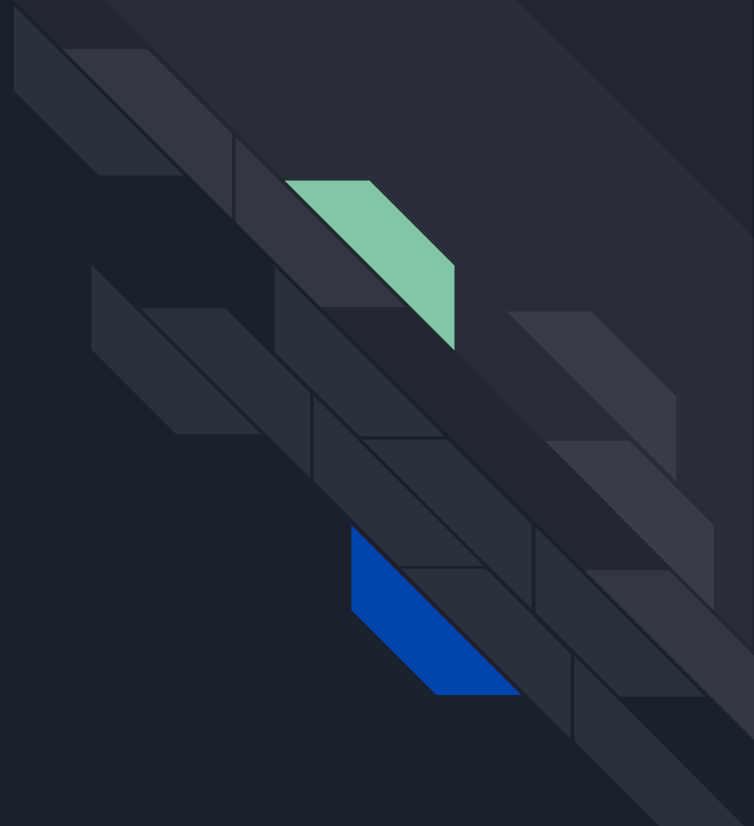
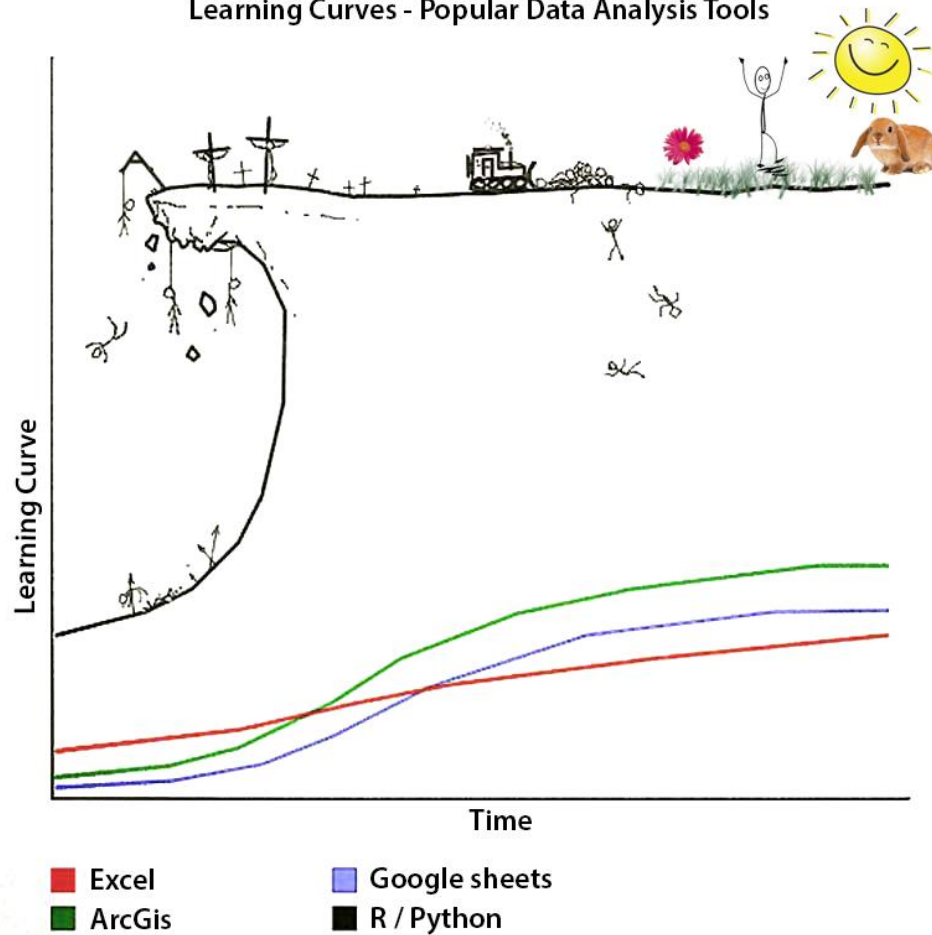


