DATA SCIENCE: SETTING UP DATAFRAMES

Presentation By: Matthew Hall

DATAFRAMES

One For Each Use.

STORE COMMON ENTRIES

- Each DataFrame Should Have Specific Purpose
- Keep Distinct Datasets Separate, Unless Necessary
- Only Merge What's Necessary, Don't Need Giant DataFrame with Everything

ROWS VS. COLUMNS

Seems Basic, But Very Important.

COLUMNS

- Common Data Points For Each Entry In DataFrame
- Homogeneous Data Type — Forced in Python & Most Databases
- Can Always Store NaN or Nothing if Datapoint Doesn’t Exist

ROWS

- **EACH ROW SHOULD BE UNIQUE, DISTINCT ENTRY**
- If The Data You Get Stores Multiple Entries Per Line, You Should Coerce Dataset into Proper Shape

INDEX

What Belongs in Here.

IDENTIFYING INFORMATION

- **Data That Identifies The Row Entry Belongs in The Index**
- **Everything Else Should Be in Columns**
- **Dates, Some Categorical Information, Unique Keys, ID Numbers, States, Countries, and Similar Belong in Index**

DATA ONLY IN COLUMNS

- Makes Representing Data as Matrix Easier
- Isolate Data Values From Identifying Data
- Improves Performance on Larger Datasets

ROW-WISE VS COLUMN-WISE OPERATIONS

What's The Difference?

GENERALLY…

Column-Wise Operations

- Create New Columns Based Upon Each Row's Data
- Process Each Row Separately

Row-Wise Operations

- Generate Summary Statistics For Columns
- Collapse Dataset Vertically
- Extract Data From Common Columns Between Entries

EXAMPLES OF DIFFERENCES

Column-Wise Operations

- Excel: C5 = SUM(C1:C5)
- Applying Logic to Each Row Individually

Row-Wise Operations

- Group By
- Mean, Median, Quantiles of Columns
- Summary Statistics
- Filters
- Majority of SQL-Like Functionality


MELIORA