

# Week 1: Exploratory Data Analysis



M A I S I



Building Interpretable AI in Healthcare  
Michigan Data Science Team - Winter 2026



# Week 1 Agenda

## 01. Meet the Leads +

### Icebreaker!

Get to know us and start to learn about each other with a fun start-of-semester icebreaker!

## 02. Intro to EDA

What is Exploratory Data Analysis, which Python libraries does it utilize, and how will it be useful going forward?

## 03. Identifying Good Data

Using reliable training data is paramount. But how can we tell if our dataset is trustworthy and representative?

## 04. Hands-On Data Science!

Let's apply these EDA skills to datasets of your choice and get our projects rolling!

# 01

## Meet the Leads / Fun Icebreaker!!

Get to know us and start to learn about each other with a fun start-of-semester icebreaker!



# Meet the Leads! – Will McKanna



**Hometown:** Rockford, MI

**Major:** DS and Statistics

**Year:** Sophomore

**Ask me about:** Studying abroad in Iceland, crocheting, trombone, Michigan and Detroit football, being a UMich Tour Guide



# Meet the Leads! - Seena Simkani



**Hometown:** Grand Blanc, MI

**Major:** Data Science

**Year:** Sophomore

**Ask me about:** Flying  
planes, playing the piano,  
mopeds, music



# Team Building - Icebreaker Bingo!

	A	B	C	D	E
1	I'm a fan of the Detroit Lions	Slept overnight at a UofM non dorm building	I can whistle	I'm part of another CS/DS club	I get the supreme slice @ Joe's
2	I'm a member of MAISI	I have season tickets to Michigan Football	Touched grass this summer (3+ outdoor activities)	I'm a Data Science major	I know the capital of Mongolia
3	I play a sport	I'm a non CS/DS major	I'm a part of MDST	I pay for guac at chipotle	I live on North
4	I'm a Computer Science major	Took Math 215 at Michigan (WCC >>)	I play an instrument	I've taken a formal statistics class (HS/college)	I've visited the Upper Peninsula
5	I'm from the state of Michigan	Skipped < half of my lectures last week	I live on Central	I've customized my VSCode	Read 3+ books this school year (since August)



# 02

## Intro to EDA

What is Exploratory Data Analysis, which Python libraries does it utilize, and how will it be useful going forward?

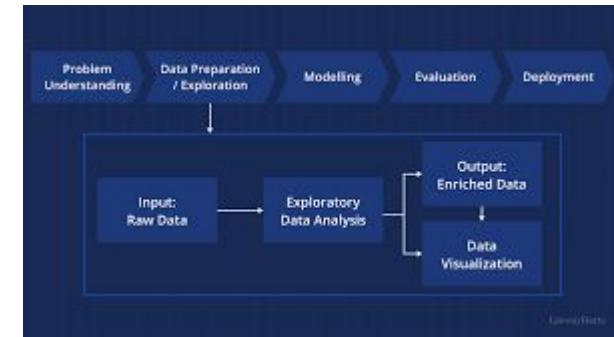


# What is Exploratory Data Analysis (EDA)?

**Definition:** “A process used to analyze datasets to summarize their main characteristics, often using visualization tools.”

## Main Goals:

- Understand Data Structure - Inspect the dataset to understand shape and types.
- Identify Patterns and Relationships - Use statistics and plots to find correlations and trends.
- Detect Data Quality Issues - Locate missing values, outliers, and inconsistencies.
- Generate Hypotheses - Develop questions from insights that can be investigated further.





# The Significance of EDA

- **EDA** is something of a philosophy of understanding data without assumptions. This enables us to obtain a thorough understanding of its **context, patterns, and limitations.**
- It reveals insights and helps in identifying **biases** and **relationships** that guide further analysis
- EDA ensures that data is **well-understood** and clean.
  - Many datasets have missing values of some sort. This can mess up your code! (and your understanding)
- This process informs **stronger hypotheses** and leads to better, **data-driven decisions.**



# Python Libraries for Data Analysis

Some of the most useful Python libraries known  
to man! (and panda)