Introduction to Data Science with Python

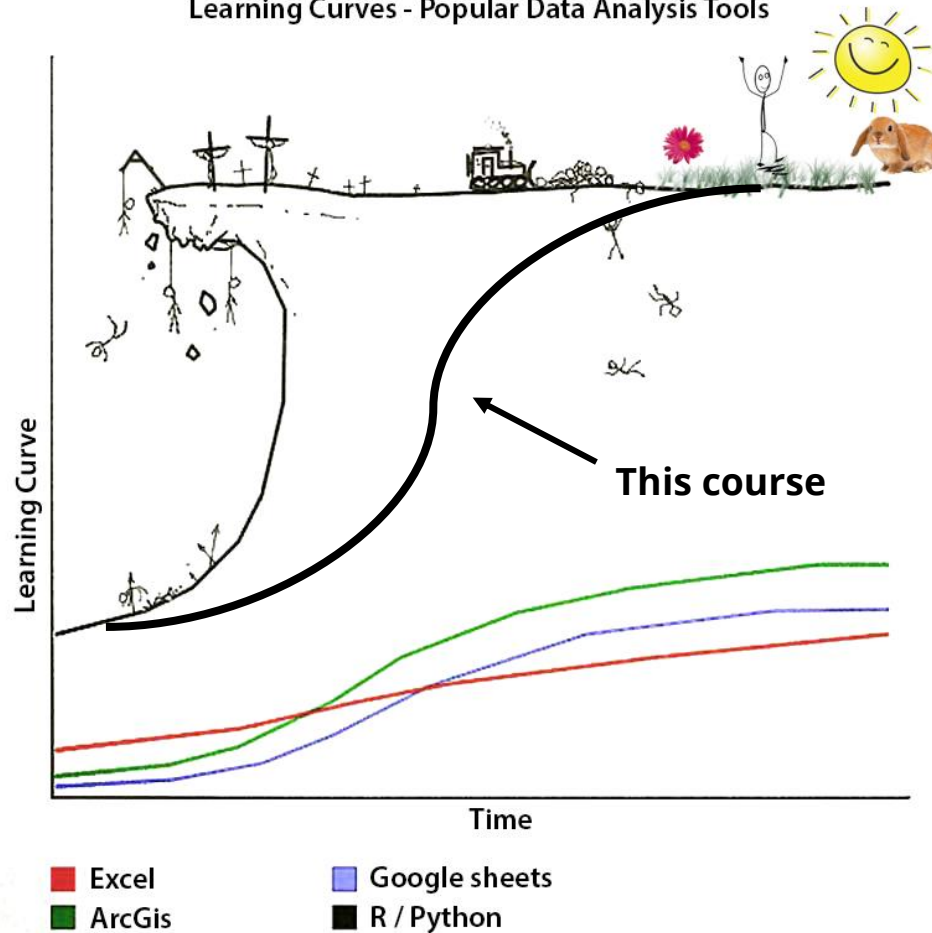
Chapter 1



Learning Curves - Popular Data Analysis Tools



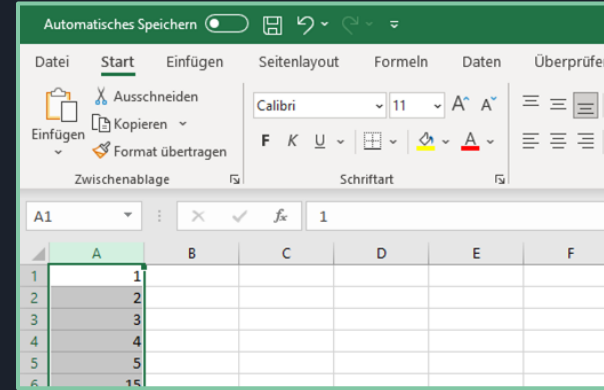
Learning Curves - Popular Data Analysis Tools



