Nobel_Prize_Analysis

July 18, 2023

1 Setup and Context

1.0.1 Introduction

On November 27, 1895, Alfred Nobel signed his last will in Paris. When it was opened after his death, the will caused a lot of controversy, as Nobel had left much of his wealth for the establishment of a prize.

Alfred Nobel dictates that his entire remaining estate should be used to endow "prizes to those who, during the preceding year, have conferred the greatest benefit to humankind".

Every year the Nobel Prize is given to scientists and scholars in the categories chemistry, literature, physics, physiology or medicine, economics, and peace.

Let's see what patterns we can find in the data of the past Nobel laureates. What can we learn about the Nobel prize and our world more generally?

1.0.2 Import Statements

[1]:
```
import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)

import pandas as pd
import numpy as np
import plotly.express as px
import seaborn as sns
import matplotlib.pyplot as plt
```

1.0.3 Notebook Presentation

[2]:
```
pd.options.display.float_format = '{:,.2f}'.format
```

1.0.4 Read the Data

[3]:
```
df_data = pd.read_csv('nobel_prize_data.csv')
```

Caveats: The exact birth dates for Michael Houghton, Venkatraman Ramakrishnan, and Nadia Murad are unknown. I've substituted them with mid-year estimate of July 2nd.

2 Data Exploration & Cleaning

Challenge: Preliminary data exploration. * What is the shape of `df_data`? How many rows and columns? * What are the column names? * In which year was the Nobel prize first awarded? * Which year is the latest year included in the dataset?

[4]:
```
df_data.shape
```

[4]:
```
(962, 16)
```

[5]:
```
df_data.head()
```

[5]:

	year	category	prize	motivation	prize_share
0	1901	Chemistry	The Nobel Prize in Chemistry 1901	"in recognition of the extraordinary services …	1/1
1	1901	Literature	The Nobel Prize in Literature 1901	"in special recognition of his poetic composit…	1/1
2	1901	Medicine	The Nobel Prize in Physiology or Medicine 1901	"for his work on serum therapy, especially its…	1/1
3	1901	Peace	The Nobel Peace Prize 1901	NaN	1/2
4	1901	Peace	The Nobel Peace Prize 1901	NaN	1/2

	laureate_type	full_name	birth_date	birth_city	birth_country	birth_country_current	sex	organization_name	organization_city	organization_country	ISO
0	Individual	Jacobus Henricus van 't Hoff	1852-08-30	Rotterdam	Netherlands	Netherlands	Male	Berlin University	Berlin	Germany	NLD
1	Individual	Sully Prudhomme	1839-03-16	Paris	France	France	Male	NaN	NaN	NaN	FRA
2	Individual	Emil Adolf von Behring	1854-03-15	Hansdorf (Lawice)	Prussia (Poland)	Poland	Male	Marburg University	Marburg	Germany	POL
3	Individual	Frédéric Passy	1822-05-20	Paris	France	France	Male	NaN	NaN	NaN	FRA
4	Individual	Jean Henry Dunant	1828-05-08	Geneva	Switzerland	Switzerland	Male	NaN	NaN	NaN	CHE

[6]:
```
df_data.tail()
```

[6]:

	year	category	prize	motivation	prize_share
957	2020	Medicine	The Nobel Prize in Physiology or Medicine 2020	"for the discovery of Hepatitis C virus"	1/3
958	2020	Peace	The Nobel Peace Prize 2020	"for its efforts to combat hunger, for its con…	1/1
959	2020	Physics	The Nobel Prize in Physics 2020	"for the discovery of a supermassive compact o…	1/4
960	2020	Physics	The Nobel Prize in Physics 2020	"for the discovery of a supermassive compact o…	1/4
961	2020	Physics	The Nobel Prize in Physics 2020	"for the discovery that black hole formation i…	1/2

	laureate_type	full_name	birth_date	birth_city	birth_country
957	Individual	Michael Houghton	1949-07-02	NaN	United Kingdom
958	Organization	World Food Programme (WFP)	NaN	NaN	NaN
959	Individual	Andrea Ghez	1965-06-16	New York, NY	United States of America
960	Individual	Reinhard Genzel	1952-03-24	Bad Homburg vor der Höhe	Germany
961	Individual	Roger Penrose	1931-08-08	Colchester	United Kingdom

	birth_country_current	sex	organization_name	organization_city	organization_country	ISO
957	United Kingdom	Male	University of Alberta	Edmonton	Canada	GBR
958	NaN	NaN	NaN	NaN	NaN	NaN
959	United States of America	Female	University of California	Berkeley, CA	United States of America	USA
960	Germany	Male	University of California	Los Angeles, CA	United States of America	DEU
961	United Kingdom	Male	University of Oxford	Oxford	United Kingdom	GBR

Challange: * Are there any duplicate values in the dataset? * Are there NaN values in the dataset? * Which columns tend to have NaN values? * How many NaN values are there per column? * Why do these columns have NaN values?

2.0.1 Check for Duplicates

[7]:
```
print(f'Any duplicates? {df_data.duplicated().values.any()}')
```

```
Any duplicates? False
```

2.0.2 Check for NaN Values

[8]:
```
print(f'Any NaN values among the data? {df_data.isna().values.any()}')
```

```
Any NaN values among the data? True
```

[9]:
```
df_data.isna().sum()
```

[9]:
```
year                     0
category                 0
prize                    0
motivation              88
prize_share              0
laureate_type            0
full_name                0
birth_date              28
birth_city              31
birth_country           28
birth_country_current   28
sex                     28
organization_name      255
organization_city      255
organization_country   254
ISO                     28
dtype: int64
```

Why are there NaN values for birth dates?

[10]:
```
# NaN values for birth date are all organisations
col_subset = ['year','category', 'laureate_type',
              'birth_date','full_name', 'organization_name']
df_data.loc[df_data.birth_date.isna()][col_subset]
```

[10]:
```
     year category laureate_type birth_date  \
24   1904    Peace  Organization        NaN
60   1910    Peace  Organization        NaN
89   1917    Peace  Organization        NaN
200  1938    Peace  Organization        NaN
215  1944    Peace  Organization        NaN
237  1947    Peace  Organization        NaN
238  1947    Peace  Organization        NaN
283  1954    Peace  Organization        NaN
348  1963    Peace  Organization        NaN
```

```
     year category        type  prize_share
349  1963    Peace Organization          NaN
366  1965    Peace Organization          NaN
399  1969    Peace Organization          NaN
479  1977    Peace Organization          NaN
523  1981    Peace Organization          NaN
558  1985    Peace Organization          NaN
588  1988    Peace Organization          NaN
659  1995    Peace Organization          NaN
682  1997    Peace Organization          NaN
703  1999    Peace Organization          NaN
730  2001    Peace Organization          NaN
778  2005    Peace Organization          NaN
788  2006    Peace Organization          NaN
801  2007    Peace Organization          NaN
860  2012    Peace Organization          NaN
873  2013    Peace Organization          NaN
897  2015    Peace Organization          NaN
919  2017    Peace Organization          NaN
958  2020    Peace Organization          NaN

                                          full_name organization_name
24   Institut de droit international (Institute of …              NaN
60   Bureau international permanent de la Paix (Per…              NaN
89   Comité international de la Croix Rouge (Intern…              NaN
200  Office international Nansen pour les Réfugiés …              NaN
215  Comité international de la Croix Rouge (Intern…              NaN
237   American Friends Service Committee (The Quakers)               NaN
238              Friends Service Council (The Quakers)               NaN
283  Office of the United Nations High Commissioner…              NaN
348  Comité international de la Croix Rouge (Intern…              NaN
349  Ligue des Sociétés de la Croix-Rouge (League o…              NaN
366            United Nations Children's Fund (UNICEF)               NaN
399         International Labour Organization (I.L.O.)               NaN
479                              Amnesty International               NaN
523  Office of the United Nations High Commissioner…              NaN
558  International Physicians for the Prevention of…              NaN
588                 United Nations Peacekeeping Forces               NaN
659   Pugwash Conferences on Science and World Affairs               NaN
682     International Campaign to Ban Landmines (ICBL)               NaN
703                            Médecins Sans Frontières               NaN
730                               United Nations (U.N.)               NaN
778          International Atomic Energy Agency (IAEA)               NaN
788                                       Grameen Bank               NaN
801   Intergovernmental Panel on Climate Change (IPCC)               NaN
860                                European Union (EU)               NaN
873  Organisation for the Prohibition of Chemical W…              NaN
897                            National Dialogue Quartet               NaN
```

```
919  International Campaign to Abolish Nuclear Weap…              NaN
958                          World Food Programme (WFP)              NaN
```

That makes sense. We also see that since the organisation's name is in the full_name column, the organisation_name column contains NaN.

