Welcome to the CoGrammar

Tutorial: Data Sets, Frames and Visualisation

The session will start shortly...

Questions? Drop them in the chat. We'll have dedicated moderators answering questions.

CoGrammar

Data Science Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Data Science Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**
- Report a **safeguarding** incident: **www.hyperiondev.com/safeguardreporting**
- We would love your **feedback** on lectures: **Feedback on Lectures**

Skills Bootcamp
8-Week Progression Overview

Fulfil 4 Criteria to Graduation

✅ Criterion 1: Initial Requirements

Timeframe: First 2 Weeks
Guided Learning Hours (GLH): Minimum of 15 hours
Task Completion: First four tasks

Due Date: 24 March 2024

✅ Criterion 2: Mid-Course Progress

60 Guided Learning Hours

Data Science - **13 tasks**
Software Engineering - **13 tasks**
Web Development - **13 tasks**

Due Date: 28 April 2024

CoGrammar

Skills Bootcamp Progression Overview

✅ Criterion 3: Course Progress

Completion: All mandatory tasks, including Build Your Brand and resubmissions by study period end
Interview Invitation: Within 4 weeks post-course
Guided Learning Hours: Minimum of 112 hours by support end date (10.5 hours average, each week)

✅ Criterion 4: Demonstrating Employability

Final Job or Apprenticeship Outcome: Document within 12 weeks post-graduation
Relevance: Progression to employment or related opportunity

CoGrammar

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar

Tutorial: Data Sets, Frames and Visualisation

April 2024

Learning objectives

- Read and manipulate data with Pandas
- Generate plots in Python using Matplotlib and Seaborn.
- Gain an understanding of more advanced graphing techniques.

CoGrammar

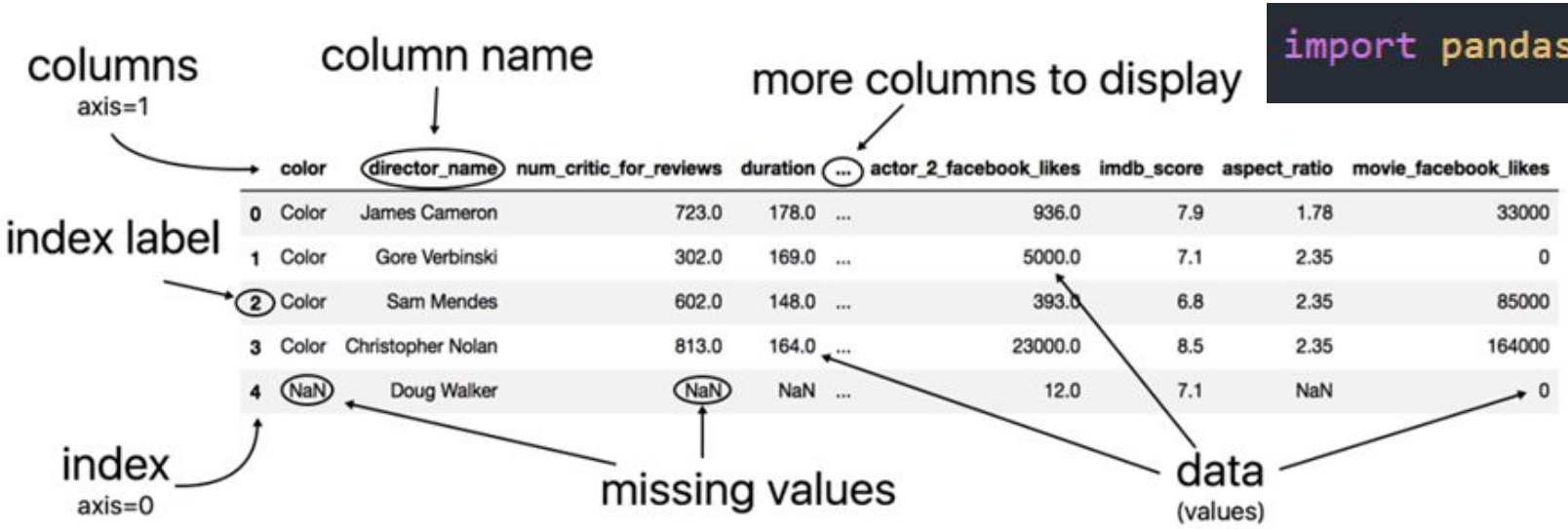
Pandas dataframe

CoGrammar

Pandas DataFrame

- The pandas' library documentation defines a DataFrame as a "two-dimensional, size-mutable, with labelled rows and columns."

```
import pandas as pd
```