# What is Pandas?

- Pandas is a Python library for data manipulation and analysis
  - The name “pandas” is derived from “Panel Data” and “Python Data Analysis” - helps us **understand data structures**
- Produces two main structures: **Series (1D)** and **DataFrame (2D)** <- Imagine a table in a database!
- Main Uses
  - Identify **missing values** and discover **patterns**
  - Enables **data cleaning and preprocessing** – crucial steps to data preparation





# What is NumPy?

- NumPy stands for “Numerical Python”
  - Provides **fast, efficient arrays** and a wide range of **mathematical functions**
  - Key tool for scientific computing
- NumPy’s Relationship with pandas and Its Role in EDA
  - Foundational for pandas: pandas is **built on top** of NumPy; pandas DataFrames internally use NumPy arrays
  - Exploratory Data Analysis: NumPy enables data manipulation, statistical analysis, and complex calculations required for understanding data before visualization or modeling

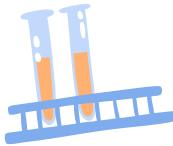




# 03

## Identifying Good Datasets

Using reliable training data is paramount. But how can we tell if our dataset is trustworthy and representative?





# Why is it important to have good data?

- Data imbued with **bias** can perpetuate harmful stereotypes, lead to misguided takeaways, and mess with the training of machine learning models.
- In the context of **healthcare**, data with accuracy or interpretability deficiencies can lead to further marginalization of at-risk populations.
  - Ex. A machine learning model is improperly trained on patient data that includes financial records and comes to the conclusion that low-income patients are less likely to be readmitted and thus must be at lower risk for their condition to reemerge. (What's the problem here?)
- There are three main types of bias that we will highlight:
  - Sampling Bias
  - Feature Selection Bias
  - Models Perpetuating Stereotypes



# Sampling Bias

- The strength of an ML model is proportional to the size of its training dataset
  - Like a human: more studying, more knowledge
- **Non-representative datasets** (datasets with sampling bias) have varying degrees of representation for people/instances in different groups
- Ex: In 2007, UMass-Amherst created a dataset called Labeled Faces in the Wild, an image dataset with thousands of human faces. However...
  - It was 77% male and 83% white
  - There were 530 images of just George W. Bush (the president at the time)
    - which is twice as many as all images of all Black women combined



George  
Robertson (22)



George W. Bush  
(530)



Gerhard  
Schroeder (109)

Q: What can we take away from this?

A: It matters what data we train ML models on (EDA would expose these flaws!)



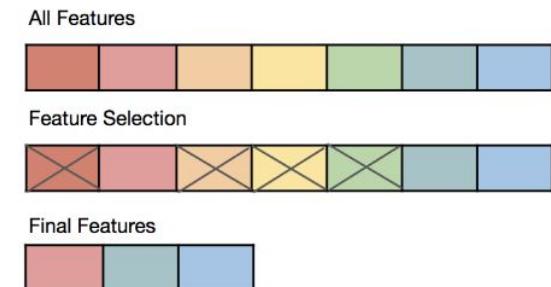
# Feature Selection Bias

- **Feature Selection** is the process of identifying and selecting the most relevant features of a dataset to use in training an ML model.
- In most ML models, this is a crucial step to reduce overfitting, improve performance, 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 models trained on your dataset took a patient's race in as a feature, this can lead to undesirable outcomes and racial profiling.

- A **proxy** is a feature that is not directly reflective of a certain group but is correlated with one.
- Four not-so-obvious examples
  - Last name
  - Skin cancer risk
  - If you worked food service in HS
  - ZIP code





# Data Perpetuating Stereotypes

- Society is inherently biased in uncountably many ways, some more subversive than others. But what happens when a dataset is “well-adjusted” to how we think? Is an algorithm biased if it makes decisions based off of biased behavior it learned from humans?
- Example: Natural Language Processing (NLP) model **word2vec**
  - Trained to calculate the relationship between words
    - However, a now-famous paper by Tolga Bolukbasi found that this model perpetuated some... unsatisfactory stereotypes
  - Some outputs included: “man” is to ‘computer scientist’ as ‘woman’ is to ‘homemaker’”
- Question: is this model sexist, or does it merely reflect how we perceive the world implicitly?

# Possible Datasets to Use

Diabetes Data from  
1999-2008

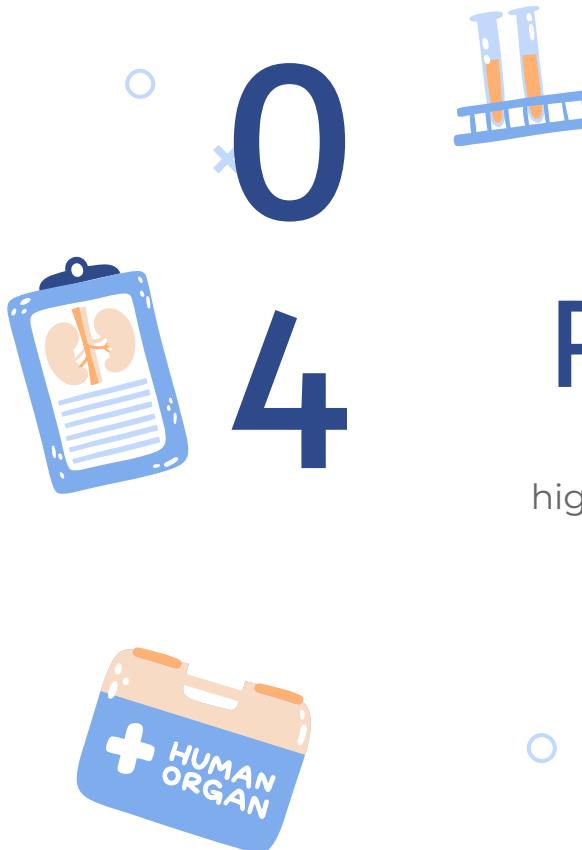
Lung Cancer Data

Cardiovascular  
Health Data

Stroke Prediction  
Data

Heart Attack Data

Sleep Health Data



# References

Here are a handful of useful guides to high-utility functions, Git demos, and Colab explanations!



# Project Timeline

**Week 1 (01/25):** Icebreaker/EDA intro, choosing a good dataset

**Week 2 (02/01):** Data Visualization, Further EDA

**Week 3 (02/08):** Feature Engineering and Intro to Machine Learning

**Week 4 (02/15):** Machine Learning and Model Optimization

**Week 5 (02/22):** Conformal Predictions, Risk Control, and Multi-Class Calibration

**NO MEETING 03/01 OR 03/08 – SPRING BREAK**

**Week 6 (03/15):** SHAP Values and Feature Importance Interpretation

**Week 7 (03/22):** Intro to Streamlit and Project Work Time

**Weeks 8 + 9 (03/29 + 04/05):** Final Project work time + Data Science Night prep!





# Important Functions to Know (pt. 1)

Syntax	Description
<code>import pandas as pd</code>	Import the <b>pandas</b> library, using the alias <b>pd</b> for convenience
<code>import numpy as np</code>	Import the <b>NumPy</b> library, using alias <b>np</b> for numerical operations
<code>import seaborn as sns</code>	Import <b>Seaborn</b> for statistical data visualization
<code>df = pd.read_csv('file.csv')</code>	Load a <b>CSV</b> file into a <b>pandas DataFrame</b> for data manipulation
<code>df.head()</code>	Returns the <b>first 5 rows</b> of the DataFrame, useful for quickly inspecting data
<code>df.info()</code>	Provides a concise <b>summary</b> of the DataFrame, including <b>data types</b> and <b>non-null values</b>



## Important Functions to Know (pt. 2)

Syntax	Description
<code>df.describe()</code>	Generates <b>descriptive statistics</b> of numerical columns (mean, median, quartiles, etc.)
<code>df['column_name']</code>	Access a <b>specific column</b> in the DataFrame, works like a key in a dictionary
<code>df.drop(columns=['col'...])</code>	<b>Drops specified columns</b> from the DataFrame, use <code>inplace=True</code> to modify the original DataFrame
<code>df.index</code>	Returns the <b>index labels</b> of the DataFrame
<code>df.isnull()</code>	returns a DataFrame of <b>boolean values</b> , where each entry indicates whether the corresponding value in df is NaN (missing)



## Important Functions to Know (pt. 3)

Syntax	Description
<code>df['column'].value_counts() ()</code>	Returns the <b>count of unique values</b> in a specific column
<code>df['column'].mean()</code>	Returns the <b>mean value</b> of a numerical column
<code>df.corr()</code>	Computes <b>correlation</b> for numerical columns to understand relationships
<code>df.columns</code>	Lists all <b>column names</b> in the <b>DataFrame</b> , useful for renaming or viewing dataset structure
<code>df.shape</code>	Returns <b># of rows and columns</b>



## Important Functions to Know (pt. 4)

Syntax	Description
<code>df.rename(columns={'old' : 'new'}, inplace=True)</code>	Renames <b>specific columns</b> to new names
<code>df.groupby('category')[‘value’]</code>	Groups the DataFrame using a specified column to perform <b>aggregate functions</b> (e.g., <code>.sum()</code> , <code>.mean()</code> )
<code>df[‘new_column’] = df[‘column’].apply(function)</code>	<b>Applies</b> a function to each element or column/row
<code>df.shape</code>	Returns a <b>tuple</b> representing the <b>dimensions</b> (rows, columns) of the DataFrame
<code>df.dropna()</code>	Removes rows or columns containing <b>missing values</b>



## Important Functions to Know (pt. 5)

Syntax	Description
<code>df.to_numpy()</code>	Converts Pandas DataFrame into a <b>NumPy array</b>
<code>np.shape</code>	Returns a <b>tuple</b> representing the <b>dimensions</b> (rows, columns) of the array (similar to <code>df.shape</code> )
<code>np.reshape()</code>	<b>Reshapes</b> an original numpy array to the specified dimensions (rows, columns)
<code>np.array()</code>	Creates an <b>N-dimensional array</b> ( <code>ndarray</code> ) which is more <b>memory efficient</b> than standard python lists
<code>np.unique()</code>	Returns the <b>unique elements</b> in a NumPy array (also returns the <b>counts of each element</b> given the keyword argument, <code>return_counts=True</code> )



# How to Upload Files into Colab

Click on the folder in the sidebar



The screenshot shows the Google Colab interface with the following elements:

- Top Bar:** Includes "Commands", "+ Code", "+ Text", and "Run all" buttons.
- File Sidebar:** Shows a "Files" section with a folder icon, a file icon, and a refresh icon. A red arrow points to the folder icon. Below it is a "sample\_data" folder.
- Code Editor:** A section titled "Week 1 - Pandas Practice" contains the following code:

```
import pandas as pd
import seaborn as sns
```

# Allows you to provide a path to a Google Drive address rather than a local file path
from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive
- Output:** Displays the output of the code execution, showing the first few rows of a CSV file named "MDST Week 1 - Pandas Practice.csv".

```
df.head()
```

Timestamp	What is your name?	What is your major?	What is your graduate?	What year will you leave blank.	If you're American, what state are you from? If not, leave blank.	If you're American, what country are you from? If not, leave blank.	(Answer in state code)	(Approximately) How many years of coding experience do you have? If you're American, what country are you from? If not, leave blank.	How many pets do you have? If you're American, what country are you from? If not, leave blank.	How many credits are you taking? If you're American, what country are you from? If not, leave blank.	How many roommates do you have? If you're American, what country are you from? If not, leave blank.	Football games have you been to?	How many have you been to?	Michigan
2023-10-01 14:23:15.123	John Doe	Computer Science	Graduate	2025	Michigan	United States	Python	4 years	1 pet	15 credits	3 roommates	1 game	10 games	Michigan
- Bottom Navigation:** Includes "Variables", "Terminal", "Python 3", and disk usage information ("Disk 68.47 GB available").



# How to Upload Files into Colab

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

The screenshot shows the Google Colab interface. On the left, there's a sidebar titled 'Files' with a red arrow pointing to the 'Upload' button (a cloud icon). Below it, there's a folder named 'sample\_data' and a file named 'MDST Week 1 - Pandas Practice.csv'. The main area shows a code cell under the heading 'Week 1 - Pandas Practice'. It contains code to import pandas and seaborn, mount Google Drive, and read a CSV file. Below the code cell, instructions say to load the Google Forms .csv into a Pandas dataframe. Another code cell shows df.read\_csv('/content/MDST Week 1 - Pandas Practice.csv'). Further down, instructions say to print out the .head() and the datatypes. A code cell shows df.head(). The resulting output is a table of survey questions and their corresponding data types. At the bottom, it shows 'Disk' with 68.47 GB available, 'Variables' tab selected, 'Terminal' tab, and 'Python 3' notebook.

```
import pandas as pd
import seaborn as sns

# Allows you to provide a path to a Google Drive address rather than a local file path
from google.colab import drive
drive.mount('/content/drive')

df = pd.read_csv('/content/MDST Week 1 - Pandas Practice.csv')

df.head()
```

Timestamp	What is your name?	What is your major?	What year will you graduate?	If you're American, what state are you from? If not, what country are you from? If answer from? If in state code	(Approximately) How many years of coding experience do you have? If you're American, what country are you from? If answer from? If in state code	How many pets do you have? If you're American, what country are you from? If answer from? If in state code	How many credits are you taking this semester? If you're American, what country are you from? If answer from? If in state code	How many roommates do you have? If you're American, what country are you from? If answer from? If in state code	How many Football games have you been to? If you're American, what country are you from? If answer from? If in state code	How many Michigan games have you been to? If you're American, what country are you from? If answer from? If in state code
2023-09-01 12:00:00	John Doe	Computer Science	2024	Michigan, USA	1 year	1 pet	1 credit	1 roommate	1 game	1 game

Disk 68.47 GB available

Variables Terminal Python 3



# How to Upload Files into Colab

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



The screenshot shows the Google Colab interface. On the left, there's a sidebar titled 'Files' with a 'Mount' section. A red arrow points to the three-dot menu next to 'Mount'. The menu options include 'Download', 'Rename file', 'Delete file', 'Copy path', 'Refresh', and 'Add to Gemini'. Below the sidebar, the main workspace shows a code cell for 'Week 1 - Pandas Practice'. It contains code to import pandas and seaborn, and to mount Google Drive. A note explains that drive.mount('/content/drive') provides a path to a Google Drive address. Another code cell shows df.read\_csv('MDST Week 1 - Pandas Practice.csv'). A third code cell displays df.head(), showing a sample of a survey dataset with columns like Timestamp, What is your name?, What is your major?, What year will you graduate?, If you're American, what state are you from?, If not, leave blank., (Answer in state code), If you're not American, what country are you from? If American, (Approximately) How many years of coding experience do you have?, How many pets do you have?, How many credits are you taking this semester?, How many roommates do you have?, How many football games have you been to?, and How many Michigan games have you been to?. At the bottom, it says '(Approximate answers are ok.)'. The bottom navigation bar includes 'Variables', 'Terminal', and 'Python 3'.

```
import pandas as pd
import seaborn as sns

# Allows you to provide a path to a Google Drive address rather than a local file path
from google.colab import drive
drive.mount('/content/drive')

df = pd.read_csv('/content/MDST Week 1 - Pandas Practice.csv')

df.head()
```

If you're American, what state are you from? If not, leave blank. (Answer in state code)

Timestamp	What is your name?	What is your major?	What year will you graduate?	If you're American, what state are you from? If not, leave blank. (Answer in state code)	If you're not American, what country are you from? If American, (Approximately) How many years of coding experience do you have?	How many pets do you have?	How many credits are you taking this semester?	How many roommates do you have?	How many football games have you been to?	How many Michigan games have you been to?
2023-09-18 14:45:23.456Z	John Doe	Computer Science	2025	United States	United States	1	15	1	1	1



# How to Mount Your Google Drive

```
from google.colab import drive  
  
drive.mount('/content/drive')  
  
pd.read_csv('/content/drive/MyDrive/[FILE NAME]')
```

**\*\*NOTE:** If you saved your dataset in a folder (not just loose in your MyDrive folder), you will need to add further code to the file path before the file name in the final line. For example, if I saved my data called “alzheimers.csv” in a folder called “MDST-W26,” my read\_csv statement would read:

```
pd.read_csv('/content/drive/MyDrive/MDST-W26/alzheimers.csv')
```