# ScHARe

## Research Think-a-Thons

National Institutes of Health

# Be a Part of the Future of Knowledge Generation
# Part II

July 17, 2024

**Deborah Duran**, PhD · NIMHD
**Luca Calzoni**, MD MS PhD Cand. · NIMHD

# Look deeper with more eyes

*"For the first time in history, we have a technology (AI) that is opening our eyes to <u>who we are</u>, is changing us as we speak, and could allow us to play a conscious role in <u>who we want to become</u>."*

*Jennifer Aue*

IBM Director for AI Transformation

AI professor at the University of Texas


- **Diverse perspectives**
- **Bias mitigation strategies**
- **Research paradigm shift to Big Data**

# ScHARe

**S**cience
**c**ollaborative for
**H**ealth disparities and
**A**rtificial intelligence bias
**Re**duction

# Outline

| | |
|---|---|
| 10’ | Introduction |
| 25’ | What is ScHARe? |
| 10’ | The ScHARe Think-a-Thons |
| 35’ | Making data AI-ready |
| 25’ | Ethical and transparent AI |
| 15’ | Computational strategies: traditional statistics |
| 25’ | Computational strategies: AI and Machine Learning |
| 5’ | Resources |

# Experience poll

**Please check your level of experience with the following:**

| | None | Some | Proficient | Expert |
|---|---|---|---|---|
| Python | ☐ | ☐ | ☐ | ☐ |
| R | ☐ | ☐ | ☐ | ☐ |
| Cloud computing | ☐ | ☐ | ☐ | ☐ |
| Terra | ☐ | ☐ | ☐ | ☐ |
| Health disparities research | ☐ | ☐ | ☐ | ☐ |
| Health outcomes research | ☐ | ☐ | ☐ | ☐ |
| Algorithmic bias mitigation | ☐ | ☐ | ☐ | ☐ |

# Interest poll

**I am interested in (check all that apply):**

- ☐ Learning about Health Disparities and Health Outcomes research to apply my data science skills
- ☐ Conducting my own research using AI/cloud computing and publishing papers
- ☐ Connecting with new collaborators to conduct research using AI/cloud computing and publish papers
- ☐ Learning to use AI tools and cloud computing to gain new skills for research using Big Data
- ☐ Learning cloud computing resources to implement my own cloud
- ☐ Developing bias mitigation and ethical AI strategies
- ☐ Other

# ScHARe

## What is ScHARe?

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

**ScHARe is a cloud-based population science data platform** designed to accelerate research in health disparities, health and healthcare delivery outcomes, and artificial intelligence (AI) bias mitigation strategies

**ScHARe aims to fill four critical gaps:**

- Increase participation of **women & underrepresented populations with health disparities** in data science through data science skills training, cross-discipline mentoring, and multi-career level collaborating on research
- Leverage population science, SDoH, and behavioral Big Data and cloud computing tools to foster a **paradigm shift** in healthy disparity, and health and healthcare delivery outcomes research
- **Advance AI bias mitigation and ethical inquiry** by developing innovative strategies and securing diverse perspectives
- Provide a **data science cloud computing resource** for community colleges and low resource minority serving institutions and organizations

# ScHARe

nimhd.nih.gov/schare

# ScHARe

## Google Platform Terra Interface

- Secure workspaces
- Data storage
- Computational resources
- Tutorials (how to)
- Copy-and-paste code in Python and R
- Learning Terra on ScHARe prepares you to use other NIH platforms

Terra recommends using **Chrome**

Must have a **Gmail** friendly account

## PREPARING FOR AI RESEARCH AND HEALTHCARE USING BIG DATA

### Mapping across cloud platforms with Terra interface for collaborative research

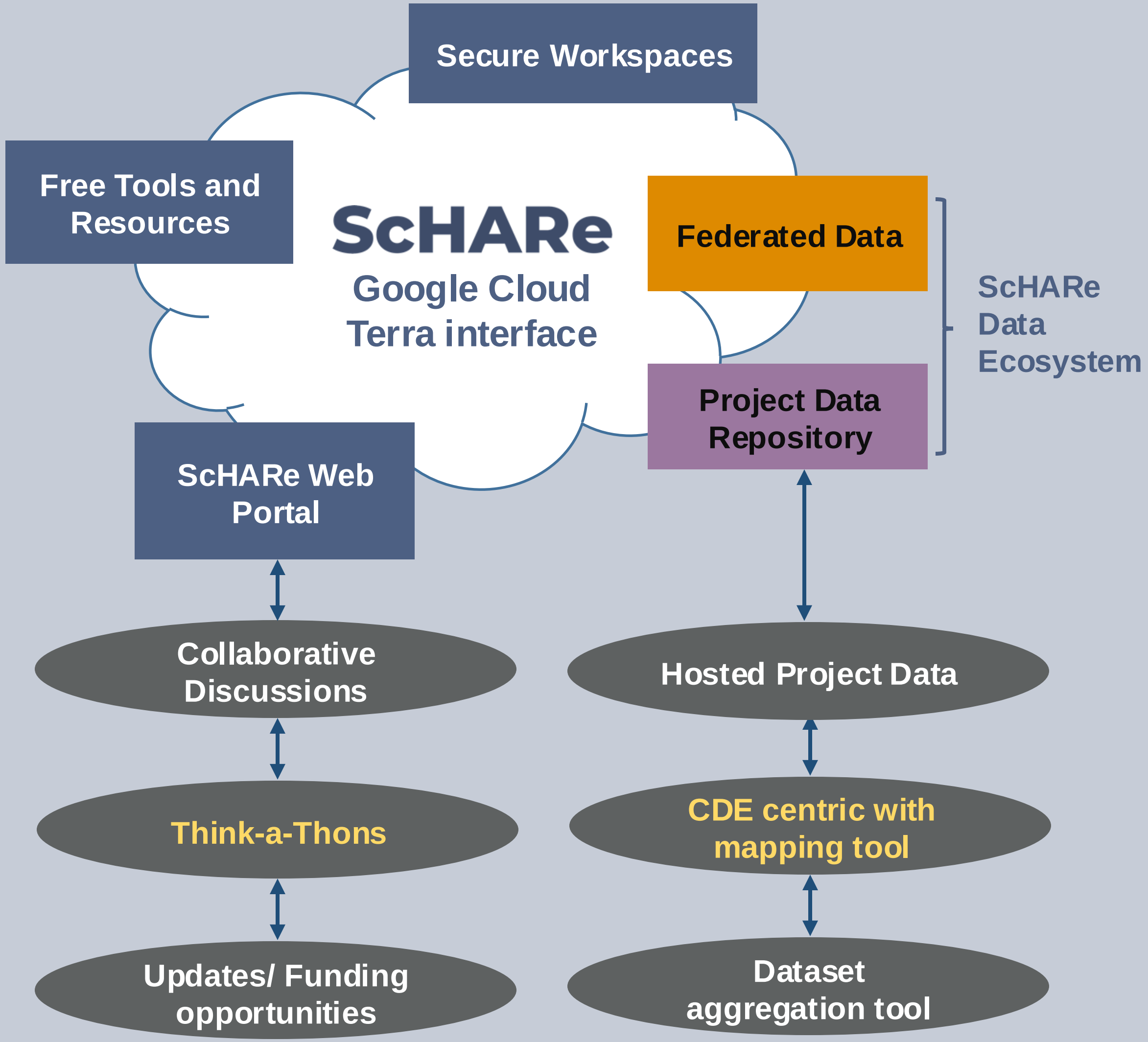
BE A PART OF THE FUTURE OF KNOWLEDGE GENERATION

# ScHARe Components

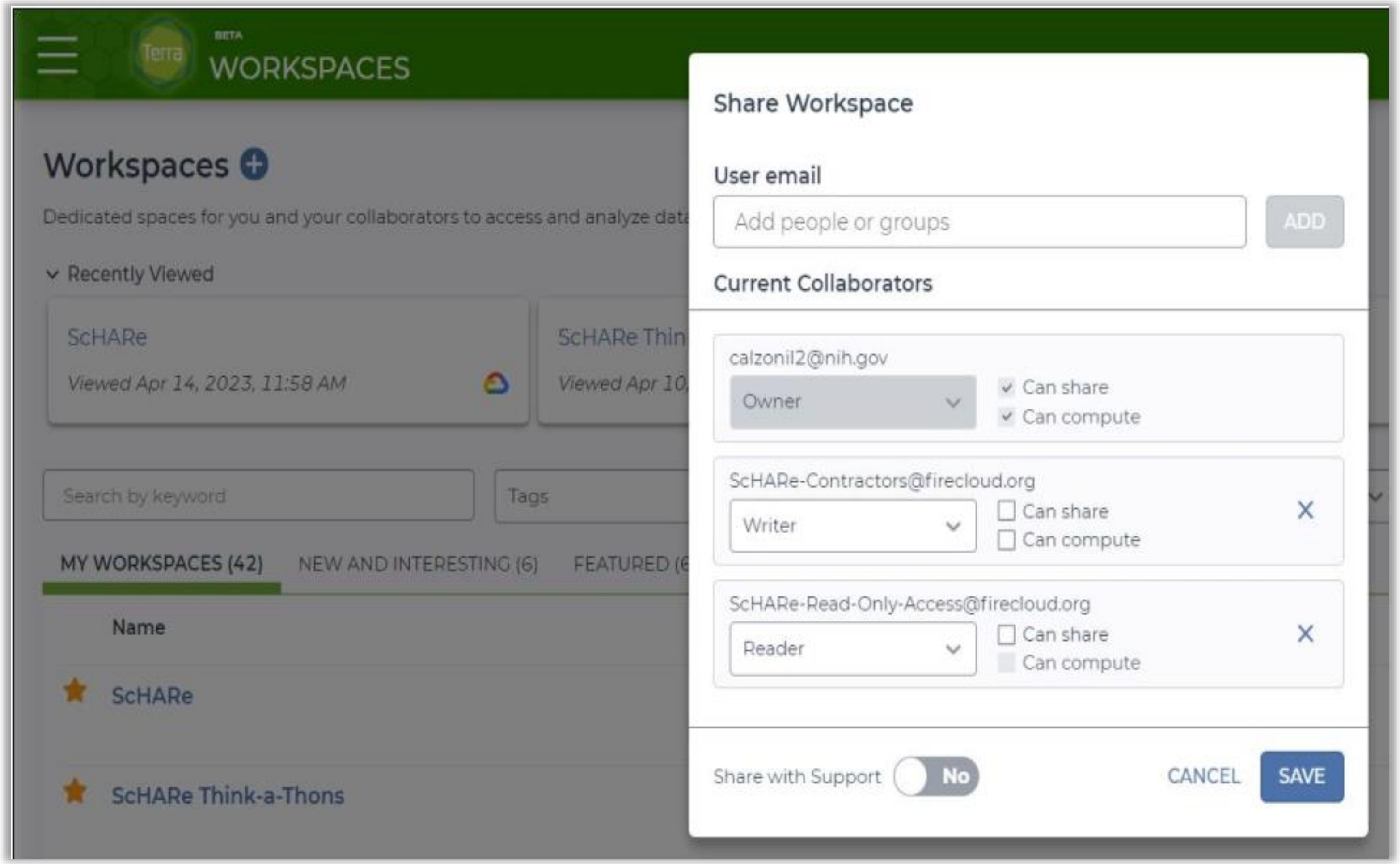
**ScHARe co-localizes within the cloud:**

1. **Datasets** (including social determinants of health and social science data) relevant to minority health, health disparities, and healthcare outcomes research
2. **CDE-focused data repository** to comply with the required hosting and sharing of data from NIMHD-/NINR-funded programs
3. **User-friendly computational capabilities and secure, collaborative workspaces** for students and all career level researchers
4. **Tools for collaboratively evaluating and mitigating biases** associated with datasets and algorithms utilized to inform healthcare and policy decisions (*upcoming*)

## Intramural and Extramural Resource



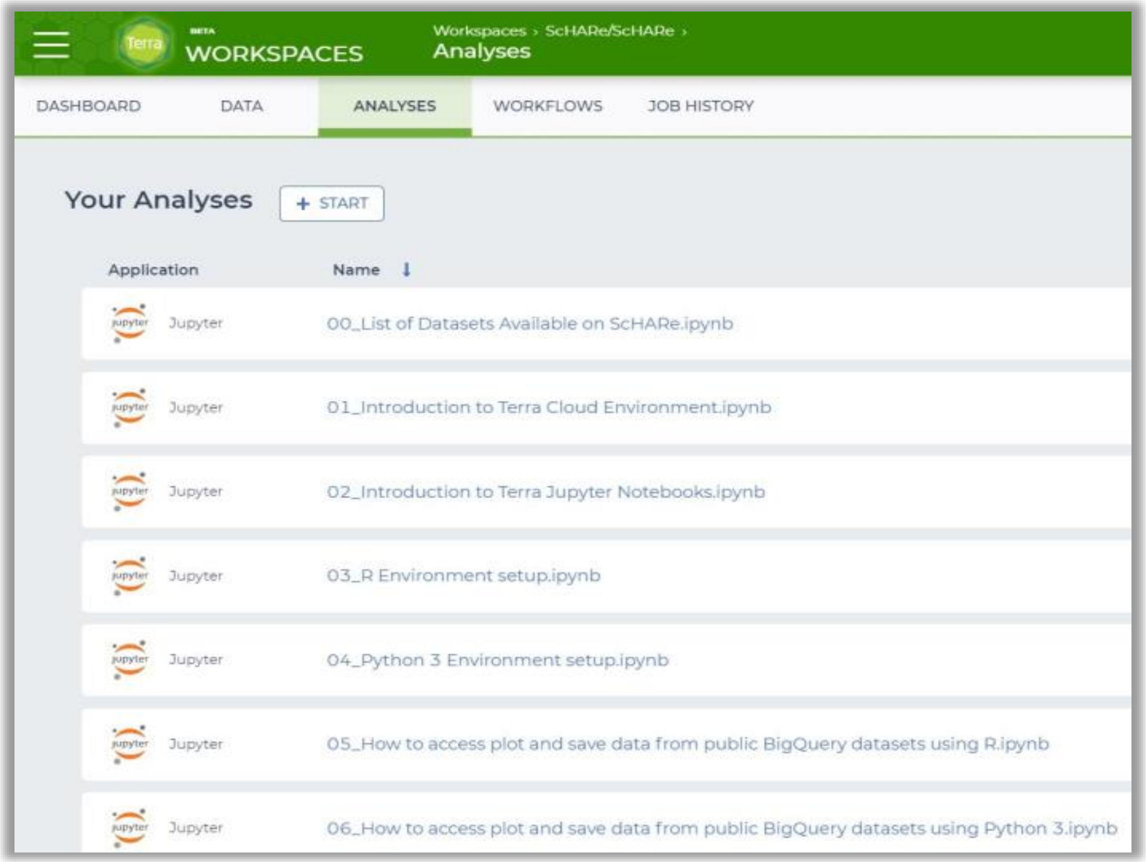
# ScHARe Terra interface: secure workspace




- Secure workspace **for self or collaborative research**
- **Assign roles**: review or admin
- **Host own data and code**

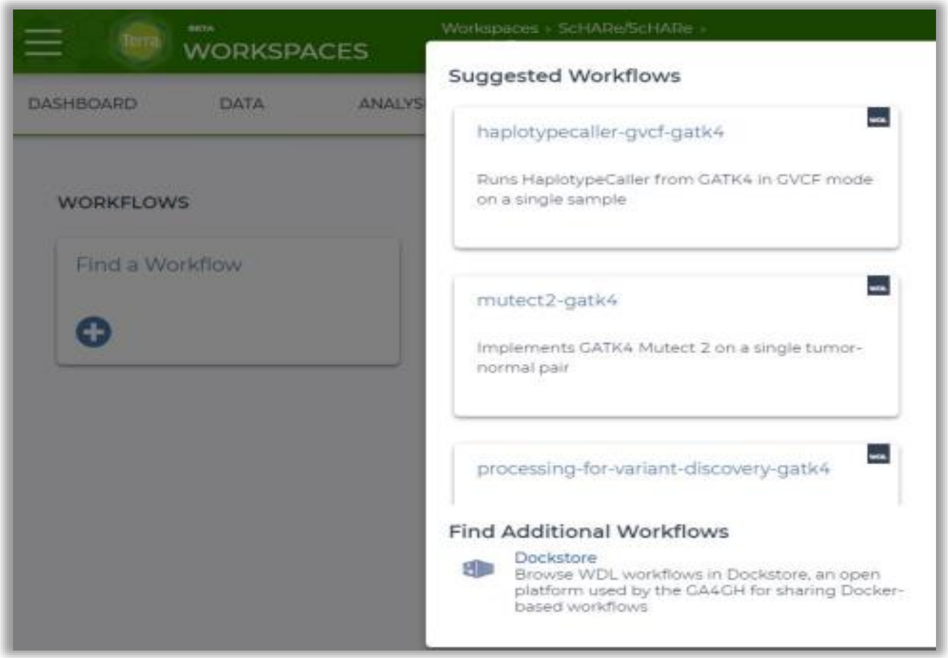
# ScHARe Terra interface: analyses

## Notebooks for analytics and tutorials




## Modular codes

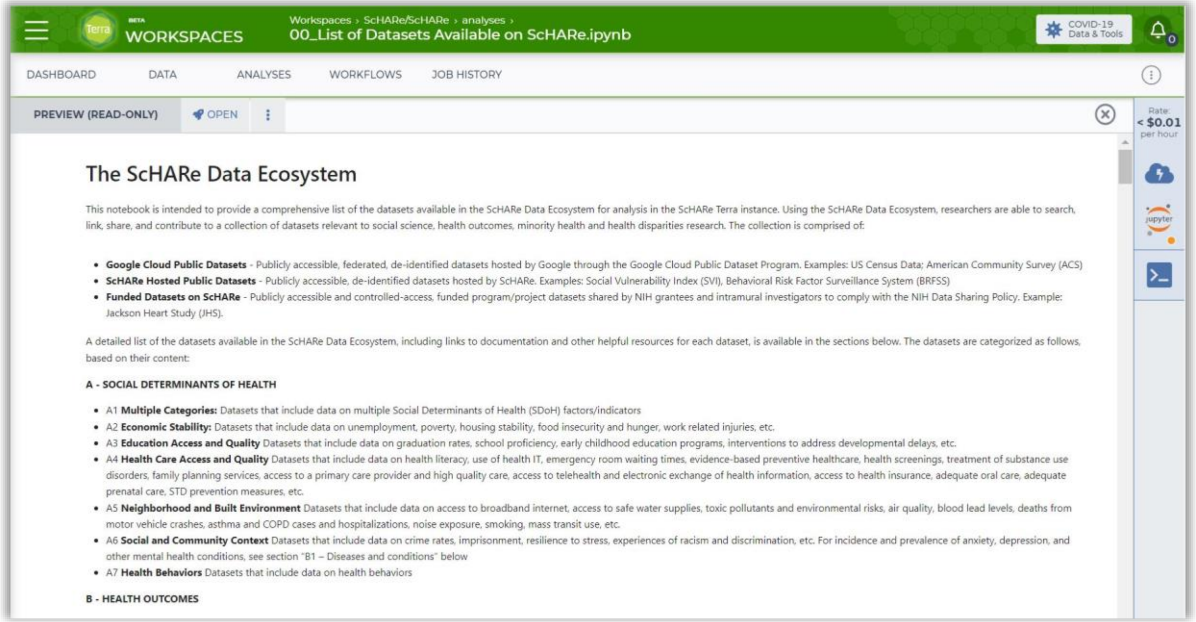
- **Easy-to-use copy-and-paste analytics**




