ICS5510 Assignment

Carmel Gafa

***Abstract*—TODO: abstract here**
***Index Terms*—TODO: keywords**

I. Introduction

This exercise will explore the well-known COMPAS dataset using several machine-learning techniques. We will also look into the ethical implications of predictive risk assessment models. We have taken the opportunity of this study to implement some of the techniques discussed in ICS5510, like imputation and encoding, to help in data preparation, linear regression, neural networks and others as the tools used for prediction.

Wherever possible, we preferred the manual implementation of some of the steps over the functionality available in popular Python libraries to appreciate the techniques implemented more thoroughly.

A. History of the COMPAS tool

The COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) dataset and tool have a controversial history rooted in its use for assessing the likelihood of recidivism among criminal defendants. Developed by Northpointe, COMPAS gained widespread adoption in the U.S. judicial system for pretrial risk assessments and sentencing decisions. This tool is helpful in various stages of the criminal justice process, including bail, sentencing, and parole decisions.

However, in 2016, an investigative report by ProPublica revealed significant racial biases in the tool's predictions. The report found that COMPAS disproportionately labelled Black defendants as high risk for reoffending while underestimating the risk for white defendants, even when both groups had similar criminal histories. This revelation sparked a broader debate about using algorithmic tools in criminal justice and their transparency and fairness.

The COMPAS tool has not been directly the subject of lawsuits, but its use in judicial decisions has led to legal challenges. For instance, in State v. Loomis (2016), the Wisconsin Supreme Court upheld using COMPAS in sentencing. However, judges must be informed about its limitations, particularly its proprietary nature and potential biases. The case highlighted the broader tension between the utility of predictive algorithms and their application's need for accountability and fairness.

B. The COMPAS dataset

The dataset that originates from the COMPAS tool is widely used in criminology and machine learning studies.

The dataset contains attributes such as demographic information, prior charges, juvenile records, and risk scores, including the widely analysed decile score, which categorises individuals into ten different risk groups.

The decile score is a critical feature, assigning a numerical value to an individual's likelihood of reoffending. Other important features include the number of prior offences `priors_count` and the type of offence `c_charge_degree`, provide context for these predictions. At the same time, the label `two_year_recid` indicates whether an individual reoffended within two years of their COMPAS assessment.

While the dataset has been instrumental in research aimed at understanding and improving risk prediction models, it has also been the subject of extensive scrutiny due to its implications for fairness and equity in the justice system. A couple of thoughts resulting from this scrutiny include:

- Multiple studies, including the influential ProPublica investigation in 2016, have highlighted racial disparities in the COMPAS predictions. African-American defendants were found to be nearly twice as likely as Caucasian defendants to be labelled as high-risk for recidivism but not reoffend. Conversely, Caucasian defendants were more likely to be classified as low-risk but later reoffend, raising concerns about systemic bias embedded in the algorithm, which could exacerbate existing inequalities in the justice system.
- The COMPAS tool operates as a proprietary black-box model, meaning its internal workings and feature weights are not disclosed to the public or even to the defendants it evaluates. This lack of transparency prevents meaningful scrutiny and accountability, leaving users unable to fully understand or challenge the tool's predictions.
- The COMPAS algorithm relies on historical criminal justice data, which may reflect social and systemic biases. For example, law enforcement practices that can result in sentencing disparities can all influence the patterns observed in the data. Using such data as input, the COMPAS tool risks perpetuating these biases into an electronic tool.
- Some features in the COMPAS dataset, such as age and criminal history, are static and cannot change over time, as this data is based on the date of the COMPAS assessment. We can argue that these features in risk predictions without considering the period after the COMPAS assessment undermines the potential for individuals to reform and leads to insensible punitive outcomes.
- The ethical implications of using predictive algorithms

in high-stakes decisions, such as sentencing and parole, constitute a significant area of concern. The potential for false positives can lead to unjustly harsher treatment, while false negatives can impact public safety.
- The dataset available for research purposes is a reduced version of the original COMPAS data, with several features anonymised or removed. Missing important data introduces limitations for academic studies aiming to replicate or validate the findings from real-world COMPAS applications.

The criticism of the COMPAS tool emphasises the challenges of deploying machine learning systems in sensitive domains like justice. These challenges are not unique to COMPAS but highlight broader issues in applying algorithmic decision-making tools in socially important contexts. They highlight the need for transparency, fairness-aware modelling techniques, and careful ethical evaluations when designing and implementing such tools.

C. Objectives of this work

The main objectives of this study are:

Analyse the COMPAS dataset and its predictions.
Prepare the dataset for machine learning through cleaning, transformation, and feature engineering.
Train and evaluate machine learning models for ethical analysis.
Investigate potential biases and ethical implications in predictions.

II. Downloading and first look at the dataset

The COMPAS dataset used in this study is publicly available through ProPublica's GitHub repository. This repository contains the dataset and other assets used by ProPublica to investigate the biases present in the COMPAS risk assessment tool.

The file chosen for this analysis is **`compas-scores-two-years.csv`**, as it provides the cleanest and most relevant data for general recidivism prediction. This CSV file contains the key data required for our study, including several attributes related to demographics, criminal history, COMPAS risk scores, and the two-year recidivism outcomes that are important for exploring the predictive capabilities and the ethical implications of machine learning models in the context of recidivism prediction.

The dataset includes important information about individuals. Following an initial analysis, a list of the key fields in the dataset is below.

- Personal Information, includes attributes such as **`age`**, **`race`**, **`age_category`**, etc.
- Case and Event-Related Details are the fields prefixed with **`c_`** that provide a timeline and details of a person's interactions with the criminal justice system.
- Violence Risk Assessment are the fields prefixed with **`v_`** and are associated with the violence risk assessment in COMPAS. This dimension predicts violent recidivism risk.
- Case-Level Details for Violent Recidivism are the fields prefixed with **`vr_`**. These fields provide additional details specific to violent recidivism events.
- Juvenile Criminal Record are the fields prefixed with **`juv_`**. These fields capture information about an individual's juvenile criminal record, which is a key predictor of future adult criminal behaviour.
- Previous Charges and Severity can be deduced from fields such as **`priors_count`** and **`juv_`** fields.
- Additional fields, including **`r_charge_`**, **`r_offense_`**, **`vr_`** fields, **`c_charge_degree`**, and **`c_charge_desc`**, provide a broader perspective on criminal history and severity.
- Two-Year Recidivism, or the **`two_year_recid`** field in the COMPAS dataset, indicates whether an individual reoffended (recidivated) within two years of their initial assessment or release. This field is critical for evaluating the predictive accuracy of the COMPAS risk assessment tool.
- Decile Score is a standardized risk score in the COMPAS dataset. It categorizes an individual's likelihood of recidivism into ten equal groups (deciles) where 1 is the lowest risk, and 10 is the highest risk. Each decile represents approximately 10% of the sample when applied to a norm group.

