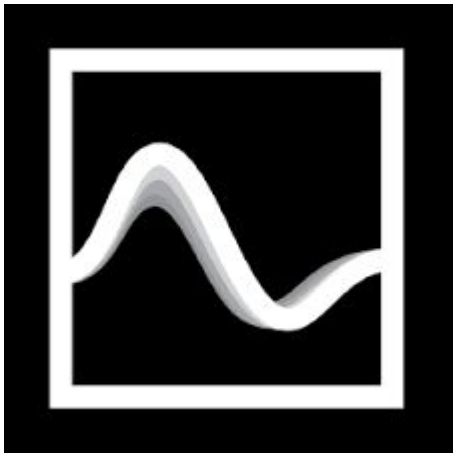


MAISI

CRA Week 3:

Error and Bias Analysis across Demographics

Michigan Data Science Team

Fall 2025

Session 3 Agenda

01

Fun Icebreaker!!

Get to know your projectmates! (and their dream jobs)

02

Intro to Scikit

What are risk assessment algorithms?

03

Intro to Performance Metrics

How was COMPAS used, and what are its flaws?

04

AI Safety Mini-Lesson

How does COMPAS relate to the use of biased AIs?

05

Practice Time!

Work on the COMPAS dataset (with a fun warmup)!

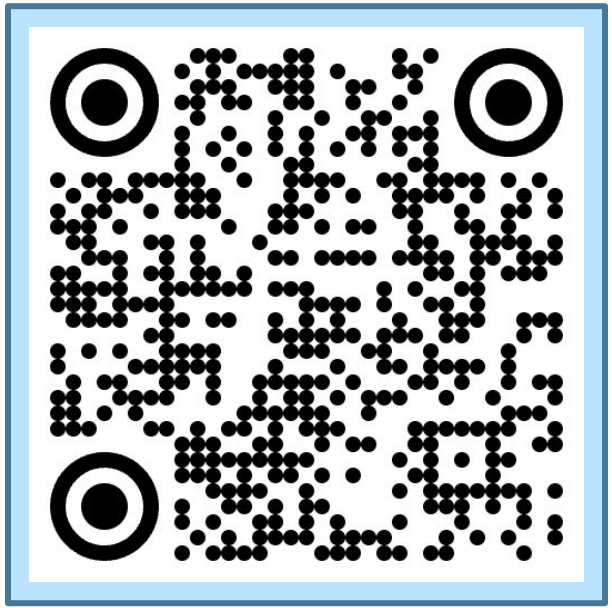
Quick Icebreaker!!

What is your unrealistic dream job? (No need to worry about money or qualifications)

Share with the people around you :)

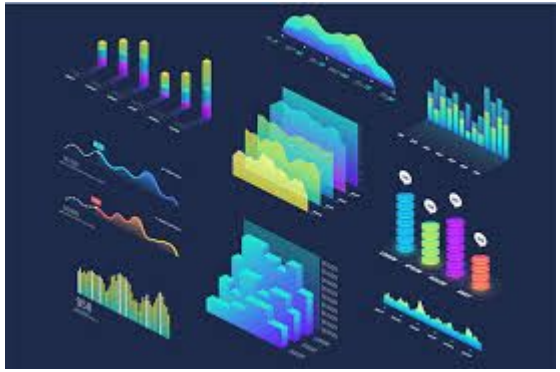
But first, a quick survey!

We were thinking that it may be fun to offer the opportunity for a team social event outside of our typical work sessions! If this sounds interesting to you, please fill out this short form letting us know your thoughts on when and what we should do.



Introduction to New Statistical Tools and Techniques

- The exercises we'll be doing today will be analyzing the types of errors that the COMPAS model makes
- We'll be seeing whether false positives and false negatives in assigned recidivism scores differ between groups (across races, genders, ages, etc.)
- We'll also be doing some more visualization and calculating some more performance metrics to see how well the model … performs



Introduction to scikit-learn

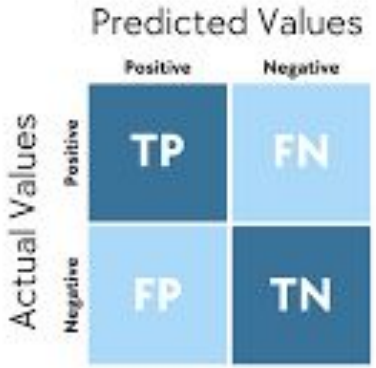
- scikit-learn is one of the most popular and powerful open-source machine learning libraries for Python.
- It provides a wide variety of tools for building, evaluating, and analyzing machine learning models, such as classification, regression, clustering, and more.
- It includes easy-to-use functions for model evaluation and performance metrics, which are essential for understanding how well a model performs.

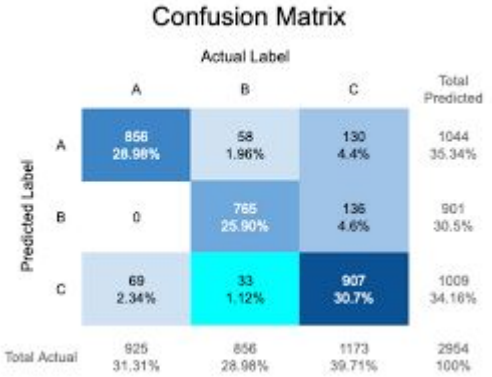
scikit-learn in Data Analysis

- **Ease of Use:** scikit-learn provides user-friendly machine learning functionality.
- **Wide Range of Features:** scikit-learn offers comprehensive tools to cover the entire workflow, from model building to evaluation
- **Community Support:** scikit-learn is well-supported and documented, making it a go-to tool for data scientists and analysts of all experience levels.
- This week, we are using scikit-learn to evaluate **how well the model predicts recidivism** and where its potential biases may lie

Key scikit-learn Function to Know

- Confusion Matrix
  - **What It Does:** The confusion matrix provides a summary of prediction results, plotting true/false positives/negatives.
  - **Why We Use It:** This breakdown helps us understand where our model is making errors (errant score in COMPAS context)





Practical Application of scikit-learn

Step 1: Building a Confusion Matrix

- Using confusion_matrix() from scikit-learn to build a matrix that shows how well the COMPAS risk assessment model differentiates between individuals who reoffend and those who do not.
- It helps visualize the distribution of correct vs. incorrect predictions.

Step 2: Calculating Performance Metrics

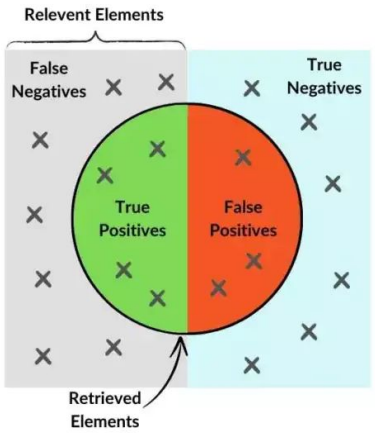
- Precision helps us assess how often high-risk predictions are correct, which is particularly relevant for evaluating fairness.
- Recall helps us understand how effectively the model identifies actual recidivists across different demographics.

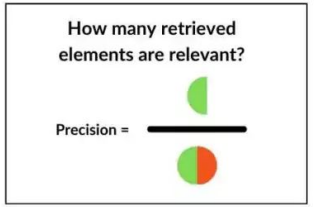
Performance Metrics

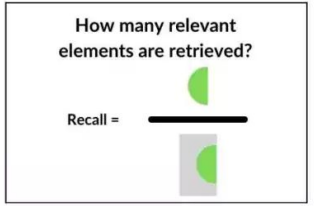
- **Accuracy**: The ratio of correct predictions to the total number of predictions.
- **Precision**: The ratio of correctly predicted positive observations to the total predicted positives. It answers the question: "Of all those predicted to reoffend, how many actually did?"
- **Recall** (Sensitivity): The ratio of correctly predicted positive observations to all actual positives. It answers the question: "Of all those who actually reoffended, how many were correctly predicted?"
- **Specificity:** High specificity means that the classifier correctly identifies most of the true negatives, minimizing false positives.
- **F1 Score**: A balanced metric that considers both precision and recall, especially useful if the class distribution is imbalanced.

Why Do We Use Performance Metrics?

These metrics help us evaluate different aspects of model performance. For example, precision is especially important in scenarios where false positives have significant consequences (e.g., mistakenly labeling someone as high-risk). Recall is important for ensuring that we do not miss actual recidivists.







Equations

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

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

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

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

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

TN = True Negative
TP = True Positive

FN = False Negative
FP = False Positive

A Note on Equalized Odds

- We'll be looking at False Positives vs. False Negatives today.
  - While doing this, it's important to keep in mind that while these cases are similar, their errors can have drastically different qualitative effects in certain contexts.

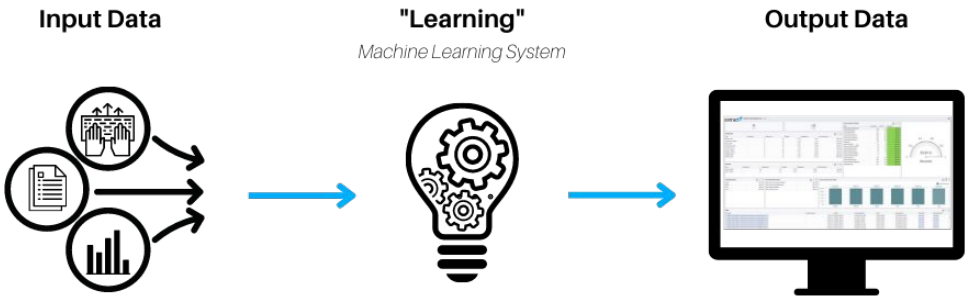
- Imagine a cancer predictor
  - FP: Patient is alarmed, but is otherwise safe
  - FN: Patient's condition worsens

- What is the difference between FP/FN in COMPAS?

The Machine Learning System

- Machine learning models work by learning patterns from data through **training on examples.**
- Possible sources of problems in machine learning
  - Bias **within the dataset** (less common)
  - Bias **within the algorithm itself** (less common)

How can this system of training on past examples go wrong? How so?



Non-Representative Datasets / Sampling Bias

The strength of an ML model is directly proportional to the size of its dataset.

- Similar to human learning: more studying = more knowledge

Non-representative datasets or datasets with sampling bias are datasets with different amounts of representation for different groups

For example: in 2007, UMass Amherst created a dataset called Labeled Faces in the Wild (LFW), with thousands of human faces.

- 77% male, 83% white
- 530 images of George W. Bush - 2x the amount of all images of all black women

...

George Robertson (22)

George W Bush (530)

Gerhard Schroeder (109)

Feature Selection Bias

Feature selection is the process of identifying and selecting the most relevant features (or variables) from a dataset to use in training a machine learning model.

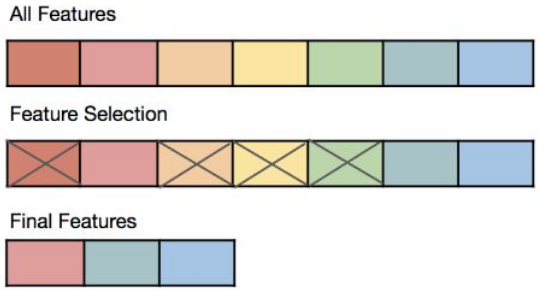
In ML models, this is a crucial step to improve performance, reduce overfitting, and help the model to generalize.

Unfair outcomes start to emerge when the features are selected improperly

- If a feature reflects a certain protected group, this can lead to unfair outcomes between groups.
  - E.g. if the COMPAS algorithm took the defendant's race as a feature (which it doesn't)

A **proxy** is a feature that is not directly reflective of a certain group but is correlated with one. Proxies are in many different fields, not just ML.

- 4 examples from obvious to not so obvious:
  - Last name
  - Chance to get skin cancer
  - Did you work food service when you were in high school
  - Zip code



Datasets Perpetuating Stereotypes

- As we all know, society is not free of its biases. What happens when a dataset is well-adjusted to how we think? Are decisions and algorithms made on this data also biased?

- Example: Natural Language Processing (NLP) model ***word2vec***
  - word2vec is a model trained to calculate the relationship between words
    - However, a famous paper by Tolga Bolukbasi found that this model perpetuated some unsatisfactory stereotypes
  - Such outputs were of the form: "'man' is to 'computer scientist' as 'woman' is to 'homemaker'"

- Question: Is this model sexist, or is it just outputting what it learned?

Link to an awesome illustration of how word2vec works - highly recommend!

Practice Time!

Let's see how the COMPAS model stacks up when reviewing its performance metrics!

Reminders

- Don’t share colab notebooks with teammates if you are working at the same time
- Even though you did this last week, you will likely need to re-upload your file and run all of your code chunks from last week before continuing.
- Where to put csv and data files
  - Google Drive
    - Need to include:

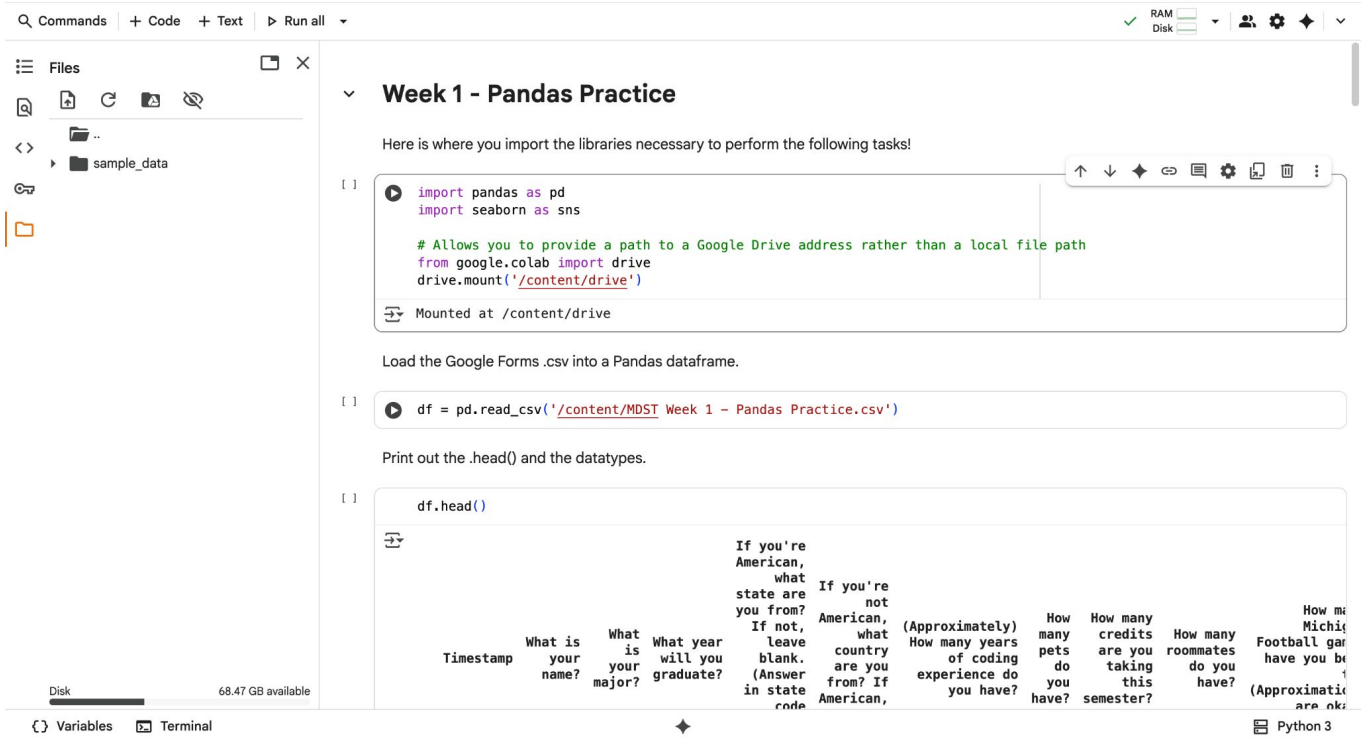
```
from google.colab import drive
drive.mount('/content/drive')
```

```
pd.read_csv('/content/drive/MyDrive/[FILE NAME]')
```

  - Colab Files
    - See next slides

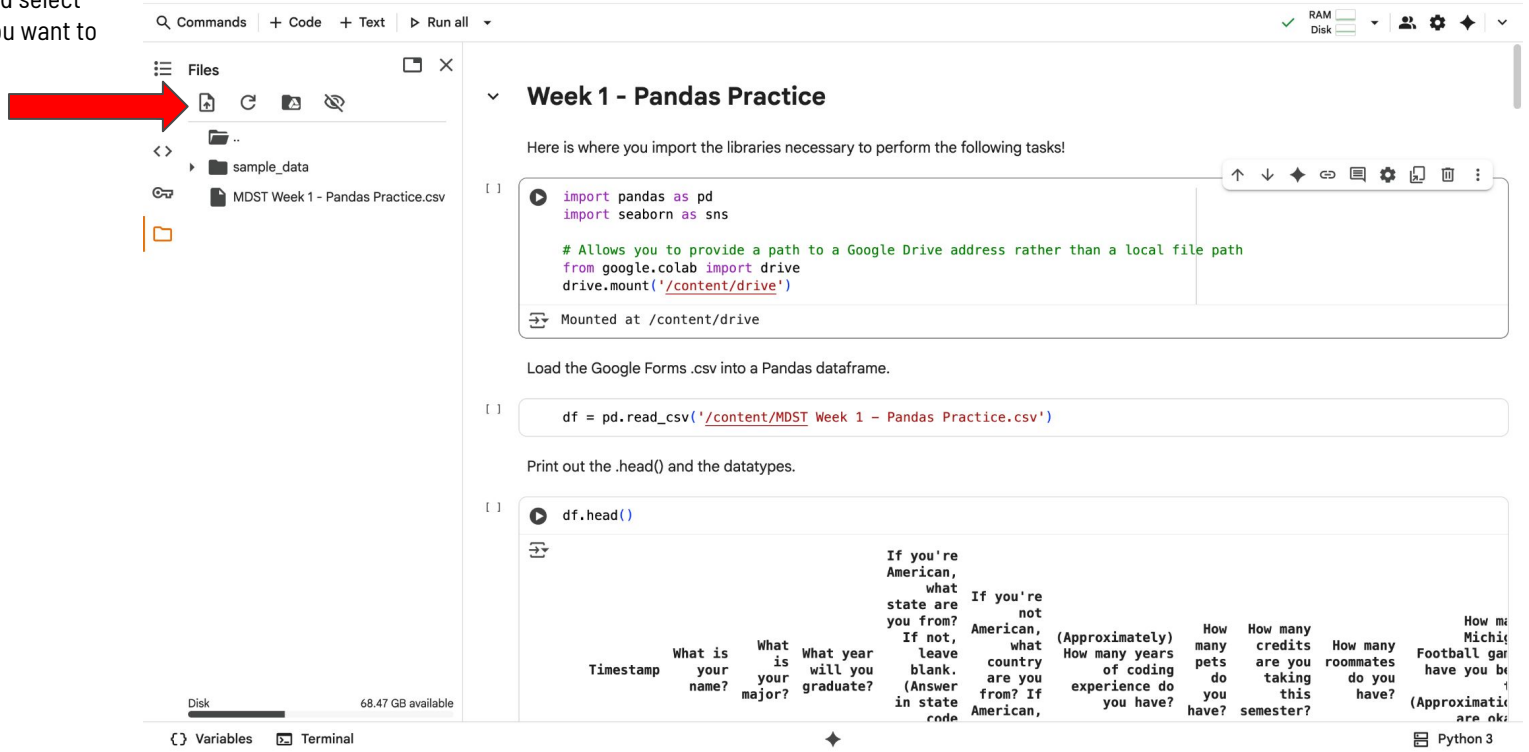
Reminders

Click on the folder in the sidebar



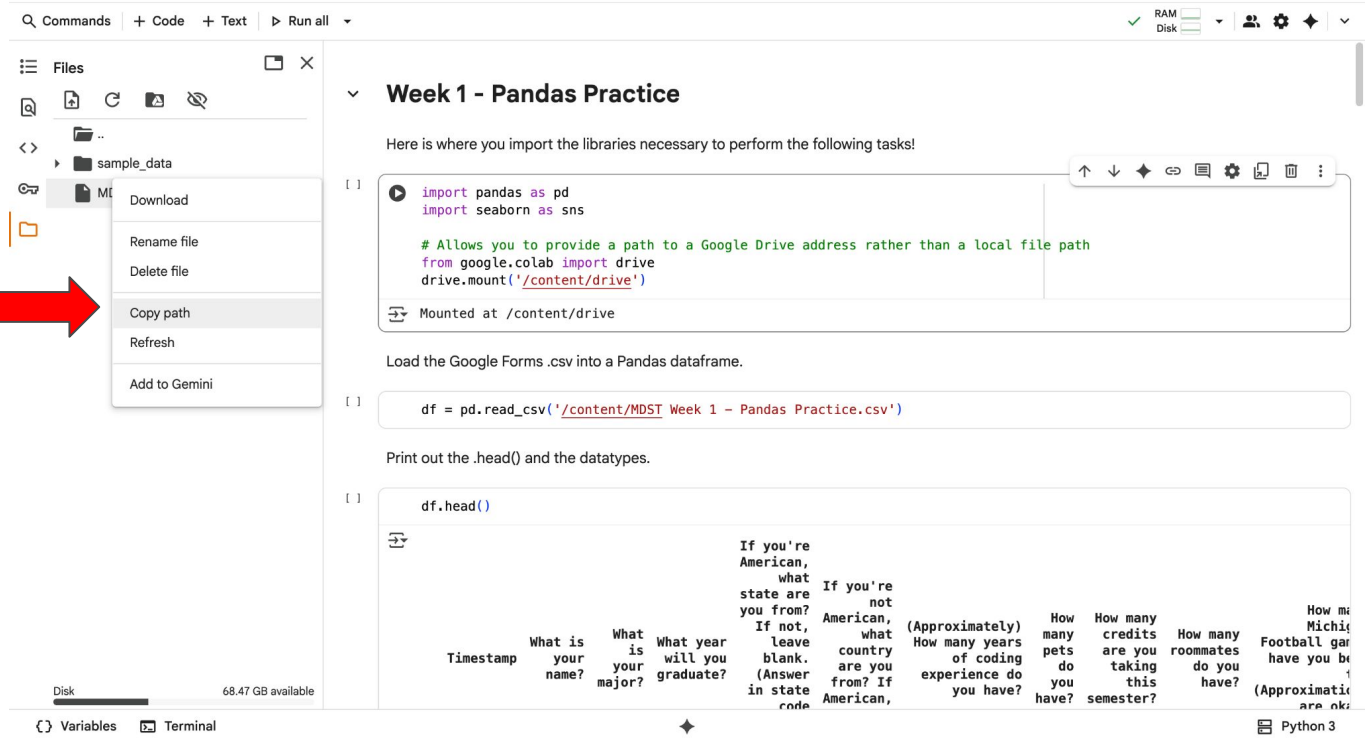
Reminders

Click the upload button and select the file you want to upload



Reminders

Click the three dots and copy the path. Put this in your read function



Once You're Done - Exporting a .csv

- Once you complete Exercise 3.7 in bias_analysis.ipynb, you'll want to take your nice and clean dataset and export it
  - This way, when you move on to do similar bias analysis for gender, you won't have to do all of that monotonous cleaning all over again

- The easiest way to do this will be by exporting your files directly to your Google Drive. The syntax for how to do this is shown below:

```
df.to_csv('/content/drive/MyDrive/compas_week3_cleaned.csv',
index=False)
```

Create a new code block after Exercise 3.7 and copy this line! (make sure your drive is mounted)

Hands-On Data Science!! :O

Next Steps:

1. Find the F25 CRA repo in the MDST GitHub and download all the files in the Week 3 directory
    a. You will need the completed key Ryan sent out for Week 2 to continue with today's work if you were not here last week.
2. Split into teams of 2-3 (new ones or same as last week)
    a. If you're working with new people, introduce yourselves!!
3. Work on the exercises in the notebooks!
    a. When you're done with Exercise 3.7, export your dataset and work on bias_analysis_gender.ipynb. Ask us if you need help!

Pandas Cheat Sheet Seaborn Cheat Sheet