In addition, when we look at for rows where the organization_name column has no value, we also see that many prizes went to people who were not affiliated with a university or research institute. This includes many of the Literature and Peace prize winners.

[11]:
```
# NaN values for organisation_name
col_subset = ['year','category', 'laureate_type','full_name',␣
 ↪'organization_name']
df_data.loc[df_data.organization_name.isna()][col_subset]
```

[11]:
```
      year    category laureate_type                          full_name  \
1     1901  Literature    Individual                      Sully Prudhomme
3     1901       Peace    Individual                       Frédéric Passy
4     1901       Peace    Individual                     Jean Henry Dunant
7     1902  Literature    Individual  Christian Matthias Theodor Mommsen
9     1902       Peace    Individual                 Charles Albert Gobat
..     …          …            …                                  …
932   2018       Peace    Individual                         Nadia Murad
942   2019  Literature    Individual                        Peter Handke
946   2019       Peace    Individual                      Abiy Ahmed Ali
954   2020  Literature    Individual                        Louise Glück
958   2020       Peace  Organization          World Food Programme (WFP)

     organization_name
1                  NaN
3                  NaN
4                  NaN
7                  NaN
9                  NaN
..                 …
932                NaN
942                NaN
946                NaN
954                NaN
958                NaN

[255 rows x 5 columns]
```

Some prizes are given to Organisations rather than individuals!

2.0.3 Type Conversions

Challenge: * Convert the `birth_date` column to Pandas `Datetime` objects * Add a Column called `share_pct` which has the laureates' share as a percentage in the form of a floating-point number.

Convert Year and Birth Date to Datetime

[12]:
```
df_data.birth_date = pd.to_datetime(df_data.birth_date)
```

Add a Column with the Prize Share as a Percentage

[13]:
```
separated_values = df_data.prize_share.str.split('/', expand=True)
numerator = pd.to_numeric(separated_values[0])
denomenator = pd.to_numeric(separated_values[1])
df_data['share_pct'] = numerator / denomenator
```

[14]:
```
df_data.info()
```

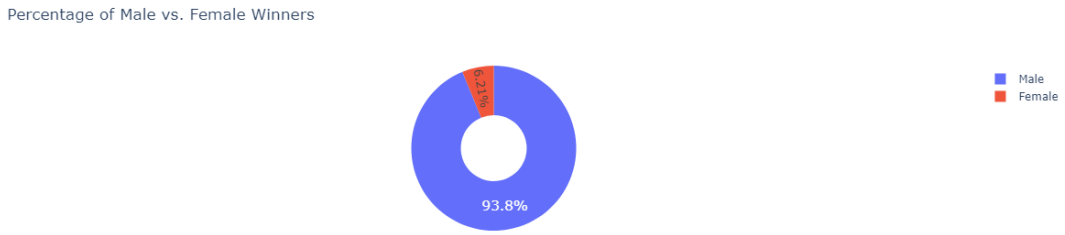
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 962 entries, 0 to 961
Data columns (total 17 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   year                   962 non-null    int64
 1   category               962 non-null    object
 2   prize                  962 non-null    object
 3   motivation             874 non-null    object
 4   prize_share            962 non-null    object
 5   laureate_type          962 non-null    object
 6   full_name              962 non-null    object
 7   birth_date             934 non-null    datetime64[ns]
 8   birth_city             931 non-null    object
 9   birth_country          934 non-null    object
 10  birth_country_current  934 non-null    object
 11  sex                    934 non-null    object
 12  organization_name      707 non-null    object
 13  organization_city      707 non-null    object
 14  organization_country   708 non-null    object
 15  ISO                    934 non-null    object
 16  share_pct              962 non-null    float64
dtypes: datetime64[ns](1), float64(1), int64(1), object(14)
memory usage: 127.9+ KB
```

3 Plotly Donut Chart: Percentage of Male vs. Female Laureates

Challenge: Create a donut chart using plotly which shows how many prizes went to men compared to how many prizes went to women. What percentage of all the prizes went to women?

[15]:
```
biology = df_data.sex.value_counts()
fig = px.pie(labels=biology.index,
             values=biology.values,
             title="Percentage of Male vs. Female Winners",
             names=biology.index,
             hole=0.4,)
```

```
fig.update_traces(textposition='inside', textfont_size=15, textinfo='percent')

fig.show()
```



4 Who were the first 3 Women to Win the Nobel Prize?

Challenge: * What are the names of the first 3 female Nobel laureates? * What did the win the prize for? * What do you see in their `birth_country`? Were they part of an organisation?

[16]:
```
df_data[df_data.sex == 'Female'].sort_values('year', ascending=True)[:3]
```

[16]:
```
    year    category                                prize  \
18  1903     Physics     The Nobel Prize in Physics 1903
29  1905       Peace          The Nobel Peace Prize 1905
51  1909  Literature  The Nobel Prize in Literature 1909

                                        motivation prize_share  \
18  "in recognition of the extraordinary services …         1/4
29                                                NaN         1/1
51  "in appreciation of the lofty idealism, vivid …         1/1

   laureate_type                                          full_name  \
18    Individual                       Marie Curie, née Sklodowska
29    Individual  Baroness Bertha Sophie Felicita von Suttner, n…
51    Individual                      Selma Ottilia Lovisa Lagerlöf

   birth_date birth_city                    birth_country  \
18 1867-11-07     Warsaw          Russian Empire (Poland)
29 1843-06-09     Prague  Austrian Empire (Czech Republic)
51 1858-11-20   Mårbacka                           Sweden

   birth_country_current     sex organization_name organization_city  \
18                Poland  Female               NaN               NaN
```

```
29        Czech Republic  Female               NaN               NaN
51                Sweden  Female               NaN               NaN

   organization_country  ISO  share_pct
18                   NaN  POL       0.25
29                   NaN  CZE       1.00
51                   NaN  SWE       1.00
```

5 Find the Repeat Winners

Challenge: Did some people get a Nobel Prize more than once? If so, who were they?

[17]:
```
is_winner = df_data.duplicated(subset=['full_name'], keep=False)
multiple_winners = df_data[is_winner]
print(f'There are {multiple_winners.full_name.nunique()}' \
      ' winners who weere awarded the prize more than once.')
```

```
There are 6 winners who weere awarded the prize more than once.
```

[18]:
```
col_subset = ['year', 'category', 'laureate_type', 'full_name']
multiple_winners[col_subset]
```

[18]:
```
     year   category laureate_type  \
18   1903    Physics    Individual
62   1911  Chemistry    Individual
89   1917      Peace  Organization
215  1944      Peace  Organization
278  1954  Chemistry    Individual
283  1954      Peace  Organization
297  1956    Physics    Individual
306  1958  Chemistry    Individual
340  1962      Peace    Individual
348  1963      Peace  Organization
424  1972    Physics    Individual
505  1980  Chemistry    Individual
523  1981      Peace  Organization

                                          full_name
18                        Marie Curie, née Sklodowska
62                        Marie Curie, née Sklodowska
89   Comité international de la Croix Rouge (Intern…
215  Comité international de la Croix Rouge (Intern…
278                                Linus Carl Pauling
283  Office of the United Nations High Commissioner…
297                                     John Bardeen
306                                Frederick Sanger
340                                Linus Carl Pauling
```

```
348  Comité international de la Croix Rouge (Intern…
424                                        John Bardeen
505                                     Frederick Sanger
523  Office of the United Nations High Commissioner…
```

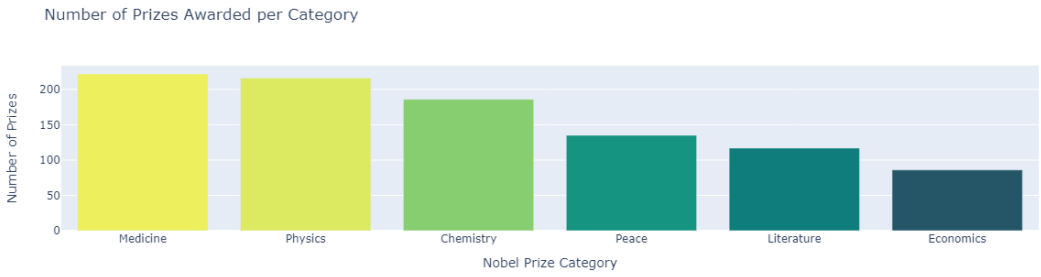
6 Number of Prizes per Category

Challenge: * In how many categories are prizes awarded? * Create a plotly bar chart with the number of prizes awarded by category. * Which category has the most number of prizes awarded? * Which category has the fewest number of prizes awarded?

[19]:
```
# Number of different categories
df_data.category.nunique()
```

[19]: 6

[20]:
```
prizes_per_category = df_data.category.value_counts()
v_bar = px.bar(
        x = prizes_per_category.index,
        y = prizes_per_category.values,
        color = prizes_per_category.values,
        color_continuous_scale='Aggrnyl',
        title='Number of Prizes Awarded per Category')

v_bar.update_layout(xaxis_title='Nobel Prize Category',
                    coloraxis_showscale=False,
                    yaxis_title='Number of Prizes')
v_bar.show()
```



Challenge: * When was the first prize in the field of Economics awarded? * Who did the prize go to?

[21]:
```
df_data[df_data.category == 'Economics'].sort_values('year')[:3]
```

[21]:
```
     year   category                                                prize  \
393  1969  Economics  The Sveriges Riksbank Prize in Economic Scienc…
394  1969  Economics  The Sveriges Riksbank Prize in Economic Scienc…
402  1970  Economics  The Sveriges Riksbank Prize in Economic Scienc…

                                          motivation prize_share  \
393  "for having developed and applied dynamic mode…         1/2
394  "for having developed and applied dynamic mode…         1/2
402  "for the scientific work through which he has …         1/1

    laureate_type          full_name birth_date birth_city  \
393    Individual      Jan Tinbergen 1903-04-12  the Hague
394    Individual      Ragnar Frisch 1895-03-03       Oslo
402    Individual  Paul A. Samuelson 1915-05-15   Gary, IN

               birth_country     birth_country_current   sex  \
393              Netherlands               Netherlands  Male
394                   Norway                    Norway  Male
402  United States of America  United States of America  Male

                           organization_name organization_city  \
393         The Netherlands School of Economics         Rotterdam
394                          University of Oslo              Oslo
402  Massachusetts Institute of Technology (MIT)     Cambridge, MA

         organization_country  ISO  share_pct
393               Netherlands  NLD       0.50
394                    Norway  NOR       0.50
402  United States of America  USA       1.00
```

7 Male and Female Winners by Category

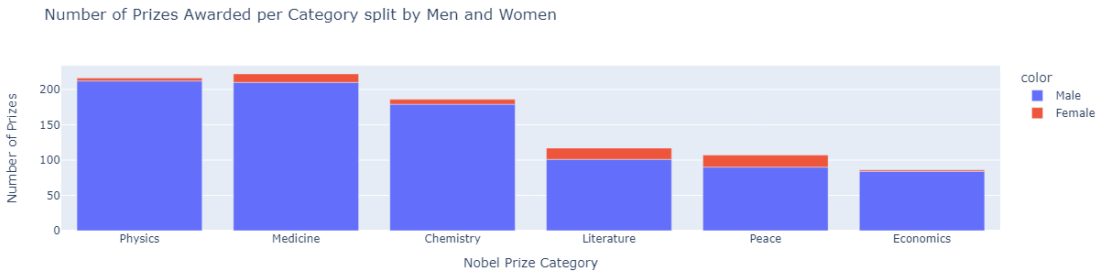
Challenge: Create a plotly bar chart that shows the split between men and women by category.

[22]:
```
cat_men_women = df_data.groupby(['category', 'sex'],
                                as_index=False).agg({'prize': pd.Series.count})
cat_men_women.sort_values('prize', ascending=False, inplace=True)
cat_men_women
```

[22]:
```
      category     sex  prize
11     Physics    Male    212
7     Medicine    Male    210
1    Chemistry    Male    179
5   Literature    Male    101
9        Peace    Male     90
3    Economics    Male     84
8        Peace  Female     17
```

```
4   Literature  Female     16
6     Medicine  Female     12
0    Chemistry  Female      7
10     Physics  Female      4
2    Economics  Female      2
```

[23]:
```
v_bar_split = px.bar(x = cat_men_women.category,
                     y = cat_men_women.prize,
                     color = cat_men_women.sex,
                     title='Number of Prizes Awarded per Category split by Men␣
 ↪and Women')

v_bar_split.update_layout(xaxis_title='Nobel Prize Category',
                          yaxis_title='Number of Prizes')
v_bar_split.show()
```



We see that overall the imbalance is pretty large with physics, economics, and chemistry. Women are somewhat more represented in categories of Medicine, Literature and Peace.

8 Number of Prizes Awarded Over Time

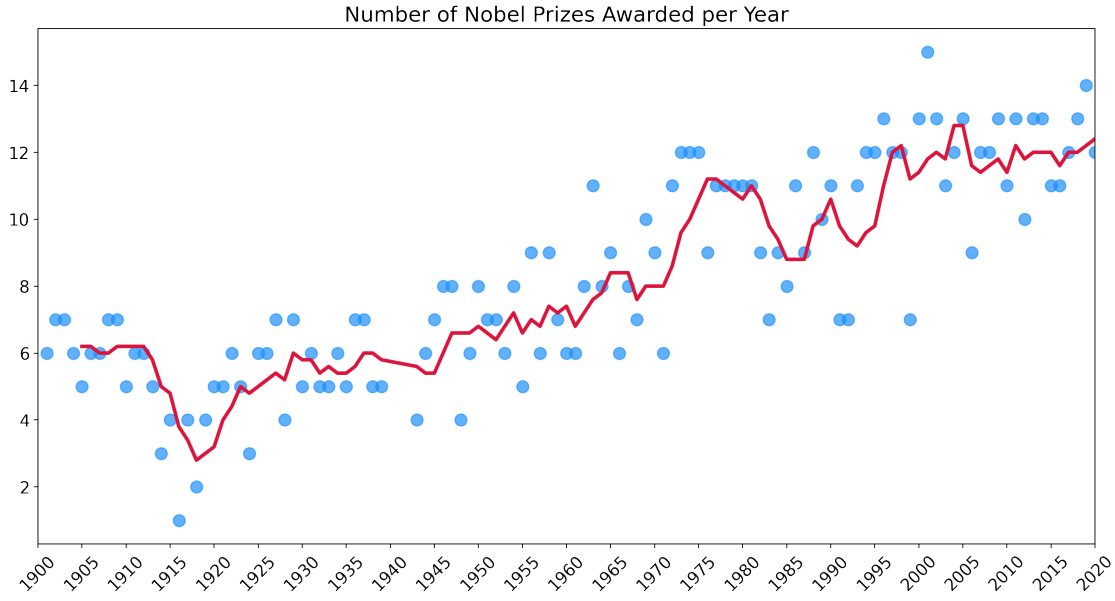
Challenge: Are more prizes awarded recently than when the prize was first created? * Count the number of prizes awarded every year. * Create a 5 year rolling average of the number of prizes. * Show a tick mark on the x-axis for every 5 years from 1900 to 2020. * Looking at the chart, did the first and second world wars have an impact on the number of prizes being given out? * What could be the reason for the trend in the chart?

[24]:
```
prize_per_year = df_data.groupby(by='year').count().prize
```

[25]:
```
moving_average = prize_per_year.rolling(window=5).mean()
```

[26]:
```
plt.figure(figsize=(16,8), dpi=200)
plt.title('Number of Nobel Prizes Awarded per Year', fontsize=18)
plt.yticks(fontsize=14)
plt.xticks(ticks=np.arange(1900, 2021, step=5),
```