Suppose that we observe the following data:

Field	Value
`priors_count`	5
`juv_felony_count`	2
`juv_misdemeanor_count`	3
`r_charge_degree`	Felony

We can interpret this as an individual who has five total prior charges, including:

- 2 juvenile felonies
- 3 juvenile misdemeanours
- The severity of previous charges includes felonies (**`r_charge_degree`**).

III. Preparing the data for further analysis and training

Before we can perform any analysis or apply machine learning techniques, it is important to pre-process and prepare the dataset so that we can handle missing values, encode categorical features, and split the data into training, testing, and validation sets. This step will produce a clean dataset for building accurate and unbiased models. The following steps outline the procedures to prepare the dataset for further analysis and training.

A. Initial look at data and missing values handling

The dataset has 7214 instances over 53 columns. The target of the dataset is **`decile_score`**, but the dataset also contains information about whether or not the person recidivised, most notably through the label **`two_year_recid`**.

The first step in data preparation is removing the features irrelevant to this exercise or with over 50% missing records. We removed all the COMPAS-administrative labels and additional recidivism information apart from **two_year_recid**, narrowing the dataset to 17 fields.

The difference between **c_jail_in** and **c_jail_out** was calculated into a new field, **days_in_jail** and the difference between **in_custody** and **out_custody**, in a new field, **days_in_custody**. We subsequently removed the features containing date information from the dataset, together with **days_in_custody**, as it contained no information. At this stage, the dataset contains thirteen features: eight numerical, four categorical, and one descriptive. It also contains two labels, decile_score, which we will treat as the leading label in this exercise and **two_year_recid**, which we are keeping to compare the prediction power of our models to the original one.

B. Imputation of missing data

While examining the resultant dataset, we noticed that **days_b_screening_arrest** has 6907 values that are not null. Whilst it is possible to eliminate the rows that contain the null values at this stage, we replaced the missing values using a KNN imputation technique by grouping the numeric values of this dataset so that we can calculate the missing values. We checked this process by plotting the distribution of **days_b_screening_arrest** before and after imputation to see if any variations occurred.

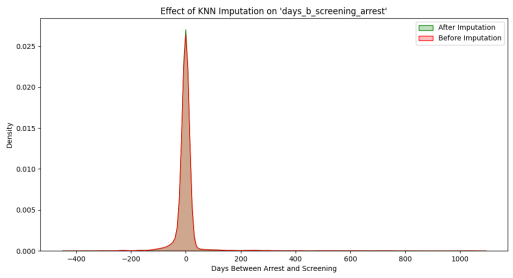

Fig. 1

At this stage, the dataset contains four categorical features that need encoding for machine learning algorithms. This section will focus on converting them into a numerical format using two encoding techniques. The categorical features and their values are listed below:

The following transformations are applied:

One-Hot Encoding on **sex**, **race**, and **c_charge_degree**, transforming them into binary columns.

Ordinal Encoding on **age_cat**. This encoding technique was preferred over one-hot in this case as it preserves order,thus respecting the inherent ranking of the category.

The original categorical columns were retained in the dataset for future use in the analysis steps.

Feature	Description	Unique Values
sex	Gender of the individual	[l]Male Female
race	Race of the individual	African-American Caucasian Hispanic Asian Native American Other
age_cat	Age category	Less than 25 25 - 45 Greater than 45
c_charge_degree	Degree of the criminal charge	F (Felony) M (Misdemeanor)

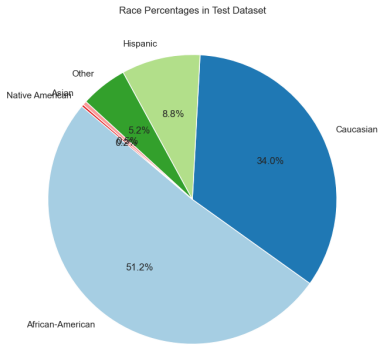

Fig. 2: Caption for the first image

C. Splitting the data into train, test and dev

A stratified shuffle split technique is preferred to create the train, test, and dev datasets whilst ensuring that the splits are proportional by **race**. In the first split, 80% Train and 20% Test are created, whilst in the Second split, The 20% Test is further divided into 10% Test and 10% Dev.

IV. Data Exploration and Visualisation

This section will examine the dataset in more detail to understand the patterns, distributions, and relationships. In this exercise, we will use more of the visual tools available through several Python libraries to identify potential biases, explore correlations between variables, and uncover insights that may influence the outcomes of predictive models.

A. Demographic analysis

We begin this analysis by segmenting the dataset based by race and gender.

By examining the racial composition of the dataset, we observe the following:

- Over half of the test dataset is composed of African-American individuals, suggesting that the dataset may be imbalanced, with a disproportionate representation of one racial group.

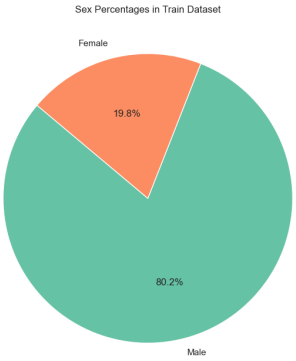

Fig. 3: Caption for the second image

- Asians and Native Americans each makeup only 0.2% of the dataset; this underrepresentation might raise some concerns as it may lead to challenges in statistical analysis or machine learning models. Such concerns include the lack of reliability or significance for these groups due to insufficient data.