Excel vs Python



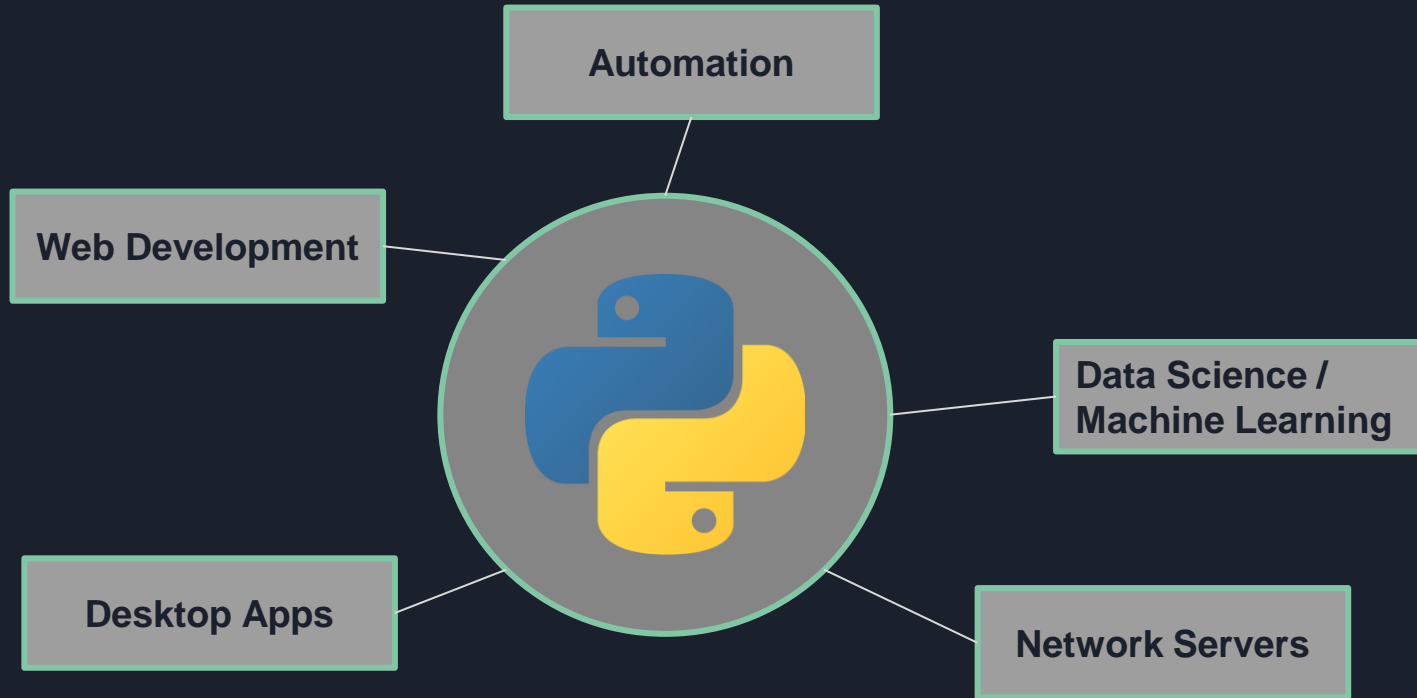
1. Select data
2. Click on buttons :)