```
            fontsize=14,
            rotation=45)

ax = plt.gca()
ax.set_xlim(1900, 2020)

ax.scatter(x=prize_per_year.index,
            y=prize_per_year.values,
            c='dodgerblue',
            alpha=0.7,
            s=100,)

ax.plot(prize_per_year.index,
        moving_average.values,
        c='crimson',
        linewidth=3,)

plt.show()
```



9 Are More Prizes Shared Than Before?

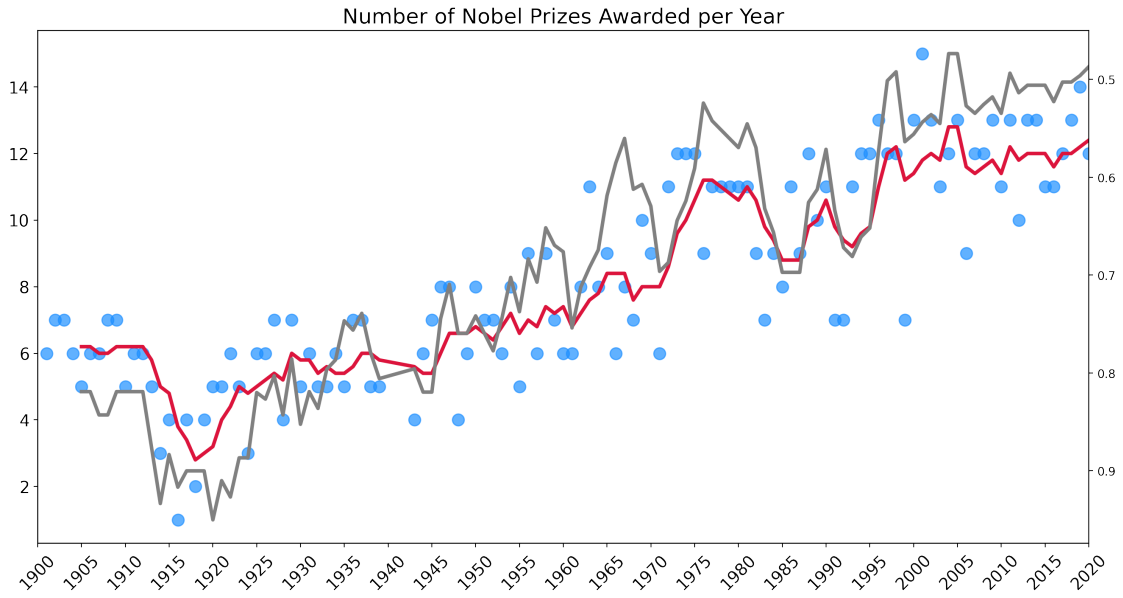
Challenge: Investigate if more prizes are shared than before.

- Calculate the average prize share of the winners on a year by year basis.
- Calculate the 5 year rolling average of the percentage share.
- Copy-paste the cell from the chart created above.
- Modify the code to add a secondary axis to our Matplotlib chart.

- Plot the rolling average of the prize share on this chart.

[27]:
```
yearly_avg_share = df_data.groupby(by='year').agg({'share_pct': pd.Series.mean})
share_moving_average = yearly_avg_share.rolling(window=5).mean()
```

[28]:
```
plt.figure(figsize=(16,8), dpi=200)
plt.title('Number of Nobel Prizes Awarded per Year', fontsize=18)
plt.yticks(fontsize=14)
plt.xticks(ticks=np.arange(1900, 2021, step=5),
           fontsize=14,
           rotation=45)

ax1 = plt.gca()
ax2 = ax1.twinx()
ax1.set_xlim(1900, 2020)

# Can invert axis
ax2.invert_yaxis()

ax1.scatter(x=prize_per_year.index,
           y=prize_per_year.values,
           c='dodgerblue',
           alpha=0.7,
           s=100,)

ax1.plot(prize_per_year.index,
        moving_average.values,
        c='crimson',
        linewidth=3,)

ax2.plot(prize_per_year.index,
        share_moving_average.values,
        c='grey',
        linewidth=3,)

plt.show()
```



What does the graph show? There's a clear upward trend in the number of prizes awarded, as more and more prizes are shared. In addition, more prizes were awarded from 1969 onwards, due to the addition of the "Economy" category. We also note that very few prizes were awarded during the First and Second World Wars.

10 The Countries with the Most Nobel Prizes

Challenge: * Create a Pandas DataFrame called `top20_countries` that has the two columns. The `prize` column should contain the total number of prizes won. * Is it best to use `birth_country`, `birth_country_current` or `organization_country`? * What are some potential problems when using `birth_country` or any of the others? Which column is the least problematic?

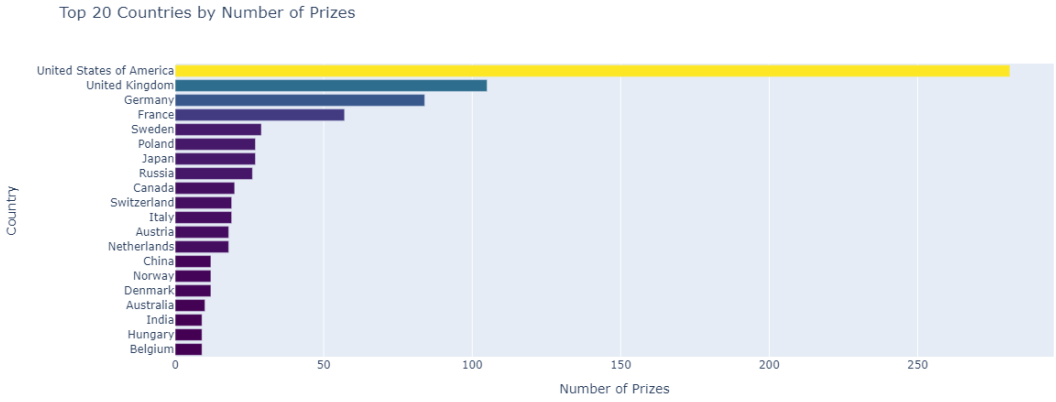
[29]:
```
top_countries = df_data.groupby(['birth_country_current'],
                                as_index=False).agg({'prize': pd.Series.
 ↪count})

top_countries.sort_values(by='prize', inplace=True)
top20_countries = top_countries[-20:]
top20_countries
```

[29]:

	birth_country_current	prize
7	Belgium	9
31	Hungary	9
33	India	9
2	Australia	10
20	Denmark	12
54	Norway	12

13	China	12
51	Netherlands	18
3	Austria	18
39	Italy	19
68	Switzerland	19
11	Canada	20
61	Russia	26
40	Japan	27
57	Poland	27
67	Sweden	29
25	France	57
26	Germany	84
73	United Kingdom	105
74	United States of America	281

[30]:
```
h_bar = px.bar(x=top20_countries.prize,
               y=top20_countries.birth_country_current,
               orientation='h',
               color=top20_countries.prize,
               color_continuous_scale='Viridis',
               title='Top 20 Countries by Number of Prizes',
               height=500)