Figure **??** also shows our dataset's male/female split, with females comprising only 19.8%. It is, therefore, evident that the female group is underrepresented, which can lead to biased models as models may overfit male patterns and underperform on females and misleading conclusions as insights derived might generalise poorly for the female subgroup.

B. Age distribution analysis

We used a boxplot to illustrate the age patterns across racial groups, helping to identify central values, spread, and any anomalies.

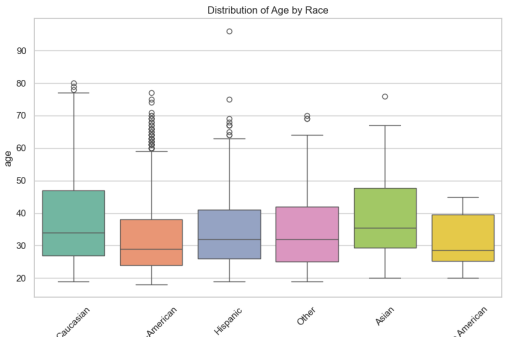

Fig. 4

Race	Median Age	Distribution Description
African-American	≈30 years	Concentrated in the 20–40 range, with a relatively narrow spread. We notice outliers above 60 years, indicating fewer older individuals.
Caucasian	≈40 years	Broader age range, from 20 to 70+ years. We notice more older individuals (upper outliers), making this group appear older on average.
Hispanic	≈30–35 years	Moderately broad spread, with most individuals between 20 and 50 years.
Asian	≈30 years	Narrow distribution, concentrated between 25 and 40 years. No outliers.
Native American	≈33 years	Very tight distribution, with all ages clustered closely around the median (little variability). It is important to note that this group accounts for a tiny portion of the population.
Other	≈35–40 years	Similar to Caucasians but with slightly fewer older individuals. The IQR shows a widespread.

C. Analysing distributions

Next, we plotted the distributions of all the features in our dataset; this is depicted in Figure 5 . From these histograms, we notice

- The age distribution shows a right-skewed pattern, with most individuals falling in the younger age ranges (20–40 years).
- There is a significant over-representation of certain racial groups, particularly African Americans, which could indicate potential biases in the dataset's sampling.
- Most individuals have zero juvenile felony counts, zero juvenile misdemeanour counts, and no recorded "other" juvenile offences, with each distribution rapidly declining for higher counts.
- A large proportion of individuals have a low number of prior offences, but there is a long tail indicating some individuals have a significant number of priors.
- Most individuals have relatively short jail durations, with a few experiencing significantly longer durations.
- The distribution of days between screening and arrest is clustered around zero, with few extreme outliers on both ends.
- The decile scores appear relatively evenly distributed, but slight patterns suggest clustering at specific score levels (e.g., lower decile scores are slightly more frequent).
- Two-year recidivism plot shows a near-equal distribution, indicating a balanced dataset for recidivism outcomes.

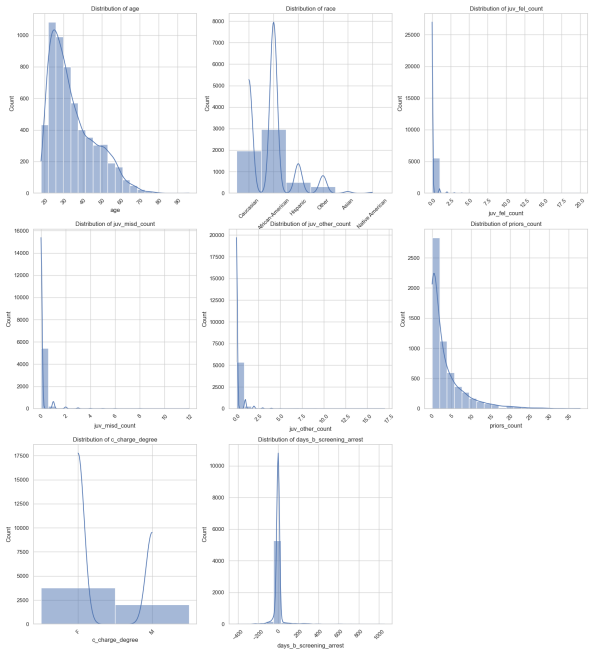

Fig. 5

D. Correlation analysis

The correlation matrix heatmap shown in Figure 6 was created to gather more insights and pinpoint the areas of high correlation. This plot raises a number of interesting observations, namely:

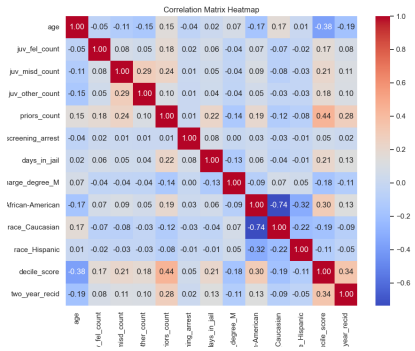

Fig. 6

- Individuals with more prior offences tend to have higher risk scores, as prior criminal behaviour is a key factor in risk assessment models. The number of prior offences also correlates positively with recidivism; individuals with more prior offences tend to re-offend more often.
- Older individuals tend to have lower risk scores, suggesting that age may be inversely related to the risk of recidivism, with younger individuals being assessed as higher risk. In addition, older individuals are also less likely to recidivate, supporting this general trend.
- Individuals with higher risk scores are likelier to recidivate within two years, suggesting that the risk score

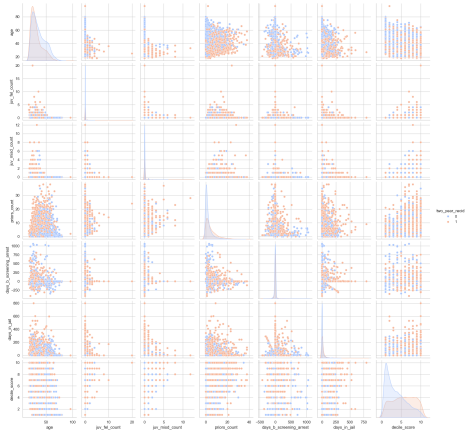

Fig. 7

(**decile_score**) is predictive to a certain extent of recidivism.
- A correlation of 0.30 indicates that being African-American is moderately associated with higher decile scores, raising potential concerns about racial bias in the scoring system. At the same time, Caucasian individuals correlate -0.19 and, therefore, are less likely to receive higher risk scores.
- The time spent in jail has only a small positive relationship with the likelihood of re-offending within two years
- Juvenile felony, misdemeanour, and other counts are positively correlated, indicating that individuals with one type of juvenile record will often have other types.

Figure 7 is a pair plot created to substantiate these observations further to show the relationships among the selected numerical features, with the recidivism outcome (**two_year_recid**) as the hue.

- **age** vs **priors_count**. Younger individuals tend to have fewer prior offences, but the prior count is scattered as age increases, showing that younger offenders tend to continue having problems with the judicial system.
- **age** vs **decile_score**. Older individuals tend to have lower decile scores. Younger individuals are associated with higher scores.
- **priors_count** vs **decile_score**. Positive trend: Higher priors count leads to higher decile scores, suggesting a strong correlation, indicating the tendency for offenders to be viewed as a risk community
- The distribution of **decile_score**. Two-year recidivists tend to cluster at higher decile scores (6–10 range). Non-recidivists are spread more evenly across scores.

E. Analysis of decile score relationship with race

The COMPAS dataset has been extensively discussed in recent years regarding the fairness of the scores assigned to individuals, especially considering the race component. Although the tool does not use race as one of the features in decile score prediction, there are concerns that race can be associated indirectly with other features. In this section, we

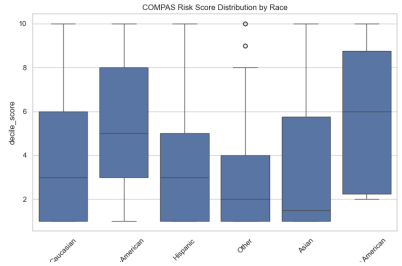

Fig. 8

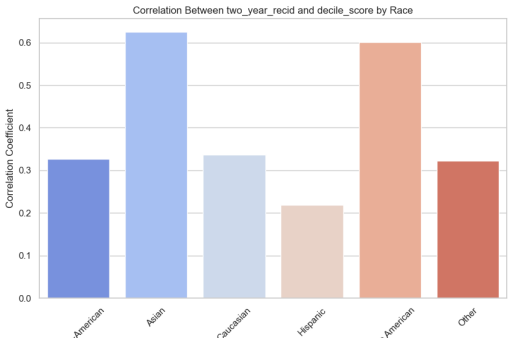

Fig. 9

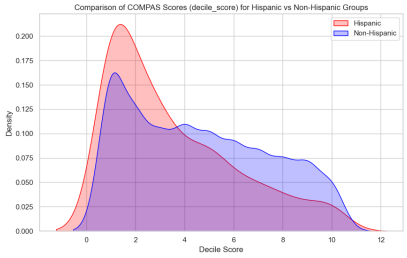

Fig. 10

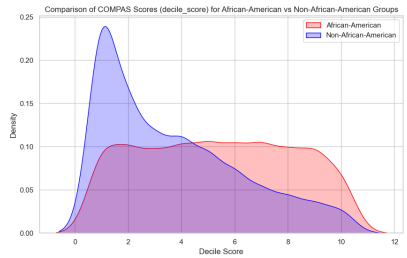

Fig. 11

will look at the data from the race point of view to see what patterns we can deduce from the dataset.

As a first step, we create a boxplot to compare the distribution of **decile_score** across different racial groups.

From this plot, we notice that African Americans have higher median scores and broader distributions, which suggests a potential bias in the COMPAS scoring system. In addition, Hispanics, Asians, and Other groups scored lower on average, which may indicate differences in the risk assessment process or underlying data inputs.

We then investigated the correlation of **decile_score** with **two_year_recid** by race:

- Asians and Native Americans show the highest correlations, indicating that, for these groups, COMPAS scores align more closely with observed recidivism outcomes. However, it is important to remember that these groups comprise a tiny percentage of the population.
- African-Americans, Caucasians, and Others have moderate correlations, so COMPAS scores are somewhat predictive for these groups but not as strongly as for Asians or Native Americans.
- The correlation between **two_year_recid** and **decile_score** for Hispanics is 0.22, the lowest among the groups, suggesting that for Hispanic individuals, the COMPAS scores are less predictive of actual recidivism outcomes than other racial groups. The low correlation for the Hispanic group is an area of concern because if the COMPAS scores do not accurately predict recidivism for this ethnicity, this could mean that the scoring system may overestimate or underestimate their actual risk, which could lead to misclassification of individuals, leading to potential unfair treatment, for example, harsher parole conditions, sentencing).

The distribution of **decile_score** for Hispanic individuals was then compared with that of other racial groups using kernel density plots (KDE).

The plots show that the peak density for Hispanic individuals occurs at a lower score of around 2. In contrast, non-Hispanic individuals have a flatter distribution extending to higher scores, albeit peaking in the same region, suggesting that Hispanic individuals are more likely to receive lower COMPAS scores than non-Hispanic individuals.

One has to note that we have previously seen that Hispanic individuals account for 8.8% of the population, which is significantly smaller compared to African-American and Caucasian groups. Although by any means not conclusive, these aspects raise questions about how risk scores are calibrated for underrepresented groups.

1) Analysis of the African-American sector: We created another density plot to compare the distribution of risk scores between African-American individuals and non-African-American individuals.

Here, we notice that African Americans are more likely to receive risk scores in the higher decile range, exceeding a value of six, than non-African Americans, suggesting that the COMPAS tool systematically assigns higher risk scores to African Americans. As noticed previously, non-African-American individuals show a strong peak at around a score of 2, with fewer individuals in this group receiving scores in the higher ranges.

This disparity in score distributions raises concerns about

potential bias in the COMPAS scoring system. African Americans appear to be disproportionately classified as higher risk, a factor which could impact downstream decisions such as sentencing or parole.

F. Robustness of **`decile_score`**

As a final analysis, we compare **`decile_score`** with **`two_year_recid`** to see how many high-risk individuals receded and how many individuals categorised as low-risk did not. This metric is very powerful, as it assesses the validity of the predicted risk metric. Throughout this study, we will also use this test to validate our predicted scores for the machine learning models.