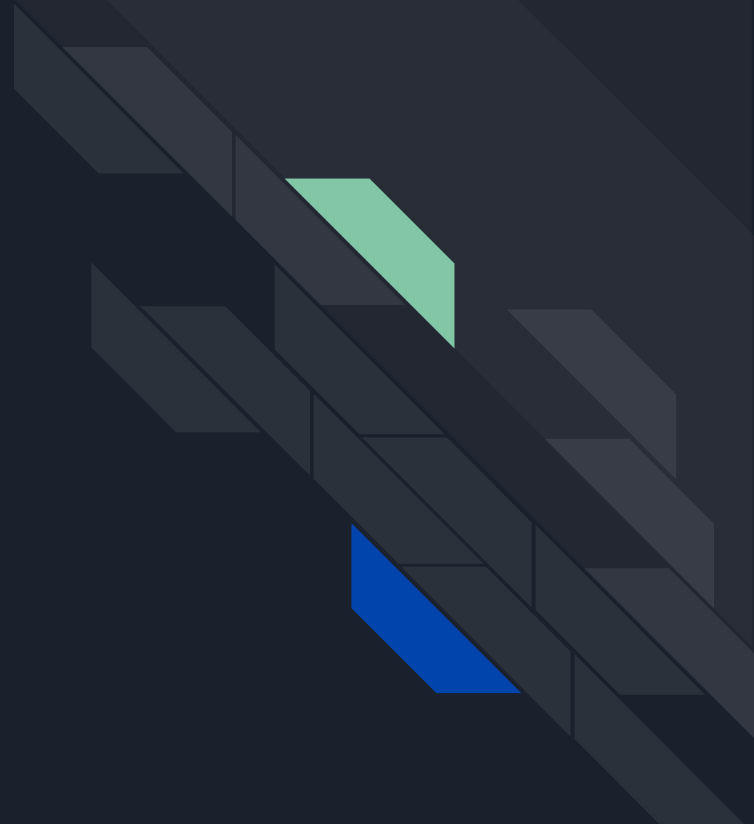
1. Write code in editor
2. Execute code with Python
3. Result will be returned

```
data = pd.read_csv(file)
mean = data.mean()
print(mean)
```

A General Purpose Coding Language



Software



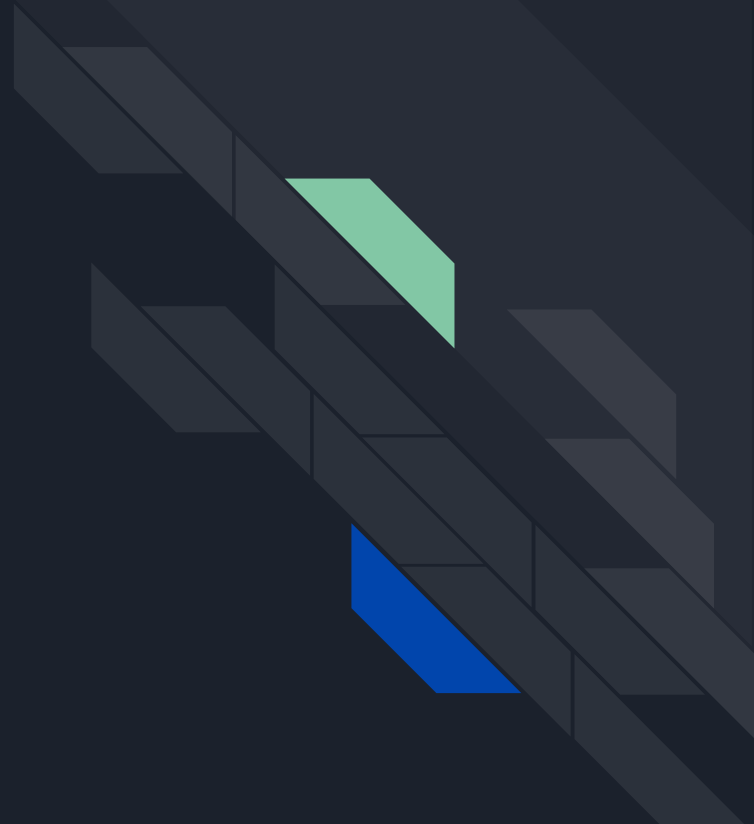



Google Colab

- Write and execute code
- Accessed via Browser (runs on Google Servers)
- No pre-configurations necessary
- Independent of your local machine
- Jupyter Notebook format heavily used in data science community

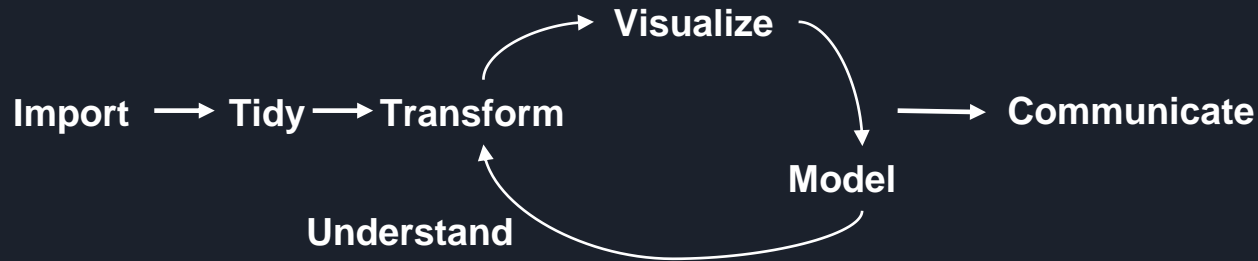
colab

Data science life cycle



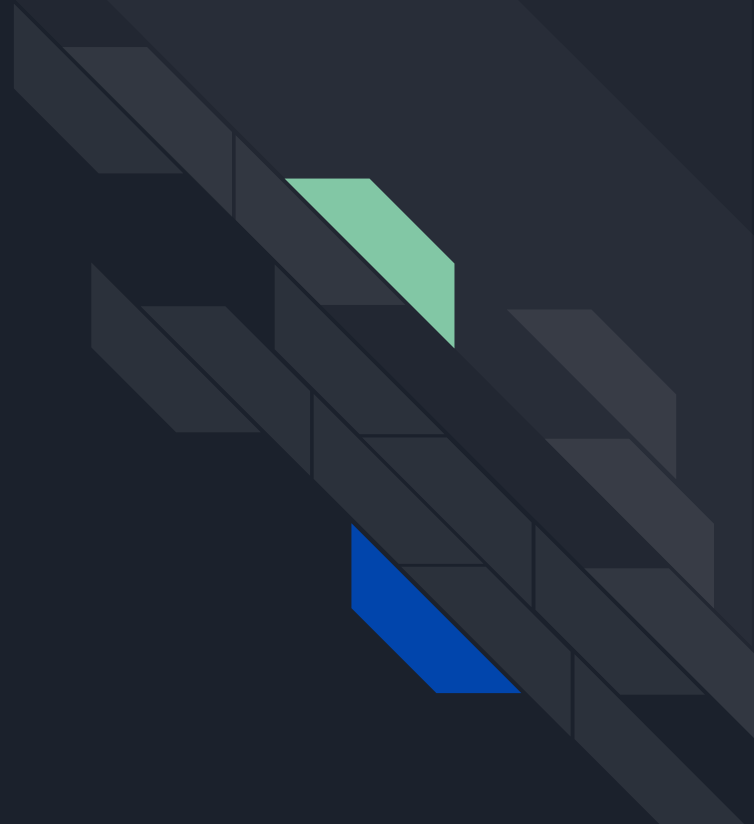


We'll walk you through the data science lifecycle and introduce the tools for each step



Program

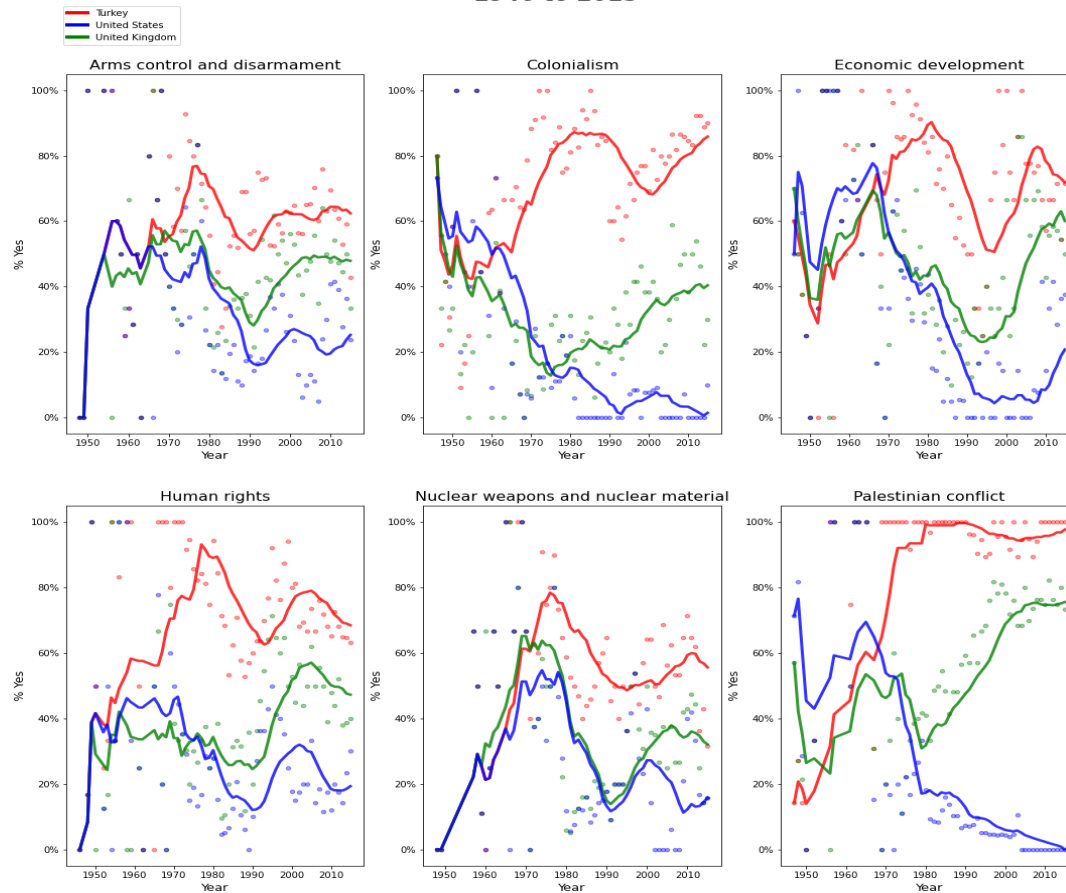
Let's dive in!





Based on data-science.org

Percentage of 'Yes' votes in the UN General Assembly 1946 to 2015



```

1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0,4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])["vote"].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                 Line2D([0], [0], color="blue", lw=4),
14                 Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
22         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
23         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8, linewidth = 3)
24         plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
25                   color = country_color[country], s=20, alpha = 0.4)
26         plt.title(issue, fontdict = {'fontsize' : 16})
27         plt.xlabel("Year", fontdict = {'fontsize' : 13})
28         plt.ylabel("% Yes", fontdict = {'fontsize' : 13})
29         plt.gca().yaxis.set_major_formatter(mtick.PercentFormatter(xmax=1.0))
30
31 plt.suptitle("Percentage of 'Yes' votes in the UN General Assembly\n1946 to 2015", weight = "bold", size = 22)
32 plt.legend(custom_lines, country_color.keys(), bbox_to_anchor=(-2, 2.4))
33 plt.show()

```



```

1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0,4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])["vote"].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                 Line2D([0], [0], color="blue", lw=4),
14                 Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
22         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
23         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8, linewidth = 3)
24         plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
25                   color = country_color[country], s=20, alpha = 0.4)
26         plt.title(issue, fontdict = {'fontsize' : 16})
27         plt.xlabel("Year", fontdict = {'fontsize' : 13})
28         plt.ylabel("% Yes", fontdict = {'fontsize' : 13})
29         plt.gca().yaxis.set_major_formatter(mtick.PercentFormatter(xmax=1.0))
30
31 plt.suptitle("Percentage of 'Yes' votes in the UN General Assembly\n1946 to 2015", weight = "bold", size = 22)
32 plt.legend(custom_lines, country_color.keys(), bbox_to_anchor=(-2, 2.4))
33 plt.show()

```

```

1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0, 4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])["vote"].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                 Line2D([0], [0], color="blue", lw=4),
14                 Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
22         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
23         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8, linewidth = 3)
24         plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
25                   color = country_color[country], s=20, alpha = 0.4)
26         plt.title(issue, fontdict = {'fontsize' : 16})
27         plt.xlabel("Year", fontdict = {'fontsize' : 13})
28         plt.ylabel("% Yes", fontdict = {'fontsize' : 13})
29         plt.gca().yaxis.set_major_formatter(mtick.PercentFormatter(xmax=1.0))
30
31 plt.suptitle("Percentage of 'Yes' votes in the UN General Assembly\n1946 to 2015", weight = "bold", size = 22)
32 plt.legend(custom_lines, country_color.keys(), bbox_to_anchor=(-2, 2.4))
33 plt.show()

```

```

1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6
7 un_votes["year"] = un_votes.date.str.slice(0,4)
8 un_votes["year"] = pd.to_numeric(un_votes["year"])
9 un_votes["vote"] = un_votes["vote"] == "yes"
10 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])["vote"].mean().to_frame().reset_index()
11 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
12
13 custom_lines = [Line2D([0], [0], color="red", lw=4),
14                  Line2D([0], [0], color="blue", lw=4),
15                  Line2D([0], [0], color="green", lw=4)]
16
17 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
18
19 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
20     plt.subplot(2,3,index + 1)
21     for country in un_votes_grouped.country.unique():
22         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
23         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
24         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8, linewidth = 3)
25         plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
26                    color = country_color[country], s=20, alpha = 0.4)
27         plt.title(issue, fontdict = {'fontsize' : 16})
28         plt.xlabel("Year", fontdict = {'fontsize' : 13})
29         plt.ylabel("% Yes", fontdict = {'fontsize' : 13})
30         plt.gca().yaxis.set_major_formatter(mtick.PercentFormatter(xmax=1.0))
31
32 plt.suptitle("Percentage of 'Yes' votes in the UN General Assembly\n1946 to 2015", weight = "bold", size = 22)
33 plt.legend(custom_lines, country_color.keys(), bbox_to_anchor=(-2, 2.4))
34 plt.show()

```



```

1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0,4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])["vote"].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                  Line2D([0], [0], color="blue", lw=4),
14                  Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
22         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
23         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8, linewidth = 3)
24         plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
25                    color = country_color[country], s=20, alpha = 0.4)
26         plt.title(issue, fontdict = {'fontsize' : 16})
27         plt.xlabel("Year", fontdict = {'fontsize' : 13})
28         plt.ylabel("% Yes", fontdict = {'fontsize' : 13})
29         plt.gca().yaxis.set_major_formatter(mtick.PercentFormatter(xmax=1.0))
30
31 plt.suptitle("Percentage of 'Yes' votes in the UN General Assembly\n1946 to 2015", weight = "bold", size = 22)
32 plt.legend(custom_lines, country_color.keys(), bbox_to_anchor=(-2, 2.4))
33 plt.show()

```

1 # UN Votes

1 ## Introduction

1 How do various countries vote in the United Nations general Assembly, how have their voting patterns evolved throughout time, and how similarly or differently do their view certain issues? Answering these questions (at a high level) is the focus of this analysis.

1 We will use **pandas**, **matplotlib**, **seaborn**, and **numpy** libraries for the data import, data wrangling, and data visualization. The data we're using come from the **unvotes** package from R.

```
In [1]: 1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 import numpy as np
5 import matplotlib.ticker as mtick
6 from matplotlib.lines import Line2D
7 plt.rcParams["figure.figsize"]=18,18
```

1 Let's create a data visualization that displays how the voting record of the UK changed over time on a variety of issues, and compares it to two other countries: US and Turkey.

```
In [13]: 1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0,4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])[["vote"]].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                  Line2D([0], [0], color="blue", lw=4),
14                  Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
```

UN Votes

Introduction

How do various countries vote in the United Nations general Assembly, how have their voting patterns evolved throughout time, and how similarly or differently do their view certain issues? Answering these questions (at a high level) is the focus of this analysis.

We will use **pandas**, **matplotlib**, **seaborn**, and **numpy** libraries for the data import, data wrangling, and data visualization. The data we're using come from the **unvotes** package from R.

```
In [1]: 1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 import numpy as np
5 import matplotlib.ticker as mtick
6 from matplotlib.lines import Line2D
7 plt.rcParams["figure.figsize"]=18,18
```

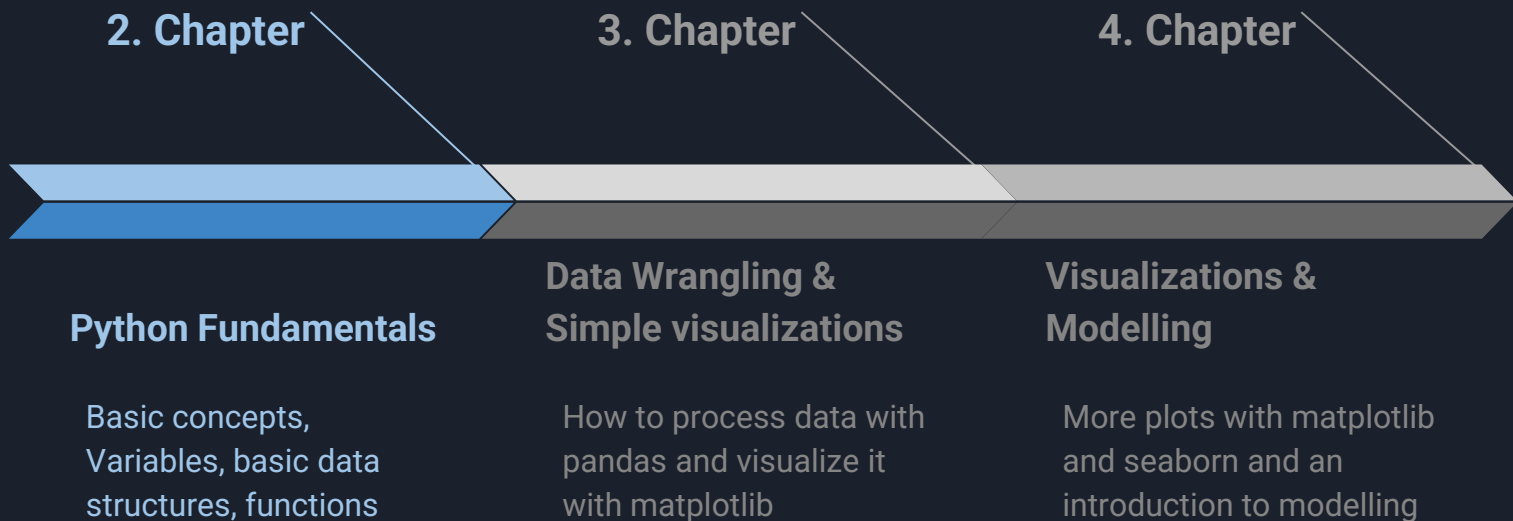
Let's create a data visualization that displays how the voting record of the UK changed over time on a variety of issues, and compares it to two other countries: US and Turkey.

```
In [13]: 1 un_votes = pd.read_csv("un_votes.csv")
2 un_roll_calls = pd.read_csv("un_roll_calls.csv")
3 un_roll_call_issues = pd.read_csv("un_roll_call_issues.csv")
4 un_votes = un_votes.merge(un_roll_calls, on = "rcid").merge(un_roll_call_issues, on = "rcid")
5 un_votes = un_votes[un_votes.country.isin(["United States", "United Kingdom", "Turkey"])]
6 un_votes["year"] = un_votes.date.str.slice(0,4)
7 un_votes["year"] = pd.to_numeric(un_votes["year"])
8 un_votes["vote"] = un_votes["vote"] == "yes"
9 un_votes_grouped = un_votes.groupby(["country", "year", "issue"])[["vote"]].mean().to_frame().reset_index()
10 un_votes_grouped = un_votes_grouped[un_votes_grouped.year < 2016]
11
12 custom_lines = [Line2D([0], [0], color="red", lw=4),
13                  Line2D([0], [0], color="blue", lw=4),
14                  Line2D([0], [0], color="green", lw=4)]
15
16 country_color = {"Turkey": "red", "United States": "blue", "United Kingdom": "green"}
17
18 for index, issue in enumerate(sorted(un_votes_grouped.issue.unique())):
19     plt.subplot(2,3,index + 1)
20     for country in un_votes_grouped.country.unique():
21         subset = (un_votes_grouped.issue == issue) & (un_votes_grouped.country == country)
22         vote_smooth = un_votes_grouped[subset].vote.rolling(12, min_periods = 0).mean()
23         plt.plot(un_votes_grouped[subset].year, vote_smooth, color = country_color[country], alpha = 0.8,
24                  plt.scatter(un_votes_grouped[subset].year, un_votes_grouped[subset].vote,
25                             color = country_color[country], s=20, alpha = 0.4)
```

Is it possible to learn all that in this course?

Yes, if you're actively coding along and invest some time.

We'll go through every step in this course:





Structure of the course

For each of 3 Chapters:

- Introduction of new concepts
- Your turn! - Small exercises (~5 minutes)
- Live coding