h_bar.update_layout(xaxis_title='Number of Prizes',
                    yaxis_title='Country',
                    coloraxis_showscale=False)
h_bar.show()
```



The United States has a massive number of prizes by this measure. The UK and Germany are in second and third place respectively.

11 Use a Choropleth Map to Show the Number of Prizes Won by Country

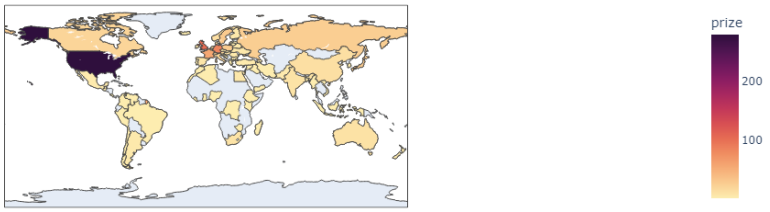
[31]:
```
df_countries = df_data.groupby(['birth_country_current', 'ISO'],
                               as_index=False).agg({'prize': pd.Series.count})
df_countries.sort_values('prize', ascending=False)
```

[31]:

	birth_country_current	ISO	prize
74	United States of America	USA	281
73	United Kingdom	GBR	105
26	Germany	DEU	84
25	France	FRA	57
67	Sweden	SWE	29
..	…	…	…
32	Iceland	ISL	1
47	Madagascar	MDG	1
34	Indonesia	IDN	1
36	Iraq	IRQ	1
78	Zimbabwe	ZWE	1

[79 rows x 3 columns]

[32]:
```
world_map = px.choropleth(df_countries,
                          locations='ISO',
                          color='prize',
                          hover_name='birth_country_current',
                          color_continuous_scale=px.colors.sequential.matter,)

world_map.update_layout(coloraxis_showscale=True,)

world_map.show()
```



12 In Which Categories are the Different Countries Winning Prizes?

Challenge: Trying to divide the bar chart created above to show the categories that represent the total number of awards:Here are the questions I need to answer: * In which category are Germany and Japan the weakest compared to the United States? * In which category does Germany have more prizes than the UK? * In which categories does France have more prizes than Germany? * Which category makes up most of Australia's nobel prizes? * Which category makes up half of the prizes in the Netherlands? * Does the United States have more prizes in Economics than all of France? What about in Physics or Medicine?

[33]:
```
cat_country = df_data.groupby(['birth_country_current', 'category'],
                              as_index=False).agg({'prize': pd.Series.count})
cat_country.sort_values(by='prize', ascending=False, inplace=True)
cat_country
```

[33]:

	birth_country_current	category	prize
204	United States of America	Medicine	78
206	United States of America	Physics	70
201	United States of America	Chemistry	55
202	United States of America	Economics	49
198	United Kingdom	Medicine	28
..	…	…	…
97	Iraq	Peace	1
99	Ireland	Medicine	1
100	Ireland	Physics	1
102	Israel	Economics	1
210	Zimbabwe	Peace	1

[211 rows x 3 columns]

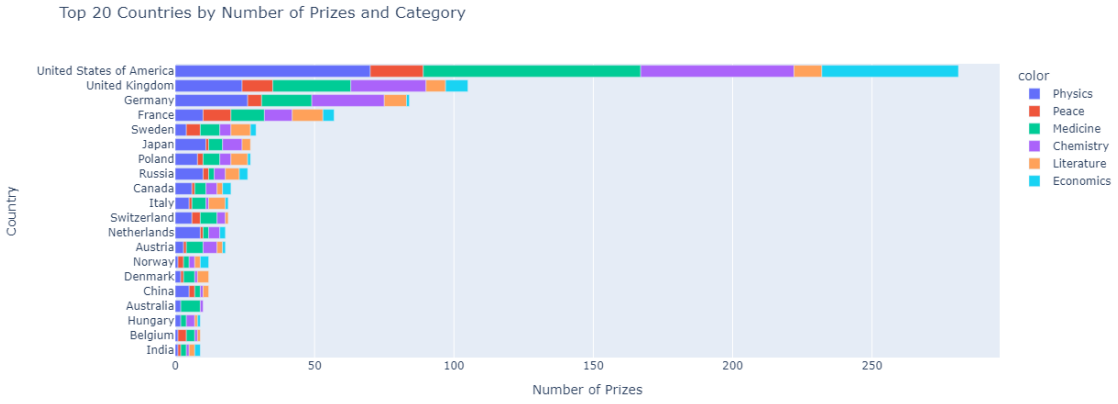
[34]:
```
merged_df = pd.merge(cat_country, top20_countries, on='birth_country_current')
# change column names
merged_df.columns = ['birth_country_current', 'category', 'cat_prize',␣
 ↪'total_prize']
merged_df.sort_values(by='total_prize', inplace=True)
merged_df
```

[34]:

	birth_country_current	category	cat_prize	total_prize
109	India	Physics	1	9
108	India	Peace	1	9
88	Belgium	Peace	3	9
89	Belgium	Medicine	3	9
90	Belgium	Chemistry	1	9
..	…	…	…	…
4	United States of America	Peace	19	281
3	United States of America	Economics	49	281

```
2    United States of America  Chemistry         55          281
1    United States of America    Physics         70          281
0    United States of America   Medicine         78          281

[110 rows x 4 columns]
```

[35]:
```
cat_cntry_bar = px.bar(x=merged_df.cat_prize,
                       y=merged_df.birth_country_current,
                       color=merged_df.category,
                       orientation='h',
                       title='Top 20 Countries by Number of Prizes and␣
↪Category',
                       height=500)

cat_cntry_bar.update_layout(xaxis_title='Number of Prizes',
                            yaxis_title='Country')
cat_cntry_bar.show()
```



Splitting the country bar chart by category allows us to get a very granular look at the data and answer a whole bunch of questions. For example, we see is that the US has won an incredible proportion of the prizes in the field of Economics. In comparison, Japan and Germany have won very few or no economics prize at all. Also, the US has more prizes in physics or medicine alone than all of France's prizes combined. On the chart, we also see that Germany won more prizes in physics than the UK and that France has won more prizes in peace and literature than Germany, even though Germany has been awarded a higher total number of prizes than France.

12.0.1 Number of Prizes Won by Each Country Over Time

- When did the United States eclipse every other country in terms of the number of prizes won?
- Which country or countries were leading previously?
- Calculate the cumulative number of prizes won by each country in every year.
- Create a plotly line chart where each country is a coloured line.

[36]:
```
prize_by_year = df_data.groupby(by=['birth_country_current', 'year'],␣
 ↪as_index=False).count()
prize_by_year = prize_by_year.sort_values('year')[['year',␣
 ↪'birth_country_current', 'prize']]
prize_by_year
```

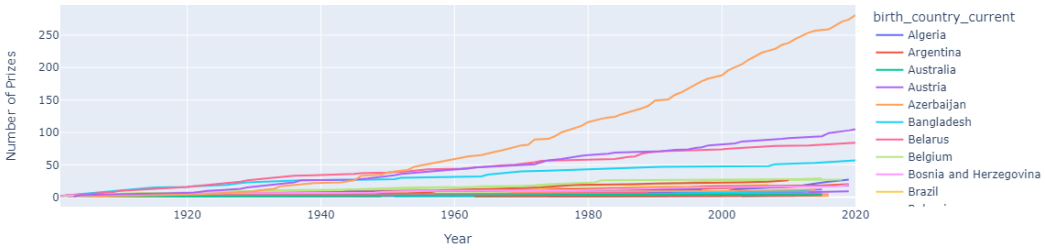
[36]:

	year	birth_country_current	prize
118	1901	France	2
346	1901	Poland	1
159	1901	Germany	1
312	1901	Netherlands	1
440	1901	Switzerland	1
..	…	…	…
31	2019	Austria	1
221	2020	Germany	1
622	2020	United States of America	7
533	2020	United Kingdom	2
158	2020	France	1

[627 rows x 3 columns]

[37]:
```
cumulative_prizes = prize_by_year.groupby(by=['birth_country_current',
                                              'year']).sum().groupby(level=[0]).
 ↪cumsum()
cumulative_prizes.reset_index(inplace=True)
```

[38]:
```
l_chart = px.line(cumulative_prizes,
                  x='year',
                  y='prize',
                  color='birth_country_current',
                  hover_name='birth_country_current')

l_chart.update_layout(xaxis_title='Year',
                      yaxis_title='Number of Prizes')