	color	director_name	num_critic_for_reviews	duration	...	actor_2_facebook_likes	imdb_score	aspect_ratio	movie_facebook_likes
0	Color	James Cameron	723.0	178.0	...	936.0	7.9	1.78	33000
1	Color	Gore Verbinski	302.0	169.0	...	5000.0	7.1	2.35	0
2	Color	Sam Mendes	602.0	148.0	...	393.0	6.8	2.35	85000
3	Color	Christopher Nolan	813.0	164.0	...	23000.0	8.5	2.35	164000
4	NaN	Doug Walker	NaN	NaN	...	12.0	7.1	NaN	0

Anatomy of a DataFrame

Pandas cheat sheet

IMPORTING DATA

- `pd.read_csv(filename)` - From a CSV file
- `pd.read_table(filename)` - From a delimited text file (like TSV)
- `pd.read_excel(filename)` - From an Excel file
- `pd.read_sql(query, connection_object)` - Reads from a SQL table/database
- `pd.read_json(json_string)` - Reads from a JSON formatted string, URL or file.
- `pd.read_html(url)` - Parses an html URL, string or file and extracts tables to a list of dataframes
- `pd.read_clipboard()` - Takes the contents of your clipboard and passes it to `read_table()`
- `pd.DataFrame(dict)` - From a dict, keys for columns names, values for data as lists

EXPORTING DATA

- `df.to_csv(filename)` - Writes to a CSV file
- `df.to_excel(filename)` - Writes to an Excel file
- `df.to_sql(table_name, connection_object)` - Writes to a SQL table
- `df.to_json(filename)` - Writes to a file in JSON format
- `df.to_html(filename)` - Saves as an HTML table
- `df.to_clipboard()` - Writes to the clipboard

SELECTION

- `df[col]` - Returns column with label `col` as Series
- `df[[col1, col2]]` - Returns Columns as a new DataFrame
- `s.iloc[0]` - Selection by position
- `s.loc[0]` - Selection by index
- `df.iloc[0,:]` - First row
- `df.iloc[0,0]` - First element of first column

VIEWING/INSPECTING DATA

- `df.head(n)` - First **n** rows of the DataFrame
- `df.tail(n)` - Last **n** rows of the DataFrame
- `df.shape()` - Number of rows and columns
- `df.info()` - Index, Datatype and Memory information
- `df.describe()` - Summary statistics for numerical columns

DATA CLEANING

- `df.columns = ['a','b','c']` - Renames columns
- `pd.isnull()` - Checks for null Values, Returns Boolean Array
- `df.rename(columns={'old_name': 'new_name'})` - Selective renaming

CoGrammar

What does a DataFrame represent in Pandas?

1. A two-dimensional array with labeled axes.
2. A single-dimensional array of data.
3. A database management system.
4. A type of Python function.

CoGrammar

What does a DataFrame represent in Pandas?

1. **A two-dimensional array with labeled axes.**
2. A single-dimensional array of data.
3. A database management system.
4. A type of Python function.

CoGrammar

Which method can be used to read a CSV file into a DataFrame?

1. pd.to_csv()
2. pd.csv_reader()
3. pd.read_csv()
4. pd.open_csv()

CoGrammar

Which method can be used to read a CSV file into a DataFrame?

1. pd.to_csv()
2. pd.csv_reader()
3. **pd.read_csv()**
4. pd.open_csv()

CoGrammar

What does the df.describe() method provide?

1. A list of all the columns in a DataFrame.
2. A count of missing values in each column.
3. The first five rows of the DataFrame.
4. Summary statistics for numerical columns.

CoGrammar

What does the df.describe() method provide?

1. A list of all the columns in a DataFrame.
2. A count of missing values in each column.
3. The first five rows of the DataFrame.
4. **Summary statistics for numerical columns.**

CoGrammar

Manipulating Data

CoGrammar

How can you select a subset of columns from a DataFrame?

1. df['column1', 'column2']
2. df[['column1', 'column2']]
3. df(column1, column2)
4. df.columns['column1', 'column2']

CoGrammar

How can you select a subset of columns from a DataFrame?

1. df['column1', 'column2']
2. **df[['column1', 'column2']]**
3. df(column1, column2)
4. df.columns['column1', 'column2']

CoGrammar

What does the groupby() method do?

1. Sorts the DataFrame based on specified columns
2. Combines several columns into a new one
3. Splits the DataFrame into groups based on some criteria
4. Applies a function to each row in a DataFrame

CoGrammar

What does the groupby() method do?

1. Sorts the DataFrame based on specified columns
2. Combines several columns into a new one
3. **Splits the DataFrame into groups based on some criteria**
4. Applies a function to each row in a DataFrame

CoGrammar

Which method is used to get a quick overview of a DataFrame's structure?

1. df.overview()
2. df.describe()
3. df.summary()
4. df.info()

CoGrammar

Which method is used to get a quick overview of a DataFrame's structure?

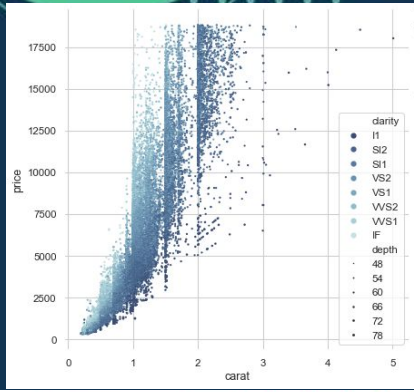
1. df.overview()
2. df.describe()
3. df.summary()
4. **df.info()**

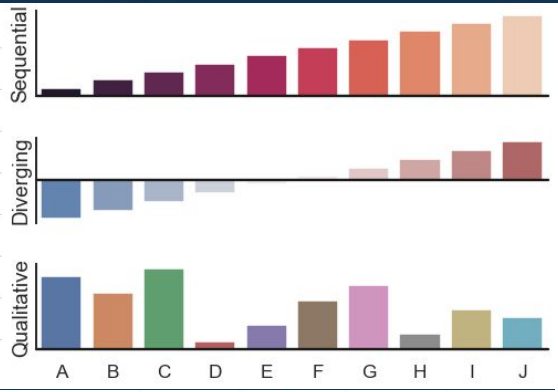
CoGrammar

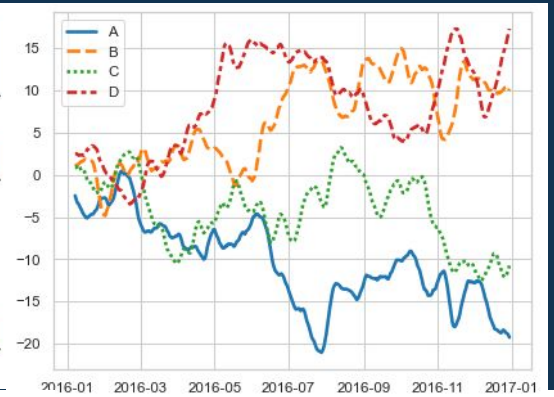
Data visualisation

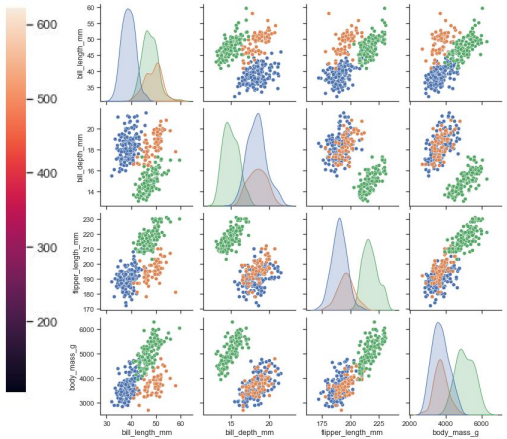
CoGrammar

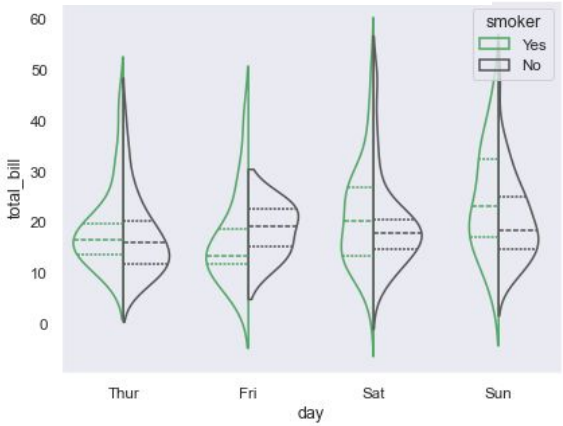
Seaborn examples











CoGrammar

Which plot is best for comparing distributions of several groups?

1. Scatter plot
2. Bar chart
3. Kernel density plots (KDEs)
4. Pie charts

CoGrammar

Which plot is best for comparing distributions of several groups?

1. Scatter plot
2. Bar chart
3. **Kernel density plots (KDEs)**
4. Pie charts

CoGrammar

Violin plots are ideal for:

1. Showing relationships over time
2. Comparing individual values
3. Highlighting correlations only
4. Showing density alongside summary statistics

CoGrammar

Violin plots are ideal for:

1. Showing relationships over time
2. Comparing individual values
3. Highlighting correlations only
4. **Showing density alongside summary statistics**

CoGrammar

Advanced Box Plots can be enhanced by adding:

1. Color-coded matrices
2. Raw data points or swarm plots
3. Vertical axes for each feature
4. Pre-attentive attributes

CoGrammar

Advanced Box Plots can be enhanced by adding:

1. Color-coded matrices
2. **Raw data points or swarm plots**
3. Vertical axes for each feature
4. Pre-attentive attributes

CoGrammar

Heatmaps are excellent for:

1. Depicting pairwise relationships
2. Revealing structure, highlighting correlations, and identifying clusters
3. Comparing distributions
4. Showing many dimensions on the same plot

CoGrammar

Heatmaps are excellent for:

1. Depicting pairwise relationships
2. **Revealing structure, highlighting correlations, and identifying clusters**
3. Comparing distributions
4. Showing many dimensions on the same plot

CoGrammar

Parallel Coordinates are used to:

1. Show many dimensions on the same plot
2. Compare individual values
3. Analyze trends over time
4. Focus on relationships between two variables

CoGrammar

Parallel Coordinates are used to:

1. **Show many dimensions on the same plot**
2. Compare individual values
3. Analyze trends over time
4. Focus on relationships between two variables

CoGrammar

Which principle emphasizes the use of clear labels, thoughtful color choice, and minimal clutter for effective visualization?

1. Visual Perception
2. Chart Choice vs. Your Question
3. Less is More
4. The Principle of Detailed Complexity

CoGrammar

Which principle emphasizes the use of clear labels, thoughtful color choice, and minimal clutter for effective visualization?

1. Visual Perception
2. Chart Choice vs. Your Question
3. **Less is More**
4. The Principle of Detailed Complexity

CoGrammar

Jupyter notebook

CoGrammar

Jupyter Notebook

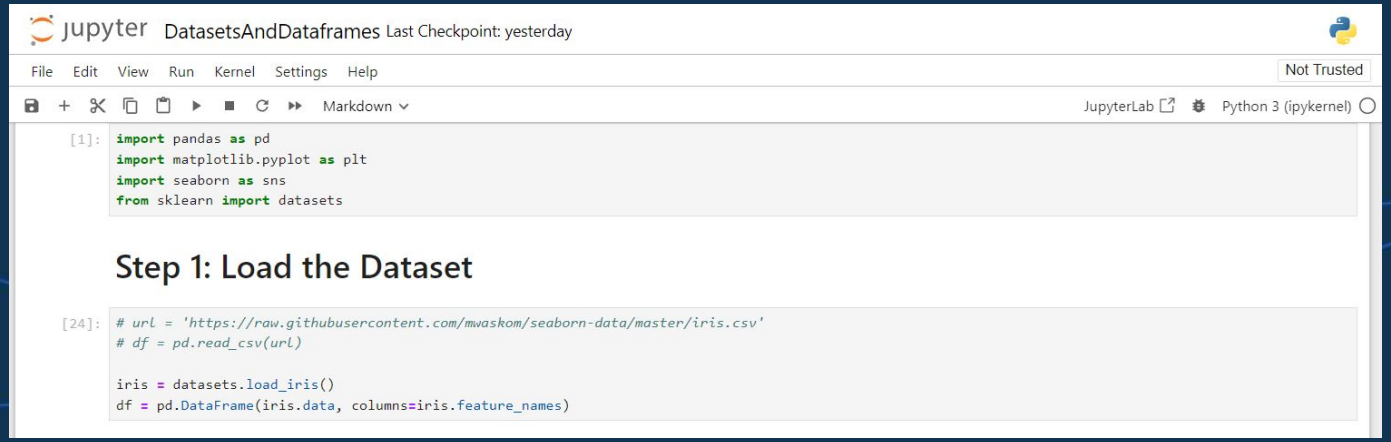
- An interactive environment perfect for data science work. They let you combine code, the results of the code (output), and explanatory text (like in a scientific report).

- This fosters clear data exploration and storytelling, all in one place

Running

```
jupyter notebook
```

```
python -m notebook
```



CoGrammar

Questions and Answers

CoGrammar

Thank you for attending

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar