Week 3 lab

COGS 108, 9:00-9:50AM (B01)

https://github.com/alexavndra/cogs108-b01

Reminders😸 □

- ➢ A1 is due TODAY at 11:59PM!
- ➢ D2 is due Friday, October 20th at 11:59PM
- ➢ Make some office hours appointments :3
    - ○ https://calendly.com/alexandrarh/office-hours

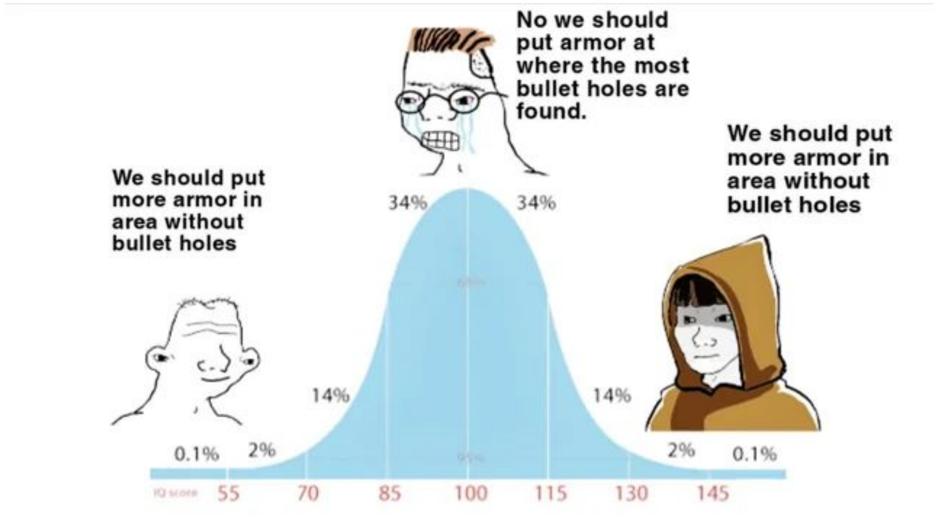
CHAINSAW MAN
Story & Art
Tatsuki Fujimoto
Homeless kid absorbs a demon dog and becomes chainsaw man
* Anime leaves off at vol 5

Pre-lab thoughts!

Survivorship bias

Definition: When you only consider the “surviving” results from a study, ignoring those that didn’t “survive” (ex: the WWII plane)

➢ May prevent you from achieving the actual BEST result!


We should put more armor in area without bullet holes
No we should put armor at where the most bullet holes are found.
We should put more armor in area without bullet holes
34%
34%
14%
14%
0.1%
2%
2%
0.1%
55
70
85
100
115
130
145

D2 Overview

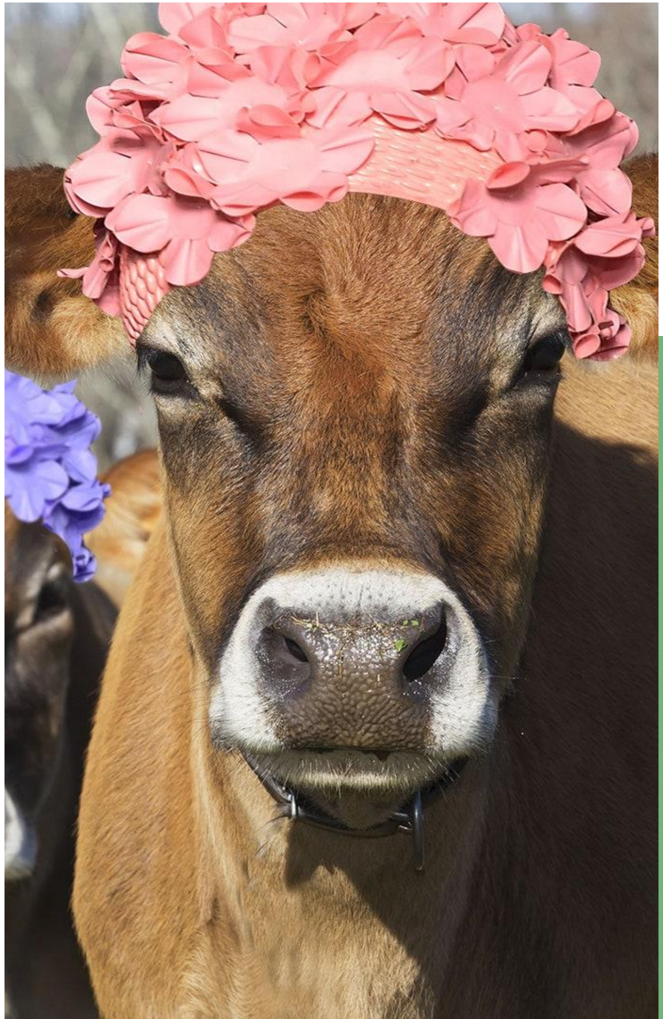
Data wrangling and cleaning

Pt I: Data wrangling🤠🐮

Important aspects of data wrangling include:

1. **Data exploration**: Trying to understand via visualizations
2. **Data reshaping**: Fitting data according to requirements
3. **Data filtering**: Filtering out unnecessary data (DON’T CONFUSE WITH SURVIVORSHIP BIAS)
4. **Missing data**: labeling them appropriately (fill in later)!

Unfortunately, no cows are included here :(...unless you are studying a dataset regarding cows!



Pt II: NumPy and Pandas

NumPy

Aka Numerical Python, it's good to use for vectorization of mathematical operations (very efficient)

When using, use the command

```
import numpy as np
```

Pandas

This Python package is good for data analysis operations (very efficient too!)

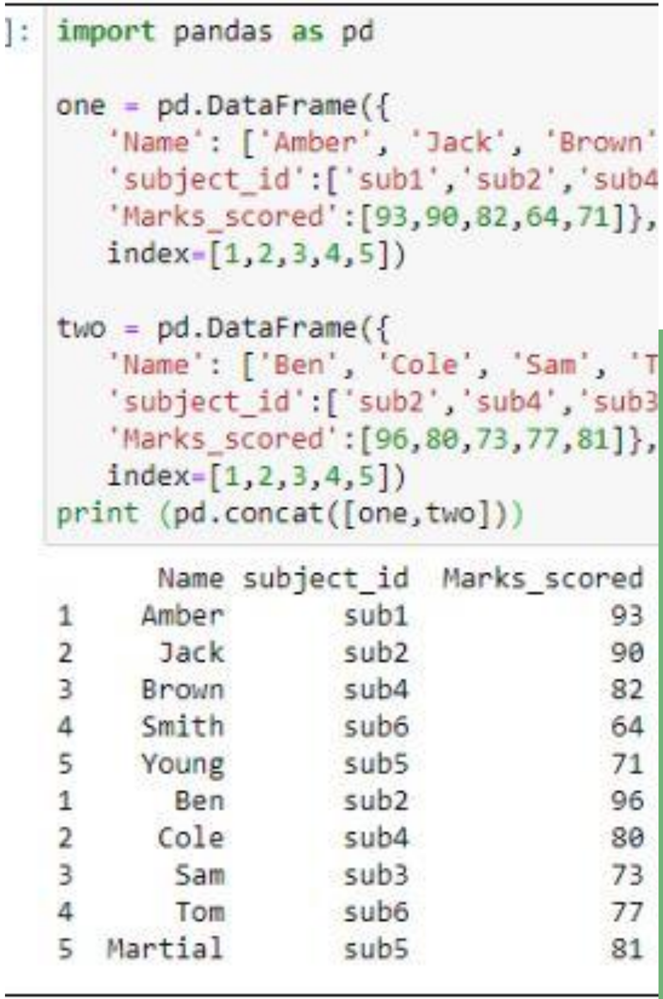
When using, use the command

```
import pandas as pd
```

Pt II, Pt A: Pandas operations

Pandas uses include…

- ➢ **`df = pd_read.csv(“file.csv”)`**: opens up a CSV file + reads it for usage
- ➢ **`df.head(n)`**: returns *n* rows from the dataset; default param = 5
- ➢ **`df.describe()`**: generates statistics from dataset (e.g. mean, median, mode)
  - ○ Could also be used on a single column (**`df[n].describe()`**)
- ➢ **`df.iloc[n,m]`**: returns a view of selected row and/or column in dataframe
- ➢ **`list(df)`**: prints dataframe column names
  - ○ Can also rename columns with **`survey.columns = [‘name1’, …]`**



```
]: import pandas as pd

   one = pd.DataFrame({
      'Name': ['Amber', 'Jack', 'Brown'
      'subject_id':['sub1','sub2','sub4
      'Marks_scored':[93,90,82,64,71]},
      index=[1,2,3,4,5])

   two = pd.DataFrame({
      'Name': ['Ben', 'Cole', 'Sam', 'T
      'subject_id':['sub2','sub4','sub3
      'Marks_scored':[96,80,73,77,81]},
      index=[1,2,3,4,5])
   print (pd.concat([one,two]))

         Name subject_id  Marks_scored
   1    Amber       sub1            93
   2     Jack       sub2            90
   3    Brown       sub4            82
   4    Smith       sub6            64
   5    Young       sub5            71
   1      Ben       sub2            96
   2     Cole       sub4            80
   3      Sam       sub3            73
   4      Tom       sub6            77
   5  Martial       sub5            81
```

Pt II, Pt B: Pandas cleaning

`messy_dataset = ick`. Here's how Pandas helps:

- ➢ `isnull(df)/isnull()`: checks if there's any null values in dataframe (boolean)
  - ○ Can be used with `null_rows = dataframeName.isnull().any(axis=1).sum()`
- ➢ `dropna()`: drop rows + columns with NaN values
- ➢ `fillna()`: fills any NaN values in dataset
  - ○ Good for few null values present
- ➢ `df = df.fillna(method='fill')`: fill null value with value of previous

Next week…

D3 + possible Project Review questions?

D2 Demo

https://datahub.ucsd.edu