l_chart.show()
```



What we see is that the United States really started to take off after the Second World War which decimated Europe. Prior to that, the Nobel prize was pretty much a European affair. Very few laureates were chosen from other parts of the world. This has changed dramatically in the last 40 years or so. There are many more countries represented today than in the early days. Interestingly we also see that the UK and Germany traded places in the 70s and 90s on the total number of prizes won. Sweden being 5th place pretty consistently over many decades is quite interesting too. Perhaps this reflects a little bit of home bias?

All this analysis of different countries makes me curious about where the actual research is happening. Where are the cities and organisations located where people actually make discoveries?

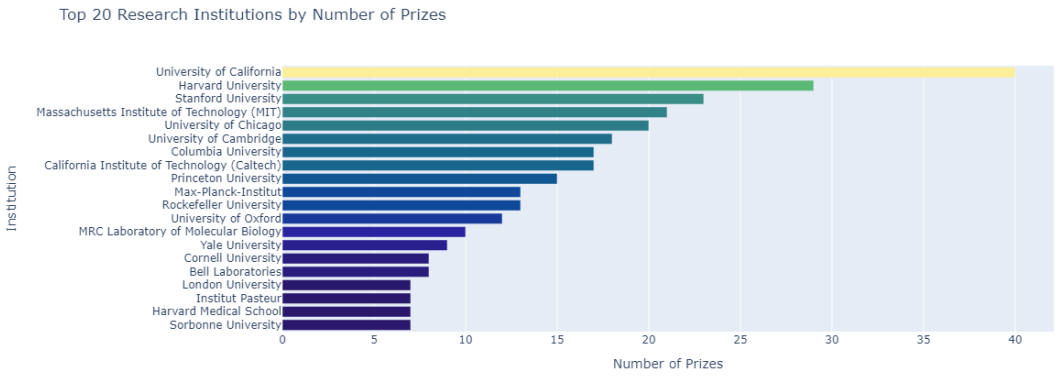
13 What are the Top Research Organisations?

Challenge: Create a bar chart showing the organisations affiliated with the Nobel laureates.

- Which organisations make up the top 20?
- How many Nobel prize winners are affiliated with the University of Chicago and Harvard University?

[39]:
```
top20_orgs = df_data.organization_name.value_counts()[:20]
top20_orgs.sort_values(ascending=True, inplace=True)
```

[40]:
```
org_bar = px.bar(x = top20_orgs.values,
                 y = top20_orgs.index,
                 orientation='h',
                 color=top20_orgs.values,
                 color_continuous_scale=px.colors.sequential.haline,
                 title='Top 20 Research Institutions by Number of Prizes',
                 height=470)

org_bar.update_layout(xaxis_title='Number of Prizes',
                      yaxis_title='Institution',
                      coloraxis_showscale=False)
org_bar.show()
```

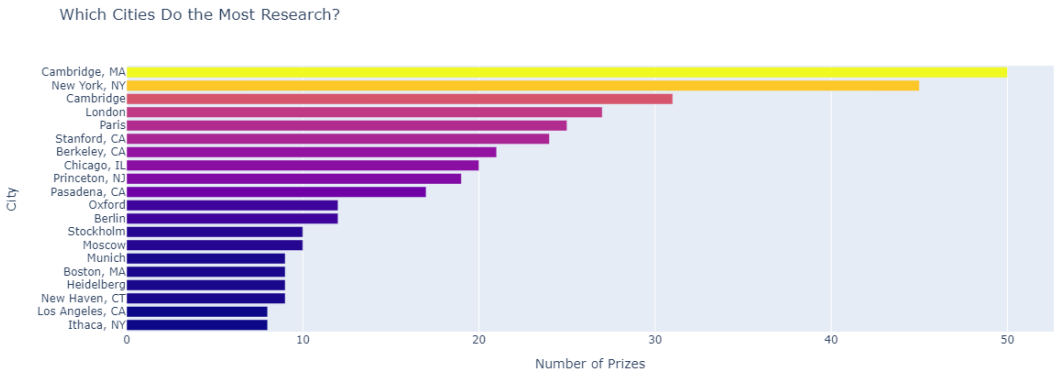


14 Which Cities Make the Most Discoveries?

Where do major discoveries take place?

Challenge: * Create another plotly bar chart graphing the top 20 organisation cities of the research institutions associated with a Nobel laureate. * Where is the number one hotspot for discoveries in the world? * Which city in Europe has had the most discoveries?

[41]:
```
top20_org_cities = df_data.organization_city.value_counts()[:20]
top20_org_cities.sort_values(ascending=True, inplace=True)
city_bar2 = px.bar(x = top20_org_cities.values,
                   y = top20_org_cities.index,
                   orientation='h',
                   color=top20_org_cities.values,
                   color_continuous_scale=px.colors.sequential.Plasma,
                   title='Which Cities Do the Most Research?',
                   height=470)

city_bar2.update_layout(xaxis_title='Number of Prizes',
                        yaxis_title='City',
                        coloraxis_showscale=False)
city_bar2.show()
```

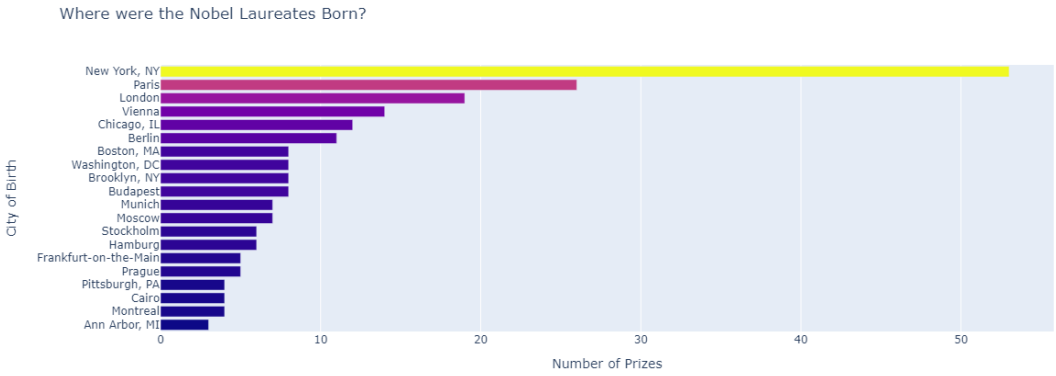


15 Where are Nobel Laureates Born? Chart the Laureate Birth Cities

Challenge: * Create a plotly bar chart graphing the top 20 birth cities of Nobel laureates. * What percentage of the United States prizes came from Nobel laureates born in New York? * How many

Nobel laureates were born in London, Paris and Vienna? * Out of the top 5 cities, how many are in the United States?

[42]:
```
top20_cities = df_data.birth_city.value_counts()[:20]
top20_cities.sort_values(ascending=True, inplace=True)
city_bar = px.bar(x=top20_cities.values,
                  y=top20_cities.index,
                  orientation='h',
                  color=top20_cities.values,
                  color_continuous_scale=px.colors.sequential.Plasma,
                  title='Where were the Nobel Laureates Born?',
                  height=470)

city_bar.update_layout(xaxis_title='Number of Prizes',
                       yaxis_title='City of Birth',
                       coloraxis_showscale=False)
city_bar.show()
```



16 Plotly Sunburst Chart: Combine Country, City, and Organisation

Challenge:

- Create a DataFrame that groups the number of prizes by organisation.
- what do you notice about Germany and France?

[43]:
```
country_city_org = df_data.groupby(by=['organization_country',
                                       'organization_city',
                                       'organization_name'], as_index=False).
↪agg({'prize': pd.Series.count})

country_city_org = country_city_org.sort_values('prize', ascending=False)
```

```
country_city_org
```

[43]:
```
         organization_country     organization_city  \
205  United States of America         Cambridge, MA
280  United States of America          Stanford, CA
206  United States of America         Cambridge, MA
209  United States of America           Chicago, IL
195  United States of America          Berkeley, CA
..                        …                     …
110                     Japan               Sapporo
111                     Japan                 Tokyo
112                     Japan                 Tokyo
113                     Japan                 Tokyo
290  United States of America  Yorktown Heights, NY

                              organization_name  prize
205                           Harvard University     29
280                          Stanford University     23
206  Massachusetts Institute of Technology (MIT)     21
209                        University of Chicago     20
195                      University of California     19
..                                            …    …
110                          Hokkaido University      1
111                       Asahi Kasei Corporation     1
112                          Kitasato University      1
113                 Tokyo Institute of Technology     1
290           IBM Thomas J. Watson Research Center     1