An important decision in this exercise is the selection of the decile score threshold that will dictate that values above it are more likely to recidivate and those below it not. In order to do this, we compared the decile score and two-year record for multiple thresholds. We examined the number of true negatives (the individuals with a decile score lower than the threshold and did not recidivate), true positives (those with a decile score higher or equal than the threshold and did recidivate) and false positives and negatives, and obtained the following results:

Threshold	True Neg	False Pos	True Pos	False Neg
4	2129	1046	964	1632
5	2433	742	1243	1353
6	2661	514	1529	1067
7	2849	326	1805	791
8	2979	196	2087	509

TABLE I: Decile Score Threshold Results

We then used these results to calculate the sensitivity, specificity, precision and accuracy:

Threshold	Sensitivity	Specificity	Precision	Accuracy
4	0.628659	0.670551	0.60941	0.651707
5	0.521186	0.766299	0.645823	0.656039
6	0.411017	0.83811	0.674889	0.645989
7	0.3047	0.897323	0.708147	0.63074
8	0.196071	0.938268	0.721986	0.604401

TABLE II: Performance Metrics for Different Decile Score Thresholds

Following this procedure, a threshold of 5 was decided to provide the best balance of these metrics. With this threshold, the total accuracy and precision are moderate, indicating that while the model performs reasonably well overall, there is room for improvement. We notice a low sensitivity compared to specificity; the model is better at identifying non-recidivists than recidivists.

The metrics for each racial group were then calculated with this threshold.

The Caucasian group has a relatively high specificity but low sensitivity, indicating the COMPAS model more effectively avoids false positives but struggles to identify true positives.

The African-American group shows higher sensitivity but lower specificity compared to Caucasians. This suggests the model identifies more recidivists among African Americans but at the cost of more false positives.

The Hispanic group has low sensitivity, meaning the model struggles significantly to identify true positives in this group. Precision is also the lowest, indicating a high false-positive rate.

The significant disparity in sensitivity and specificity across racial groups highlights potential fairness issues in the model's predictions. For example, the model disproportionately favours Caucasians and Asians in terms of specificity while penalizing African-Americans with higher false-positive rates.

V. Predicting **`decile_score`** using linear regression

A. Measurement

Quantitative measurements numerically represent attributes and are fundamental for evaluating machine learning models by providing an objective means to assess the model's performance. We can analyse the model's effectiveness by comparing model predictions to actual outcomes. Appropriate selection and reporting of measurement methods, including their reliability, validity, and potential biases, are essential to ensure accurate interpretation and meaningful results. This section will look at the measurements used in this study.

The function used in machine learning and statistical modelling to quantify the difference between the predicted outputs of a model and the actual target values is called the loss function. It serves as a measure of the model's performance, guiding the optimisation process to improve predictions.

For a single prediction:

$$\text{Loss}(\hat{y}, y) = f(\hat{y}, y)$$

Where:

$\hat{y}$: Predicted value.
y : Actual value.
f : Specific loss function formula

The aggregation of the loss function across the entire dataset is called the cost function so that;

$$\text{Cost}(\hat{y}, y) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\widehat{y}_i, y_i)$$

In the problem we are considering, we will try to predict whether a person will recidivate given information about his or her previous criminal history is a binary classifier problem. A binary classifier is a function that can be applied to features X such as $(x_1, x_2, x_3, \ldots, x_n)$ and maps them to an output Y, where $Y \in \{0, 1\}$. It is a supervised learning technique; therefore, a test set is extracted from the available data to validate the model before being deployed in production.

$$f(x_1, x_2, x_3, \ldots, x_n) = Y \in \{0, 1\}$$

The function will return a value between 0 and 1; therefore, a threshold value is operated to classify the result as true or

false. The model will subsequently classify predictions as true or false according to the threshold value.

For the classification problem that we have in hand, we will primarily use the log-likelihood cost function defined as:

$$\mathcal{L}_{\text{log-likelihood}} = -\sum_{i=1}^{N} [y_i \ln(\hat{y}_i) + (1 - y_i) \ln(1 - \hat{y}_i)]$$

Negative log-likelihood focuses on the distance between the labels y_i and the predicted probabilities $\hat{y}_i$. If we consider a sample where:

- y_i is one and $\hat{y}_i$ is close to 1: $y_i \ln(\hat{y}_i)$ is close to zero, while $(1-y_i)\ln(1-\hat{y}_i)$ is zero, so that the resultant loss is close to zero.
- y_i is 1 and $\hat{y}_i$ is far from 1: $y_i \ln(\hat{y}_i)$ is large, while $(1 - y_i)\ln(1 - \hat{y}_i)$ is zero so that the resultant loss is large.
- y_i is zero and $\hat{y}_i$ is close to 0: $y_i \ln(\hat{y}_i)$ is zero, while $(1-y_i)\ln(1-\hat{y}_i)$ is also close to zero so that the resultant loss is close to zero.
- y_i is zero and $\hat{y}_i$ is far from 0: $y_i \ln(\hat{y}_i)$ is zero, while $(1 - y_i)\ln(1 - \hat{y}_i)$ is large so that the resultant loss is large.

In this scenario, we can obtain four kinds of results:

Prediction	Classif	Outcome	Description
1	1	True Positive (TP)	The model correctly predicted the positive class.
1	0	False Positive (FP)	The model incorrectly predicted the positive class (a "false alarm").
0	1	False Negative (FN)	The model incorrectly predicted the negative class (a "miss").
0	0	True Negative (TN)	The model correctly predicted the negative class.

The number of samples that are TP, TN, FP or FN can be organised in what is known as a confusion matrix, that is shown in Figure 12. This tool makes it easy to perform calculations that determine the validity of the model at hand.

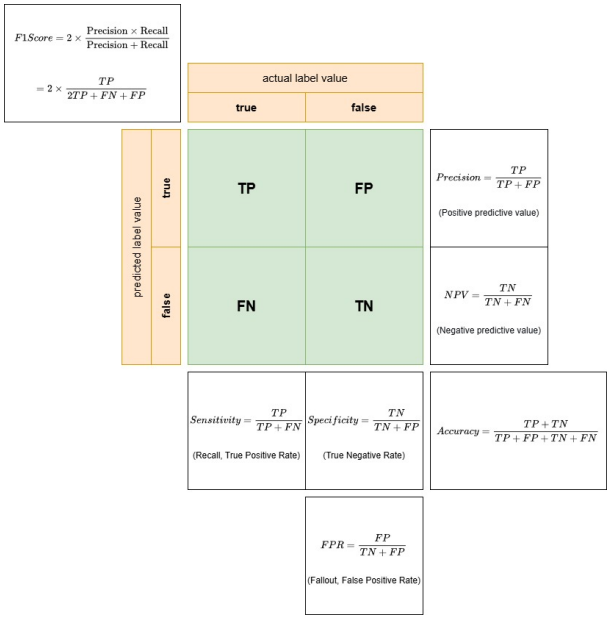

Fig. 12: Confusion matrix and associated equations

There are several key performance metrics derived from the confusion matrix, each offering a different perspective on evaluating the effectiveness of a binary classification model.

The **accuracy** of a model is a straightforward measure that evaluates the proportion of correct predictions (both True Positives and True Negatives) out of the total number of predictions. It can be mathematically defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

for a dataset,

$$Accuracy = \frac{1}{N} \sum_{i=1}^{N} 1(\hat{y}_i = y_i)$$

Precision answers the question: "Out of all the observations predicted to be positive, how many were positive?" In other words, it measures the model's ability to avoid false positives.

$$Precision = \frac{TP}{TP + FP}$$

Recall or **sensitivity** measures the model's ability to correctly identify positive cases. It answers the question: "Out of all the actual positive instances, how many did the model correctly identify as positive?"

$$Recall = \frac{TP}{TP + FN}$$

Specificity measures the model's ability to identify negative cases correctly. It answers the question: "Out of all the actual negative instances, how many did the model correctly identify as negative?"

The formula for specificity is:

$$Specificity = \frac{TN}{TN + FP}$$

The F1-score is the harmonic mean of [precision](model-performance-precision) and [recall](model-performance-recall). It provides a single metric that balances precision and recall, which is useful when there is an uneven class distribution or when false positives and false negatives are important.

The formula for the F1 score is:

$$\begin{aligned} F1 &= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \\ &= 2 \times \frac{TP}{2TP + FN + FP} \end{aligned}$$

The ROC curve is plotted by varying the decision threshold of the classifier and plotting the corresponding values of FPR (on the x-axis) and sensitivity or TPR (on the y-axis). Each point on the ROC curve represents a (FPR, TPR) pair for a particular threshold value.

![[img-receiver-operator-characteristic-curve.png]]

The Area Under the ROC Curve (AUC) is a key indicator of model performance. The value of the AUC ranges from 0 to 1:

- A perfect classifier has an AUC of 1, indicating it achieves a TPR of 1 while keeping the FPR at 0.
- A model with an AUC of 0.5 performs no better than random guessing.
- Higher AUC values indicate better overall performance.

The goal of a classifier is to maximise the TPR (correctly predicting positive instances) while minimising the FPR (incorrectly classifying negative instances as positive). Ideally, the ROC curve should approach the top-left corner of the plot, indicating a high TPR with a low FPR.

In summary, the ROC curve helps to visualise and compare the trade-offs between true positives and false positives across different thresholds, and the AUC provides a single number summarising the model's ability to discriminate between positive and negative classes.

B. Cross-validation

Cross-validation is a technique that can be applied to any ML algorithm that is aimed to reduce overfitting by estimating how well each hypothesis generalizes to unseen data. In practice, a portion of the data is reserved for this purpose.

The K-fold cross validation technique splits the dataset into k folds, training that model on $k-1$ folds and testing on the remaining one. The test fold is rotated and the process is repeated so that each fold will act as the test set once. Metrics like mean square error are averaged across folds, ensuring a robust estimate.

Therefore, if for example, we split the data into 10 folds, so that each fold will have $n/10$ records, we train the model on 9 folds and test on the remaining one. We then rotate the test fold and repeat this process 10 times. The final performance metric is computed as the mean of the metrics across the 10 folds.

In stratified K-fold cross validation, the folds have the same proportion of the classes as in the original dataset.

Other types of cross validation include Leave one out cross-validation, where each data point is used as a test set once.

C. Feature scaling

Feature scaling is an important data transformation process. It is a very important aspect of many machine learning algorithms including logistic regression, support vector machines and neural networks, as, the performance of such algorithms is adversely impacted when the numerical features have different scales. The two methods to transform features on the same scale are normalization and rescaling.

During normalization or rescaling, values are scaled and shifted so that they are mapped onto the $[0, 1]$ interval. This is achieved by applying the following transformation;

$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Normalization is also sometimes referred to as min-max scaling.

Standardization is particularly suited for algorithms that assume Gaussian distributions, like linear regression and logistic regression. It is achieved by first subtracting values from the mean so that the mean of the normalized values is zero, and then dividing by the standard deviation so that the variance of the normalized distribution is one. Thus;

$$x_{standardized} = \frac{x - \mu}{\sigma}$$

Unlike normalization, the range of standardization is not fixed, and this can sometimes be an issue if a value in the $[0, 1]$ interval is expected. Standardization is however more resilient to the effect of outliers.

Tree based machine learning models, like decision trees and random forests do not normally require feature scaling.

D. Logistic regression

Logistic regression is a machine learning algorithm commonly used for binary classification tasks.

Given a feature vector $X \in \mathbb{R}^{n_x}$, the goal of logistic regression is to predict the probability $\hat{y}$ that a binary output variable y takes the value 1, given X, that is $\hat{y} = P(y = 1|X)$, $0 \leq y \leq 1$. For example, in the case of image classification, logistic regression can be used to predict the probability that an image contains a cat.

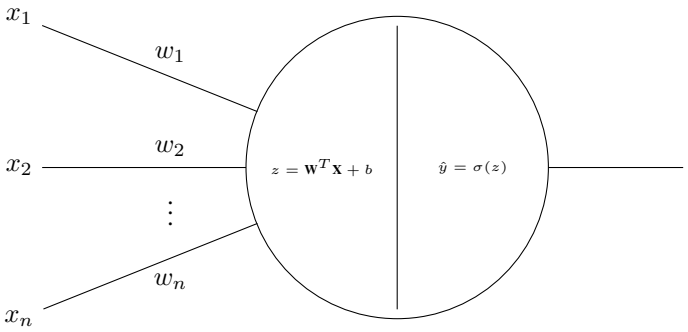

Fig. 13: A logistic regression model

Logistic regression can be visualized as the model shown in Figure 13. It consists of several main components:

Inputs. The input vector to the model $\mathbf{X} \in \mathbb{R}^{n_x}$.

Parameters. A weight vector $\mathbf{W} \in \mathbb{R}^{n_x}$ and a bias term $b \in \mathbb{R}$. These will form the coefficients of a linear equation that gives the log odds ratio.

Pre-activation result: The result is obtained by multiplying the transpose of the weights with the inputs and then adding the bias.

$$z = \mathbf{W}^T\mathbf{X} + b = \begin{bmatrix} w_1 & w_2 & \dots & w_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + b$$

Sigmoid function. A function as shown in Figure 14, $\sigma(z) = \frac{1}{1+e^{-z}}$, which maps any real number z to the range $(0, 1)$. This function is used to ensure that the predicted probability $\hat{y}$ is always between 0 and 1.

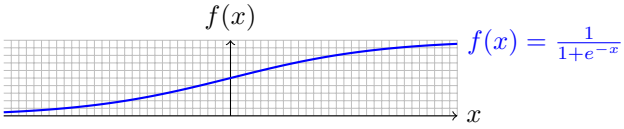

Fig. 14: Sigmoid activation function

Output. The predicted probability $\hat{y}$ is computed as $\hat{y} = \sigma(z) = \sigma(\mathbf{W}^T\mathbf{X} + b)$.

A term that is often encountered in this scenario is the log-odds ratio or logit. Logistic regression models the probability $P(y = 1 \mid X)$, of the binary dependent variable Y given the predictor variables $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$. The goal is to find a relationship between $P(y = 1)$, and the predictors $\mathbf{X}$.

The probability is modelled using the sigmoid function:

$$P(y = 1 \mid X) = \frac{1}{1 + e^{-\eta}}$$

Where: $\eta = b + w_1x_1 + w_2x_2 + \cdots + w_nx_n$ is the linear regression function

The odds of $y = 1$ are defined as the ratio of the probability of success to the probability of failure:

$$\text{Odds} = \frac{P(y = 1)}{1 - P(y = 1)}$$

Taking the natural logarithm of the odds gives the [log-odds-ratio]] or logit:

$$\text{Log-Odds} = \ln\left(\frac{P(y = 1)}{1 - P(y = 1)}\right)$$

From the sigmoid function, we can derive the relationship between the probability and the log-odds-ratio:

$$P(y = 1) = \frac{1}{1 + e^{-\eta}}$$

$$1 - P(y = 1) = 1 - \frac{1}{1 + e^{-\eta}} = \frac{e^{-\eta}}{1 + e^{-\eta}}$$

The odds, therefore, are

$$\text{Odds} = \frac{P(y = 1)}{1 - P(y = 1)} = \frac{\frac{1}{1+e^{-\eta}}}{\frac{e^{-\eta}}{1+e^{-\eta}}} = e^{\eta}$$

Taking the natural logarithm of both sides gives the log-odds:

$$ln\left(\frac{P(y = 1)}{1 - P(y = 1)}\right) = \eta$$

Substituting $\eta = b + w_1x_1 + w_2x_2 + \cdots + w_nx_n$

$$\ln\left(\frac{P(y = 1)}{1 - P(y = 1)}\right) = \eta = b + w_1x_1 + w_2x_2 + \cdots + w_nx_n \quad (1)$$

We conclude this overview of logistic regression by looking at how they work and learn. The weight vector $\mathbf{W}$ and the bias term b are learned from a labelled training set by minimizing a suitable loss function using techniques such as gradient descent or its variants. Once trained, the logistic regression model can be used to predict the probability of the binary output variable for new input examples.

The **feedforward process** for logistic regression can be described as follows:

Compute z as the dot product of the weight vector $\mathbf{W}$ and the input features, plus the bias term b, transforming the input features info a single scalar z that represents the log-odds of the output being $y = 1$:

$$z = \mathbf{W}^T\mathbf{X} + b \quad (2)$$

Pass z through the sigmoid function to map the log-odds z to a probability $\hat{y} = P(y = 1 \mid X)$, ensuring the output is between 0 and 1:

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}} \quad (3)$$

During training, we define the loss function $\mathcal{L}$ as the negative log-likelihood of the predicted output given the true label:

$$\mathcal{L} = -\left(y \ln(\hat{y}) + (1 - y) \ln(1 - \hat{y})\right) \quad (4)$$

For a trained system, we compare $\hat{y}$ to a threshold to convert the probabilistic output into the final binary classification.

We now look at **feedback process** for logistic regression. To optimize the weight vector $\mathbf{W}$, we compute the derivatives of the loss function with respect to each weight and the bias term and use these derivatives to update the weights in the opposite direction of the gradient. This is known as gradient descent.

To compute the derivatives, we use the chain rule:

$$\frac{\partial \mathcal{L}}{\partial w_i} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial w_i}$$

and

$$\frac{\partial \mathcal{L}}{\partial b} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial b}$$

We can then use these derivatives to update the weights as follows:

$$w_i \leftarrow w_i - \alpha \frac{\partial \mathcal{L}}{\partial w_i} \tag{5}$$

and

$$b \leftarrow b - \alpha \frac{\partial \mathcal{L}}{\partial b} \tag{6}$$

Where α is the learning rate, which controls the step size of the updates. By iteratively performing these updates on a training set, we can find the optimal weight vector $\mathbf{W}$ that minimizes the loss function on the training set.

