CM4125 Topic 4 - Data Loading and Exploration

Lecture objectives

Lecture objectives

1. Understand how we define data (for the purpose of this module) and where to get it

Lecture objectives

1. Understand how we define data (for the purpose of this module) and where to get it

2. Learn different methods to import/export data into practical representations

Lecture objectives

1. Understand how we define data (for the purpose of this module) and where to get it
2. Learn different methods to import/export data into practical representations
3. Apply basic commands and concepts to explore it

What is data (for data viz purposes)?

Information or characteristics collected through observation or experimentation

Information or characteristics collected through observation or experimentation

Using a tabular approach...

Information or characteristics collected through observation or experimentation

Using a tabular approach...

Each entry/observation corresponds to a `row`

Information or characteristics collected through observation or experimentation

Using a tabular approach...

Each entry/observation corresponds to a `row`

Each feature/characteristic corresponds to a `column`

Where to get data from?

”From a friend”

”From a friend”

We are given access to a file or register containing the data

”From a friend”

We are given access to a file or register containing the data

There are many formats, but mostly we will deal with:

```
.csv
.tsv
.txt
.xlsx
.json
```

In [1]:

```
## Example: Reading a .csv file into python
## This file contains the age and height of some participants
import pandas as pd
df = pd.read_csv('https://www.dropbox.com/s/9aiiad9j6zxs07i/data.csv?raw=1')
df
```

Out[1]:

	Col 1	Col 2	Col 3
0	Nick	21	1.85
1	Chris	29	1.79
2	Tim	28	1.75
3	Ron	34	1.81
4	Monica	35	1.69
5	Cassandra	21	1.66

Online repos

Online repos

- There is a huge amount of free data out there!
- Just be careful where you get it!
- Websites such as Kaggle, UCI or Keel contain thousands of examples that you can download and import to your preferred tool
- Sometimes they even contain ways to connect to your data in faster and more secure ways

Online repos

- There is a huge amount of free data out there!
- Just be careful where you get it!
- Websites such as Kaggle, UCI or Keel contain thousands of examples that you can download and import to your preferred tool
- Sometimes they even contain ways to connect to your data in faster and more secure ways

Modules and packages

Modules and packages

Languages such as Python and R already contain some preloaded data repositories

Modules and packages

Languages such as Python and R already contain some preloaded data repositories

They are sometimes in weird shapes and forms

Modules and packages

Languages such as Python and R already contain some preloaded data repositories

They are sometimes in weird shapes and forms

Nonetheless, they are good to start experimenting!

In [2]:

```
## Loading the IRIS dataset from the SCIKIT LEARN module
from sklearn.datasets import load_iris
iris = load_iris()
iris
```