[291 rows x 4 columns]
```

[44]:
```
burst = px.sunburst(country_city_org,
                    path=['organization_country', 'organization_city',␣
 ↪'organization_name'],
                    values='prize',
                    title='Where do Discoveries Take Place?',
                    height=800)

burst.update_layout(xaxis_title='Number of Prizes',
                    yaxis_title='City',
                    coloraxis_showscale=False)

burst.show()
```

France is an excellent example of concentration. Virtually all the organizations associated with Nobel Prize winners are based in Paris. By contrast, scientific discoveries are much more dispersed in Germany. The UK, meanwhile, is dominated by Cambridge and London.

17 Patterns in the Laureate Age at the Time of the Award

How Old Are the Laureates When the Win the Prize?

Challenge: Calculate the age of the laureate in the year of the ceremony and add this as a column called `winning_age` to the `df_data` DataFrame.

[45]:
```
# Use Datetime object
birth_years = df_data.birth_date.dt.year
birth_years
```

[45]:
```
0      1,852.00
1      1,839.00
2      1,854.00
3      1,822.00
4      1,828.00
         …
957    1,949.00
958         NaN
959    1,965.00
960    1,952.00
961    1,931.00
Name: birth_date, Length: 962, dtype: float64
```

[46]:
```
df_data['winning_age'] = df_data.year - birth_years
df_data.winning_age
```

[46]:
```
0      49.00
1      62.00
2      47.00
3      79.00
4      73.00
        …
957    71.00
958      NaN
959    55.00
960    68.00
961    89.00
Name: winning_age, Length: 962, dtype: float64
```

17.0.1 Who were the oldest and youngest winners?

Challenge: * What are the names of the youngest and oldest Nobel laureate? * What did they win the prize for? * What is the average age of a winner? * 75% of laureates are younger than what age when they receive the prize? * Use Seaborn to create histogram to visualise the distribution of laureate age at the time of winning.

[47]:
```
display(df_data.nlargest(n=1, columns='winning_age'))
display(df_data.nsmallest(n=1, columns='winning_age'))
```

```
     year   category                                prize  \
937  2019  Chemistry  The Nobel Prize in Chemistry 2019

                                        motivation prize_share laureate_type  \
937  "for the development of lithium-ion batteries"         1/3    Individual

           full_name birth_date birth_city birth_country  \
937  John Goodenough 1922-07-25       Jena       Germany

    birth_country_current   sex    organization_name organization_city  \
937               Germany  Male  University of Texas         Austin TX

         organization_country  ISO  share_pct  winning_age
937  United States of America  DEU       0.33        97.00
```

```
     year category                        prize  \
885  2014    Peace  The Nobel Peace Prize 2014

                                          motivation prize_share  \
885  "for their struggle against the suppression of…         1/2

    laureate_type         full_name birth_date birth_city birth_country  \
885    Individual  Malala Yousafzai 1997-07-12    Mingora      Pakistan

    birth_country_current     sex organization_name organization_city  \
885              Pakistan  Female               NaN                NaN

    organization_country  ISO  share_pct  winning_age
885                  NaN  PAK       0.50        17.00
```

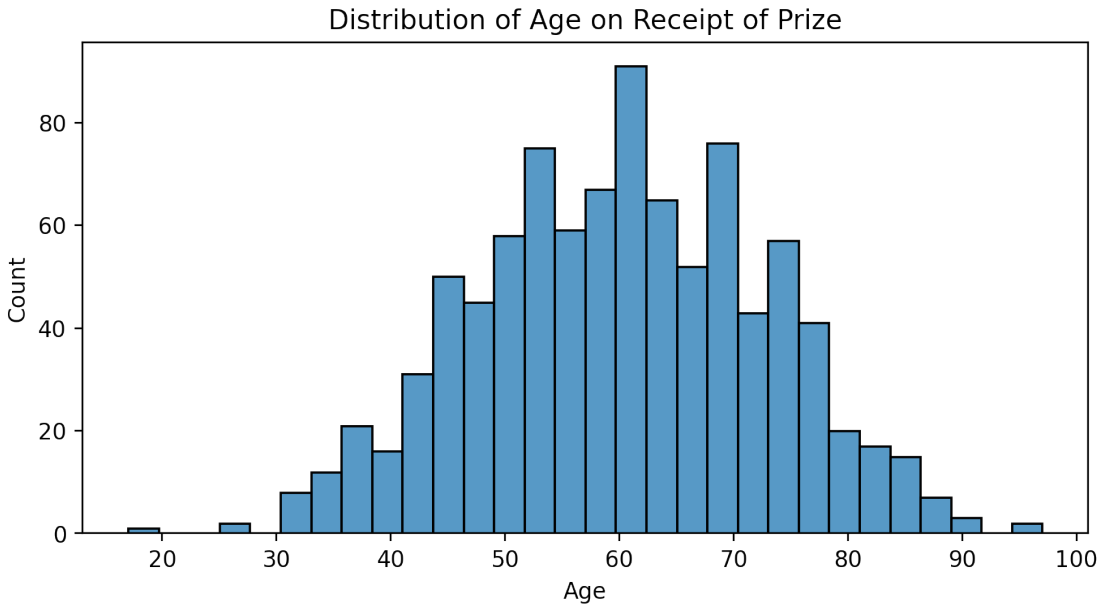
17.0.2 Descriptive Statistics for the Laureate Age at Time of Award

- Calculate the descriptive statistics for the age at the time of the award.
- Then visualise the distribution in the form of a histogram using Seaborn's .histplot() function.

[48]:
```
df_data.winning_age.describe()
```

[48]:
```
count   934.00
mean     59.95
std      12.62
min      17.00
25%      51.00
50%      60.00
75%      69.00
max      97.00
Name: winning_age, dtype: float64
```

[49]:
```
plt.figure(figsize=(8, 4), dpi=200)
sns.histplot(data=df_data,
             x=df_data.winning_age,
             bins=30)
plt.xlabel('Age')
plt.title('Distribution of Age on Receipt of Prize')
plt.show()
```



17.0.3 Age at Time of Award throughout History

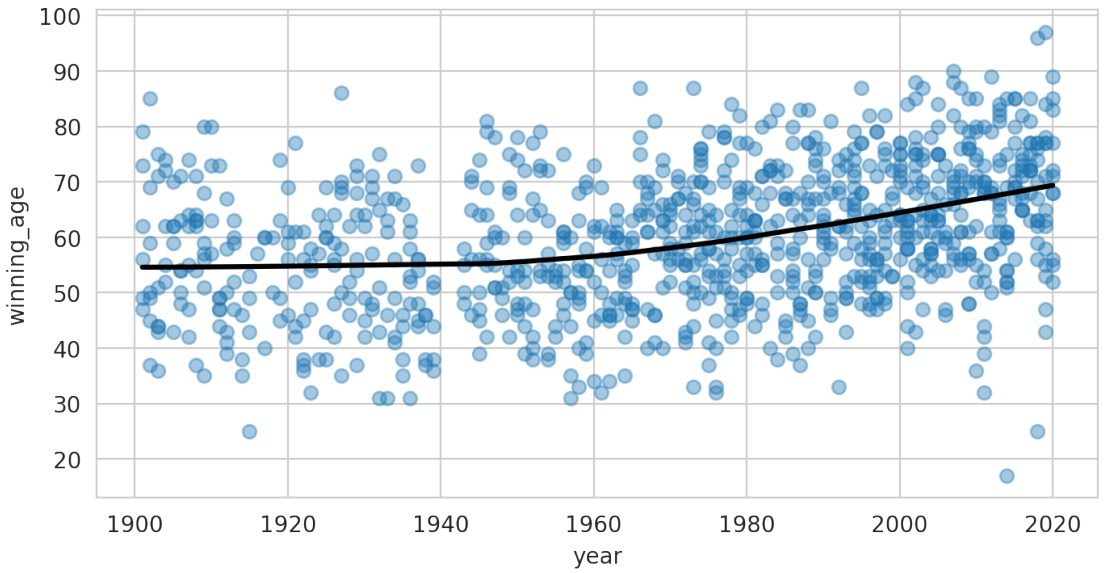
Are Nobel laureates being nominated later in life than before? Have the ages of laureates at the time of the award increased or decreased over time?

Challenge

- Use Seaborn to create a .regplot with a trendline.
- Set the `lowess` parameter to `True` to show a moving average of the linear fit.
- According to the best fit line, how old were Nobel laureates in the years 1900-1940 when they were awarded the prize?
- According to the best fit line, what age would it predict for a Nobel laureate in 2020?

[50]:
```
plt.figure(figsize=(8,4), dpi=200)
with sns.axes_style("whitegrid"):
    sns.regplot(data=df_data,
                x='year',
                y='winning_age',
                lowess=True,
                scatter_kws = {'alpha': 0.4},
```