- Modular codes developed for reuse
- **Adding SAS**

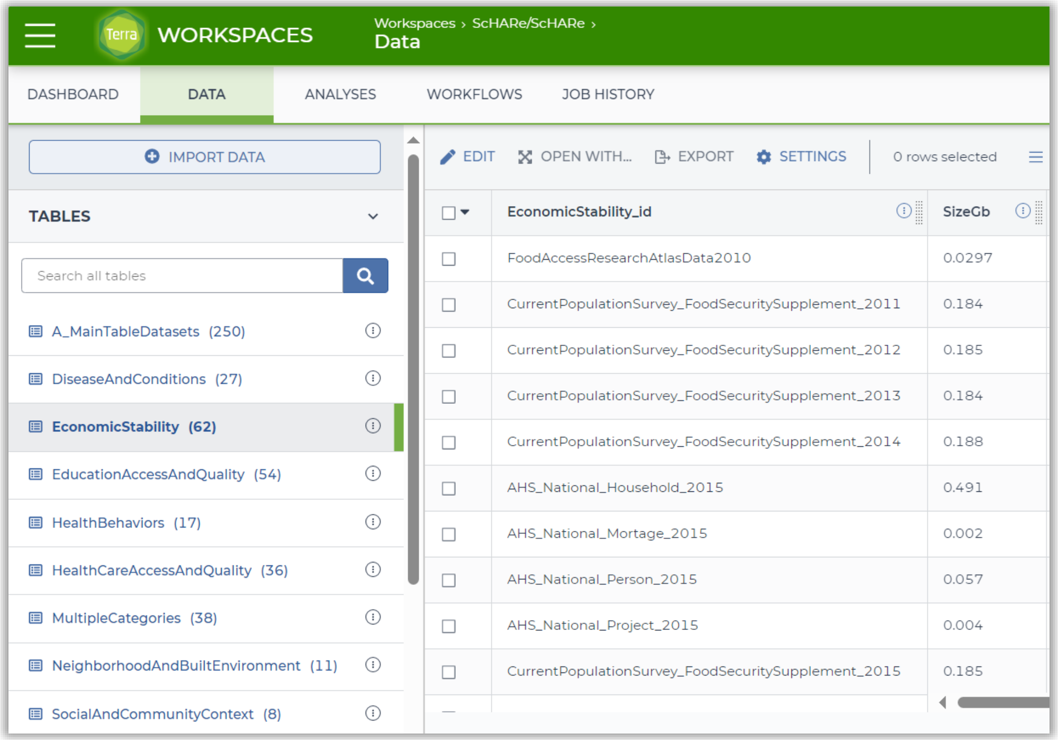
# **ScHARe** Terra interface: access to datasets

## What data?




In the **Analyses tab**, the notebook **00_List of Datasets Available on ScHARe** lists all datasets

## Where?




In the **Data tab**, data tables help access data

# ScHARe Ecosystem structure

Researchers can access, link, analyze, and export **a wealth of SDoH and population science related datasets** within and across platforms relevant to research about health disparities, health care delivery, health outcomes and bias mitigation, including:

**250+**
**FEDERATED PUBLIC DATASETS**

## Public datasets

Publicly accessible, federated, de-identified datasets hosted by ScHARe or hosted by Google through the Google Cloud Public Dataset Program

**ScHARe** **e.g.:** *Behavioral Risk Factor Surveillance System (BRFSS)*

**Google** **e.g.:** *American Community Survey (ACS)*

**CDE**
**FOCUSED REPOSITORY**

## Funded datasets

Publicly accessible and controlled-access, funded program/project datasets using Common Data Elements shared by NIH grantees and intramural investigators to comply with the NIH Data Sharing Policy

**e.g.:** *Jackson Heart Study (JHS)*
*Extramural Grant Data*
*Intramural Project Data*

**Innovative Approach:**
CDE Concept Codes
Uniform Resource Identifier (**URI**)

OVER 260 DATA SETS CENTRALIZED

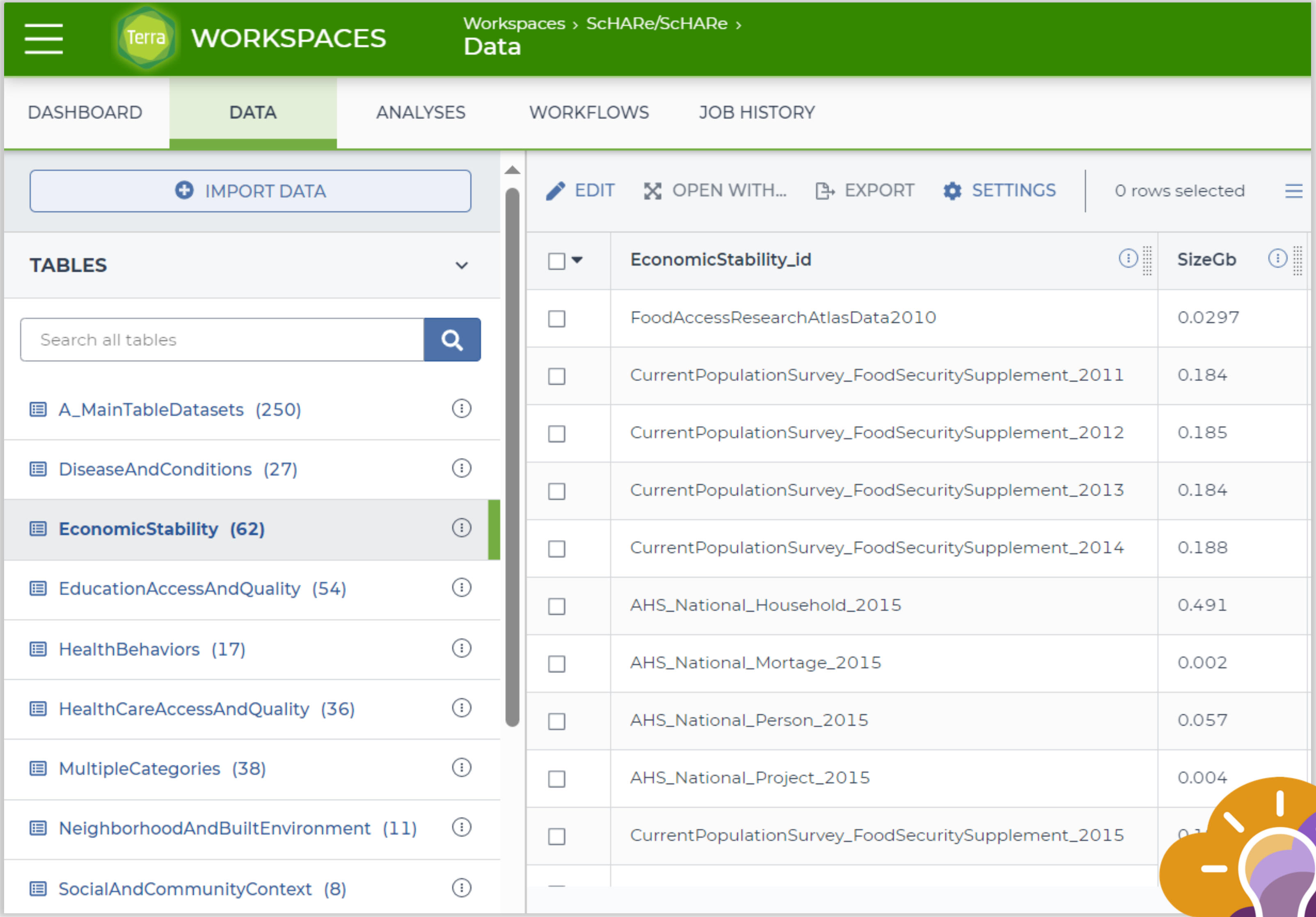
# ScHARe Ecosystem

Datasets are categorized by content based on the CDC **Social Determinants of Health categories**:

1. Economic Stability
2. Education Access and Quality
3. Health Care Access and Quality
4. Neighborhood and Built Environment
5. Social and Community Context

with the addition of:

- **Health Behaviors**
- **Diseases and Conditions**



# ScHARe Ecosystem: ScHARe hosted datasets

Organized based on the **CDC SDoH categories**, with the addition of *Health Behaviors* and *Diseases and Conditions*:

**260+** datasets

- **What are the Social Determinants of Health?**

Social determinants of health (SDoH) are the **nonmedical factors that influence health outcomes**

They are the **conditions in which people are born, grow, work, live, and age, and the wider set of forces and systems shaping the conditions of daily life**


www.cdc.gov/about/sdoh/index.html

# ScHARe Ecosystem: ScHARe hosted datasets

**Education access and quality**

Data on graduation rates, school proficiency, early childhood education programs, interventions to address developmental delays, etc.

**Economic stability**

Data on unemployment, poverty, housing stability, food insecurity and hunger, work related injuries, etc.

**Health care access and quality**

Data on health literacy, use of health IT, preventive healthcare, access to health insurance, etc.

*** Health behaviors**

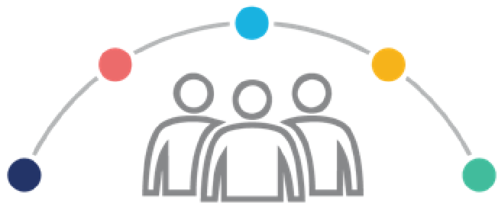
Data on health-related practices that can directly affect health outcomes.

**Neighborhood and built environment**

Data on access to safe water supplies, toxic pollutants and environmental risks, air quality, blood lead levels, noise exposure, smoking, mass transit use, etc.

*** Diseases and conditions**

Data on incidence and prevalence of specific diseases and health conditions.



**Social and community context**

Data on crime rates, imprisonment, resilience to stress, experiences of racism and discrimination, etc.

** Not Social Determinants of Health*

# ScHARe Ecosystem: Google hosted datasets

Examples of interesting datasets include:

- **American Community Survey** (U.S. Census Bureau)
- **US Census Data** (U.S. Census Bureau)
- **Area Deprivation Index** (BroadStreet)
- **GDP and Income by County** (Bureau of Economic Analysis)
- **US Inflation and Unemployment** (U.S. Bureau of Labor Statistics)
- **Quarterly Census of Employment and Wages** (U.S. Bureau of Labor Statistics)
- **Point-in-Time Homelessness Count** (U.S. Dept. of Housing and Urban Development)
- **Low Income Housing Tax Credit Program** (U.S. Dept. of Housing and Urban Development)
- **US Residential Real Estate Data** (House Canary)
- **Center for Medicare and Medicaid Services - Dual Enrollment** (U.S. Dept. of Health & Human Services)
- **Medicare** (U.S. Dept. of Health & Human Services)
- **Health Professional Shortage Areas** (U.S. Dept. of Health & Human Services)
- **CDC Births Data Summary** (Centers for Disease Control)
- **COVID-19 Data Repository by CSSE at JHU** (Johns Hopkins University)
- **COVID-19 Mobility Impact** (Geotab)
- **COVID-19 Open Data** (Google BigQuery Public Datasets Program)
- **COVID-19 Vaccination Access** (Google BigQuery Public Datasets Program)

# How to access Google hosted datasets

## Big Query

The Google public datasets are **available for access on Terra using BigQuery**

- BigQuery is the Google Cloud storage solution for structured data
- It is easy to use, works with large amounts of data and offers fast data retrieval and analysis
- Our **instructional notebooks in the Analyses tab** provide code and instructions on using Big Query to access Google datasets

Jupyter 06_How to access plot and save data from public BigQuery datasets using Python 3.ipynb

The following Python code will read a BigQuery table into a Pandas dataframe.

From https://cloud.google.com/community/tutorials/bigquery-ibis

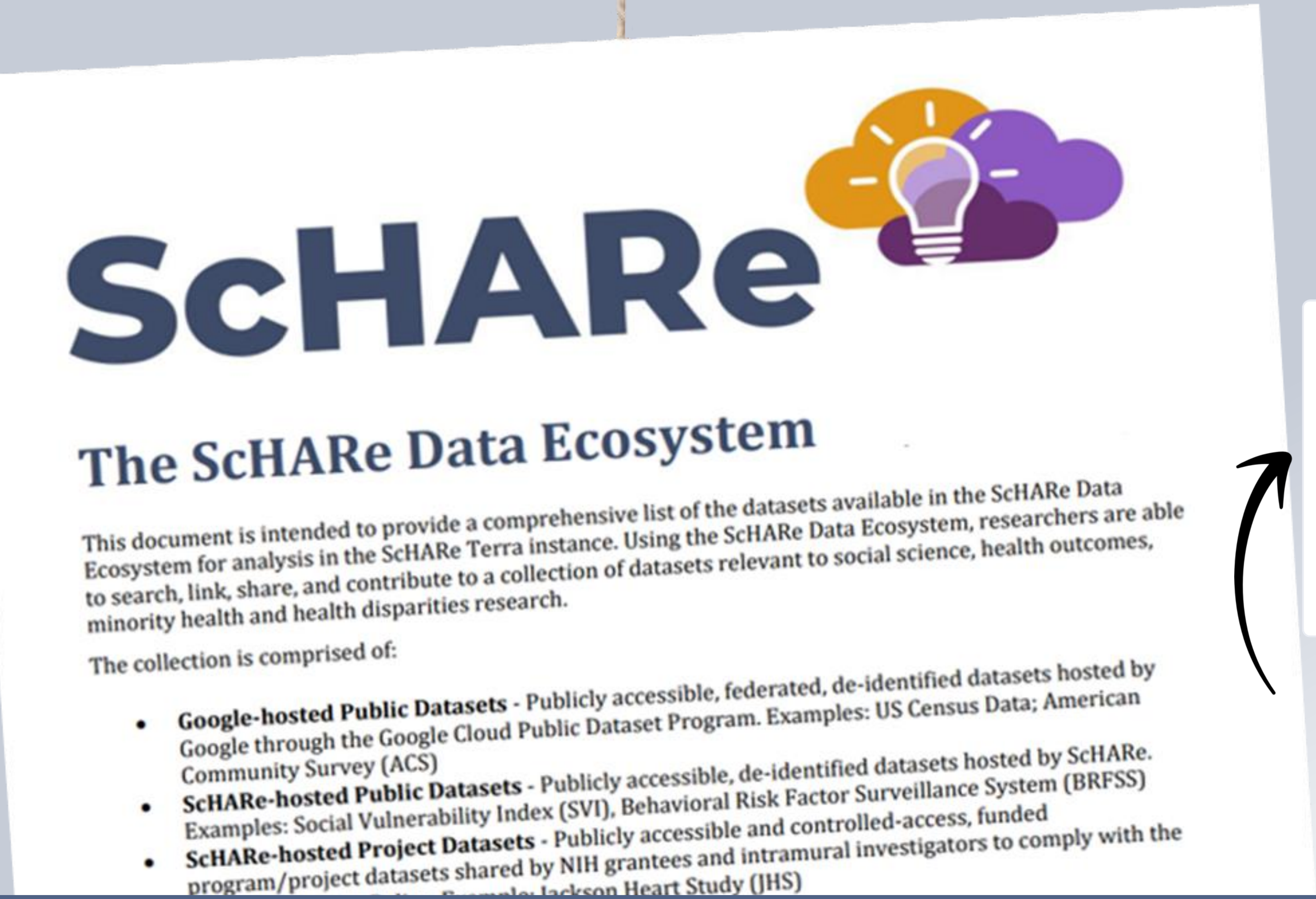
*Ibis is a Python library for doing data analysis. It offers a Pandas-like environment for executing data analysis composable, and familiar replacement for SQL.*

In [9]:
```
# Connect to the dataset
conn = ibis.bigquery.connect(dataset_id='bigquery-public-data.broadstreet_adi')
```

In [10]:
```
# Read table
ADI_table_2 = conn.table('area_deprivation_index_by_census_block_group')
ADI_table_2
```

Out[10]:
```
BigQueryTable[table]
  name: bigquery-public-data.broadstreet_adi.area_deprivation_index_by_census_block_group
  schema:
    geo_id : string
    state_fips_code : string
    county_fips_code : string
    block_group_fips_code : string
    description : string
    county_name : string
    state_name : string
    state : string
    year : int64
    area_deprivation_index_percent : float64
```

# ScHARe
## Datasets PDF list



# ScHARe

## The ScHARe Data Ecosystem

This document is intended to provide a comprehensive list of the datasets available in the ScHARe Data Ecosystem for analysis in the ScHARe Terra instance. Using the ScHARe Data Ecosystem, researchers are able to search, link, share, and contribute to a collection of datasets relevant to social science, health outcomes, minority health and health disparities research.

The collection is comprised of:

- **Google-hosted Public Datasets** - Publicly accessible, federated, de-identified datasets hosted by Google through the Google Cloud Public Dataset Program. Examples: US Census Data; American Community Survey (ACS)
- **ScHARe-hosted Public Datasets** - Publicly accessible, de-identified datasets hosted by ScHARe. Examples: Social Vulnerability Index (SVI), Behavioral Risk Factor Surveillance System (BRFSS)
- **ScHARe-hosted Project Datasets** - Publicly accessible and controlled-access, funded program/project datasets shared by NIH grantees and intramural investigators to comply with the ... Jackson Heart Study (JHS)

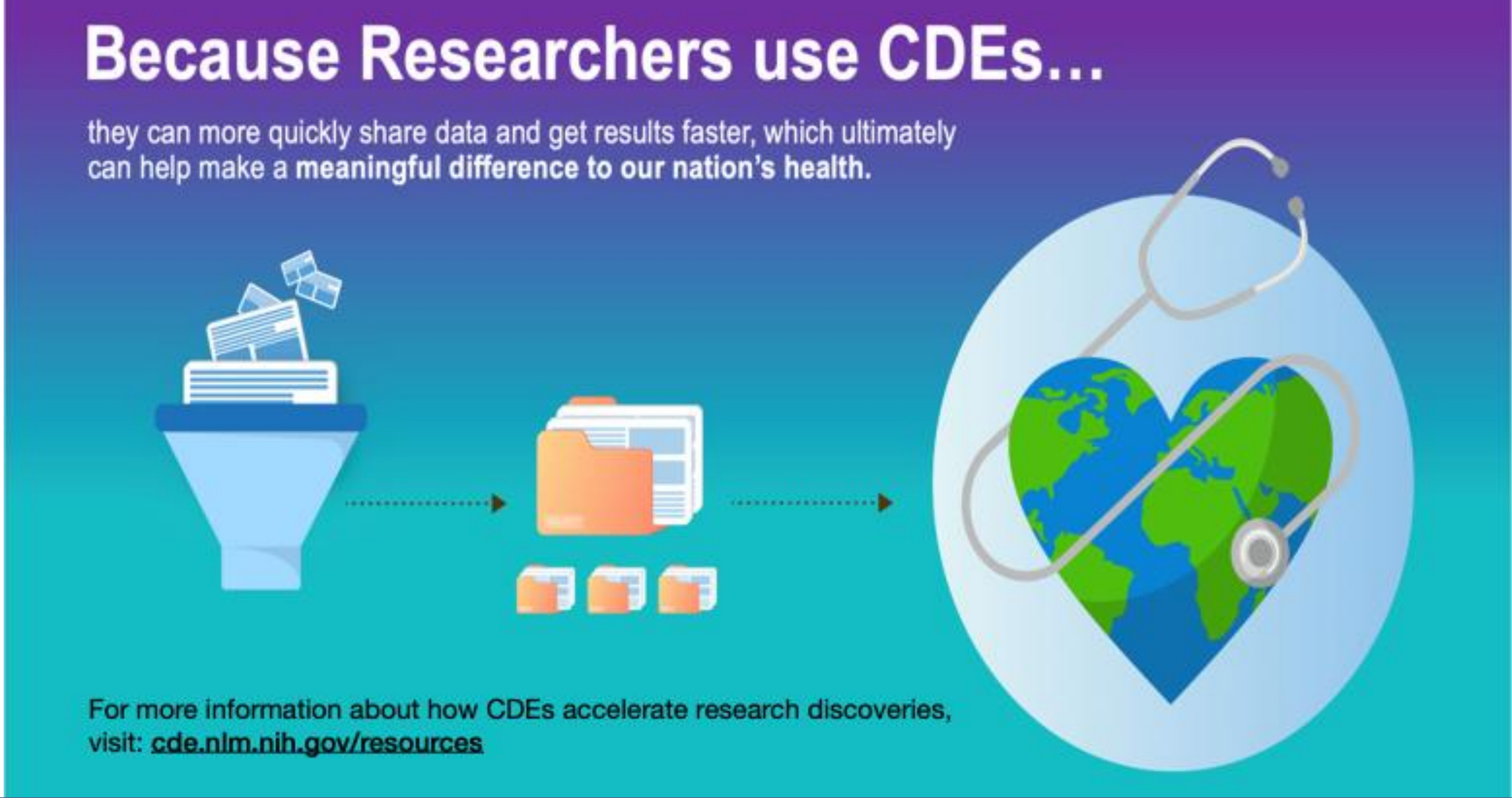
Scan me

bit.ly/ScHARe-datasets

**CDE benefits:**

- Faster start-up for project
- Better data aggregation across projects
- Shared meaning
- Concept-focused to allow questions/answers variations
- Coding enables an URI approach for better data interoperability

A **Common Data Element** (CDE) is a standardized, precisely defined question, paired with a set of allowable responses, used systematically across different sites, studies, or clinical trials to ensure consistent data collection



# ScHARe Core CDEs

PhenX Toolkit

**NIH Endorsed**

- Age
- Birthplace
- Zip Code
- Race and Ethnicity
- Sex
- Gender
- Sexual Orientation
- Marital Status
- Education
- Annual Household Income
- Household Size

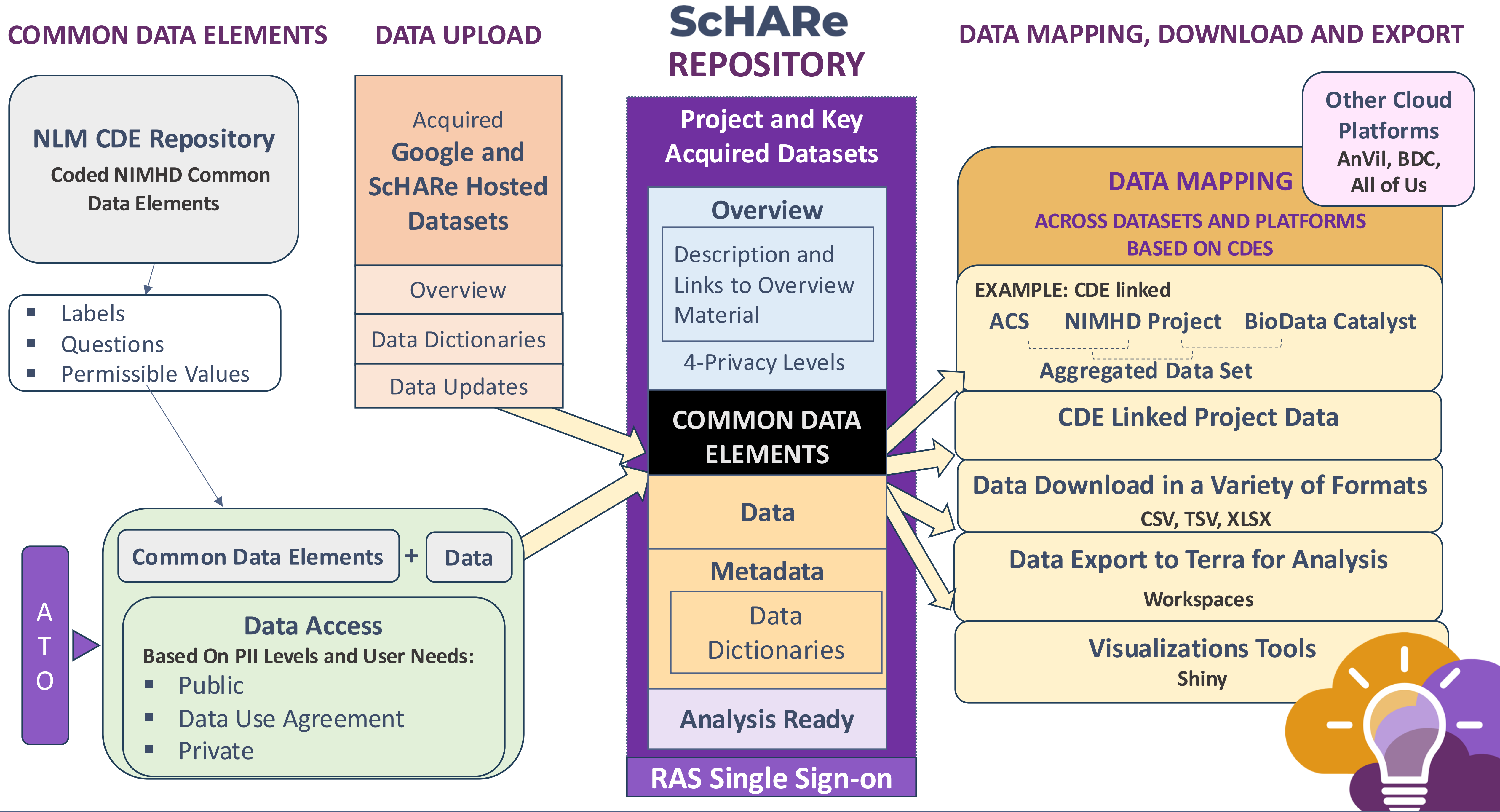
- English Proficiency
- Disabilities
- Health Insurance
- Employment Status
- Usual Place of Health Care
- Financial Security / Social Needs
- Self-Reported Health
- Health Conditions (and Associated Medications/Treatments)

- **NIMHD Framework***
- **Health Disparity Outcomes***

* Project Level CDEs

ScHARe has developed **Common Data Elements** to ensure consistent data collection across studies, facilitate interoperability, and link data from different sources

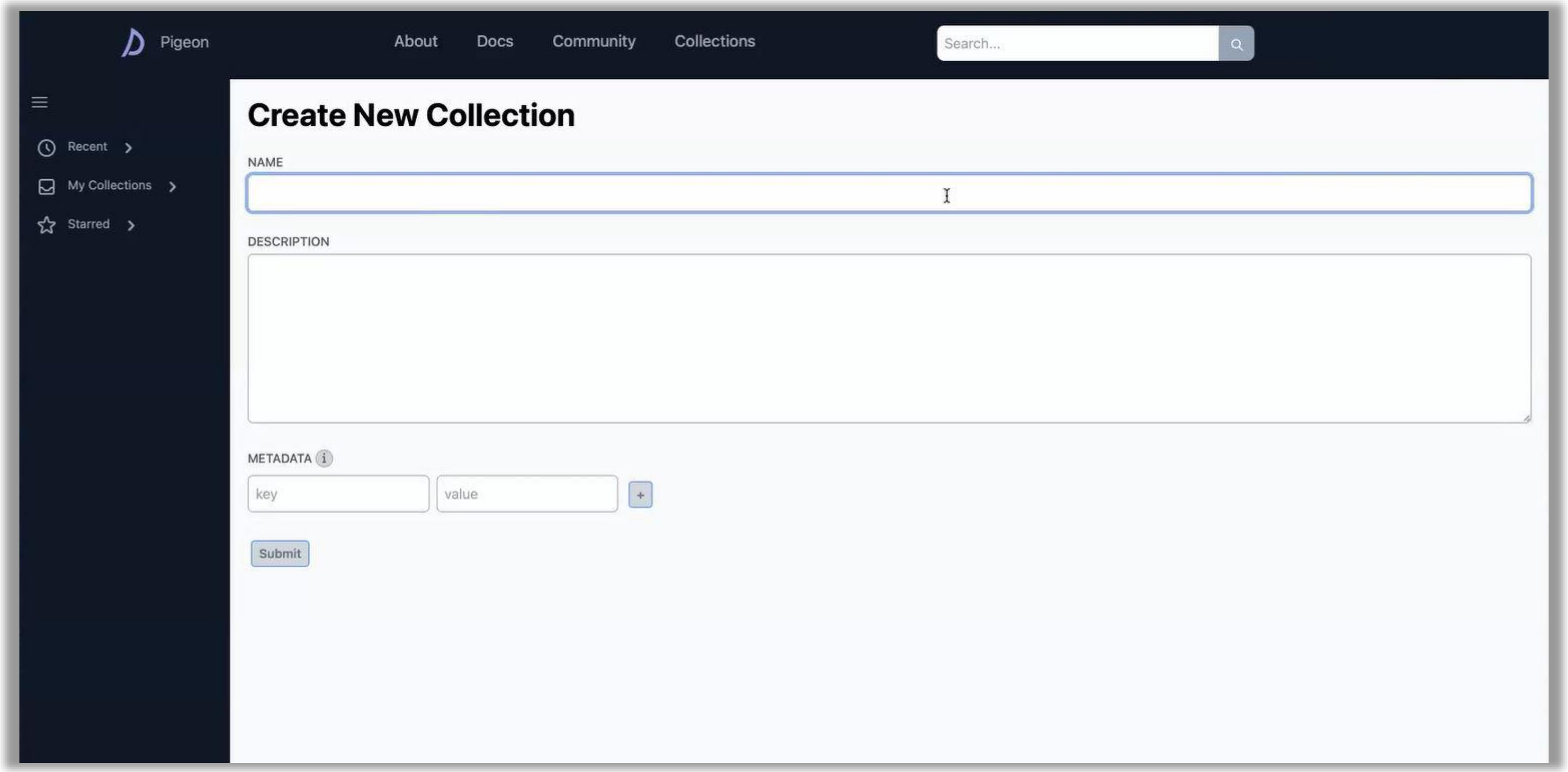
**NIH CDE Repository:**
cde.nlm.nih.gov/home
**PhenX Toolkit:**
www.nimhd.nih.gov/resources/phenx/



COMMON DATA ELEMENTS
DATA UPLOAD
ScHARe REPOSITORY
DATA MAPPING, DOWNLOAD AND EXPORT
NLM CDE Repository
Coded NIMHD Common Data Elements
Labels
Questions
Permissible Values
Acquired Google and ScHARe Hosted Datasets
Overview
Data Dictionaries
Data Updates
Common Data Elements
+
Data
Data Access
Based On PII Levels and User Needs:
Public
Data Use Agreement
Private
A T O
Project and Key Acquired Datasets
Overview
Description and Links to Overview Material
4-Privacy Levels
COMMON DATA ELEMENTS
Data
Metadata
Data Dictionaries
Analysis Ready
RAS Single Sign-on
Other Cloud Platforms
AnVil, BDC, All of Us
DATA MAPPING
ACROSS DATASETS AND PLATFORMS BASED ON CDES
EXAMPLE: CDE linked
ACS
NIMHD Project
BioData Catalyst
Aggregated Data Set
CDE Linked Project Data
Data Download in a Variety of Formats
CSV, TSV, XLSX
Data Export to Terra for Analysis
Workspaces
Visualizations Tools
Shiny

# ScHARe Repository

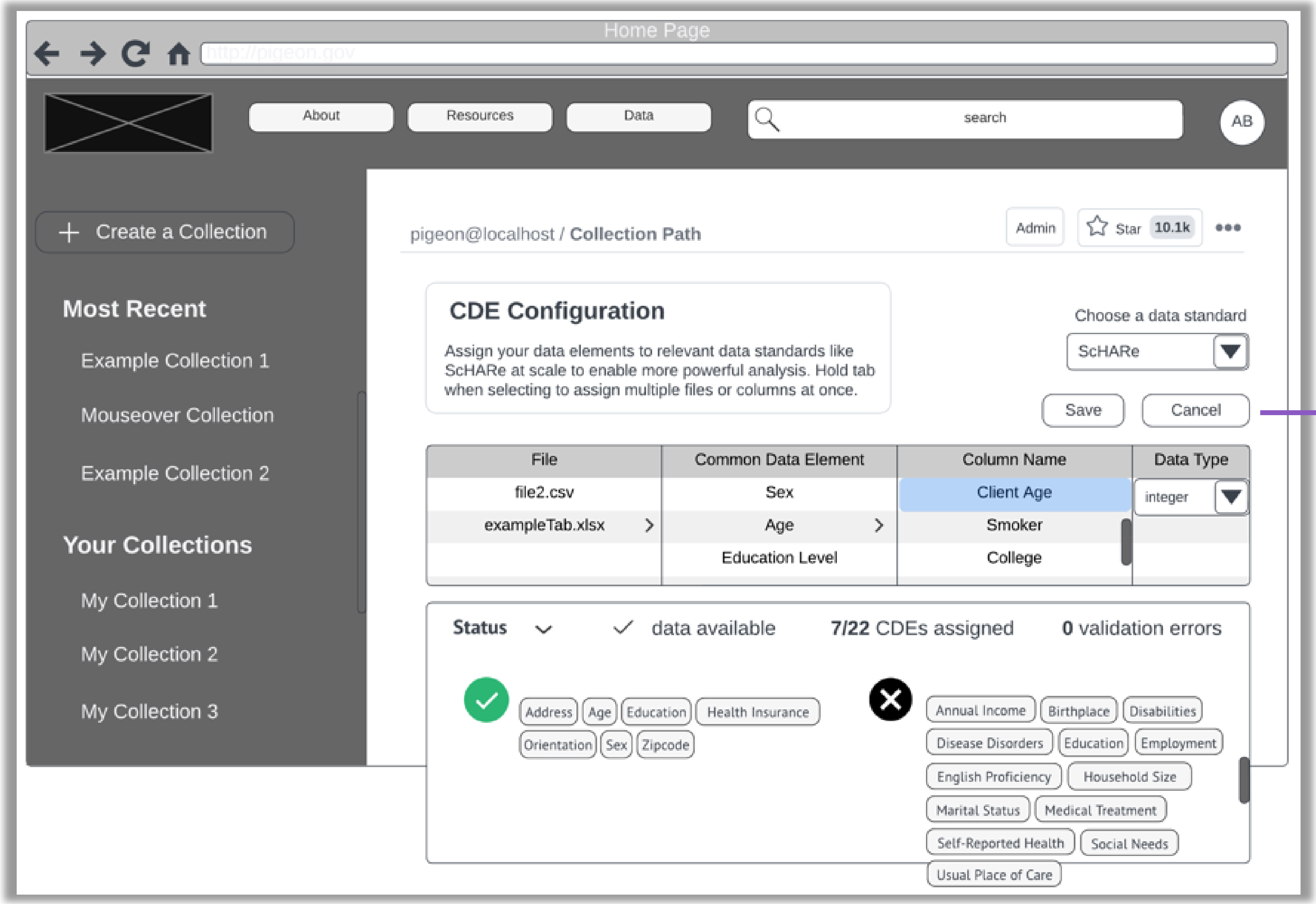
PUBLICLY AVAILABLE FALL 2024




- Host your project data in a **safe space** with privacy levels, secure workspaces, collaboration platform
- **CDE centric**
- **Focus**: Social Science, SDoH, Health Disparities, Health Outcomes Research
- Comply with **NIH Data Management and Data Sharing Policy**
- **Link your data** with others and federated data