Out[2]:

```
{'data': array([[5.1, 3.5, 1.4, 0.2],
        [4.9, 3. , 1.4, 0.2],
        [4.7, 3.2, 1.3, 0.2],
        [4.6, 3.1, 1.5, 0.2],
        [5. , 3.6, 1.4, 0.2],
        [5.4, 3.9, 1.7, 0.4],
        [4.6, 3.4, 1.4, 0.3],
        [5. , 3.4, 1.5, 0.2],
        [4.4, 2.9, 1.4, 0.2],
        [4.9, 3.1, 1.5, 0.1],
        [5.4, 3.7, 1.5, 0.2],
        [4.8, 3.4, 1.6, 0.2],
        [4.8, 3. , 1.4, 0.1],
        [4.3, 3. , 1.1, 0.1],
        [5.8, 4. , 1.2, 0.2],
        [5.7, 4.4, 1.5, 0.4],
        [5.4, 3.9, 1.3, 0.4],
        [5.1, 3.5, 1.4, 0.3],
        [5.7, 3.8, 1.7, 0.3],
        [5.1, 3.8, 1.5, 0.3],
        [5.4, 3.4, 1.7, 0.2],
        [5.1, 3.7, 1.5, 0.4],
        [4.6, 3.6, 1. , 0.2],
        [5.1, 3.3, 1.7, 0.5],
        [4.8, 3.4, 1.9, 0.2],
        [5. , 3. , 1.6, 0.2],
        [5. , 3.4, 1.6, 0.4],
        [5.2, 3.5, 1.5, 0.2],
        [5.2, 3.4, 1.4, 0.2],
        [4.7, 3.2, 1.6, 0.2],
        [4.8, 3.1, 1.6, 0.2],
```

```
[5.4, 3.4, 1.5, 0.4],
[5.2, 4.1, 1.5, 0.1],
[5.5, 4.2, 1.4, 0.2],
[4.9, 3.1, 1.5, 0.2],
[5. , 3.2, 1.2, 0.2],
[5.5, 3.5, 1.3, 0.2],
[4.9, 3.6, 1.4, 0.1],
[4.4, 3. , 1.3, 0.2],
[5.1, 3.4, 1.5, 0.2],
[5. , 3.5, 1.3, 0.3],
[4.5, 2.3, 1.3, 0.3],
[4.4, 3.2, 1.3, 0.2],
[5. , 3.5, 1.6, 0.6],
[5.1, 3.8, 1.9, 0.4],
[4.8, 3. , 1.4, 0.3],
[5.1, 3.8, 1.6, 0.2],
[4.6, 3.2, 1.4, 0.2],
[5.3, 3.7, 1.5, 0.2],
[5. , 3.3, 1.4, 0.2],
[7. , 3.2, 4.7, 1.4],
[6.4, 3.2, 4.5, 1.5],
[6.9, 3.1, 4.9, 1.5],
[5.5, 2.3, 4. , 1.3],
[6.5, 2.8, 4.6, 1.5],
[5.7, 2.8, 4.5, 1.3],
[6.3, 3.3, 4.7, 1.6],
[4.9, 2.4, 3.3, 1. ],
[6.6, 2.9, 4.6, 1.3],
[5.2, 2.7, 3.9, 1.4],
[5. , 2. , 3.5, 1. ],
[5.9, 3. , 4.2, 1.5],
```

```
[6. , 2.2, 4. , 1. ],
[6.1, 2.9, 4.7, 1.4],
[5.6, 2.9, 3.6, 1.3],
[6.7, 3.1, 4.4, 1.4],
[5.6, 3. , 4.5, 1.5],
[5.8, 2.7, 4.1, 1. ],
[6.2, 2.2, 4.5, 1.5],
[5.6, 2.5, 3.9, 1.1],
[5.9, 3.2, 4.8, 1.8],
[6.1, 2.8, 4. , 1.3],
[6.3, 2.5, 4.9, 1.5],
[6.1, 2.8, 4.7, 1.2],
[6.4, 2.9, 4.3, 1.3],
[6.6, 3. , 4.4, 1.4],
[6.8, 2.8, 4.8, 1.4],
[6.7, 3. , 5. , 1.7],
[6. , 2.9, 4.5, 1.5],
[5.7, 2.6, 3.5, 1. ],
[5.5, 2.4, 3.8, 1.1],
[5.5, 2.4, 3.7, 1. ],
[5.8, 2.7, 3.9, 1.2],
[6. , 2.7, 5.1, 1.6],
[5.4, 3. , 4.5, 1.5],
[6. , 3.4, 4.5, 1.6],
[6.7, 3.1, 4.7, 1.5],
[6.3, 2.3, 4.4, 1.3],
[5.6, 3. , 4.1, 1.3],
[5.5, 2.5, 4. , 1.3],
[5.5, 2.6, 4.4, 1.2],
[6.1, 3. , 4.6, 1.4],
[5.8, 2.6, 4. , 1.2],
```

```
[5. , 2.3, 3.3, 1. ],
[5.6, 2.7, 4.2, 1.3],
[5.7, 3. , 4.2, 1.2],
[5.7, 2.9, 4.2, 1.3],
[6.2, 2.9, 4.3, 1.3],
[5.1, 2.5, 3. , 1.1],
[5.7, 2.8, 4.1, 1.3],
[6.3, 3.3, 6. , 2.5],
[5.8, 2.7, 5.1, 1.9],
[7.1, 3. , 5.9, 2.1],
[6.3, 2.9, 5.6, 1.8],
[6.5, 3. , 5.8, 2.2],
[7.6, 3. , 6.6, 2.1],
[4.9, 2.5, 4.5, 1.7],
[7.3, 2.9, 6.3, 1.8],
[6.7, 2.5, 5.8, 1.8],
[7.2, 3.6, 6.1, 2.5],
[6.5, 3.2, 5.1, 2. ],
[6.4, 2.7, 5.3, 1.9],
[6.8, 3. , 5.5, 2.1],
[5.7, 2.5, 5. , 2. ],
[5.8, 2.8, 5.1, 2.4],
[6.4, 3.2, 5.3, 2.3],
[6.5, 3. , 5.5, 1.8],
[7.7, 3.8, 6.7, 2.2],
[7.7, 2.6, 6.9, 2.3],
[6. , 2.2, 5. , 1.5],
[6.9, 3.2, 5.7, 2.3],
[5.6, 2.8, 4.9, 2. ],
[7.7, 2.8, 6.7, 2. ],
[6.3, 2.7, 4.9, 1.8],
```

```
        [6.7, 3.3, 5.7, 2.1],
        [7.2, 3.2, 6. , 1.8],
        [6.2, 2.8, 4.8, 1.8],
        [6.1, 3. , 4.9, 1.8],
        [6.4, 2.8, 5.6, 2.1],
        [7.2, 3. , 5.8, 1.6],
        [7.4, 2.8, 6.1, 1.9],
        [7.9, 3.8, 6.4, 2. ],
        [6.4, 2.8, 5.6, 2.2],
        [6.3, 2.8, 5.1, 1.5],
        [6.1, 2.6, 5.6, 1.4],
        [7.7, 3. , 6.1, 2.3],
        [6.3, 3.4, 5.6, 2.4],
        [6.4, 3.1, 5.5, 1.8],
        [6. , 3. , 4.8, 1.8],
        [6.9, 3.1, 5.4, 2.1],
        [6.7, 3.1, 5.6, 2.4],
        [6.9, 3.1, 5.1, 2.3],
        [5.8, 2.7, 5.1, 1.9],
        [6.8, 3.2, 5.9, 2.3],
        [6.7, 3.3, 5.7, 2.5],
        [6.7, 3. , 5.2, 2.3],
        [6.3, 2.5, 5. , 1.9],
        [6.5, 3. , 5.2, 2. ],
        [6.2, 3.4, 5.4, 2.3],
        [5.9, 3. , 5.1, 1.8]]),
 'target': array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0,
        0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
```

```
, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2
, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2
, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]),
 'frame': None,
 'target_names': array(['setosa', 'versicolor', 'virginica'], dtype
='<U10'),
 'DESCR': '.. _iris_dataset:\n\nIris plants dataset
\n--------------------\n\n**Data Set Characteristics:**\n\n:Number of
Instances: 150 (50 in each of three classes)\n:Number of Attributes:
4 numeric, predictive attributes and the class\n:Attribute Informatio
n:\n    - sepal length in cm\n    - sepal width in cm\n    - petal le
ngth in cm\n    - petal width in cm\n    - class:\n            - Iri
s-Setosa\n            - Iris-Versicolour\n            - Iris-Virginic
a\n\n:Summary Statistics:\n\n============== ==== ==== ======= ===== =
==================\n                Min  Max   Mean    SD   Class Co
rrelation\n============== ==== ==== ======= ===== ==================
=\nsepal length:   4.3  7.9   5.84   0.83    0.7826\nsepal width:
2.0  4.4   3.05   0.43   -0.4194\npetal length:   1.0  6.9   3.76   1
.76    0.9490  (high!)\npetal width:    0.1  2.5   1.20   0.76    0.9
565  (high!)\n============== ==== ==== ======= ===== ================
====\n\n:Missing Attribute Values: None\n:Class Distribution: 33.3% f
or each of 3 classes.\n:Creator: R.A. Fisher\n:Donor: Michael Marshal
l (MARSHALL%PLU@io.arc.nasa.gov)\n:Date: July, 1988\n\nThe famous Iri
s database, first used by Sir R.A. Fisher. The dataset is taken\nfrom
Fisher\'s paper. Note that it\'s the same as in R, but not as in the
UCI\nMachine Learning Repository, which has two wrong data points.\n
\nThis is perhaps the best known database to be found in the\npattern
```

Exploring Data by Rows

Now that we have data in a tabular form, let's see how to access certain positions

Now that we have data in a tabular form, let's see how to access certain positions

To do so, let's use a larger dataset of Netflix Original series contained in a .csv file called `netflix.csv`

In [3]:
```
netflix = pd.read_csv('https://www.dropbox.com/s/pwqaqftq2m9pgdv/netflix.csv?r
netflix
```

Out[3]:

	Title	Genre	GenreLabels	Premiere	Seasons
0	House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes
1	Hemlock Grove	Horror/thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes
2	Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes
3	Marco Polo	Historical drama	historical,drama	12-Dec-14	2 seasons, 20 episodes
4	Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes
...	...	...	...	...	...
353	Busted!	Korean language variety	korean,language,variety,show	4-May-18	1 season,

	Title	Genre	GenreLabels	Premiere	Seasons
		show			10 episodes
354	The Break with Michelle Wolf	Late-night	late-night	27-May-18	1 season, 10 episodes
355	Norm Macdonald Has a Show	Talk show	talk,show	14-Sep-18	1 season, 10 episodes
356	Patriot Act with Hasan Minhaj	Talk show	talk,show	28-Oct-18	3 volumes, 19 episodes
357	The Fix	Panel show	panel,show	14-Dec-18	1 season, 10 episodes

358 rows × 14 columns

In [3]:
```
netflix = pd.read_csv('https://www.dropbox.com/s/pwqaqftq2m9pgdv/netflix.csv?r
netflix
```

Out[3]:

	Title	Genre	GenreLabels	Premiere	Seasons
0	House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes
1	Hemlock Grove	Horror/thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes
2	Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes
3	Marco Polo	Historical drama	historical,drama	12-Dec-14	2 seasons, 20 episodes
4	Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes
...	...	...	...	...	...
353	Busted!	Korean language variety	korean,language,variety,show	4-May-18	1 season,

	Title	Genre	GenreLabels	Premiere	Seasons
		show			10 episodes
354	The Break with Michelle Wolf	Late-night	late-night	27-May-18	1 season, 10 episodes
355	Norm Macdonald Has a Show	Talk show	talk,show	14-Sep-18	1 season, 10 episodes
356	Patriot Act with Hasan Minhaj	Talk show	talk,show	28-Oct-18	3 volumes, 19 episodes
357	The Fix	Panel show	panel,show	14-Dec-18	1 season, 10 episodes

358 rows × 14 columns

Notice that when the `DataFrame` is shown in Jupyter, it displays `358 rows × 14 columns` at the bottom to tell us how large the dataset is.

Using a Non-Numerical Index

We can actually use any of the unique-entry columns (for example `Title`, which is column `0`)

We can actually use any of the unique-entry columns (for example `Title`, which is column `0`)

In [4]:
```
# reloading the dataset, now stating the first column as the index one
netflix = pd.read_csv('https://www.dropbox.com/s/pwqaqftq2m9pgdv/netflix.csv?r
                      index_col=0)
netflix
```

Out[4]:

Title	Genre	GenreLabels	Premiere	Seasons	Seas
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	
Hemlock Grove	Horror/thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes	
Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes	
Marco Polo	Historical drama	historical,drama	12-Dec-14	2 seasons, 20 episodes	
Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes	
...	...	...	...	...	
Busted!	Korean	korean,language,variety,show	4-	1	

Title	Genre	GenreLabels	Premiere	Seasons	Seas
	language variety show		May-18	season, 10 episodes	
The Break with Michelle Wolf	Late-night	late-night	27-May-18	1 season, 10 episodes	
Norm Macdonald Has a Show	Talk show	talk,show	14-Sep-18	1 season, 10 episodes	
Patriot Act with Hasan Minhaj	Talk show	talk,show	28-Oct-18	3 volumes, 19 episodes	
The Fix	Panel show	panel,show	14-Dec-18	1 season, 10 episodes	

358 rows × 13 columns

It's up to you to decide whether to use default numerical indexing or to use an uniquely-identifying column from your dataset.

Locating an item

In [5]:
```
# Locating all info of an entry
netflix.loc['Orange Is the New Black']
```

Out[5]:
```
Genre                         Comedy-drama
GenreLabels                   comedy-drama
Premiere                       11-Jul-13
Seasons            6 seasons, 78 episodes
SeasonsParsed                          6
EpisodesParsed                        78
Length                       50–92 min.
MinLength                              50
MaxLength                              92
Status                           Renewed
Active                                 1
Table                              Drama
Language                         English
Name: Orange Is the New Black, dtype: object
```

In [6]:
```
# Locating particular info of that entry
netflix.at['Orange Is the New Black', 'Seasons']
```

Out[6]:
```
'6 seasons, 78 episodes'
```

In [7]:
```
# This one allows you to access by number, even if the table has no numerical
netflix.iloc[0]
```

Out[7]:

```
Genre                    Political drama
GenreLabels              political,drama
Premiere                        1-Feb-13
Seasons           6 seasons, 73 episodes
SeasonsParsed                          6
EpisodesParsed                        73
Length                       42–59 min.
MinLength                              42
MaxLength                              59
Status                             Ended
Active                                 0
Table                              Drama
Language                         English
Name: House of Cards, dtype: object
```

Getting a Subset of the Data

In [8]:
```
netflix.head(5)
```

Out[8]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesP
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	6	
Hemlock Grove	Horror/ thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes	3	
Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes	6	
Marco Polo	Historical drama	historical,drama	12-Dec-14	2 seasons, 20 episodes	2	
Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes	3	

In [9]:
```
netflix.tail(3)
```

Out[9]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Norm Macdonald Has a Show	Talk show	talk,show	14-Sep-18	1 season, 10 episodes	1	
Patriot Act with Hasan Minhaj	Talk show	talk,show	28-Oct-18	3 volumes, 19 episodes	0	
The Fix	Panel show	panel,show	14-Dec-18	1 season, 10 episodes	1	

In [11]:
```
netflix.sample(5)
```

Out[11]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsP
The Protector	Supernatural mystery	supernatural,mystery	14-Dec-18	2 seasons, 18 episodes	
Lego Friends: The Power of Friendship	childrens-animation	childrens-animation	4-Mar-16	2 seasons, 4 episodes	
The Boss Baby: Back in Business	childrens-animation	childrens-animation	6-Apr-18	2 seasons, 26 episodes	
First and Last	Reality	reality	7-Sep-18	1 season, 6 episodes	
Grace and Frankie	Comedy-drama	comedy-drama	8-May-15	5 seasons, 65 episodes	

Choosing Specific Rows

In [12]:
```
netflix.loc[['Diablero', 'Motown Magic', 'Typewriter']] # by index
```

Out[12]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsF
Diablero	Horror fantasy thriller	horror,fantasy,thriller	21-Dec-18	1 season, 8 episodes	
Motown Magic	childrens-animation	childrens-animation	20-Nov-18	1 season, 25 episodes	
Typewriter	Horror	horror	19-Jul-19	TBA	

In [13]:
```
netflix.iloc[[40, 12, 106, 79]] # by numerical index
```

Out[13]:

Title	Genre	GenreLabels	Premiere	Seasons	Seasons
Jessica Jones	Neo-noir/ psychological thriller	neo-noir,psychological,thriller	20-Nov-15	3 seasons, 39 episodes	
A Series of Unfortunate Events	Black-comedy mystery	black-comedy,mystery	13-Jan-17	3 seasons, 25 episodes	
Lost Song	Musical fantasy	musical,fantasy	31-Mar-18	1 season, 12 episodes	
After Life	Comedy	comedy	8-Mar-19	1 season, 6 episodes	

In [14]:
```
netflix.loc['Maniac':'After Life'] # selecting a range by index
```

Out[14]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	Episode
Maniac	Dark comedy	dark,comedy	21-Sep-18	10 episodes	0	
The Kominsky Method	Comedy	comedy	16-Nov-18	1 season, 8 episodes	1	
Sex Education	Coming-of-age comedy-drama	coming-of-age,comedy-drama	11-Jan-19	1 season, 8 episodes	1	
Russian Doll	Comedy	comedy	1-Feb-19	1 season, 8 episodes	1	
After Life	Comedy	comedy	8-Mar-19	1 season, 6 episodes	1	

In [15]:
```
netflix.iloc[75:79] # selecting a range by numerical index
```

Out[15]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	Episode
Maniac	Dark comedy	dark,comedy	21-Sep-18	10 episodes	0	
The Kominsky Method	Comedy	comedy	16-Nov-18	1 season, 8 episodes	1	
Sex Education	Coming-of-age comedy-drama	coming-of-age,comedy-drama	11-Jan-19	1 season, 8 episodes	1	
Russian Doll	Comedy	comedy	1-Feb-19	1 season, 8 episodes	1	

Filtering Rows by Values in Columns

One of the most important features you need is to be able to filter observations out

One of the most important features you need is to be able to filter observations out

We can define a **condition** so that it can be checked for all entries

In [16]:
```
our_condition = netflix['SeasonsParsed'] == 6
our_condition
```

Out[16]:
```
Title
House of Cards                        True
Hemlock Grove                        False
Orange Is the New Black               True
Marco Polo                           False
Bloodline                            False
                                     ...
Busted!                              False
The Break with Michelle Wolf         False
Norm Macdonald Has a Show            False
Patriot Act with Hasan Minhaj        False
The Fix                              False
Name: SeasonsParsed, Length: 358, dtype: bool
```

Then, we can ask for that condition to be the filter of the original dataset

Then, we can ask for that condition to be the filter of the original dataset

In [17]:
```
netflix[our_condition]
```

Out[17]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	Episode
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	6	
Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes	6	
The Adventures of Puss in Boots	childrens-animation	childrens-animation	16-Jan-15	6 seasons, 78 episodes	6	
Dragons: Race to the Edge	childrens-animation	childrens-animation	26-Jun-15	6 seasons, 78 episodes	6	
Trolls: The Beat Goes On!	childrens-animation	childrens-animation	19-Jan-18	6 seasons, 38 episodes	6	

In [18]:
```
netflix[netflix['SeasonsParsed'] == 6] # the same, but in one line
```

Out[18]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	Episode
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	6	
Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes	6	
The Adventures of Puss in Boots	childrens-animation	childrens-animation	16-Jan-15	6 seasons, 78 episodes	6	
Dragons: Race to the Edge	childrens-animation	childrens-animation	26-Jun-15	6 seasons, 78 episodes	6	
Trolls: The Beat Goes On!	childrens-animation	childrens-animation	19-Jan-18	6 seasons, 38 episodes	6	

In [19]:
```
netflix[netflix['MaxLength'] > 100] # series longer than 100 minutes
```

Out[19]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesParsed
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	24
Gilmore Girls: A Year in the Life	Family drama	family,drama	25-Nov-16	4 episodes	0	4

In [20]:

```
# series with a number of episodes between 30 and 34
netflix[(netflix['EpisodesParsed'] >= 30) & (netflix['EpisodesParsed'] < 35)]
```

Out[20]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	Episode
Hemlock Grove	Horror/ thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes	3	
Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes	3	
Narcos	Crime drama	crime,drama	28-Aug-15	3 seasons, 30 episodes	3	
Love	Romantic comedy	romantic,comedy	19-Feb-16	3 seasons, 34 episodes	3	
Santa Clarita Diet	Comedy-horror	comedy-horror	3-Feb-17	3 seasons, 30 episodes	3	
Go! Live Your Wav	Musical	musical	22-Feb-19	2 seasons, 30	2	

In [21]:
```
# with 33 or 73 episodes
netflix[(netflix['EpisodesParsed'] == 33) | (netflix['EpisodesParsed'] == 73)]
```

Out[21]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPar
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	6	
Hemlock Grove	Horror/ thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes	3	
Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes	3	

Sorting Rows

Let's create a new dataframe called `variable_length_shows` that contains series where longest episode is 45 minutes longer than the shortest

Let's create a new dataframe called `variable_length_shows` that contains series where longest episode is 45 minutes longer than the shortest

In [22]:
```
variable_length_shows = netflix[netflix['MaxLength'] > netflix['MinLength'] +
variable_length_shows
```

Out[22]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	
Kong: King of the Apes	childrens-animation	childrens-animation	15-Apr-16	2 seasons, 23 episodes	2	
Club de Cuervos	Comedy-drama	comedy-drama	7-Aug-15	4 seasons, 45 episodes	4	

We can use the `sort_values` method to sort on a chosen variable.

We can use the `sort_values` method to sort on a chosen variable.

In [23]:
```
variable_length_shows.sort_values('MaxLength')
```

Out[23]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Kong: King of the Apes	childrens-animation	childrens-animation	15-Apr-16	2 seasons, 23 episodes	2	
Club de Cuervos	Comedy-drama	comedy-drama	7-Aug-15	4 seasons, 45 episodes	4	
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	

We can also sort in reverse (descending) order using `ascending=False`.

We can also sort in reverse (descending) order using `ascending=False`.

In [24]:
```
variable_length_shows.sort_values('MaxLength', ascending=False)
```

Out[24]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	
Club de Cuervos	Comedy-drama	comedy-drama	7-Aug-15	4 seasons, 45 episodes	4	
Kong: King of the Apes	childrens-animation	childrens-animation	15-Apr-16	2 seasons, 23 episodes	2	

Note that `sort_values` does **not** actually modify the contents of the data frame!

Note that `sort_values` does **not** actually modify the contents of the data frame!

If we use the ascendig order, we will see it is not sorted!

In [25]:
```
variable_length_shows.sort_values('MaxLength')
variable_length_shows
```

Out[25]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	
Kong: King of the Apes	childrens-animation	childrens-animation	15-Apr-16	2 seasons, 23 episodes	2	
Club de Cuervos	Comedy-drama	comedy-drama	7-Aug-15	4 seasons, 45 episodes	4	

Solution: Create a new variable!

Solution: Create a new variable!

In [26]:
```
new = variable_length_shows.sort_values('MaxLength')
new
```

Out[26]:

Title	Genre	GenreLabels	Premiere	Seasons	SeasonsParsed	EpisodesPars
Kong: King of the Apes	childrens-animation	childrens-animation	15-Apr-16	2 seasons, 23 episodes	2	
Club de Cuervos	Comedy-drama	comedy-drama	7-Aug-15	4 seasons, 45 episodes	4	
Sense8	Science fiction	science-fiction	5-Jun-15	2 seasons, 24 episodes	2	

Creating a new dataset only with certain columns

Creating a new dataset only with certain columns

In [27]:
```
shorter_netflix = netflix[['Genre', 'SeasonsParsed']]
shorter_netflix
```

Out[27]:

Title	Genre	SeasonsParsed
House of Cards	Political drama	6
Hemlock Grove	Horror/thriller	3
Orange Is the New Black	Comedy-drama	6
Marco Polo	Historical drama	2
Bloodline	Thriller	3
...	...	...
Busted!	Korean language variety show	1
The Break with Michelle Wolf	Late-night	1
Norm Macdonald Has a Show	Talk show	1
Patriot Act with Hasan Minhaj	Talk show	0
The Fix	Panel show	1

358 rows × 2 columns

Adding new columns to a data frame (we will continue discussing this on the next Topic)

Adding new columns to a data frame (we will continue discussing this on the next Topic)

In [28]:
```
# creating a new column named 'watched' filled with no
netflix['Watched'] = ['No']*len(netflix)
netflix
```

Out[28]:

Title	Genre	GenreLabels	Premiere	Seasons	Seas
House of Cards	Political drama	political,drama	1-Feb-13	6 seasons, 73 episodes	
Hemlock Grove	Horror/thriller	horror,thriller	19-Apr-13	3 seasons, 33 episodes	
Orange Is the New Black	Comedy-drama	comedy-drama	11-Jul-13	6 seasons, 78 episodes	
Marco Polo	Historical drama	historical,drama	12-Dec-14	2 seasons, 20 episodes	
Bloodline	Thriller	thriller	20-Mar-15	3 seasons, 33 episodes	
...	...	...	...	...	
Busted!	Korean	korean,language,variety,show	4-	1	

Title	Genre	GenreLabels	Premiere	Seasons	Seas
	language variety show		May-18	season, 10 episodes	
The Break with Michelle Wolf	Late-night	late-night	27-May-18	1 season, 10 episodes	
Norm Macdonald Has a Show	Talk show	talk,show	14-Sep-18	1 season, 10 episodes	
Patriot Act with Hasan Minhaj	Talk show	talk,show	28-Oct-18	3 volumes, 19 episodes	
The Fix	Panel show	panel,show	14-Dec-18	1 season, 10 episodes	

358 rows × 14 columns

Finally

Talk about **harnessing the Power of AI** in Google Colab!

Lab Activity

Lab Activity

If you are going to use Python, open `CM4125_Lab4.ipynb` using **Google Colab** and follow the steps.

Lab Activity

If you are going to use Python, open `CM4125_Lab4.ipynb` using **Google Colab** and follow the steps.

Or, you can read `CM4125_Lab4_guided.html` to follow the logic used on the data.

- Try to replicate the steps in your preferred tool!