To calculate the derivatives, let us begin by computing the derivative of the loss function with respect to the predicted output $\hat{y}$:

$$\frac{\partial \mathcal{L}}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} \left(-\left(y \ln(\hat{y}) + (1-y) \ln(1-\hat{y}) \right) \right)$$

since

$$\frac{d(ln(x))}{dx} = \frac{1}{x}$$

we get:

$$\frac{\partial \mathcal{L}}{\partial \hat{y}} = -\left(\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}} \right)$$

The derivative of the predicted output $\hat{y}$ with respect to z is solved using the quotient rule, that is

$$\frac{d}{dx}\left(\frac{f(x)}{g(x)}\right) = \frac{f'(x)g(x) - g'(x)f(x)}{g^2(x)}$$

So, if we let

$f(z) = 1$	$f'(z) = 0$
$g(z) = 1 + e^{-z}$	$g'(z) = -e^{-z}$

$$\begin{aligned}
\frac{\partial \hat{y}}{\partial z} &= \frac{\partial}{\partial z}\left(\frac{1}{1+e^{-z}}\right) \\
&= \frac{e^{-z}}{(1+e^{-z})^2} \\
&= \frac{1}{(1+e^{-z})}\frac{e^{-z}}{(1+e^{-z})} \\
&= \frac{1}{(1+e^{-z})}\frac{1+e^{-z}-1}{1+e^{-z}} \\
&= \frac{1}{(1+e^{-z})}\left(\frac{1+e^{-z}}{1+e^{-z}} - \frac{1}{1+e^{-z}}\right) \\
&= \hat{y}(1-\hat{y})
\end{aligned}$$

The derivative of z with respect to w_i:

$$\begin{aligned}
\frac{\partial z}{\partial w_i} &= \frac{\partial}{\partial w_i}\mathbf{W}^T\mathbf{X} + b \\
&= \frac{\partial}{\partial w_i}(w_1x_1 + \cdots + w_ix_i + \cdots + w_nx_n + b) \\
&= x_i
\end{aligned} \tag{7}$$

Similarly,

$$\begin{aligned}
\frac{\partial z}{\partial b} &= \frac{\partial}{\partial b}\mathbf{W}^T\mathbf{X} + b \\
&= \frac{\partial}{\partial b}(w_1x_1 + \cdots + w_ix_i + \cdots + w_nx_n + b) \\
&= 1
\end{aligned} \tag{8}$$

Therefore

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w_i} &= \frac{\partial \mathcal{L}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial w_i} \\
&= -\left(\frac{y}{\hat{y}} - \frac{1-y}{1-\hat{y}}\right) \cdot \hat{y}(1-\hat{y}) \cdot x_i \\
&= -\left(\frac{-y(1-\hat{y}) + (1-y)\hat{y}}{\hat{y}(1-\hat{y})}\right) \cdot \hat{y}(1-\hat{y}) \cdot x_i \\
&= \left[(1-y)\hat{y} - y(1-\hat{y})\right] x_i \\
&= \left[\hat{y} - y\hat{y} - y + y\hat{y}\right] x_i
\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial w_i} = (\hat{y} - y)x_i \tag{9}$$

and similarly

$$\frac{\partial \mathcal{L}}{\partial b} = (\hat{y} - y) \tag{10}$$

REFERENCES

[1] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and machine learning: Limitations and opportunities*. MIT press, 2023.

APPENDIX

Field Name	Description	Type	Options (if Categorical)	Used
id	Unique identifier for each individual.	Numeric	N/A	No
name	Full name of the defendant	Text	N/A	No
first	First name of the defendant (anonymized).	Text	N/A	No
last	Last name of the defendant (anonymized).	Text	N/A	No
compas_screening_date	Date of the COMPAS assessment.	Date	N/A	Yes
sex	Gender of the defendant.	Categorical	Male, Female	Yes
dob	Date of birth of the defendant.	Date	N/A	No
age	Age of the defendant at the time of assessment.	Numeric	N/A	Yes
age_cat	Age category of the defendant.	Categorical	Less than 25 25 - 45 Greater than 45	Yes
race	Race of the defendant.	Categorical	African-American Caucasian Hispanic Asian Native American Other	Yes
juv_fel_count	Number of juvenile felony offenses.	Numeric	N/A	Yes
juv_misd_count	Number of juvenile misdemeanor offenses.	Numeric	N/A	Yes
juv_other_count	Number of other juvenile offenses.	Numeric	N/A	Yes
priors_count	Number of prior offenses (adult and juvenile).	Numeric	N/A	Yes
days_b_screening_arrest	Days between arrest and COMPAS screening.	Numeric	N/A	Yes
c_jail_in	Jail booking date for the charge.	Date	N/A	No
c_jail_out	Jail release date for the charge.	Date	N/A	No
c_case_number	Case number associated with the charge.	Text	N/A	No
c_offense_date	Date of the alleged offense.	Date	N/A	No
c_arrest_date	Arrest date for the charge.	Date	N/A	No
c_charge_degree	Degree of the charge.	Categorical	F (Felony) M (Misdemeanor)	Yes
c_charge_desc	Description of the charge.	Text	Free text	No
is_recid	reoffended after COMPAS screening.	Binary	0 (No), 1 (Yes)	No
r_case_number	Case number for the re-offense.	Text	N/A	No
r_charge_degree	Degree of the re-offense charge.	Categorical	F (Felony) M (Misdemeanor)	No
r_charge_desc	Description of the re-offense charge.	Text	Free text	No
r_jail_in	Jail booking date for the re-offense.	Date	N/A	No
r_jail_out	Jail release date for the re-offense.	Date	N/A	No
two_year_recid	Label: offense within two years.	Binary	0 (No), 1 (Yes)	Yes
decile_score	COMPAS risk score (1-10).	Numeric	1–10	Yes
score_text	Risk category for general recidivism.	Categorical	Low, Medium, High	Yes
v_type_of_assessment	Type of COMPAS assessment conducted.	Text	Risk of Recidivism	No
v_decile_score	Violent recidivism COMPAS score (1-10).	Numeric	1–10	No
v_score_text	Risk category for violent recidivism.	Categorical	Low, Medium, High	No
start	Start date of the two-year recidivism period.	Date	N/A	No
end	End date of the two-year recidivism period.	Date	N/A	No
event	offense during the two-year period.	Binary	0 (No), 1 (Yes)	No