# ScHARe Repository

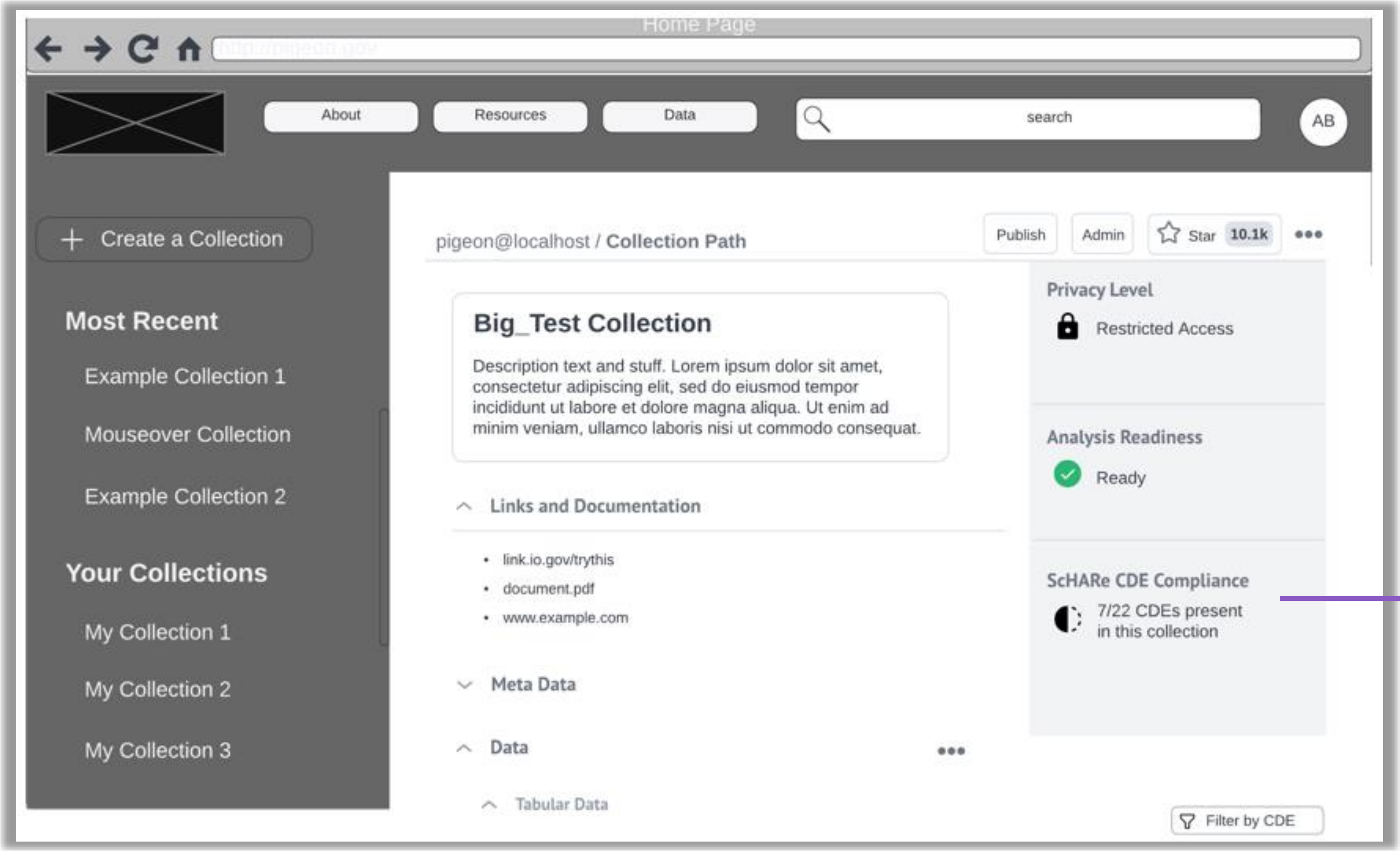
PUBLICLY AVAILABLE FALL 2024




Map project CDEs or variables to ScHARe-PhenX CDEs

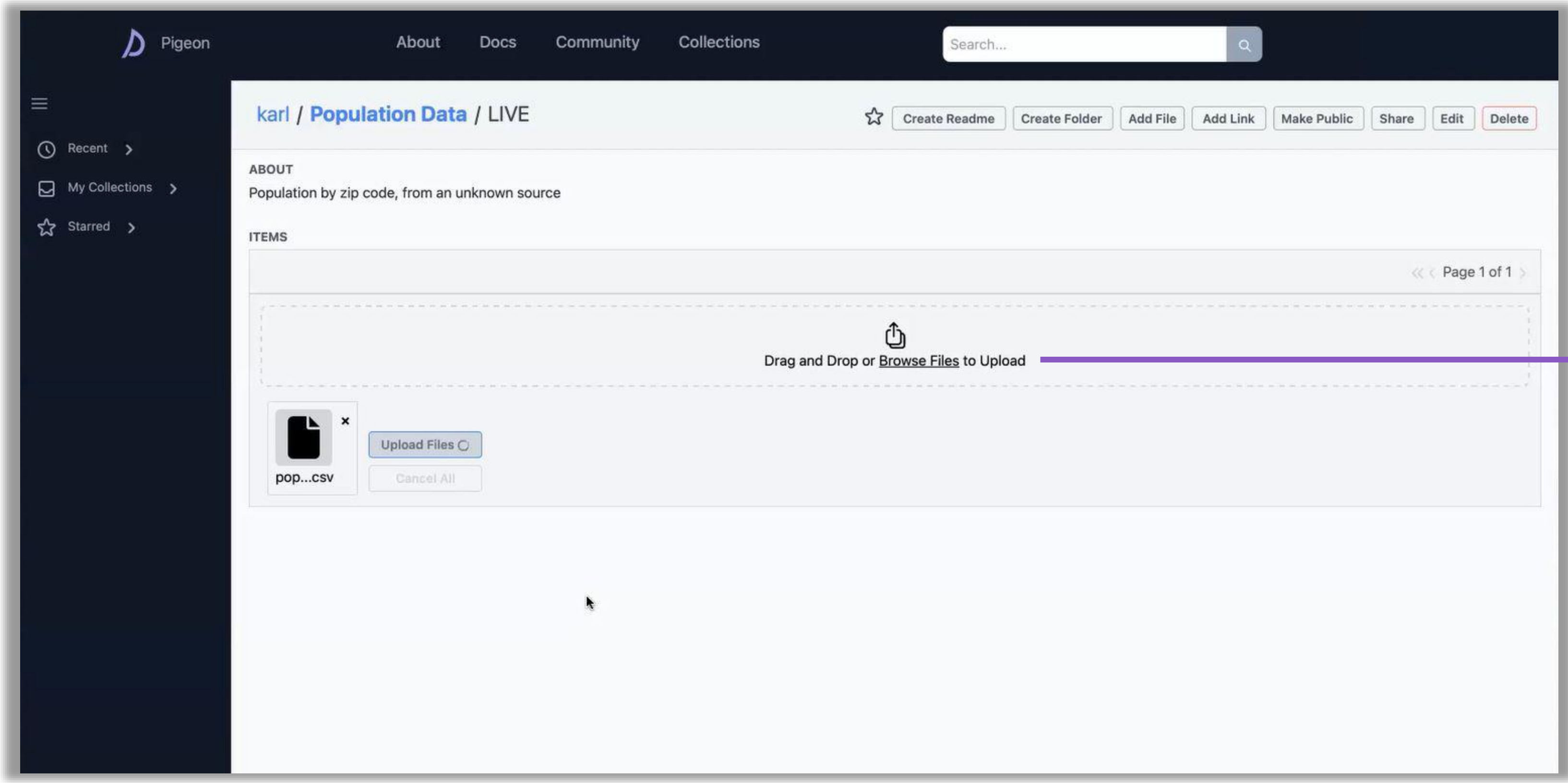
# ScHARe Repository

PUBLICLY AVAILABLE FALL 2024




Shows number of project CDEs that match or can map to ScHARe-PhenX CDEs

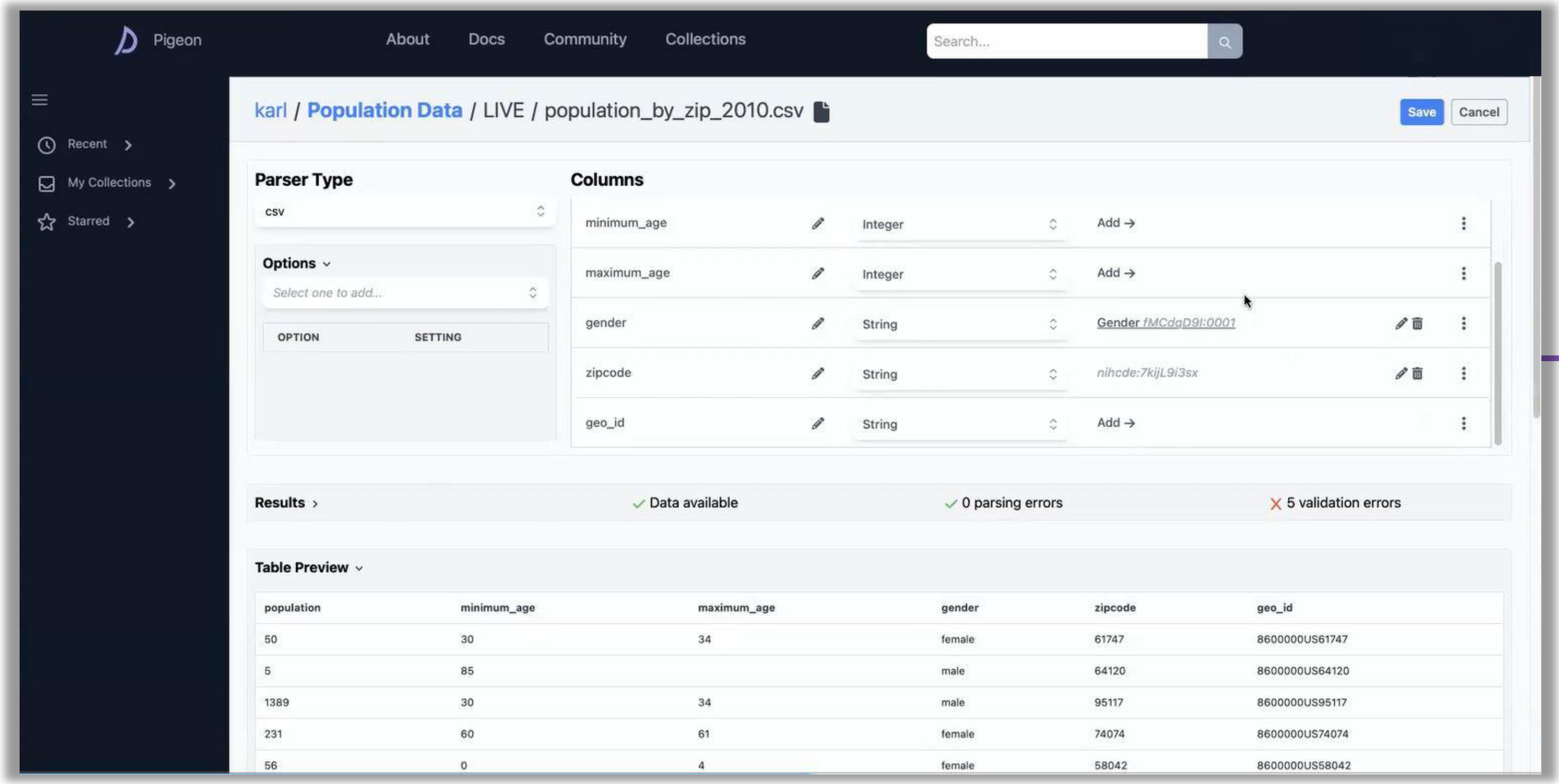
# ScHARe Repository




Aggregate datasets with drag-and-drop features

# ScHARe Repository

PUBLICLY AVAILABLE FALL 2024




View aggregated dataset

# ScHARe

## Research Think-a-Thons

- Novice **training webinars** for data science, cloud computing and research using Big Data
- **Target:** underrepresented populations, women, racial/ethnic and sexual gender minorities, rural and poor populations

# Generational career & discipline exchange

# Think-a-Thons

3rd Wednesday of every month
2 pm

## Goals:

- Upskill underrepresented populations in data science and cloud computing
- Foster a research paradigm shift to use Big Data in health disparities/health outcomes research
- Promote use of Dark Data

## 1. TUTORIAL AND TARGETED THINK-A-THONS

Launched April 2024

- Monthly sessions (2 1/2 hours)
- Instructional/interactive
- Designed for new/experienced users
- Networking
- Mentoring and coaching
- Topics include:
  - Data Science 101
  - Terra
  - Social Determinants of Health analytics
  - Common Data Elements
  - AI readiness
  - Ethical and transparent AI
  - Bias mitigation

## 2. RESEARCH THINK-A-THONS

- Multi-career (students to senior investigators)
- Multi-discipline (data scientists and researchers)
- Featured datasets with guest experts leads
- Guest experts in topic areas, analytics, data sources etc. to provide guidance
- Generate research idea - decide design, datasets and analytics
- Learn Ethical AI
- Publications

**Register:**

**bit.ly/think-a-thons**

# Think-a-Thon tutorials

bit.ly/think-a-thons

| | |
|---|---|
| February | **Artificial Intelligence and Cloud Computing 101** |
| March | **ScHARe 1 – Accounts and Workspaces** |
| April | **ScHARe 2 – Terra Datasets** |
| May | **ScHARe 3 – Terra Google-hosted Datasets** |
| June | **ScHARe 4 – Terra ScHARe-hosted Datasets** |
| July | **An Introduction to Python for Data Science – Part 1** |
| August | **An Introduction to Python for Data Science – Part 2** |
| September | **ScHARe 5: A Review of the ScHARe Platform and Data Ecosystem** |
| October | **Preparing for AI 1: Common Data Elements and Data Aggregation** |
| November | **Preparing for AI 2: An Introduction to FAIR Data and AI-ready Datasets** |
| January | **Preparing for AI 3: Computational Data Science Strategies 101** |
| February/March | **Preparing for AI 4: Overview Prep for AI Summary with Transparency, Privacy, Ethics** |
| April | **Research Teams – SDoH and Health Disparities** |
| May | **Be a Part of the Future of Knowledge Generation 1: AI/Cloud Computing Basics and CDEs** |
| July | **Be a Part of the Future of Knowledge Generation 2: AI-Ready Datasets and Computations** |

## SPECIAL EVENTS

- ScHARe for **Educators** (Community Colleges and low-resource MSIs)
- ScHARe for **American Indian/ Alaska Native Researchers**
- ScHARe for **Coders and Programmers** to conduct research

# Experience conducting ethical AI

## Transparency

*Public perception and understanding of how AI works*

- **Technical documentation for duplication/re-use**
- **Tools:**
  - Data dictionary
  - Health sheet (Data sheet)
  - Model cards (capabilities and purpose of algorithms are openly and clearly communicated to relevant stakeholders)

## Fairness

***F indable:*** *providing metadata, documentation, and clear identifiers*
***A ccessible:*** *wide audience*
***I nteroperable:*** *standardized formats and APIs enable seamless integration*
***R eusable:*** *clear documentation, licensing, reduce redundancy*

- Metadata and data should be easy to find for both humans and computers
- Ensure that data represents relevant populations

# ScHARe

## Training pipeline

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Think-a-Thons training/mentoring pipeline

NLM
OIC Experts
Fellows

+

Think-a-Thons

✓ Instructional
✓ Research

+

AIM-AHEAD

+

AnVil | HEAL
N3C | All of US
BioData Catalyst

*Using AI experts*

*to train and mentor novice AI users*

*to upskill and mentor diverse perspectives in AI*

*to increase diverse perspectives in biomedical research*

**Goal: "Upskilling"**

- ✓ Data science specialists into health disparities and health outcomes research
- ✓ Health disparities/outcomes researchers into using big data and cloud computing

**Target Audience:**

- ✓ Underrepresented populations (women, race/ethnic) users not trained in data science
- ✓ Data scientists with no or little research experience
- ✓ Resource and tool for Community Colleges and low-resource MSIs and organizations

# Join AIM-AHEAD Connect

- AIM-AHEAD's community, networking, mentoring, and career development platform
- Virtual space to engage with the entire AIM-AHEAD Consortium and build community!
- Custom tools available to the AIM-AHEAD Coordinating Center:
    - Connect with experts, learners, stakeholders, etc.
    - Mentoring, Q&A, video calls, groups, funding & jobs board, etc.
    - SignUp: Event registration & information solicitation
    - Surveys: Request feedback on various activities
    - HelpDesk: Respond to topic-specific questions
    - Programs: Collaborative space, exclusive content, and mentor matching

**Scan QR Code to Join AIM-AHEAD Connect**

# ScHARe

## Making data AI-ready

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Working with Big Data

**Extremely large datasets** that are statistically analyzed to gain detailed insights, often **using AI** and substantial **computer-processing power**

Datasets can be **linked together (data integration)** to provide a comprehensive perspective for research knowledge generation (this includes data from RO1s, U54s, PARs, KO1s, etc.)

**Data integrity (data quality)** is the overarching completeness, accuracy, consistency, accessibility, and security of the data for its intended purpose.
This should always be assessed before using a dataset

**FAIR data** are data which meet machine-actionability principles of:

- Findability
- Accessibility
- Interoperability
- Reusability


Big Data is characterized by the 4 V's:

1. **Volume**: Enormous amounts of data
2. **Variety**: Diverse data types and structures
3. **Velocity**: High-speed data generation
4. **Veracity**: Challenges in ensuring data accuracy and reliability

**Big data is difficult to process using traditional methods**

# Big Data: structured and unstructured data

**Structured data** is quantitative data that is organized and easily searchable

Some tools used to work with structured data include:
- OLAP
- MySQL
- PostgreSQL
- Oracle Database

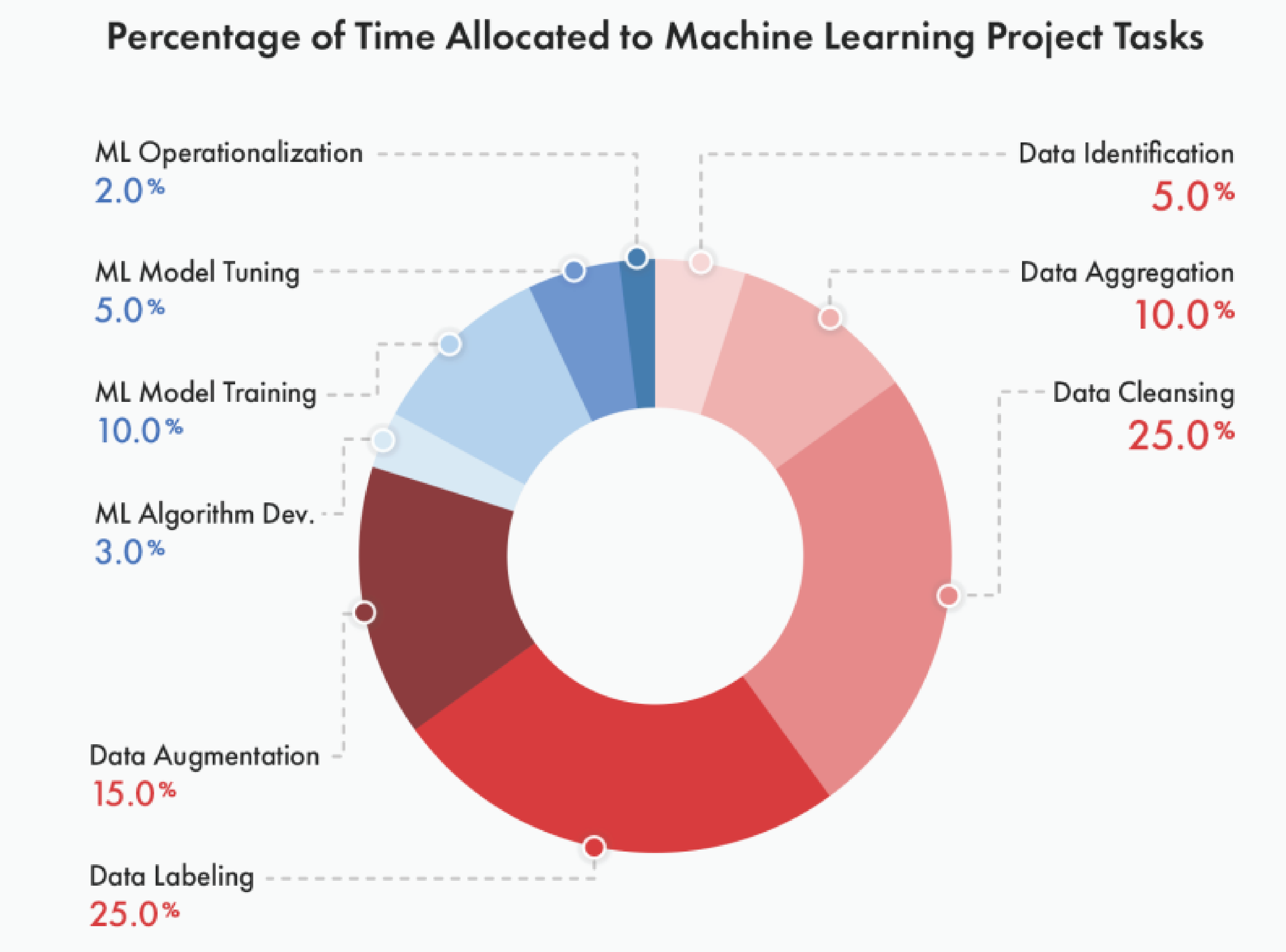

**Unstructured data** is every other type of data that is not structured.

Some tools used to manage unstructured data include:
- MongoDB
- Hadoop
- Azure


| | Structured data | Unstructured data |
|---|---|---|
| Main characteristics | Searchable<br>Usually text format<br>Quantitative | Difficult to search<br>Many data formats<br>Qualitative |
| Storage | Relational databases<br>Data warehouses | Data lakes<br>Non-relational databases<br>Data warehouses<br>NoSQL databases<br>Applications |
| Used for | Inventory control<br>CRM systems<br>ERP systems | Presentation or word processing software<br>Tools for viewing or editing media |
| Examples | Dates, phone numbers, bank account numbers, product SKUs | Emails, songs, videos, photos, reports, presentations |

**Big data is difficult to process using traditional methods**

# Preparing for AI




**Clean data** leads to more accurate, reliable, and effective AI models. Cleaning data is crucial for AI use because:

- **Accuracy:** Ensures the data is correct, improving the reliability of AI models.
- **Consistency:** Eliminates discrepancies, making the data uniform and easier to analyze.
- **Performance:** Reduces noise and irrelevant information, enhancing model efficiency.
- **Trustworthiness:** Increases confidence in the results produced by AI systems.
- **Compliance:** Helps in adhering to data quality standards and regulations.
- **Bias reduction:** Minimizes biases, leading to fairer outcomes.

Quality data can also be **aggregated with other quality data** for AI use

**Good data starts with your dataset**

# Making datasets AI-ready: a multifaceted approach

Making datasets AI-ready involves ensuring they are suitable for use in machine learning and artificial intelligence applications.

Key aspects of AI-ready datasets:

- **Data quality:** Ensure data accuracy, completeness, and consistency. Address missing values, outliers, and inconsistencies that could impact model performance.
- **Data cleaning and pre-processing:** Apply techniques like normalization, scaling, and encoding to prepare the data for machine learning algorithms.
- **Feature engineering:** Create new features from existing data or transform existing features to improve model performance.
- **Documentation:** Provide clear and detailed documentation about the dataset, including variable definitions, data collection methods, and any transformations applied.

# Why quality checks are essential for AI-ready data

Datasets are the lifeblood of AI models. Their quality directly impacts the performance, fairness, and reliability of the resulting models.

**Poor quality data can lead to:**

• **Biased models:** Unrepresentative or skewed data can lead to models that perpetuate existing biases and produce discriminatory outcomes.

• **Inaccurate results:** Inconsistent or erroneous data can cause models to learn incorrect patterns and generate unreliable predictions.

• **Wasted resources:** Training models on low-quality data is a waste of time, computational power, and financial resources.

# Overview of quality checks

Quality checks for AI-ready datasets encompass various aspects, categorized into these key areas:

1. **Data completeness:**
   1. **Missing values:** Identifying and handling missing data points through imputation or removal.
   2. **Outliers:** Detecting and addressing unusual data points that might skew model training.
2. **Data consistency:**
   1. **Formatting:** Ensuring consistent data formats across the entire dataset.
   2. **Units and labels:** Maintaining consistency in units of measurement and data labeling.
3. **Data accuracy:**
   1. **Verification:** Cross-checking data with reliable sources to identify and correct errors.
   2. **Validation:** Comparing data against expected values or domain knowledge to ensure accuracy.

# Overview of quality checks

4. **Data representativeness:**
    1. **Bias:** Analyzing the data for potential biases in sampling, labeling, or other aspects.
    2. **Generalizability:** Assessing whether the data adequately represents the target population for the intended AI application.

5. **Data documentation:**
    1. **Metadata:** Providing comprehensive information about the data, including its origin, collection methods, and usage guidelines.
    2. **Version control:** Maintaining clear versioning of the data to track changes and ensure consistency.

# Poll

What is the primary purpose of verifying data against reliable sources?

a) To identify missing values

b) To ensure data accuracy

c) To check for outliers

d) To maintain data consistency

# Checklist for quality checks

**Data completeness:**

Check for missing values and implement appropriate handling strategies.Identify and address outliers.

**Data consistency:**

Ensure consistent formatting throughout the dataset.

Verify consistency in units and labels.

**Data accuracy:**

Perform data verification against reliable sources.

Validate data against expected values or domain knowledge.

**Data representativeness:**

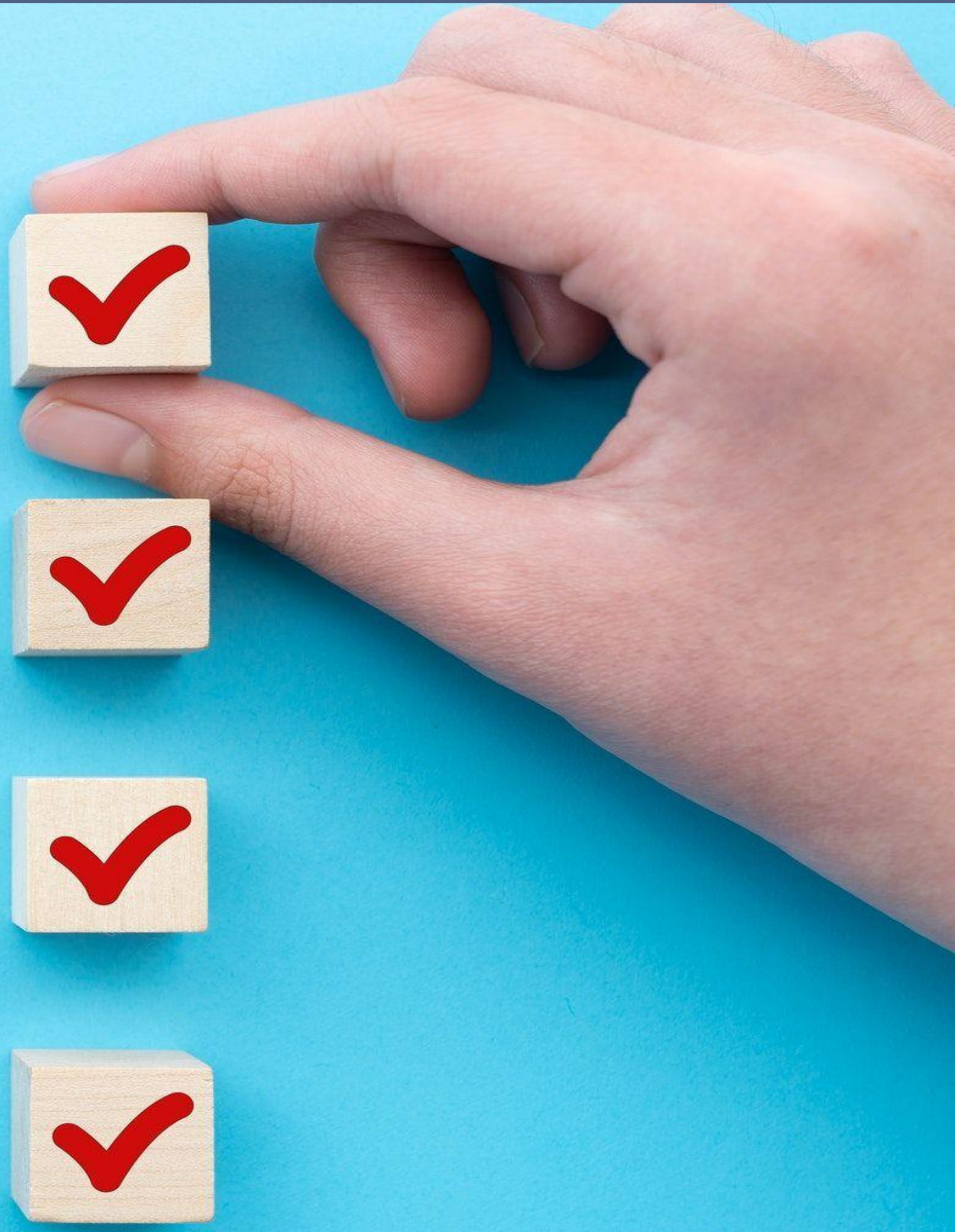
Analyze the data for potential biases.

Assess the generalizability of the data to the target population.

**Data documentation:**

Create comprehensive metadata describing the data.

Implement version control mechanisms.



# Importance of completeness and data dictionaries for AI-ready datasets

Two critical aspects of ensuring datasets are AI-ready are completeness and data dictionaries. Let's explore why each is crucial:

**1. Completeness:**

A complete dataset refers to one with minimal missing values or outliers that could significantly impact the training and performance of AI models. Missing data can lead to:

- **Biased models:** if specific data points are consistently missing, the model might learn skewed patterns and produce unfair results.
- **Inaccurate predictions:** missing data can hinder the model's ability to capture the full picture and lead to unreliable outputs.
- **Inefficient training:** training models on incomplete data can be computationally expensive and inefficient, yielding suboptimal results.

# Importance of completeness and data dictionaries for AI-ready datasets

**2. Data dictionaries:**

Data dictionaries act as the instruction manuals for your dataset, providing crucial information about each variable. They define:

- **Variable names:** clear and consistent names that facilitate understanding and avoid confusion.
- **Data types:** specifying the format of data (e.g., Numerical, categorical, text) ensures proper interpretation by the model.
- **Descriptions:** explanations of the meaning and potential values of each variable, promoting clarity and reducing ambiguity.
- **Units of measurement:** standardizing units (e.g., Meters, kilometers) ensures consistent interpretation and analysis.

# Addressing missing data: strategies for imputation

- Missing data is a common challenge in datasets, and how you handle it can significantly impact your research findings.
- Strategies for handling missing data:

    - **Deletion:** Remove rows or columns with a high percentage of missing values, but this can lead to information loss.
    - **Mean/median imputation:** Replace missing values with the mean or median of the respective variable.
    - **Model-based imputation:** Use statistical models to predict missing values based on other variables in the dataset.

# Understanding and addressing missing data

### Data Missingness Strategies: Understanding and Addressing Missing Data

Missing data, where values are absent from a dataset, is a prevalent challenge in various fields. It can significantly impact the results of data analysis and machine learning models. Fortunately, various strategies exist to address missing data

### Understanding Missing Data:

Before delving into strategies, it's crucial to understand the **types of missing data**:

- **Missing Completely at Random (MCAR):** Missingness occurs randomly and is unrelated to any other variables in the dataset.
- **Missing at Random (MAR):** Missingness depends on observable variables in the dataset but not on the missing values themselves.
- **Missing Not at Random (MNAR):** Missingness is related to the missing values themselves, often due to unobserved factors.

# Understanding and addressing missing data

**Addressing Missing Data:**

Several strategies can be employed to handle missing data, depending on the nature and extent of missingness:

1. **Deletion:**

- ❑ **Listwise deletion:** Removes entire rows with missing values, potentially reducing sample size and introducing bias if MCAR doesn't hold.
- ❑ **Pairwise deletion:** Removes only the data points with missing values for the variable being analyzed, potentially wasting information.

# Understanding and addressing missing data

**2. Imputation:**

- ❑ **Mean/Median/Mode imputation:** Replaces missing values with the average, median, or most frequent value of the variable, respectively. Simple but may introduce bias, especially for skewed distributions.

- ❑ **Hot Deck imputation:** Replaces missing values with values from existing observations with similar characteristics, reducing bias but potentially introducing noise.

- ❑ **Model-based imputation:** Uses statistical models like regression or machine learning to predict missing values based on other variables, potentially more accurate but computationally expensive.

# Poll

What strategies do you find most effective in handling missing values and outliers in datasets?

# Dealing with proxies and small sample sizes: alternative approaches

- **Not all research questions may have readily available data for every variable**. In such cases, researchers might need to employ **proxy variables** or navigate situations with small sample sizes.

- Strategies for addressing proxies and small sample sizes:

  - **Proxy variables:** Carefully select proxy variables that are <u>demonstrably linked to the desired variable</u>, but be mindful of potential limitations and biases.

  - **Small sample size analysis:** Utilize appropriate <u>statistical methods designed for small datasets</u>, such as non-parametric tests or bootstrapping techniques.

# Synthetic/AI Generated DATA

- Information that is **artificially generated** rather than produced by real-world events.
- Generated to meet **specific needs or certain conditions that may not be found in the original, real data**
- Typically created using **algorithms**, synthetic data can be deployed to **validate mathematical models** and to **train machine learning models**
- Often used for **underrepresented populations** in datasets

# Digital Twins

**Digital model** of an intended or actual real-world physical product, system, or process (a physical twin) that serves as the **effectively indistinguishable digital counterpart** of it for practical purposes, such as simulation, integration, testing, monitoring and maintenance

The digital twin of a person, based on such computer simulations, could help drug developers design, test and monitor, and aid doctors in applying, the **safest and most effective treatments or therapies** that are specific and tailored to our genetics or biochemistry.

**Not the answers to poor quality or missing data**

# Exploring the ethical considerations of using synthetic data

While synthetic data offers certain advantages, its use raises **ethical considerations** that researchers must address responsibly:

- **Transparency and disclosure:** Clearly communicate the use of synthetic data, including the number of actual people used to generate it, and its limitations to avoid misinterpretations.

- **Responsible use:** Ensure the synthetic data is used ethically and does not perpetuate harmful stereotypes or discriminatory practices.

- **Potential biases:** Be mindful of generalizability limitations and potential biases that might be introduced during the synthetic data generation process, and take steps to mitigate them.

# Poll

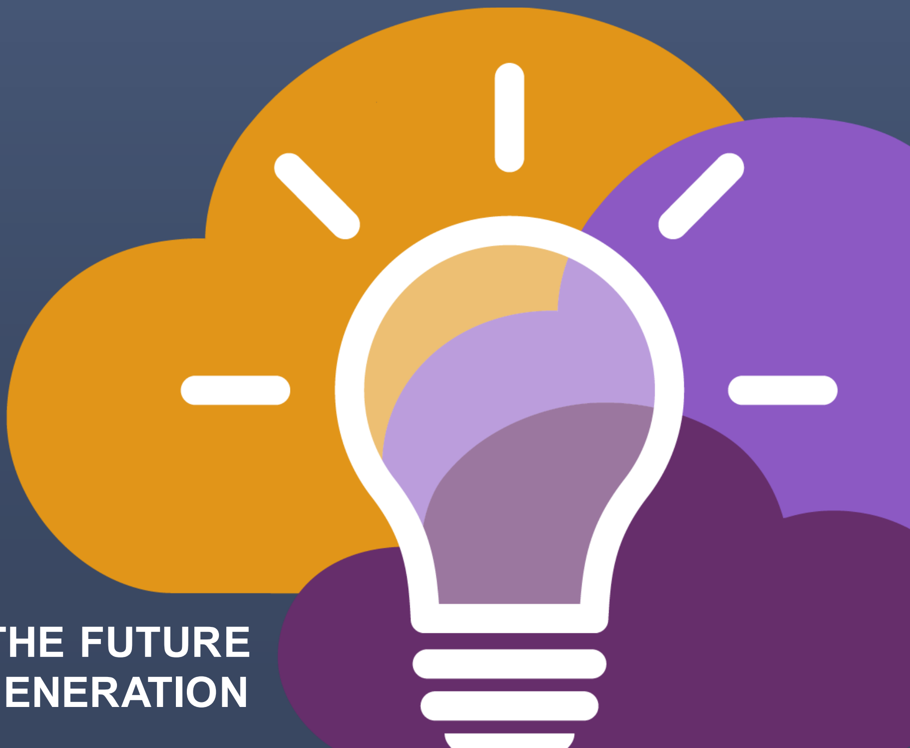
What other ethical considerations should researchers keep in mind when using synthetic data in their studies?

# Poll

In your opinion, what are the biggest challenges researchers face in ensuring their datasets are truly 'AI-ready' beyond the technical aspects?

# ScHARe

## Ethical and transparent Artificial Intelligence



BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Ethical AI

It is crucial that **AI algorithms respect basic human values** and undertake their analysis and decision-making in a trustworthy manner.

Ethical AI builds tools that are faithful to values such as **accountability, privacy, safety, security, and transparency**.

Taken together with explainable AI, it is a way to **deploy AI in ways that further human values**.

# Explainable AI (XAI)

One of the complaints about AI is the **lack of transparency** in how it operates. Many developers don’t reveal the data used or how various factors are weighted. Outsiders cannot tell how AI reached the decision that it did.

This lack of explainability can lead people to **not trust AI**.

XAI seeks to help **describe either the overall function of AI or the specific way it reaches decisions**, to make AI more understandable and trustworthy.

# Artificial Intelligence Bias

Algorithms are widely used in healthcare- and policy-related decisions. However, many operate as “**black boxes**”, offering little opportunity for testing to identify biases.

Biases can result from:

- **social/cultural context not considered**
- **design limitations**
- **data missingness and quality problems**
- **algorithm development and model training**

If not identified, biased algorithms may result in decisions that lead to discrimination, unequitable healthcare, and/or health disparities.

# Trust in AI

**Caution Against:**

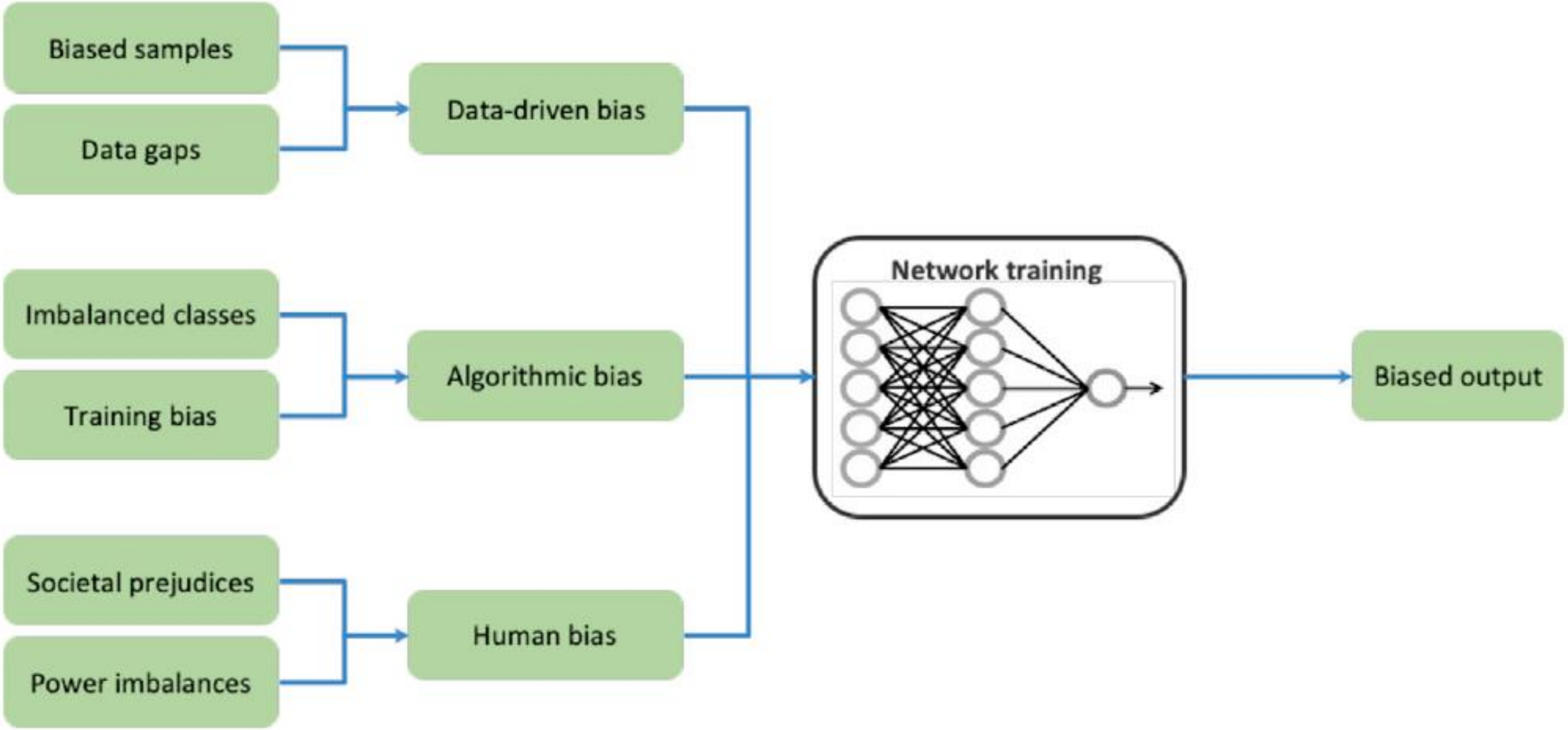
- **Epistemic trust**, which describes the willingness to accept new information from another person or entity as trustworthy, generalizable, and relevant.
- **Synthetic trust**, a misplaced belief in the model's capabilities and fairness.

**Mistrust of AI**

- Fear of misuses
- Fear because of harmful impacts of biases
- Lack of underrepresented populations/community trust

# Algorithmic bias mechanisms

**Bias can originate from unrepresentative/incomplete training data** that reflects historical inequalities, or manifest at various points in the algorithm development process



# Poll

Which of the following factors can contribute to bias in AI algorithms?

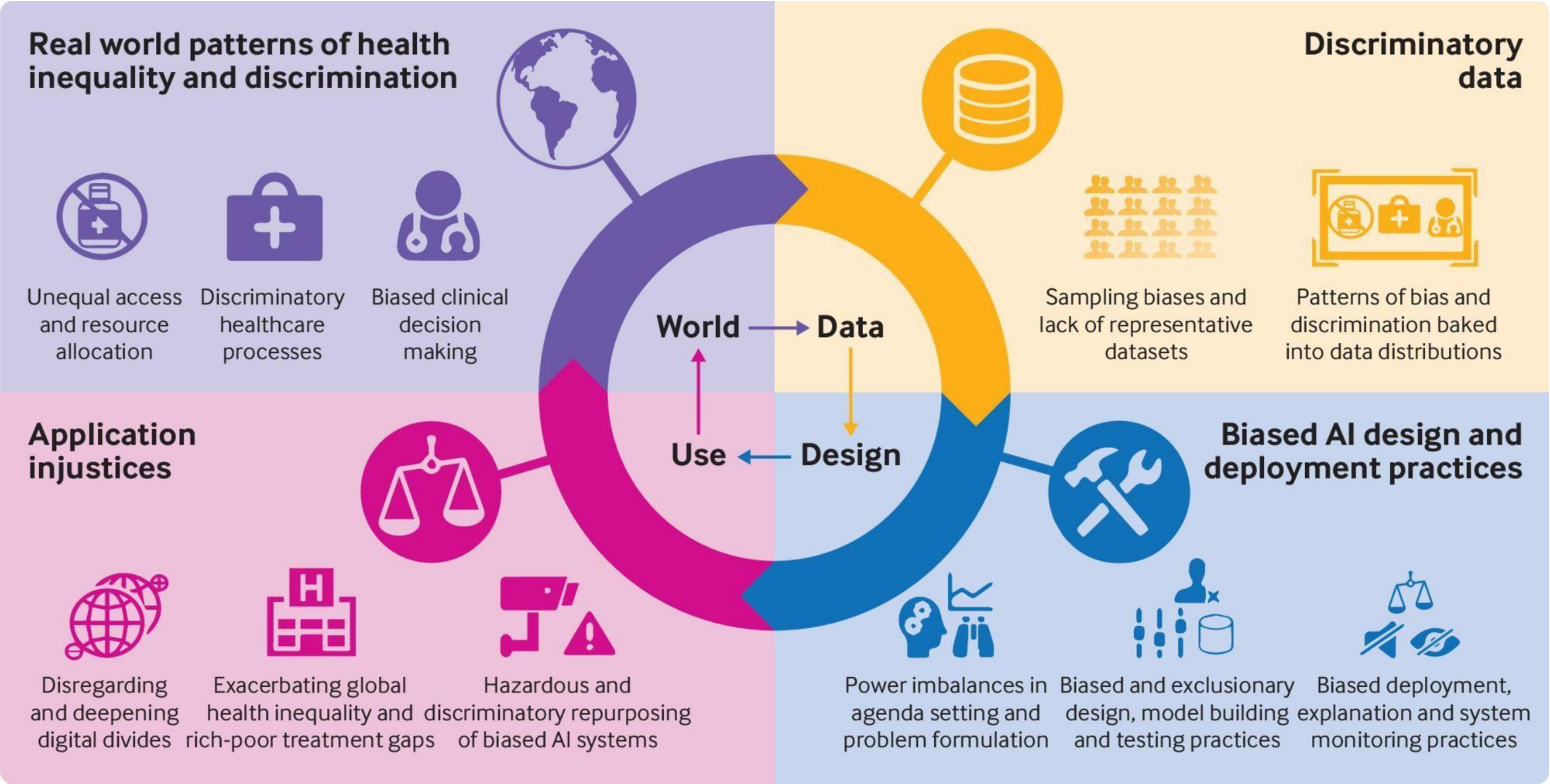
a) Data representativeness

b) Design limitations

c) Data documentation

d) Generalizability

# The big picture



# Example: AI-driven dermatology leaves dark-skinned patients behind

- Machine Learning has been used to create **programs capable of distinguishing between images of benign and malignant moles**.
- However, the algorithms used are basing most of their knowledge on a repository of **skin images from primarily fair-skinned populations.**
- **Bias emanates from unrepresentative training data that reflects historical inequalities:** decades of clinical research have focused primarily on people with light skin.
- The solution: **expand the archive to include as many skin types as possible**

**The issue**

**Lesions on patients of color are less likely to be diagnosed.** The algorithms provide advancement for the Caucasian population, which already has the highest survival rate.

Adamson AS, Smith A. Machine Learning and Health Care Disparities in Dermatology. JAMA Dermatol. 2018;154(11):1247. doi:10.1001/jamadermatol.2018.2348

# U.S. lacks a comprehensive federal AI law

## EU sets global standards with first major AI regulations

- **Europe became the first major world power to enact comprehensive AI regulations**, covering areas like transparency, use of AI in public spaces, and high-risk systems.
- **High-impact models with systemic risks** face stricter requirements, including model evaluation, risk mitigation, and incident reporting.
- Requires **models to comply with transparency obligations before they are put on the market**: drawing up documentation, complying with EU law and disseminating summaries about the content used for training.

## Federal AI Governance Policy:

- The **White House**, **Congress**, and various federal agencies have been actively shaping AI governance.
- The **Federal Trade Commission**, the **Consumer Financial Protection Bureau**, and the **National Institute of Standards and Technology** have all contributed to AI-related initiatives and policies.
- <u>Notably, existing laws do apply to AI technology, and the focus is on understanding how these laws intersect with AI rather than creating entirely new AI-specific legislation</u>
- NIST – New guidance

# Avoiding perpetuating bad AI: mitigating bias in datasets

Strategies to mitigate bias in datasets:

1. **Identify potential sources of bias:** Analyze data collection methods, sampling procedures, and variable selection for potential biases. Testing for biases in datasets and algorithmic models is **crucial for ensuring fairness and reliability** in data science.

2. **Utilize bias mitigation techniques:** Apply techniques like data balancing, weighting, or fairness-aware algorithms to mitigate bias in the data.

3. **Promote transparency and responsible AI practices:** Document the limitations of the data and potential biases to ensure responsible use of AI models trained on the dataset.

# Testing for biases in datasets

1. **Exploratory Data Analysis (EDA):**

   - **Explanation:** EDA involves visualizing and summarizing the main characteristics of the dataset using histograms, box plots, and summary statistics. The goal is to understand the data distribution
   - **Importance:** EDA helps identify outliers, imbalances, and biases
   - **Example:** If EDA reveals a dataset on job applicants is heavily skewed towards a specific gender, it might indicate a bias in the sampling process
   - **Python Libraries:** Pandas, Matplotlib, Seaborn

# Testing for biases in datasets

2. **Demographic Analysis (DA):**

   - **Explanation:** Break down the dataset based on demographic attributes (e.g., age, gender, ethnicity) and analyze the distribution within each group
   - **Importance:** DA can identify imbalances/over-representations in specific groups
   - **Example:** In a healthcare dataset, if one demographic group is over-represented, it may lead to biased predictions
   - **Python Libraries:** Pandas, Matplotlib, Seaborn

# Testing for biases in datasets

3. **Data Stratification:**

    - **Explanation:** Divide the dataset into subgroups based on relevant features and analyze each subgroup independently
    - **Importance:** This helps detect biases that may exist disproportionately in specific subgroups
    - **Example:** In a credit scoring dataset, stratifying by income levels can reveal biases in credit approval rates
    - **Python Libraries:** Pandas

# Testing for biases in datasets

4. **Bias Detection Tools:**

    - **Explanation:** Use tools like IBM's AI Fairness 360 or Google's What-If Tool that offer automated metrics for assessing bias in datasets and models
    - **Importance:** Automated tools efficiently identify subtle biases and provide quantitative measures, facilitating a systematic approach to bias detection
    - **Examples:**
        - AI Fairness 360 provides a set of algorithms to evaluate fairness across various demographic groups
        - Google's What-If Tool allows interactive exploration of model predictions and visualization of outcomes across different subsets of data
    - **Tools:** AI Fairness 360, What-If Tool

# Fixing biases in datasets

Several techniques can be employed to address bias in datasets:

- **Oversampling** involves increasing the representation of underrepresented groups in the dataset, ensuring a more balanced distribution
- **Undersampling** reduces overrepresented groups
- **Using synthetic data** generation introduces artificially generated data points to mitigate imbalances
- **Reweighting** or adjusting the importance of specific instances during model training helps address bias
- Regularly **updating and expanding datasets** with diverse, representative samples further contribute to minimizing bias

# Poll

What techniques would you prioritize to address bias in datasets, and why?

# Poll

Which technique involves increasing the representation of underrepresented groups in a dataset?

a) Undersampling

b) Oversampling

c) Reweighting

d) Hypothesis testing

# Testing for biases in algorithms

1. **Performance Metrics Disaggregation:**

   - **Explanation:** Evaluate model performance metrics (e.g., accuracy, precision) separately for different subgroups defined by sensitive attributes
   - **Importance:** Disparities in performance metrics across groups may indicate bias
   - **Example:** Testing a healthcare algorithm disaggregating accuracy by racial groups reveals slightly lower accuracy for Black patients. Fixes: root cause analysis and algorithm adjustments
   - **Python Libraries:** Scikit-learn

# Testing for biases in algorithms

2. **Confusion Matrix Analysis:**
   - **Explanation:** Analyze the confusion matrix (a table that summarizes the performance of a classification algorithm by comparing predicted and actual values) for different subgroups to identify disparities in model predictions, particularly for false positives and false negatives
   - **Importance:** Disparities in errors can pinpoint areas where bias may exist
   - **Example:** Analyzing a medical diagnosis algorithm using a confusion matrix to evaluate the model's effectiveness in making medical diagnoses. Differences in false positives between genders might indicate bias. Fix: adjusting decision thresholds, retraining with balanced data, consulting domain experts
   - **Python Libraries:** Scikit-learn

# Testing for biases in algorithms

3. **Fairness Indicators:**

   - **Explanation:** Integrate fairness indicators (measures that assess whether a model's predictions treat different groups equitably) into the model evaluation process to identify bias
   - **Importance:** Fairness indicators provide a structured approach to measure bias
   - **Example:** Using Google's TensorFlow Fairness Indicators to compare prediction accuracies of a healthcare decision support algorithm across different racial groups. Fixes: retraining the algorithm with balanced data, adjusting decision thresholds
   - **Python Libraries:** TensorFlow Fairness Indicators

# Testing for biases in algorithms

4. **Sensitivity Analysis:**

    - **Explanation:** Assess how changes in input features impact model predictions. This involves tweaking one feature at a time and observing the model's response
    - **Importance:** It helps identify features that disproportionately influence the model, potentially leading to biases
    - **Example:** In a healthcare decision support algorithm predicting diabetes risk, assessing how variations in input variables (e.g., age, BMI) impact predictions for different racial groups. The analysis reveals that the algorithm disproportionately relies on a single variable affecting certain groups. Fixes: recalibrating the model to minimize the influence of that variable, retraining with a more diverse dataset
    - **Python Libraries:** Scikit-learn

# Testing for biases in algorithms

5. **Counterfactual Analysis:**

    - **Explanation:** Counterfactual analysis involves exploring hypothetical scenarios by determining the minimal changes needed in input features to alter a model's prediction
    - **Importance:** It helps understand the model's decision boundaries and can highlight biases
    - **Example:** In a credit approval algorithm, if a loan application from a certain racial group is denied, the analysis involves identifying the minimal changes needed in the application features (income, credit score) for approval, shedding light on potential biases. Fixes: adjusting the decision thresholds, mitigating the impact of sensitive features, or retraining the model
    - **Python Libraries:** Alibi Counterfactual

# ScHARe

## Computational strategies: traditional statistics

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Computational strategies

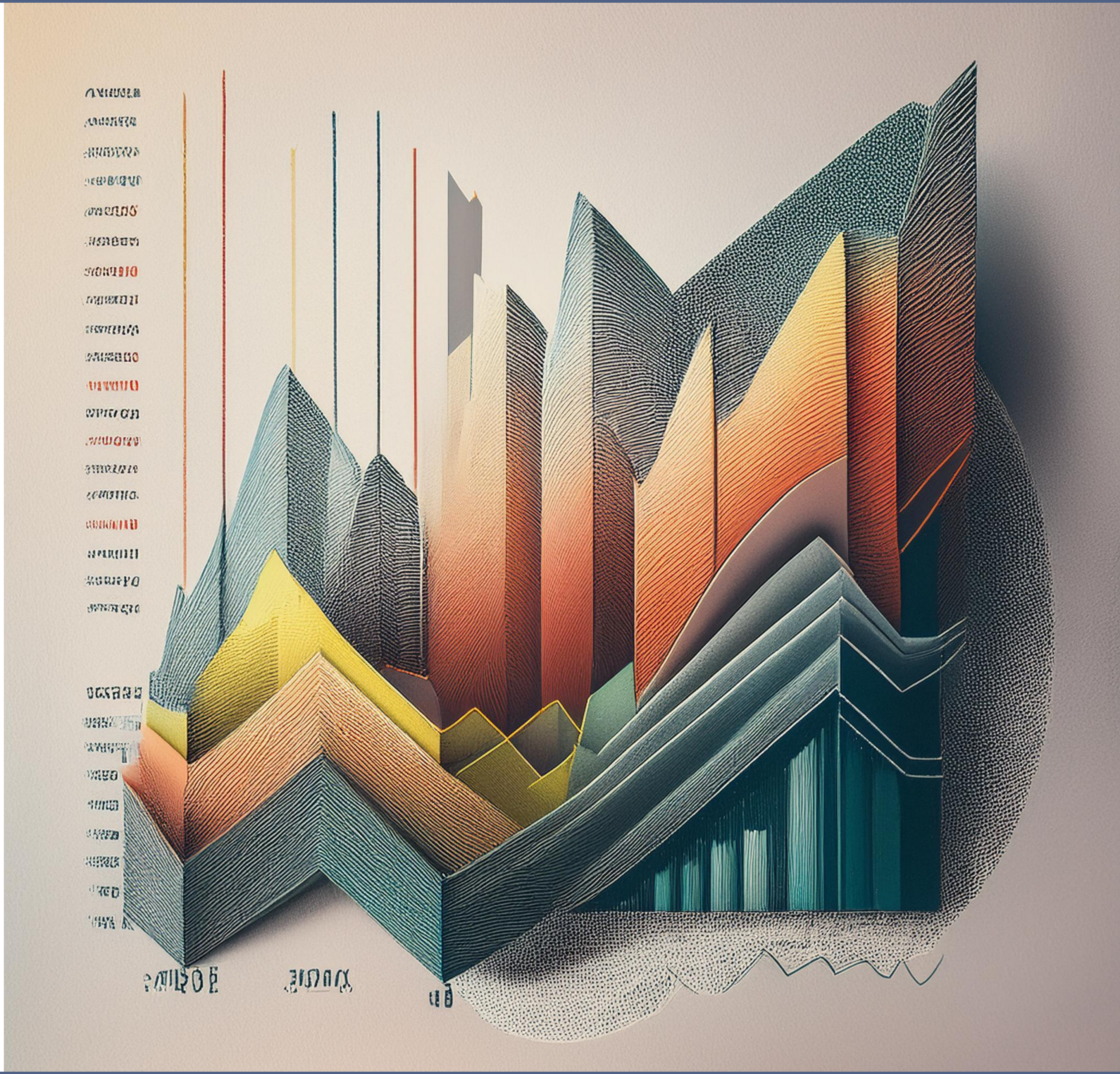
We will provide examples of **computational strategies used in healthcare disparities research**

**Objectives:**

- **Clarify the decision-making process for choosing between traditional statistics and Artificial Intelligence/Machine Learning**
- Explain the differences between these approaches and help you select the most suitable strategies for your analysis goals

# Traditional statistics

- **Strengths:** robust, interpretable, well-established methodology
- **Weaknesses:** limited predictive power, assumption-dependent, often focused on hypothesis testing
- **Data types & use cases:** numerical data, identifying trends, correlations, causal relationships
- **Popular Python libraries:** NumPy, SciPy, Pandas



# 1. Descriptive statistics

- **Strategy:** Summarizing and describing key features of healthcare data, such as mean, median, standard deviation, and percentiles
- **Applications:** Understanding the central tendency and variability in healthcare variables
- **Python Libraries:** NumPy, pandas

# 2. Inferential statistics

- **Strategy:** Making predictions or inferences about a population based on a sample from that population
- **Applications:** Drawing conclusions about healthcare disparities from a subset of relevant data
- **Python Libraries:** SciPy, statsmodels

# 3. Hypothesis testing

- **Strategy:** Evaluating statistical significance to determine whether observed differences are likely to be real or due to chance
- **Applications:** Testing hypotheses about healthcare interventions or disparities
- **Python Libraries:** SciPy, statsmodels

# 4. Analysis of variance (ANOVA)

- **Strategy:** Assessing the statistical significance of differences among group means in healthcare data
- **Applications:** Comparing means across multiple categories to identify significant differences
- **Python Libraries:** SciPy, statsmodels

# 5. Chi-Square test

- **Strategy:** Assessing the association between categorical variables in healthcare datasets
- **Applications:** Examining relationships between demographic factors and health outcomes
- **Python Libraries:** SciPy, pandas

# 6. Regression analysis

- **Strategy:** Modeling the relationship between dependent and independent variables in healthcare data
- **Applications:** Predicting health outcomes based on various factors, identifying disparities
- **Python Libraries:** Statsmodels, scikit-learn

# 7. Survival analysis

- **Strategy:** Analyzing time-to-event data, such as the time until a patient experiences a particular health event
- **Applications:** Studying disparities in disease progression or survival rates
- **Python Libraries:** Lifelines, statsmodels

# 8. Correlation analysis

- **Strategy:** Examining the strength and direction of relationships between two continuous variables in healthcare datasets
- **Applications:** Assessing associations between risk factors and health outcomes
- **Python Libraries:** NumPy, pandas

# 9. Logistic regression

- **Strategy:** Modeling the probability of a binary outcome in healthcare data
- **Applications:** Analyzing factors influencing the likelihood of specific health events
- **Python Libraries:** Statsmodels, scikit-learn

# 10. Bayesian statistics

- **Strategy:** Updating beliefs about parameters based on new evidence in a probabilistic framework
- **Applications:** Incorporating prior knowledge into healthcare disparities research
- **Python Libraries:** PyMC3, Stan

# 11. Time series analysis

- **Strategy:** Analyzing temporal patterns and trends in healthcare data
- **Applications:** Studying disparities over time in health outcomes or interventions
- **Python Libraries:** Statsmodels, Pandas

# Poll

What statistical method can assess the statistical significance of differences among group means in healthcare data?

a) Correlation analysis
b) Regression analysis
c) Analysis of variance (ANOVA)
d) Chi-square test

# ScHARe

## Computational strategies:
## AI and Machine Learning

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Artificial Intelligence (AI)

AI is defined as:

*“machines that respond to stimulation* ***consistent with traditional responses from humans****, given the human capacity for contemplation, judgment, and intention.”*

This definition emphasizes several qualities that separate AI from traditional computer software:

- **Intentionality**
- **Intelligence**
- **Adaptability**

AI-based computer systems **can learn from data, text, or images and make intentional and intelligent decisions** based on that analysis.

ScHARe

Many AI/ML projects are built using Python.

ScHARe fully supports the **Python libraries** most commonly used for AI tasks.

# The role of AI

***AI is an outcome****—the ability of machines to perform tasks that typically require human-level intelligence*

| | perception | notification | suggestion | automation | prediction | prevention | situational awareness |
|---|---|---|---|---|---|---|---|
| | *Describe and understand surroundings* | *Provide alerts, reminders, etc.* | *Build on past preferences and modify over time* | *Follow routine steps to accomplish an objective* | *Forecast the likelihood of future events based on past events* | *Apply cognitive reckoning to identify potential threats* | *Summarize the current, and likely future, environment* |
| **Key Questions Answered** | What's happening now? | What do I need to know? | What do you recommend? | What should I do? | What can I expect to happen? | What can/should I avoid? | What do I need to do now? |

**THE CURRENT ROLE OF AI:**

Curator — Recommender — Orchestrator

**NOT THE ROLE OF AI:**

Critical Thinker — Decision Maker

# Machine Learning (ML)

Machine learning is a subset of artificial intelligence (AI) that involves training algorithms to recognize patterns and make decisions based on data.

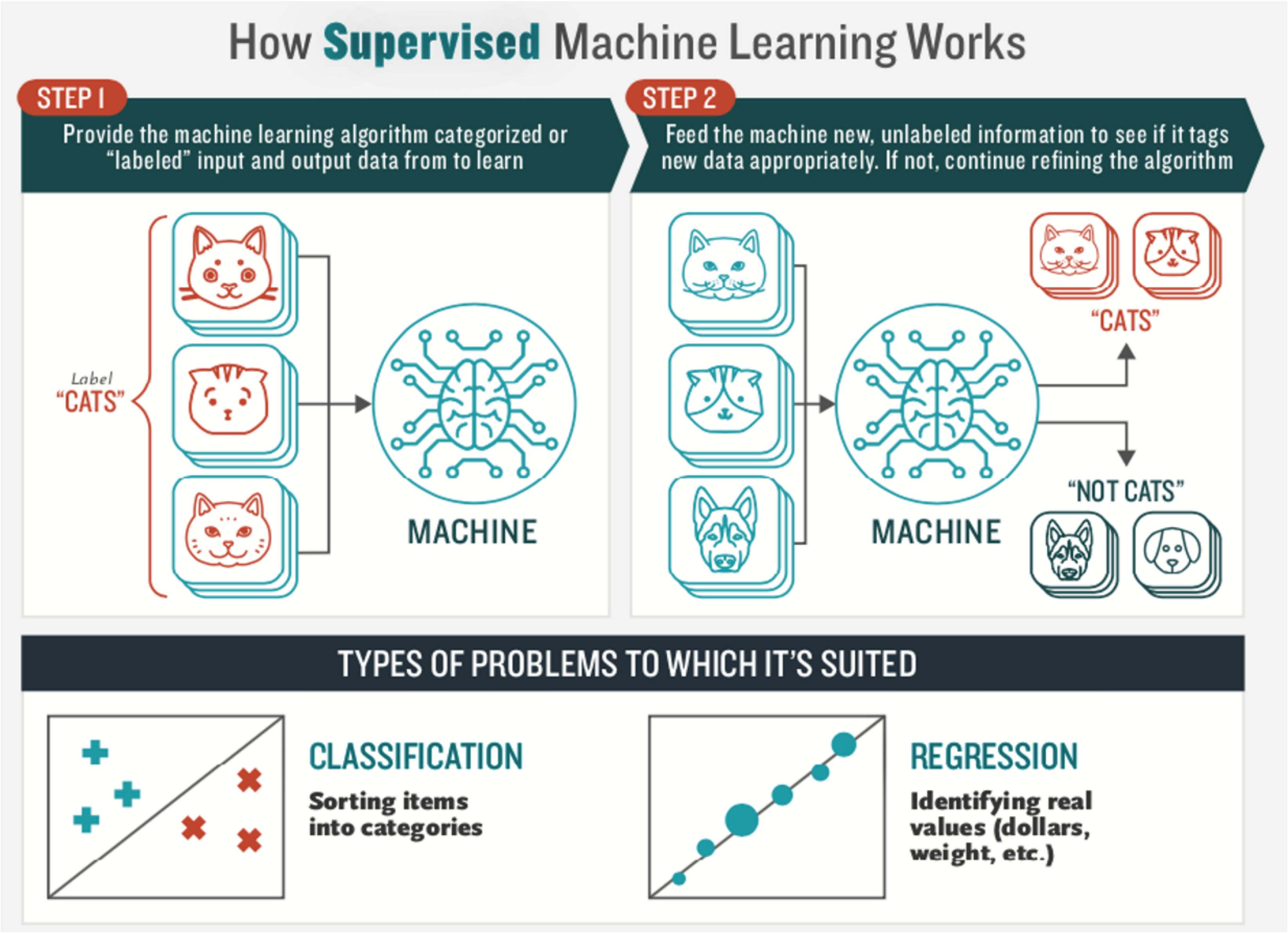
It represents **a way to classify data/objects without detailed instruction.**

**The algorithm learns in the process** so that new objects can be identified using the learned info.

ML is “based on **algorithms that can learn from data** without relying on rules-based programming.”

Unlike **traditional programming, where explicit instructions are given**, machine learning models **learn from examples and improve their performance** over time.

# Supervised Learning



# Supervised Learning: regression

Supervised learning utilizes a dataset which includes both input features as well as the output class or target which are **labeled at the start of training**. Supervised ML algorithms subsequently train on the input data set to produce a model which will differentiate among the output labels.

**Common Algorithms:** Linear Regression, Support Vector Regression, Random Forest Regression, Decision Tree Regression, Ridge Regression, Support Vector Regression (SVR)

## Machine-learning Prediction For Hospital Length Of Stay Using A French Medico-administrative Database

**Objective**: The objective of this study is to explore ML models that best predict Prolonged Hospital Length of Stay for patients based on clinical and demographic features.

**Algorithm Used**: Random Forest (RF), Neural Networks (NN), Gradient Boosting (GB), Decision Trees (CART), Logistic Regression (LR).

**Dataset**: 27 predictor variables including sociodemographic features (age, gender, state-funded medical assistance), disease category, patient origin (home or other hospital institution), hospitalization via emergency departments, destination after hospital discharge, and hospitalization via emergency departments in the previous 6 months.

www.ncbi.nlm.nih.gov/pmc/articles/PMC9707380

# Supervised Learning: classification

In classification based supervised learning, the model learns to **map input data to specific categories (classes) by studying examples where the correct output is known**. This process involves creating a set of rules or a classifier that can predict the correct category for new, unseen data based on the patterns learned from the training examples.

**Common Algorithms**: Logistic Regression, Decision Trees, Random Forest, Support Vector Machine (SVM), k-Nearest Neighbors (k-NN), Naive Bayes, Neural Networks

## Machine Learning-Based Prediction Model of Preterm Birth Using Electronic Health Record

**Objective**: Preterm birth (PTB) was one of the leading causes of neonatal death. Predicting PTB in the first trimester and second trimester will help improve pregnancy outcomes. The aim of this study is to propose a prediction model based on machine learning algorithms for PTB.

**Algorithm Used**: Six algorithms, including Naive Bayesian (NBM), support vector machine (SVM), random forest tree (RF), artificial neural networks (ANN), K-means, and logistic regression, were used to predict PTB.

**Dataset**: Demographic factors (i.e., age), physical examination, blood test, white blood cell count, and plateletcrit, urine test strip (urine pH, urine WBC, and glycosuria), and gynecological examination.

pubmed.ncbi.nlm.nih.gov/35463669/

# Supervised Learning: regression & classification

**Ensemble methods combine the predictions of several base estimators built with a given learning algorithm to improve generalizability and robustness over a single estimator**. They are often used for classification and regression tasks, where they can reduce bias and variance to improve model accuracy

**Common Algorithms:** Ensemble boosting methods (ADA Boost, Gradient Boosting) Ensemble Bagging Methods (Random Forest), Artificial Neural Networks

## A Study Of Generalizability Of Recurrent Neural Network-based Predictive Models For Heart Failure Onset Risk Using A Large And Heterogeneous EHR Data Set

**Objective**: Recurrent neural networks (RNNs) have been applied in predicting disease onset risks with Electronic Health Record (EHR) data to test generalizability and transferability of existing models and its applicability to different patient populations across hospitals.

**Algorithm Used**: Recurrent Neural Networks

**Dataset**: Number of Visits, Diagnoses, Medication, Surgery, Gender, Race, Age

pubmed.ncbi.nlm.nih.gov/29908902/

# Supervised Learning: Python libraries

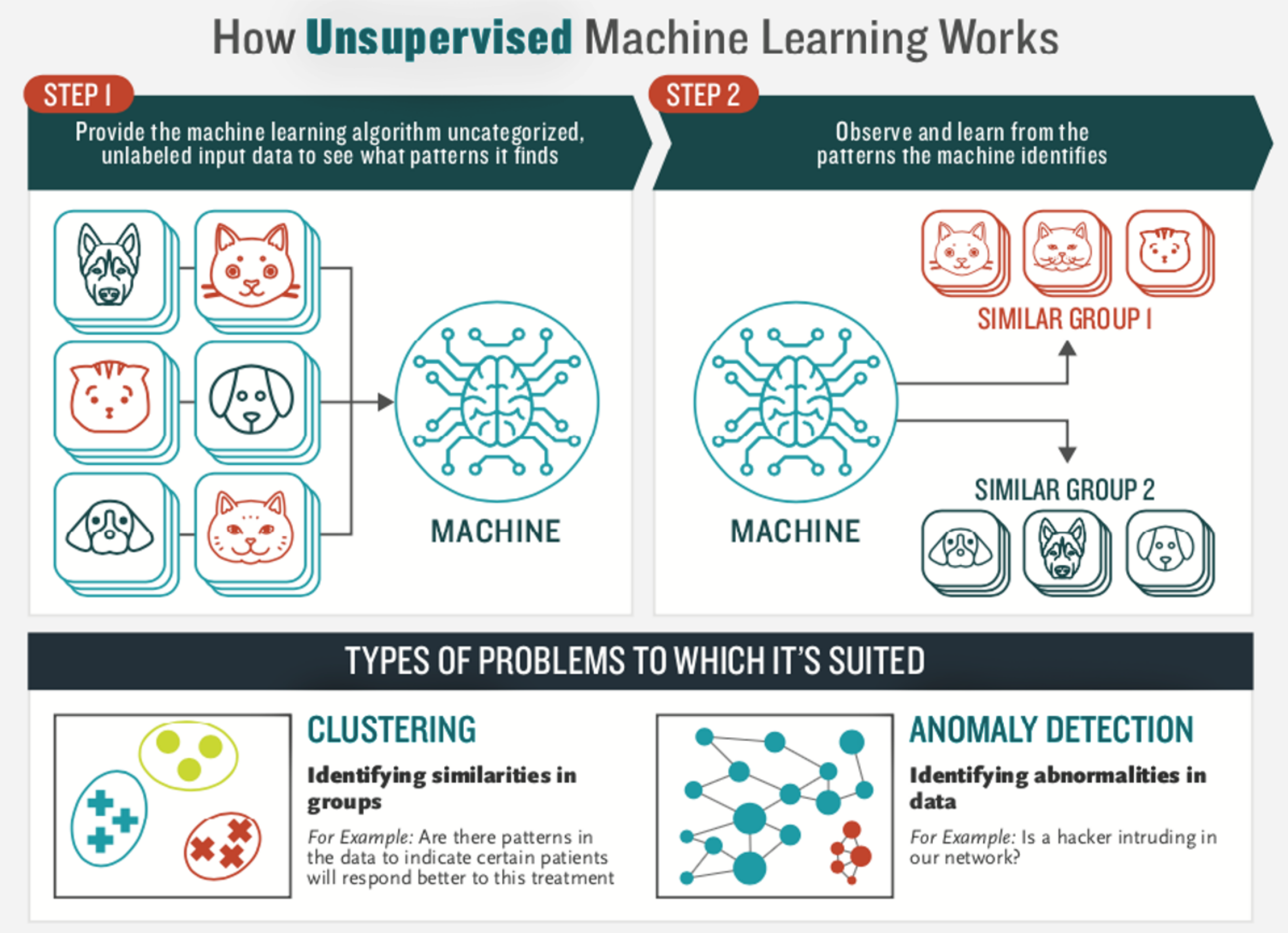
Python offers a range of data science libraries suitable for supervised learning:

1. **scikit-learn**:
    1. Comprehensive library for machine learning with tools for classification, regression, and clustering.
    2. Provides efficient implementations of algorithms like SVM, decision trees, random forests, and more.
2. **TensorFlow**:
    1. Open-source library for deep learning developed by Google.
    2. Suitable for building and training neural networks for tasks like image and speech recognition.
3. **Keras**:
    1. High-level neural networks API, running on top of TensorFlow.
    2. Simplifies the creation of deep learning models with an intuitive interface.
4. **PyTorch**:
    1. Open-source deep learning library developed by Facebook’s AI Research lab.
    2. Known for its dynamic computation graph and ease of use in research and production.
5. **XGBoost**:
    1. Optimized gradient boosting library designed for speed and performance.
    2. Effective for regression and classification problems, often used in machine learning competitions.

# Supervised Learning: Python libraries

6. **LightGBM**:
   1. Gradient boosting framework that uses tree-based learning algorithms.
   2. Known for its efficiency and scalability, suitable for large datasets.
7. **CatBoost**:
   1. Gradient boosting library that handles categorical features well.
   2. Developed by Yandex, it is user-friendly and powerful for classification and regression tasks.
8. **pandas**:
   1. Essential for data manipulation and analysis.
   2. Provides data structures like DataFrame, making it easier to clean and preprocess data for supervised learning.
9. **NumPy**:
   1. Fundamental package for numerical computing.
   2. Provides support for arrays, matrices, and many mathematical functions, often used for preprocessing.
10. **Matplotlib** and **Seaborn**:
    1. Visualization libraries useful for exploring and understanding data.
    2. Helps in identifying patterns and preparing data for modeling.

# Unsupervised Learning



# Unsupervised Learning: Classification

Unsupervised learning with classification involves **grouping similar data points together based on their features without using labeled outcomes**. The goal is to find inherent structures or patterns in the data.

Unsupervised ML **can complement supervised ML approaches, since it can be used to initially determine the most critical features** prior to supervised ML approaches which will build models to discriminate among the classes of interest.

**Common Algorithms**: K-Means Clustering, Hierarchical Clustering, Gaussian Mixture Models (GMM)

## Using Unsupervised Learning To Identify Clinical Subtypes Of Alzheimer's Disease In Electronic Health Records

**Objective**: Primary care electronic health records from the CALIBER resource were used to identify and characterize clinically-meaningful clusters of patients using unsupervised learning approaches.

**Algorithm Used**: MCA and K-Means Clustering

**Dataset**: Symptoms, comorbidities and demographic and lifestyle factors including age of onset, gender, drinking status and smoking status.

pubmed.ncbi.nlm.nih.gov/32570434/

# Unsupervised Learning: dimension reduction

Dimensionality reduction in unsupervised learning **involves reducing the number of random variables under consideration, making the dataset easier to explore** and visualize while preserving as much variance as possible.

**Common Algorithms:** Principal Component Analysis (PCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), Linear Discriminant Analysis (LDA), Autoencoders, Similarity Network Fusion (SNF)

## A Fusion Framework To Extract Typical Treatment Patterns From Electronic Medical Records

**Objective**: Reduce the dimensionality of Electronic Medical Records (EMRs) contain temporal and heterogeneous doctor order information to identify “right patient”, “right drug”, “right dose”, “right route”, and “right time” from doctor order information

**Algorithm Used**: Multi-view similarity Network Fusion (SNF) method

**Dataset**: The EMR data included five types of information including demographic information, laboratory indicators, diagnostic information, doctor orders, and treatment outcome.

www.sciencedirect.com/science/article/abs/pii/S0933365718304184

# Unsupervised Learning: Python libraries

Python offers several data science libraries that are well-suited for unsupervised learning:

1. **scikit-learn**:
    1. Provides a wide range of tools for clustering (e.g., K-means, DBSCAN, hierarchical clustering) and dimensionality reduction (e.g., PCA, t-SNE).
    2. Comprehensive and user-friendly, suitable for many unsupervised learning tasks.
2. **TensorFlow**:
    1. Deep learning library that can be used for advanced unsupervised learning techniques, such as autoencoders and generative models.
    2. Offers flexibility for building custom unsupervised learning algorithms.
3. **PyTorch**:
    1. Similar to TensorFlow, used for building and training deep learning models, including those for unsupervised learning.
    2. Known for its dynamic computation graph, making it easier to experiment with new models.
4. **hdbscan**:
    1. Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) is a clustering algorithm that can find clusters of varying densities.
    2. Effective for identifying complex cluster structures in data.

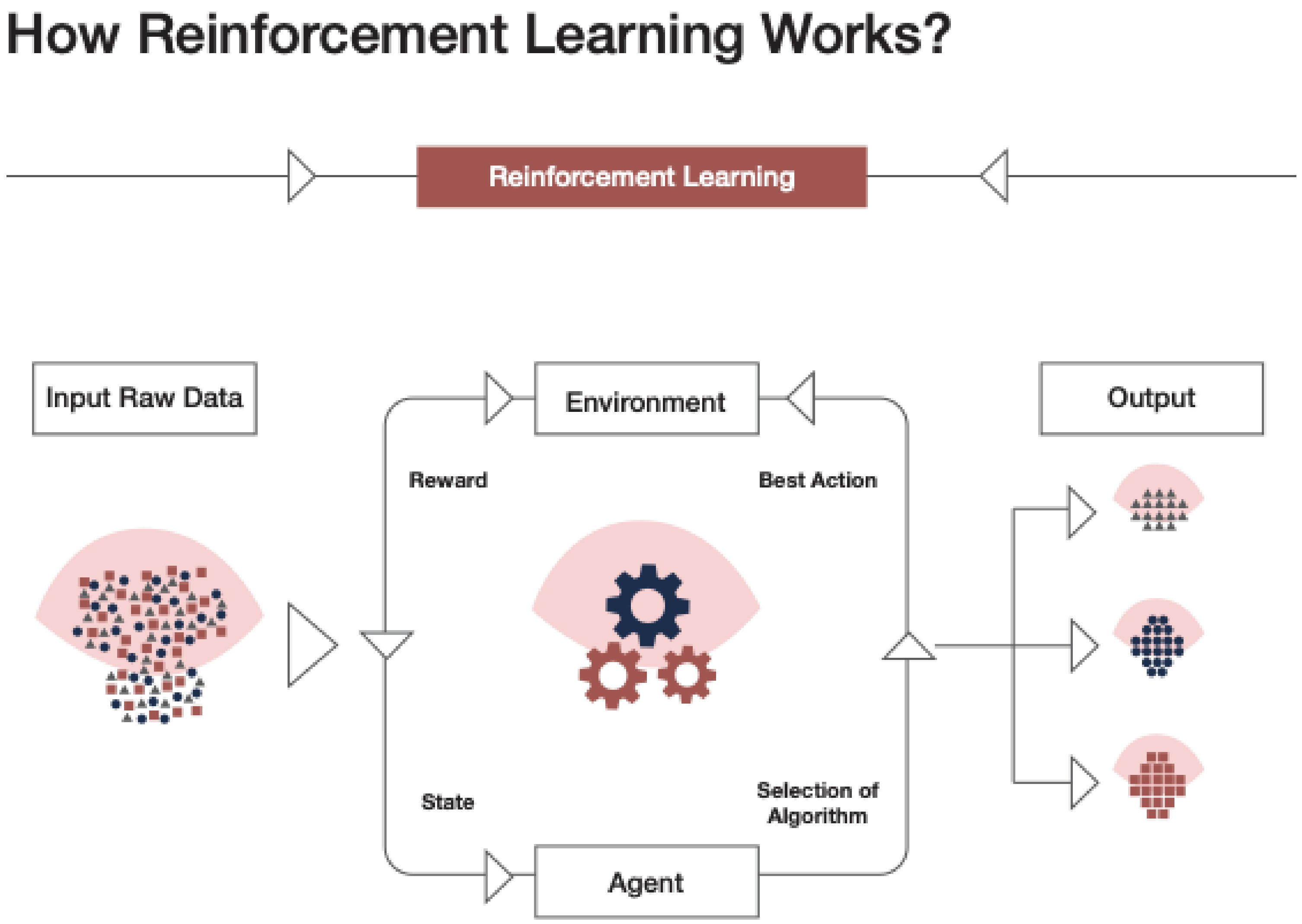
# Unsupervised Learning: Python libraries

5. **OpenCV**:
    1. Primarily used for computer vision tasks, OpenCV also offers tools for unsupervised learning like K-means clustering.
    2. Useful for image processing and pattern recognition tasks.
6. **Yellowbrick**:
    1. A visualization library that extends scikit-learn and provides visual diagnostic tools for model selection, including tools for unsupervised learning.
    2. Helps in visualizing the performance of clustering and dimensionality reduction techniques.
7. **SciPy**:
    1. A scientific computing library that includes clustering and hierarchical clustering algorithms.
    2. Often used in conjunction with NumPy for numerical operations.
8. **NMF (Non-Negative Matrix Factorization)**:
    1. Used for dimensionality reduction and feature extraction.
    2. Available through libraries like scikit-learn and specialized packages.
9. **UMAP (Uniform Manifold Approximation and Projection)**:
    1. A dimensionality reduction technique that is particularly good for visualizing high-dimensional data.
    2. Often used as an alternative to t-SNE.

# Unsupervised Learning: Python libraries

10. **Gensim**:
    1. A library for topic modeling and document similarity analysis, particularly useful for natural language processing (NLP) tasks.
    2. Implements algorithms like Latent Dirichlet Allocation (LDA) for discovering topics in text data.
11. **PyCaret**:
    1. An open-source, low-code machine learning library that simplifies the end-to-end machine learning process.
    2. Includes modules for clustering and anomaly detection.

# Reinforcement Learning



# Reinforcement Learning

Reinforcement Learning (RL) **involves training an agent to make a sequence of decisions by rewarding it for good actions and penalizing it for bad ones.** The agent learns to maximize cumulative rewards through trial and error interactions with the environment.

**Common Algorithms:** Q-Learning, Deep Q-Networks (DQN), Policy Gradient Methods, Actor-Critic Methods

## A Value-based Deep Reinforcement Learning Model With Human Expertise In Optimal Treatment Of Sepsis

**Objective**: Develop personalized treatment strategies for sepsis patients using RL to maximize patient outcomes. The RL model learns to suggest personalized treatment plans that can improve survival rates and reduce recovery times for sepsis patients.

**Algorithm Used**: Deep Q-Network (DQN).

**Dataset**: EHR data from sepsis patients, including treatment actions, physiological measurements, and outcomes.

www.nature.com/articles/s41746-023-00755-5

# Reinforcement Learning: Python libraries

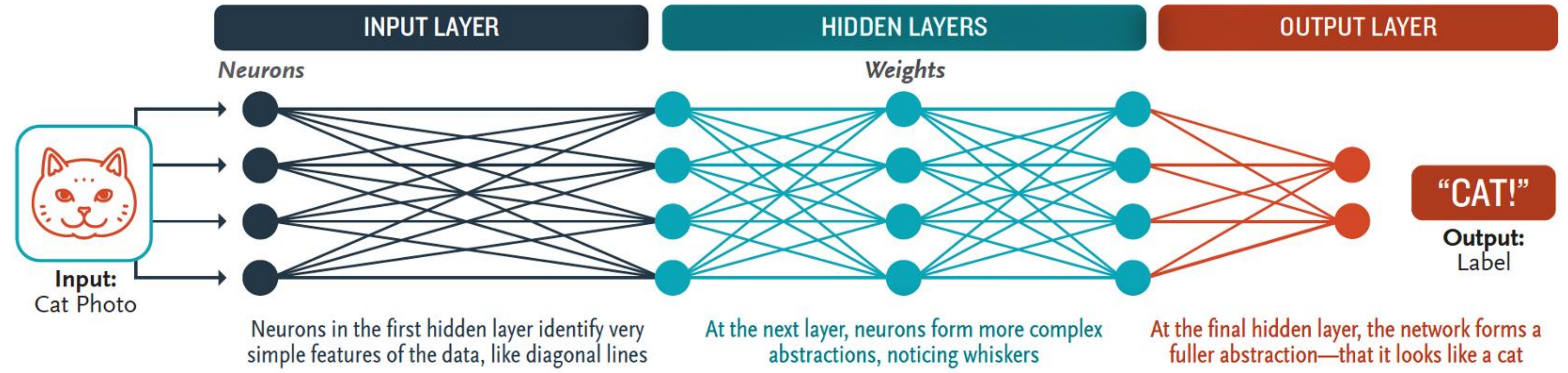
Python offers several libraries that are well-suited for reinforcement learning:

1. **OpenAI Gym**:
   1. A toolkit for developing and comparing RL algorithms.
   2. Provides a variety of environments (e.g., classic control tasks, Atari games) to test and benchmark RL algorithms.
2. **Stable Baselines3**:
   1. A set of reliable implementations of reinforcement learning algorithms in PyTorch.
   2. Built on top of OpenAI Baselines, it offers implementations of algorithms like PPO, A2C, DDPG, and more.
3. **RLlib**:
   1. Part of the Ray framework, RLlib is a scalable reinforcement learning library.
   2. Supports a wide range of RL algorithms and is designed for both single-machine and distributed training.
4. **Keras-RL2**:
   1. Builds on Keras and TensorFlow, providing simple and easy-to-use RL algorithms.
   2. Supports a variety of RL algorithms, making it easy to experiment and integrate with Keras models.

# Reinforcement Learning: Python libraries

5. **TF-Agents**:
    1. A library for reinforcement learning in TensorFlow.
    2. Provides well-documented components for building, training, and evaluating RL agents.
6. **Baselines**:
    1. Originally developed by OpenAI, it offers high-quality implementations of standard RL algorithms.
    2. Useful for researchers and practitioners to benchmark new algorithms against established ones.
7. **Dopamine**:
    1. A research framework for fast prototyping of RL algorithms, particularly focused on simplicity and flexibility.
    2. Developed by Google Research, it provides implementations of several baseline algorithms.
8. **Coach**:
    1. An RL research framework by Intel AI Lab.
    2. Provides a collection of RL algorithms and environments with a focus on usability and performance.
9. **Horizon**:
    1. An open-source RL platform developed by Facebook.
    2. Designed for production use cases, particularly for large-scale applications.
10. **Tianshou**:
    1. A reinforcement learning library based on PyTorch.
    2. Designed to be efficient, flexible, and easily extensible.

# Deep Learning



# Deep Learning

Deep Learning is a subset of machine learning that uses **neural networks with many layers (deep neural networks) to model complex patterns in data**. It can be used for identifying objects within images, transcribing spoken language into text, analyzing medical scans for diagnostics, and understanding and generating human language.

**Common Algorithms:** Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs)

## Clinical Relation Extraction Toward Drug Safety Surveillance Using Electronic Health Record Narratives: Classical Learning Versus Deep Learning

**Objective**: To evaluate natural language processing and machine learning approaches using the expert-annotated medical entities and relations in the context of drug safety surveillance, and investigate how different learning approaches perform under different configurations.

**Algorithm Used**: Support vector machines, Deep neural networks, Supervised Descriptive Rule Induction.

**Dataset**: Healthcare notes with medication, indication, severity, and adverse drug events (ADE), medication-dosage, medication-ADE, and severity-ADE.

publichealth.jmir.org/2018/2/e29/

# Deep Learning: Python libraries

Python offers a robust ecosystem of libraries for deep learning:

1. **TensorFlow**:
    1. Developed by Google, TensorFlow is an open-source deep learning library.
    2. Supports building and training neural networks for various tasks such as image recognition, natural language processing, and more.
    3. TensorFlow 2.x integrates Keras as its high-level API, making it easier to develop models.
2. **Keras**:
    1. A high-level neural networks API, written in Python and capable of running on top of TensorFlow, Microsoft Cognitive Toolkit (CNTK), or Theano.
    2. Simplifies building and training deep learning models with a user-friendly and modular approach.
3. **PyTorch**:
    1. Developed by Facebook's AI Research lab, PyTorch is an open-source deep learning library.
    2. Known for its dynamic computation graph, which makes it easier and more intuitive to build and modify neural networks.
    3. Popular in both academia and industry for research and production.

# Deep Learning: Python libraries

4. **MXNet**:
   1. An open-source deep learning framework developed by Apache.
   2. Supports a wide range of languages including Python, and provides efficient tools for building and training deep learning models.
   3. Known for its scalability and performance, especially in distributed computing environments.
5. **Caffe**:
   1. Developed by the Berkeley Vision and Learning Center (BVLC), Caffe is a deep learning framework focused on speed and modularity.
   2. Suitable for image classification and convolutional neural networks (CNNs).
6. **Theano**:
   1. One of the earliest deep learning libraries, developed by the Montreal Institute for Learning Algorithms (MILA) at the University of Montreal.
   2. While it has been succeeded by other libraries like TensorFlow and PyTorch, Theano laid the groundwork for many subsequent frameworks.
7. **Chainer**:
   1. A flexible and intuitive deep learning framework that supports dynamic computation graphs (define-by-run).
   2. Particularly well-suited for complex and varied network architectures.

# Deep Learning: Python libraries

8. **Fastai**:
    1. Built on top of PyTorch, Fastai provides high-level components that make it easy to train state-of-the-art deep learning models.
    2. Known for its user-friendly API and focus on making deep learning accessible.
9. **DL4J (DeepLearning4J)**:
    1. An open-source, distributed deep learning library for the Java Virtual Machine (JVM), but also offers APIs in Python.
    2. Integrates well with big data tools like Apache Hadoop and Apache Spark.
10. **Gluon**:
    1. A deep learning library jointly developed by AWS and Microsoft, providing an easy-to-use interface for building neural networks.
    2. Integrates with Apache MXNet to combine ease of use with performance.

# Poll

Which type of machine learning involves the model being trained on a labeled dataset where output for each input is known?

a) Supervised learning

b) Unsupervised learning

c) Semi-supervised learning

d) Reinforcement learning

# Poll

Which type of machine learning involves the model learning by interacting with an environment and receiving rewards or penalties based on its actions?

a) Supervised learning
b) Unsupervised learning
c) Semi-supervised learning
d) Reinforcement learning

# ScHARe

## Resources

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# ScHARe resources

Support made available to users:

**ScHARe-specific**

- ScHARe documentation
- Email support

**Platform-specific**

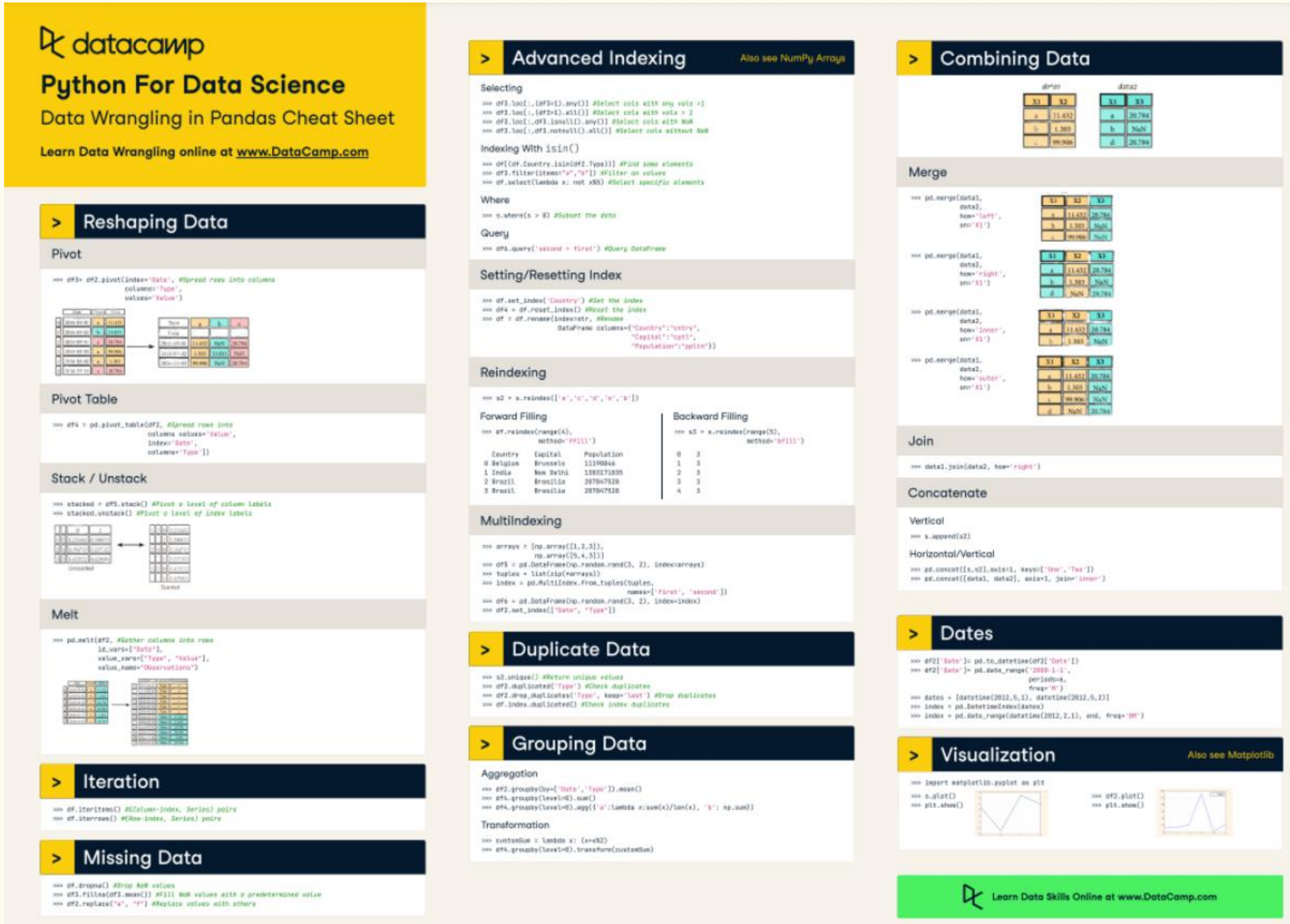
- Terra-specific support
- Terra-specific documentation

# ScHARe resources

Training opportunities made available to users:

- Monthly **Think-a-Thons**
- **Instructional materials** and slides made available online on NIMHD website
- **YouTube videos**
- **Links to relevant online resources** and training on NIMHD website
- **Pilot credits** for testing ScHARe for research needs
- **Instructional Notebooks** in ScHARe Workspace with instructions for:
    - Exploring the data ecosystem
    - Setting your workspace up for use
    - Accessing and interacting with the categories of data accessible through ScHARe

# ScHARe resources: cheatsheets




Credits: datacamp.com

# Terra resources

If you are new to Terra, we recommend exploring the following resources:

- Overview Articles: Review high-level docs that outline what you can do in Terra, how to set up an account and account billing, and how to access, manage, and analyze data in the cloud
- Video Guides: Watch live demos of the Terra platform's useful features
- Terra Courses: Learn about Terra with free modules on the Leanpub online learning platform
- Data Tables QuickStart Tutorial: Learn what data tables are and how to create, modify, and use them in analyses
- Notebooks QuickStart Tutorial**:** Learn how to access and visualize data using a notebook
- Machine Learning Advanced Tutorial: Learn how Terra can support machine learning-based analysis

# ScHARe

## Thank you

BE A PART OF THE FUTURE
OF KNOWLEDGE GENERATION

# Think-a-Thon poll

1. **Rate how useful this session was:**

- ☐ Very useful
- ☐ Useful
- ☐ Somewhat useful
- ☐ Not at all useful

# Think-a-Thon poll

**2. Rate the pace of the instruction for yourself:**

- ☐ Too fast
- ☐ Adequate for me
- ☐ Too slow

# Think-a-Thon poll

**3. How likely will you participate in the next Think-a-Thon?**

- ☐ Very interested, will definitely attend
- ☐ Interested, likely will attend
- ☐ Interested, but not available
- ☐ Not interested in attending any others

# ScHARe

**Next Think-a-Thons:**

bit.ly/think-a-thons

**Register for ScHARe:**

bit.ly/join-schare

schare@mail.nih.gov