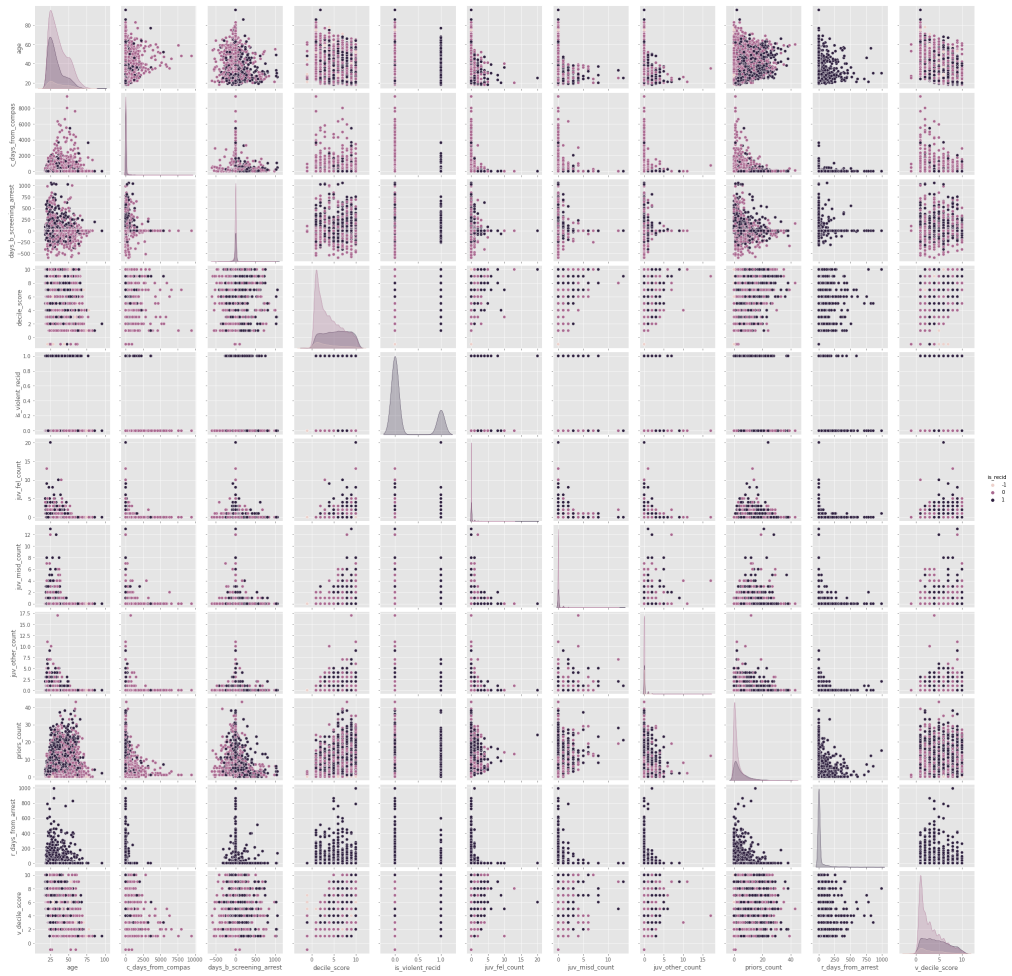


Figure 9. *Correlation between categorical attributes and* `is_violent_recid`

Histograms, bar plots and scatter plots according to the class label

Before passing to the Data Quality Report, we want to show some useful plots that will summarize the relation between the class labels and the other attributes.



Plot 2. *Scatter Plot with class label* `is_recid`

Plot 3. *Scatter plot with class label* `is_violent_recid`

Plot 4. *Correlation Matrix*

Attributes regarding dates

The attributes that represent dates are the following:

- dob
- compas_screening_date
- c_jail_in
- c_jail_out
- c_offense_date
- c_arrest_date
- r_offense_date
- r_jail_in
- r_jail_out
- vr_offense_date
- v_screening_date
- screening_date

Since we noticed a high variability for these attributes, we had some problems in plotting them in a sustainable time range. So, we decided to show the `values_count()` and their statistics description in order to complete our Data Understanding report.

Attribute name	values_count() result	Attribute name	values_count() result
dob	count 7800.000000 mean 1.507308 std 0.822150 min 1.000000 25% 1.000000 50% 1.000000 75% 2.000000 max 6.000000	c_arrest_date	count 802.000000 mean 2.316708 std 1.641212 min 1.000000 25% 1.000000 50% 2.000000 75% 3.000000 max 9.000000
compas_screening_date	count 704.000000 mean 16.700284 std 6.775800 min 1.000000 25% 12.000000 50% 16.000000 75% 21.000000 max 39.000000	r_offense_date	count 1090.000000 mean 3.397248 std 1.957536 min 1.000000 25% 2.000000 50% 3.000000 75% 4.000000 max 12.000000
c_jail_in	count 10577.0 mean 1.0 std 0.0 min 1.0 25% 1.0 50% 1.0 75% 1.0 max 1.0	r_jail_in	count 984.000000 mean 2.500000 std 1.493288 min 1.000000 25% 1.000000 50% 2.000000 75% 3.000000 max 9.000000
c_jail_out	count 10517.000000 mean 1.005705 std 0.090252 min 1.000000 25% 1.000000 50% 1.000000 75% 1.000000 max 4.000000	r_jail_out	count 953.000000 mean 2.581322 std 1.596328 min 1.000000 25% 1.000000 50% 2.000000 75% 3.000000 max 10.000000
c_offense_date	count 1036.000000 mean 8.838803 std 6.645770 min 1.000000 25% 1.000000 50% 9.000000 75% 14.000000 max 29.000000	vr_offense_date	count 599.000000 mean 1.472454 std 0.777286 min 1.000000 25% 1.000000 50% 1.000000 75% 2.000000 max 6.000000

v_screening_date	count 704.000000 mean 16.700284 std 6.775800 min 1.000000 25% 12.000000 50% 16.000000 75% 21.000000 max 39.000000	screening_date	count 704.000000 mean 16.700284 std 6.775800 min 1.000000 25% 12.000000 50% 16.000000 75% 21.000000 max 39.000000

Table 4. *Count values for the attributes regarding dates*

Data Quality

Now we will present our Data Quality Report. According to the literature and the previous steps of Data Understanding we can describe and report the main issues of the dataset.

Missing values

The following table shows the number and the percentage of null values (>0%). Also, we highlighted a high percentage of missing values in orange and red.

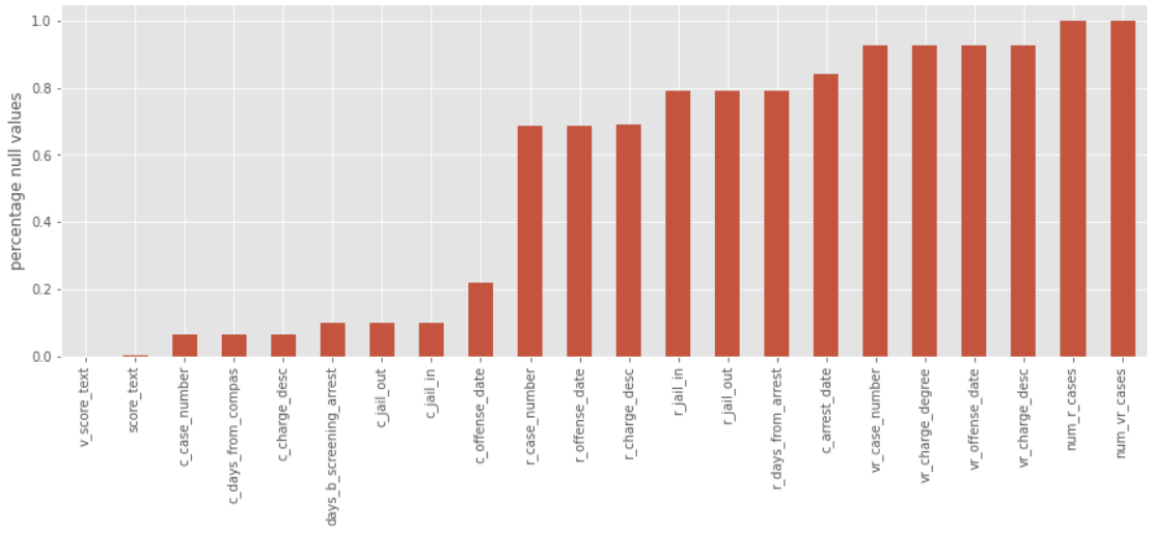
Attribute name	NULL values	Percentage
days_b_screening_arrest	1180	10%
c_jail_in	1180	10%
c_jail_out	1180	10%
c_case_number	742	6%
c_offense_date	2600	22%
c_arrest_date	9899	84%
c_days_from_compas	742	6%
c_charge_desc	749	6%
num_r_cases	11757	100%
r_case_number	8054	69%
r_days_from_arrest	9297	79%
r_offense_date	8054	69%
r_charge_desc	8114	69%

r_jail_in	9297	79%
r_jail_out	9297	79%
num_vr_cases	11757	100%
vr_case_number	10875	92%
vr_charge_degree	10875	92%
vr_offense_date	10875	92%
vr_charge_desc	10875	92%
score_text	15	0.13%
v_score_text	5	0.04%

Table 5. *Missing values, count and percentage.*

By analyzing the percentage of NULL values for each attribute, we intuitively saw that `num_r_cases` and `num_vr_cases` are useless for our analysis because they are totally NULL.

We also present a graphical representation with the following bar plot:



Plot 5. *Null values*

Duplicated attributes

Another useless attribute is `decile_score.1` cause it is a duplicate, so we did not consider it for our analysis.

Incorrect values

As suggested by the article "How We Analyzed the COMPAS Recidivism Algorithm", sometimes people's names or dates of birth were incorrectly entered in some records, which led to incorrect matches between an individual's COMPAS score and his or her criminal records. We attempted to determine how many records were affected. In a random sample of 400 cases, we found an error rate of 3.75 percent (CI: +/- 1.8 percent).

Other considerations

The attributes `is_recid` and `is_violent_recid` are our class labels and even if we are working on a binary classification task, we did not transform them into boolean attributes since their possible values are only `0` and `1`. In the next steps we will erase the record with values `-1` (assuming that this value indicates an unknown value).

Another consideration is about the attribute `race` that we renamed into `ethnicity` for moral reasons.

DATA PREPARATION

Data Set Description

This is an important phase in which we can apply strategies and mechanisms in order to select, clean and transform our data. Like the previous phases we used, again, the `Pandas` library to reach this goal and so `Python` as language.

As shown before, the dataset contains some inconsistencies, missing values and not significant attributes for the scope of our analysis.

Data Selection

Based on the previous steps done in the **Data Understanding** phase, we identified the most relevant information to reach our goal. For example, we decided to exclude some data like the `dob` attribute because it is inconsistent respect to the defendant's current age, reported in the `age` attribute, and also the `name` attribute for the same reason as suggested in the "How We Analyzed the COMPAS Recidivism Algorithm" article.

Also, we decided to delete the attributes with a high percentage of null values (greater than 50%) reported in the **Table 2** and high variability (like the `id` and other attributes about dates).

The following table shows the list of excluded and included attributes:

Included attributes
• `age_cat` • `c_offense_date` • `is_recid` • `is_violent_recid` • `r_offense_date` • `race` • `sex` • `score_text` • `v_score_text` • `vr_offense_date`

Table 6. *List of the attributes.*

Data Cleaning

To clean the dataset we used the function `columns.difference()` from `pandas`. We

reduced the shape from 47 attributes to 10 attributes. Noting that some values for the attribute `is_recid` are equal to "-1", we decided to drop these records by assuming that it stands for an unknown value.

We decided to not binarize the attributes `is_recid` and `is_violent_recid` since the only possible values are `0` and `1`, even if we are working on a binary classification problem.

Data Construction

We did not add new attributes or new rows in our dataset, there was no need for the analysis that we are going to do.

As we can see from the Jupyter Notebook, there are still missing values but we maintained that to make a new dataset to work with dates only for rows with `is_recid=1` and `is_violent_recid=1`. In this way it will be easy to predict how many days pass to become a recidivist.

Data Integration

We did not integrate or merge anything in our dataset, there was no need for the analysis that we are going to do.

Reformatted Data

In the previous phase we already renamed the attribute `race` in `ethnicity` for ethical reasons. As a Data Preparation step we decided to rename the values of the attribute `age_cat` to improve readability. In particular:

- `Less than 25` becomes `young`;
- `25-45` becomes `adult`;
- `Greater than 45` becomes `senior`.

MODELING

After the Data Understanding and the Data Preparation steps, we are ready to proceed with the Modeling phase. In this part we worked on several models design, we measured their performance based on accuracy, precision, recall and F-measure.

The following table describes these measures:

Measure Name	Description
Accuracy	the number of all the well-predicted observations over the cardinality of the dataset
Precision	the confidence of a model
Recall	the coverage of a model
F-measure	the harmonic mean of precision and recall

***Table** 7. Description of the metrics.*

Modeling Technique Selection

Since we worked on a binary classification task and since the final dataset contains mainly categorical attributes, some of the most common algorithms used to solve this type of problem are the following:

- Naïve Bayes Classifier
- Decision Tree Classifier
- K-Nearest-Neighbors Classifier
- Random Forest Classifier
- AdaBoost Classifier

Naïve Bayes Classifier

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Decision Tree Classifier

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node

(terminal node) holds a class label.

K-Nearest-Neighbors Classifier

KNN is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data.

Random Forest Classifier

The Random forest is a supervised Machine learning algorithm used for classification, regression, and other tasks using decision trees. The Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then it collects the votes from different decision trees to decide the final prediction.

AdaBoost Classifier

Ada-boost classifier combines weak classifier algorithms to form strong classifiers. A single algorithm may classify the objects poorly. But if we combine multiple classifiers with a selection of training sets at every iteration and assign the right amount of weight in final voting, we can have a good accuracy score for the overall classifier.

Test Design

We splitted into Training Set and Test Set, respectively with a percentage of 75 and 25. Then we performed Undersampling or Oversampling in order to balance the recid dataset and the violent recid dataset. In the first case we chose *Undersampling* because we already had too many records, so we decided to not add more. Instead, in the second one we wanted to perform the models with more data and records, so we chose *Oversampling* procedure.

Build Model

To find out the best parameters for each model's algorithm we used a function called `GridSearchCV`. In particular this function implements a "fit" and a "score" method. It also implements "score_samples", "predict", "predict_proba", "decision_function", "transform" and "inverse_transform" if they are implemented in the estimator used. The parameters of the estimator used to apply these methods are optimized by cross-validated grid-search over a parameter grid.

Model Assessment

After having found the parameters, we fitted the model and, based on the predictions set, we got the values of the quality measure: *accuracy*, *recall*, *precision* and *F-measure*. We decided to look at *F-measure* to select the best model for our dataset.

This table shows the output of the measures:

Model	**Output** recid dataset	**Output** violent recid dataset
Naïve Bayes	Accuracy: 0.5934 Recall: 0.6670 Precision: 0.4315 F-measure: 0.5240	Accuracy: 0.5929 Recall: 0.3805 Precision: 0.2663 F-measure: 0.3133
Decision Tree	Accuracy: 0.6402 Recall: 0.5816 Precision: 0.4707 F-measure: 0.5203	Accuracy: 0.5702 Recall: 0.3850 Precision: 0.2514 F-measure: 0.3042
KNN	Accuracy: 0.3442 Recall: 0.9903 Precision: 0.3374 F-measure: 0.5033	Accuracy: 0.6771 Recall: 0.1593 Precision: 0.2483 F-measure: 0.1941
Random Forest	Accuracy: 0.6380 Recall: 0.6151 Precision: 0.4699 F-measure: 0.5328	Accuracy: 0.5713 Recall: 0.3850 Precision: 0.2522 F-measure: 0.3047
AdaBoost	Accuracy: 0.6369 Recall: 0.5838 Precision: 0.4671 F-measure: 0.5190	Accuracy: 0.6199 Recall: 0.3584 Precision: 0.2812 F-measure: 0.3152

Table 8. *Metrics of the models.*

EVALUATION

Results Evaluation

Based on our analysis, regarding the main goal, the best model appears to be *Random Forest Classifier* for the recid dataset and *AdaBoost Classifier* for the violent recid dataset. We can see from the Notebook (Modeling phase) that the *Random Forest* had the highest *F-measure* equal to 0.5328, while the *AdaBoost* has an *F-measure* equal to 0.3152.

This following figures show the two confusion Matrix:

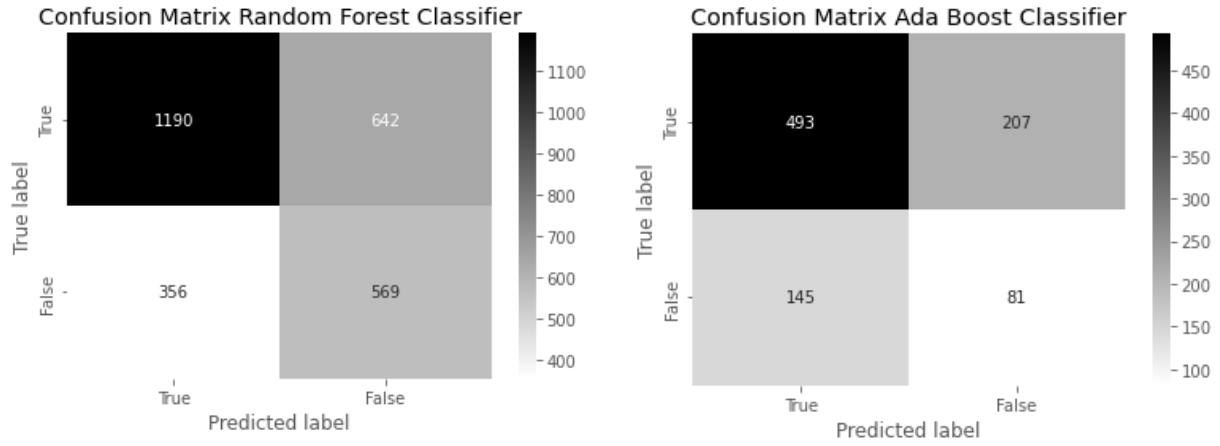


Figure 10. *Confusion Matrix Random Forest (recid dataset) and Confusion Matrix AdaBoost (violent recid dataset)*

The *ROC Curves* show that the best model according to the higher value of area under the curve is the *AdaBoost* for the recid dataset (0.67 vs 0.66 of the *Random Forest*) and both the *Naïve Bayes* and the *AdaBoost* (0.64) for the violent recid dataset.

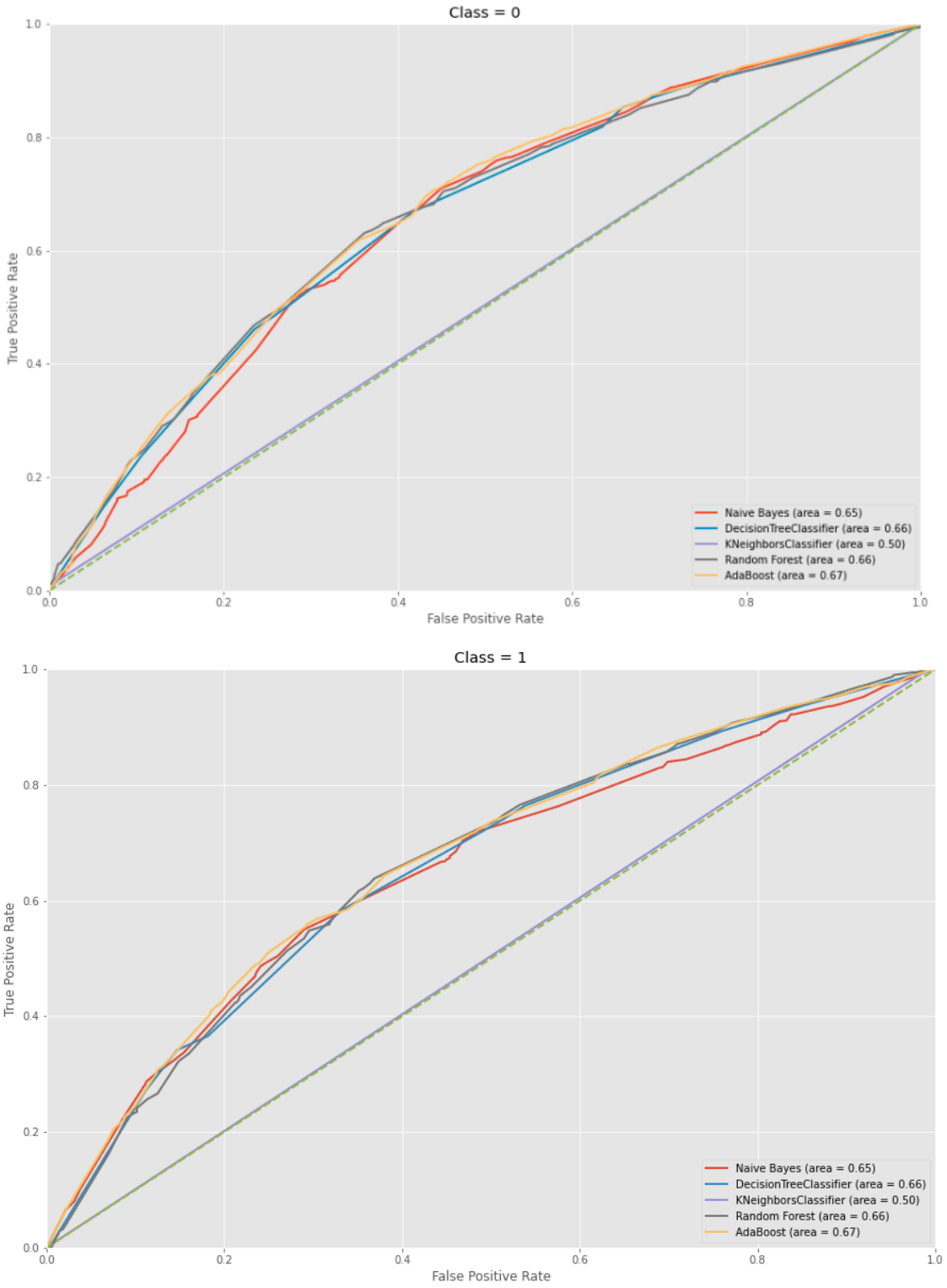


Figure 11. *ROC Curve for the recid dataset*

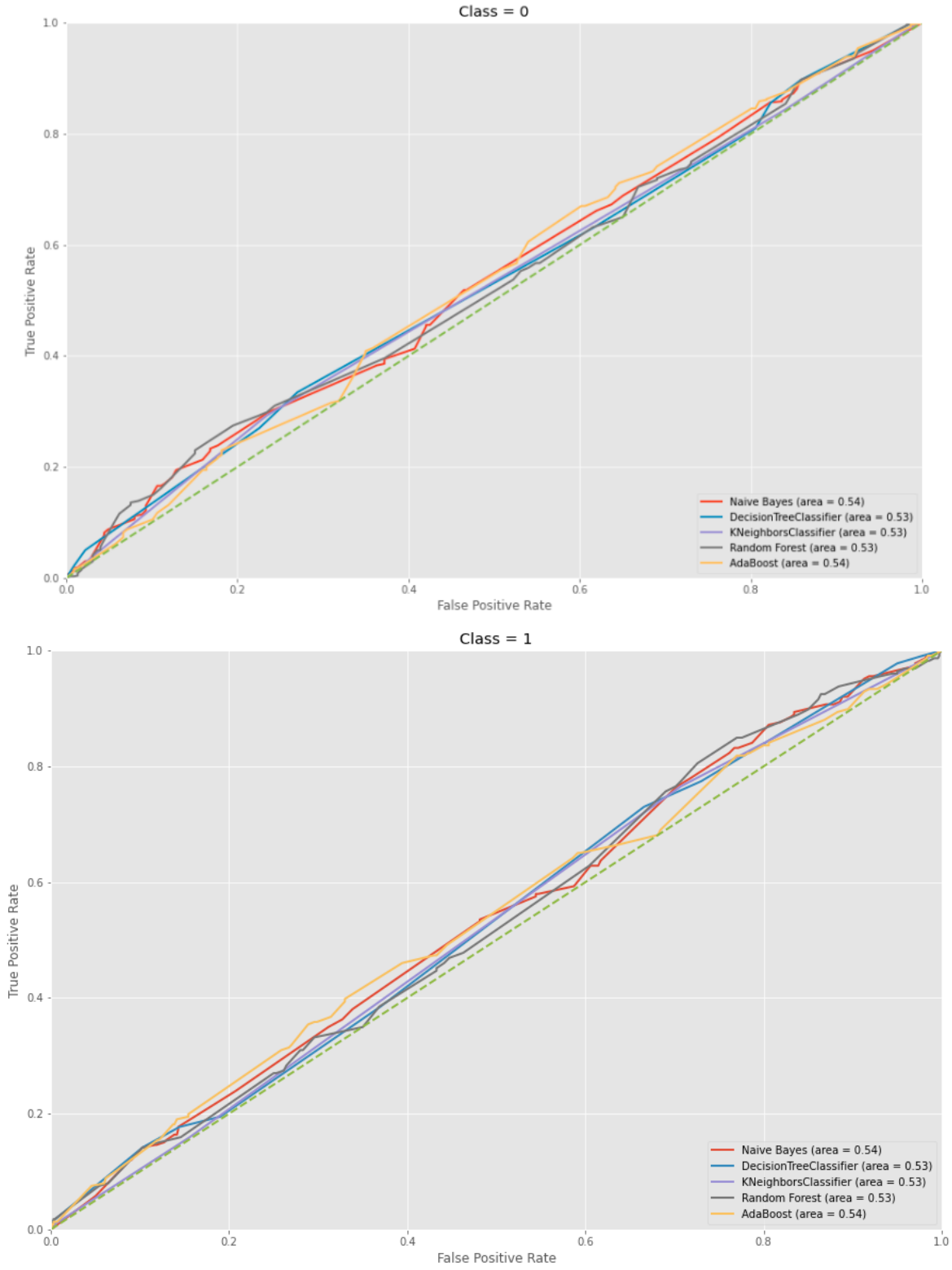


Figure 12. *ROC Curve for the violent recid dataset*

Since we noticed this inconsistency between the *F-measure* value and the *ROC Curve*, we decided to compare the Classification Reports for the *Random Forest* and the *AdaBoost Classifiers* regarding the recid dataset:

Classification Report - Random Forest Classifier

	precision	recall	f1-score	support
0	0.77	0.65	0.70	1832
1	0.47	0.62	0.53	925
accuracy			0.64	2757
macro avg	0.62	0.63	0.62	2757
weighted avg	0.67	0.64	0.65	2757

Classification Report - Ada Boost Classifier

	precision	recall	f1-score	support
0	0.76	0.66	0.71	1832
1	0.47	0.58	0.52	925
accuracy			0.64	2757
macro avg	0.61	0.62	0.61	2757
weighted avg	0.66	0.64	0.64	2757

Figure 13. *Classification reports (recid dataset)*

As is possible to see, although the AUC is higher for *AdaBoost Classifier*, if we look at the *f1-score* value, the best model for Classification equal to 1 (recidivist) is *Random Forest*.

In addition, for completeness, we compared the Classification Reports of the *AdaBoost* and the *Naïve Bayes Classifiers* regarding the violent recid dataset:

Classification Report - Ada Boost Classifier

	precision	recall	f1-score	support
0	0.77	0.70	0.74	700
1	0.28	0.36	0.32	226
accuracy			0.62	926
macro avg	0.53	0.53	0.53	926
weighted avg	0.65	0.62	0.63	926

Classification Report - Naive Bayes Classifier

	precision	recall	f1-score	support
0	0.77	0.66	0.71	700
1	0.27	0.38	0.31	226
accuracy			0.59	926
macro avg	0.52	0.52	0.51	926
weighted avg	0.65	0.59	0.61	926

Figure 14. *Classification reports (violent recid dataset)*

Also in this case, although the AUC is the same for the *AdaBoost* and the *Naïve Bayes Classifiers*, if we look at the *f1-score* value, the best model for Classification is the *AdaBoost Classifier*.

Regarding the secondary goal of predicting the number of days between the date of the first crime and the date of the recidivist or violent recidivist offense, our response is the attribute `dates_diff_in_days`. We worked on a linear regression model but we didn't find any linear correlation.

The following table shows the result of the coefficients and their interpretation for each

target variable (after the addition of the dummies):

Coefficients - dt_date_r	Interpretation - dt_date_r
<table><tr><th></th><th>coefficient</th></tr><tr><td>age_cat_senior</td><td>47.857954</td></tr><tr><td>age_cat_young</td><td>-17.088324</td></tr><tr><td>ethnicity_Asian</td><td>89.822557</td></tr><tr><td>ethnicity_Caucasian</td><td>-27.749853</td></tr><tr><td>ethnicity_Hispanic</td><td>-4.900140</td></tr><tr><td>ethnicity_Native American</td><td>-0.566236</td></tr><tr><td>ethnicity_Other</td><td>-6.337896</td></tr><tr><td>score_text_Low</td><td>26.047265</td></tr><tr><td>score_text_Medium</td><td>15.459139</td></tr><tr><td>sex_Male</td><td>-0.796888</td></tr></table>	• the average expected date_diff_in_days for a senior recidivist is +47.85 respect to an adult recidivist • the average expected date_diff_in_days for a young recidivist is -17.08 respect to an adult recidivist • the average expected date_diff_in_days for an Asian recidivist is +89.82 respect to an African-American recidivist • the average expected date_diff_in_days for an Caucasian recidivist is -27.74 respect to an African-American recidivist • the average expected date_diff_in_days for an Hispanic recidivist is -4.90 respect to an African-American recidivist • the average expected date_diff_in_days for an Native American recidivist is -0.56 respect to an African-American recidivist • the average expected date_diff_in_days for a recidivist with a Low decile score is +26.05 respect to a recidivist with a High decile score • the average expected date_diff_in_days for a recidivist with a Medium decile score is +15.05 respect to a recidivist with a High decile score • the average expected date_diff_in_days for a Male recidivist is -0.79 respect to a Female recidivist
Coefficients - dt_date_v	**Interpretation - dt_date_v**
<table><tr><th></th><th>coefficient</th></tr><tr><td>age_cat_senior</td><td>124.789211</td></tr><tr><td>age_cat_young</td><td>-8.889943</td></tr><tr><td>ethnicity_Asian</td><td>-42.569574</td></tr><tr><td>ethnicity_Caucasian</td><td>-75.149561</td></tr><tr><td>ethnicity_Hispanic</td><td>45.466648</td></tr><tr><td>ethnicity_Native American</td><td>364.672239</td></tr><tr><td>ethnicity_Other</td><td>-84.999001</td></tr><tr><td>v_score_text_Low</td><td>-32.905414</td></tr><tr><td>v_score_text_Medium</td><td>23.736487</td></tr><tr><td>sex_Male</td><td>-45.110255</td></tr></table>	• the average expected date_diff_in_days for a senior violent recidivist is +124.79 respect to an adult violent recidivist • the average expected date_diff_in_days for a young violent recidivist is -8.88 respect to an adult violent recidivist • the average expected date_diff_in_days for an Asian violent recidivist is +42.57 respect to an African-American violent recidivist • the average expected date_diff_in_days for an Caucasian violent recidivist is -75.14 respect to an African-American violent recidivist • the average expected date_diff_in_days for an Hispanic violent recidivist is +45.47 respect to an African-American violent recidivist • the average expected date_diff_in_days for an Native American violent recidivist is +364.67 respect to an African-American violent recidivist • the average expected date_diff_in_days for a

	violent recidivist with a Low violent decile score is -32.90 respect to a violent recidivist with a High violent decile score • the average expected date_diff_in_days for a violent recidivist with a Medium violent decile score is +23.73 respect to a violent recidivist with a High violent decile score • the average expected date_diff_in_days for a Male violent recidivist is -45.11 respect to a Female violent recidivist

Table 9. *Coefficients of the attributes and their meaning*

We also performed the residual plots for each attribute. It is possible to find them in the attached Jupyter Notebook (Modeling Phase).

Process Review

In this first step of Modeling and Evaluation we did not find a high level of accuracy (>0.90). Moreover, we did not find a regression model that fits well our dataset about the number of days, so it would be interesting to go deeper into this process of model finding.

We worked on different types of models, neither too many nor too few. Maybe we did not deepen the phase of searching for the best models' parameters.

Possibile Actions Decision

As next steps we want to improve the phase of parameter search and maybe consider also other classification models, in order to find a better model with a higher value of accuracy, specially in the study case of predictions of number of days before the recid or the violent crime.

DEPLOYMENT

As the last step, we will present our deployment phase (documentation and Jupyter Notebook) through a visual presentation.

CONCLUSION

All of the choices made are focused on our initial goal, defined in the Business Understanding section. Moreover, since this dataset is not well studied in literature or online, we wanted to analyze it in a simple and readable way in order to erase all noise

records and to decrease the bias of ethnicity attribute .

Even though we worked following step by step the CRISP-DM Methodology, the high biases present in the dataset does not allow us to reach a good level of accuracy. But this represents a deeper social issue based on prejudice, intolerance or plain ignorance. In that case we can not do more than hope that these social biases will decrease day by day and do our best to reach this community goal.

The conclusion we reached is that with this dataset it is possible to reach a discrete level of accuracy and goodness of predictions, but to improve the metrics it will be necessary to add more significant data and records.

RESOURCES

1. COMPAS (software) - Wikipedia
2. Practitioner-s-Guide-to-COMPAS-Core.pdf
3. Compas Scores - Kaggle
4. GitHub - propublica/compas-analysis: Data and analysis for 'Machine Bias'
5. CRISP-DM - Data Science Process Alliance
6. GeeksforGeeks - Classifiers definition
7. Linear Regression in Scikit-Learn (sklearn): An Introduction • datagy
8. statsmodels v0.10.2 documentation