```
                line_kws={'color': 'black'})

plt.show()
```



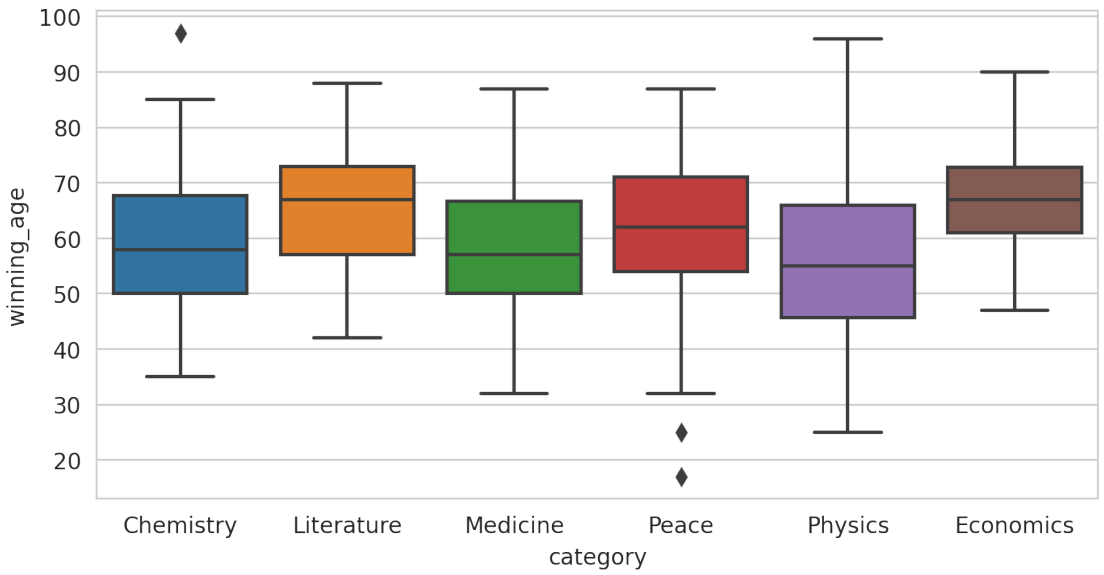
Using the lowess parameter allows us to plot a local linear regression. This means that the line of best fit is still linear, but more like a moving average, giving us a non-linear shape over the whole series. This is very interesting, as it clearly shows that Nobel Prize winners are receiving their awards later and later in life. From around 1900 to 1950, laureates were aged around 55, whereas today, they are closer to 70 when they receive their prize! The chart also shows that the gap has widened over the last ten years. There have been more very young and very old winners. In the 1950s/60s, prizewinners ranged in age from 30 to 80. More recently, this range has widened.

17.0.4 Winning Age Across the Nobel Prize Categories

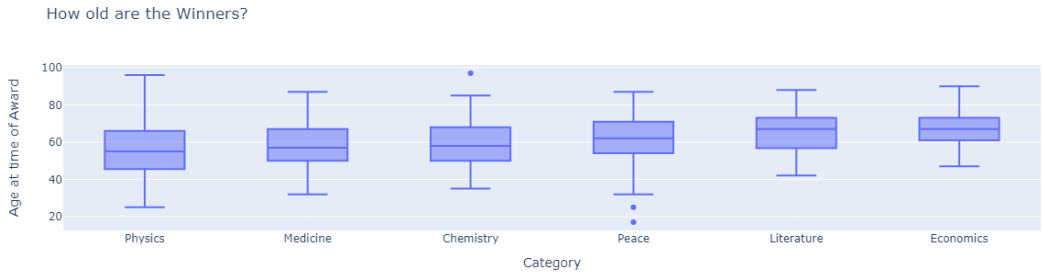
How does the age of laureates vary by category?

- Use Seaborn's `.boxplot()` to show how the mean, quartiles, max, and minimum values vary across categories. Which category has the longest "whiskers"?
- In which prize category are the average winners the oldest?
- In which prize category are the average winners the youngest?

[51]:
```
plt.figure(figsize=(8,4), dpi=200)
with sns.axes_style("whitegrid"):
    sns.boxplot(data=df_data,
                x='category',
                y='winning_age')

plt.show()
```



[52]:
```
# Box plot using plotly instead
box = px.box(df_data,
             x='category',
             y='winning_age',
             title='How old are the Winners?')

box.update_layout(xaxis_title='Category',
                  yaxis_title='Age at time of Award',
                  xaxis={'categoryorder':'mean ascending'},)

box.show()
```



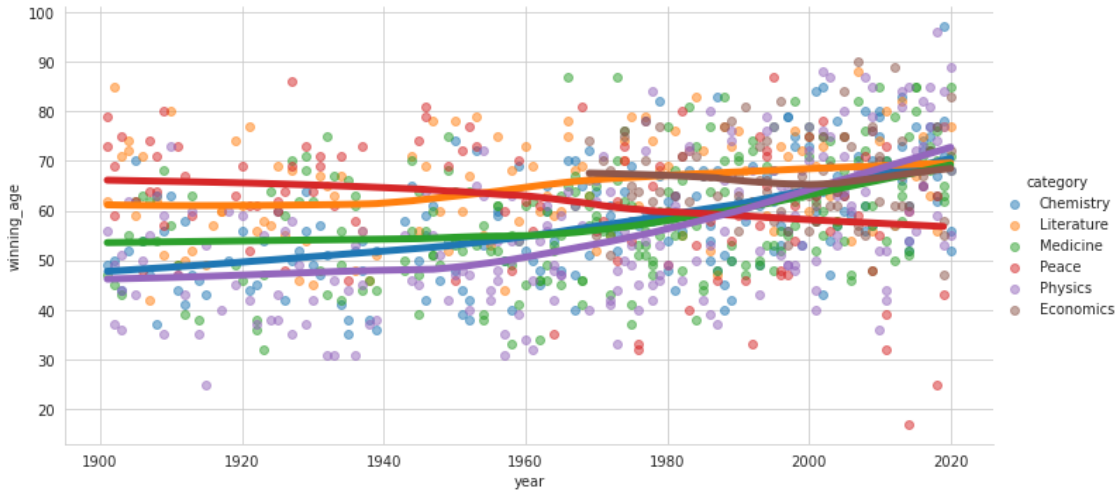
We note that winners in physics, chemistry and medicine have aged over time. The aging trend is strongest in physics. The average age used to be under 50, but is now over 70. Economics, the most recent category, is much more stable by comparison. The Peace Prize shows the opposite

trend: the winners are getting younger and younger! So, our scatterplots showing the best-fitting lines over time and our box plot of all the data can tell very different stories!

[53]:
```
with sns.axes_style('whitegrid'):
    sns.lmplot(data=df_data,
               x='year',
               y='winning_age',
               row = 'category',
               lowess=True,
               aspect=2,
               scatter_kws = {'alpha': 0.6},
               line_kws = {'color': 'black'},)

plt.show()
```


category = Chemistry
category = Literature
category = Medicine
category = Peace
category = Physics
category = Economics
winning_age
year
100
90
80
70
60
50
40
30
20
1900
1920
1940
1960
1980
2000
2020

[54]:
```
with sns.axes_style("whitegrid"):
    sns.lmplot(data=df_data,
               x='year',
               y='winning_age',
               hue='category',
               lowess=True,
               aspect=2,
               scatter_kws={'alpha': 0.5},
               line_kws={'linewidth': 5})

plt.show()
```



[ ]: