chapter06

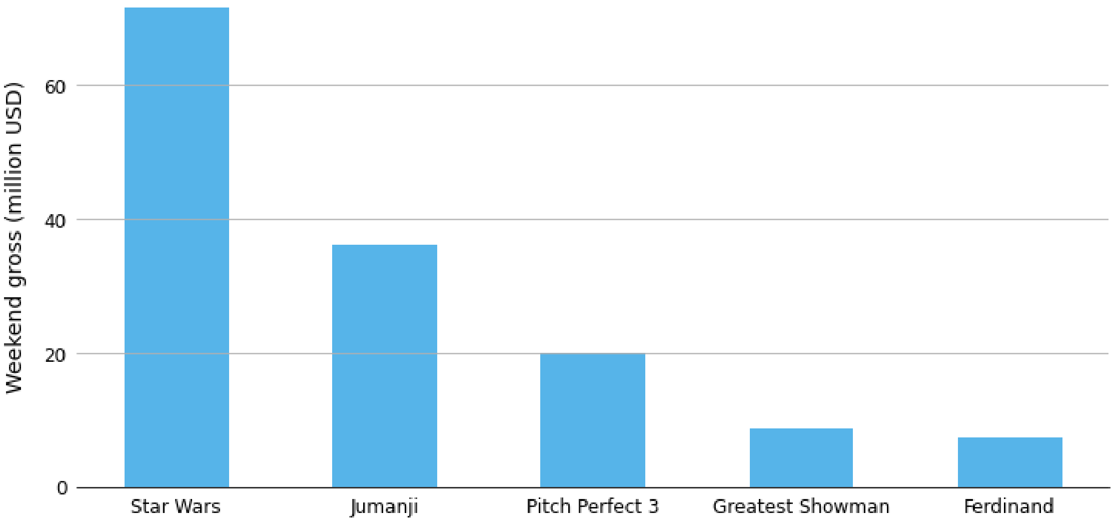
August 4, 2023

[2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import ticker
```

[3]:
```
boxoffice = pd.DataFrame(
    {"Rank":[1,2,3,4,5],
     'Title':['Star Wars: The Last Jedi','Jumanji:Welcome to the Jungle','Pitch␣
 ↪Perfect 3','The Greatest Showman','Perdinan'],
     'Short Title':['Star Wars','Jumanji','Pitch Perfect 3','Greatest␣
 ↪Showman','Ferdinand'],
     'Amount Text':['$71,565,498', '$36,169,328', '$19,928,525', '$8,805,843',␣
 ↪'$7,316,746'],
     'Amount':[71565498, 36169328, 19928525, 8805843, 7316746]
     }
    )
```

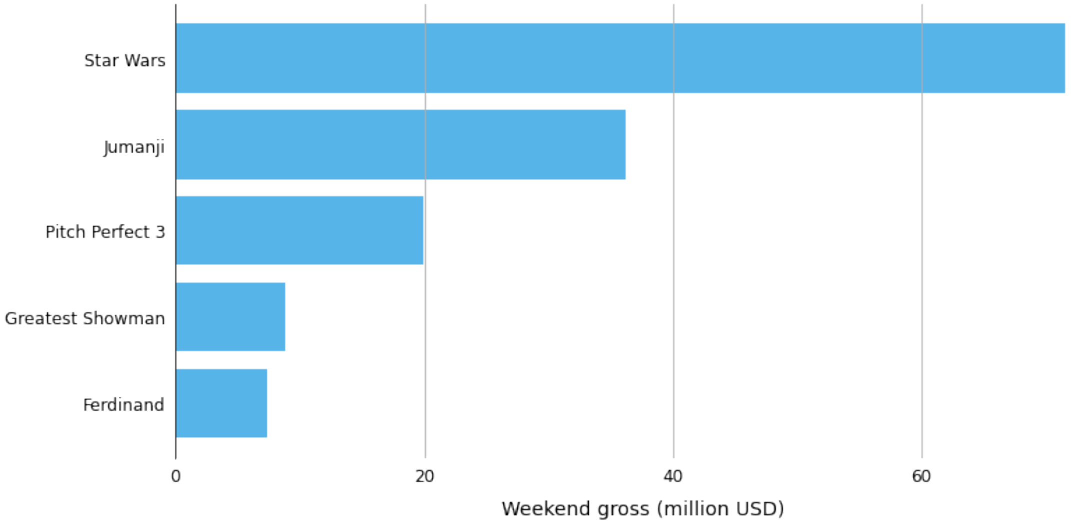
[4]:
```
amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()

fig, ax = plt.subplots(1,1,figsize = (12,6))

x_pos = [0, 3, 6, 9, 12] #set the positions of the bars
axes = ax.bar(
    x_pos,
    amount,
    # alpha=0.60,
    color = '#56B4E9',
    # color = ['#56B4E9', '#56B466', '#56B4E4', '#56B422','#56B4E1'],
    width=1.5)

ax.set_xticks(x_pos)
ax.set_xticklabels(movies)

ax.xaxis.set_ticks_position('none') # remove little ticks
ax.yaxis.set_ticks_position('none')

ax.set_yticks([0, 20, 40, 60])
ax.yaxis.grid()

ax.spines[:].set_visible(False)
ax.spines['bottom'].set_visible(True)

ax.tick_params(axis='both', which='major', labelsize=12) #set axes, major tick label size
ax.set_ylabel('Weekend gross (million USD)', fontsize=14, labelpad=10)
```

```
[4]: Text(0, 0.5, 'Weekend gross (million USD)')
```



```
[5]: fig, ax = plt.subplots(1,1,figsize = (12,6))

amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()

axes = ax.barh(
    y = movies,
    width = amount,
    # alpha = 0.6,
    color = '#56B4E9'
    )

ax.invert_yaxis()

ax.spines[:].set_visible(False)
ax.spines['left'].set_visible(True)
ax.set_xticks([0, 20, 40, 60])
ax.xaxis.grid()

ax.tick_params(axis='both', which='major', labelsize=12) #set axes, major tick␣
↪label size
ax.xaxis.set_ticks_position('none') # remove little ticks
ax.yaxis.set_ticks_position('none')

ax.set_xlabel('Weekend gross (million USD)', fontsize=14, labelpad=10)
```

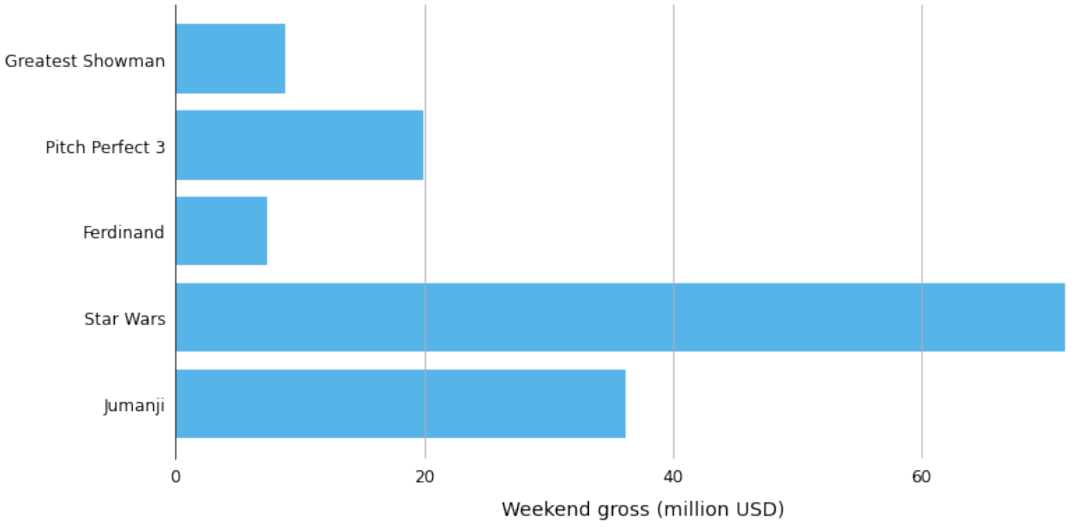
```
[5]: Text(0.5, 0, 'Weekend gross (million USD)')
```



```
[6]: fig, ax = plt.subplots(1,1,figsize = (12,6))

amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()

movies = ['Greatest Showman', 'Pitch Perfect 3', 'Ferdinand', 'Star Wars',␣
↪'Jumanji',]
amount = [8.805843, 19.928525, 7.316746, 71.565498, 36.169328]

axes = ax.barh(
    y = movies,
    width = amount,
    # alpha = 0.6,
    color = '#56B4E9'
    )
```

```
ax.invert_yaxis()

ax.spines[:].set_visible(False)
ax.spines['left'].set_visible(True)

ax.set_xticks([0, 20, 40, 60])
ax.xaxis.grid()

ax.tick_params(axis='both', which='major', labelsize=12) #set axes, major tick␣
 ↪label size
ax.xaxis.set_ticks_position('none') # remove little ticks
ax.yaxis.set_ticks_position('none')

ax.set_xlabel('Weekend gross (million USD)', fontsize=14, labelpad=10)
```

[6]:
```
Text(0.5, 0, 'Weekend gross (million USD)')
```

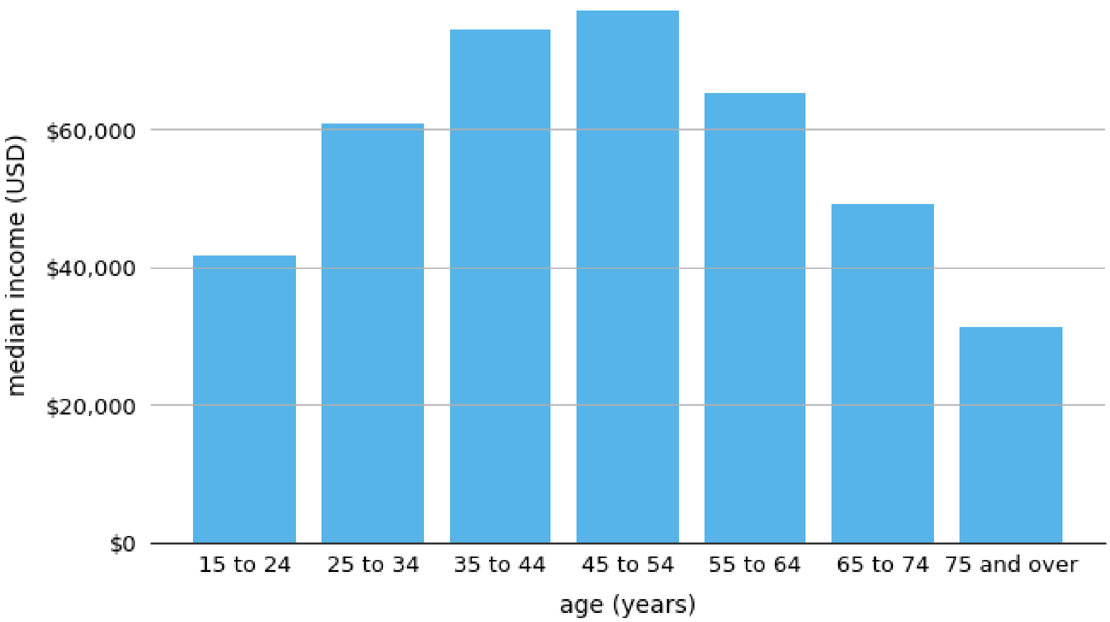


[7]:
```
import os
```

[8]:
```
incomedf = pd.read_csv(os.path.join('..','data','income_by_age.csv'))
```

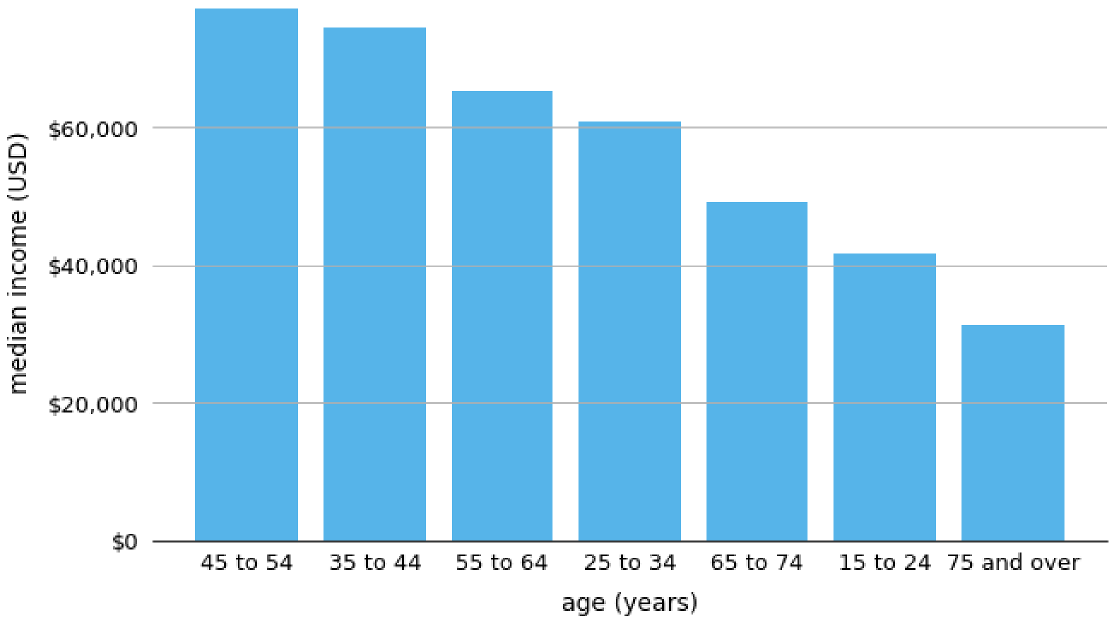
[9]:
```
incomeByAgedf = incomedf[incomedf.race=='all']
ageRange = ['15 to 24','25 to 34','35 to 44','45 to 54','55 to 64','65 to␣
 ↪74','75 and over']
median_income = [41655, 60932, 74481, 77213, 65239, 49072, 31313]
fig, ax = plt.subplots(1,1,figsize=(10,6))
axes = ax.bar(
    ageRange,
    median_income,
    color = '#56B4E9',
    )
ax.set_yticks([0, 20000, 40000, 60000])
ax.set_yticklabels(['$0', '$20,000', '$40,000', '$60,000'])
ax.tick_params(axis='both', which = 'major', labelsize =13)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.yaxis.grid()
ax.spines[:].set_visible(False)
ax.spines['bottom'].set_visible(True)
ax.set_ylabel('median income (USD)', fontsize=14, labelpad = 10)
ax.set_xlabel('age (years)', fontsize=14, labelpad = 10)
```

```
[9]: Text(0.5, 0, 'age (years)')
```



```
[10]: incomeByAgedf = incomeByAgedf.sort_values(by='median_income', ascending=False)
ageRange = incomeByAgedf['age'].values.tolist()
median_income=incomeByAgedf['median_income'].values.tolist()

fig, ax = plt.subplots(1,1,figsize=(10,6))
axes = ax.bar(
    ageRange,
    median_income,
    color = '#56B4E9',
    )
ax.set_yticks([0, 20000, 40000, 60000])
ax.set_yticklabels(['$0', '$20,000', '$40,000', '$60,000'])
ax.tick_params(axis='both', which = 'major', labelsize =13)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.yaxis.grid()
ax.spines[:].set_visible(False)
ax.spines['bottom'].set_visible(True)
ax.set_ylabel('median income (USD)', fontsize=14, labelpad = 10)
ax.set_xlabel('age (years)', fontsize=14, labelpad = 10)
```

```
[10]: Text(0.5, 0, 'age (years)')
```

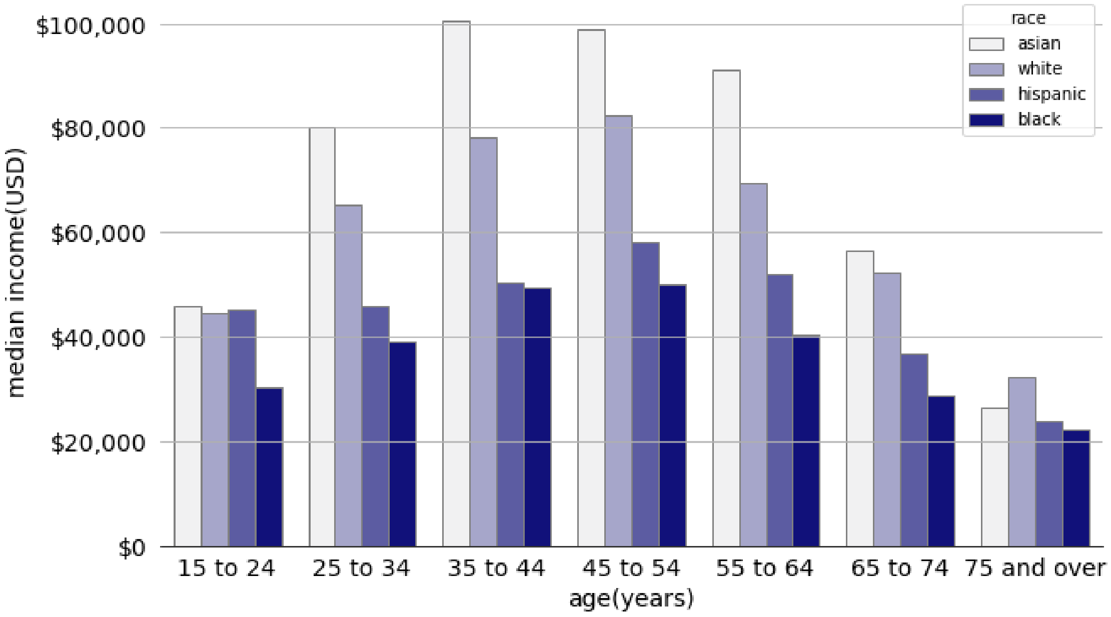


```
[11]: import seaborn as sns
```

```
[12]: incomeByRacedf = incomedf[incomedf.race.
       ↪isin(['asian','white','hispanic','black'])]
      # incomeByRacedf['race'] = incomeByRacedf['race'].astype('category').cat.
       ↪set_categories(['asian','white','hispanic','black'],ordered=True)
      # incomeByRacedf = incomeByRacedf.sort_values(by='race')
```

```
[59]: fig, ax1 = plt.subplots(1,1, figsize = (10,6))
      hue_order = ['asian','white','hispanic','black']
      axes = sns.barplot(
          data = incomeByRacedf,
          x = 'age',
          y = 'median_income',
          hue = 'race',
          hue_order =hue_order,
          color='darkblue',
          linewidth=1,
          edgecolor=".5",
          ax=ax1
      )
      ax1.tick_params(axis='both', which = 'major', labelsize = 14)
      ax1.xaxis.set_ticks_position('none')
      ax1.yaxis.set_ticks_position('none')

      y_pos = [0,20000,40000,60000,80000,100000]
      y_labels = ['$0','$20,000','$40,000','$60,000','$80,000','$100,000']
      ax1.yaxis.set_major_locator(ticker.FixedLocator(y_pos))
      ax1.yaxis.set_major_formatter(ticker.FixedFormatter(y_labels))

      ax1.spines[:].set_visible(False)
      ax1.spines['bottom'].set_visible(True)
      ax1.yaxis.grid()
      ax1.set_xlabel("age(years)", fontsize =14)
      ax1.set_ylabel('median income(USD)', fontsize =14)
```

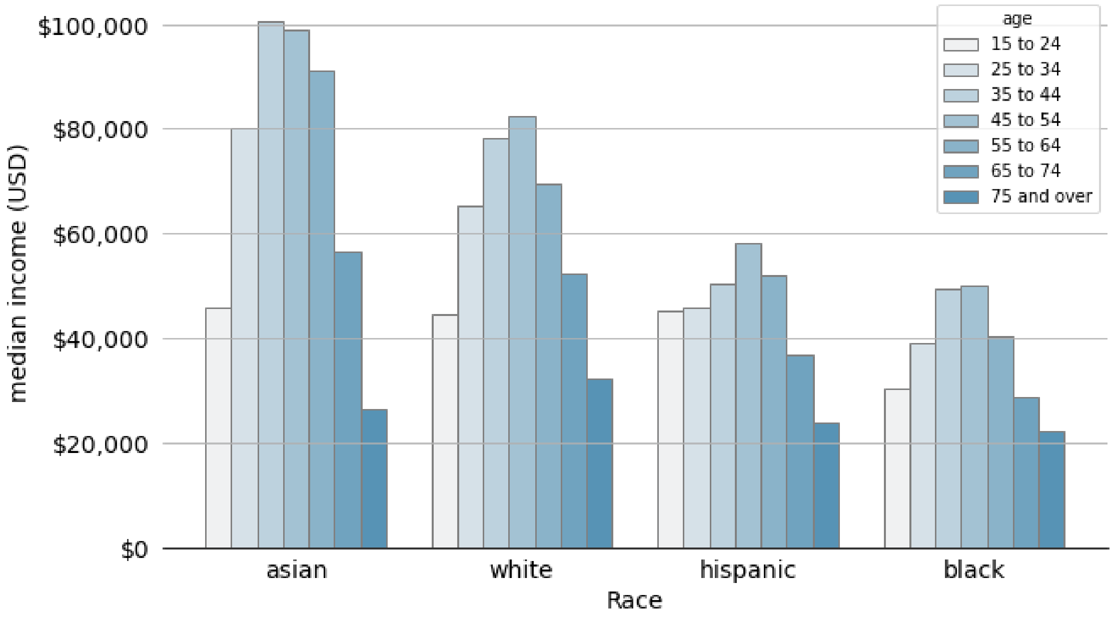
```
[59]: Text(0, 0.5, 'median income(USD)')
```



```
[13]: fig, ax1 = plt.subplots(1,1, figsize = (10,6))
```

```
hue_order = ['15 to 24','25 to 34','35 to 44','45 to 54','55 to 64','65 to␣
 ↪74','75 and over']
# sns.color_palette("ch:start=.2,rot=-.3", as_cmap=True)
sns.dark_palette("#4798c5", reverse=True, as_cmap=True)
axes = sns.barplot(
    data= incomeByRacedf,
    x = 'race',
    y ='median_income',
    hue = 'age',
    hue_order=hue_order,
    ax=ax1,
    color='#4798c5',
    edgecolor='.5',
    order=['asian','white','hispanic','black']
)

ax1.spines[:].set_visible(False)
ax1.spines['bottom'].set_visible(True)

ax1.xaxis.set_ticks_position('none')
ax1.yaxis.set_ticks_position('none')

ax1.tick_params(axis='both', which='major', labelsize=14)

ax1.yaxis.grid()

ax1.set_yticks([0, 20000, 40000, 60000, 80000, 100000],['$0', '$20,000',␣
 ↪'$40,000', '$60,000', '$80,000', '$100,000'], fontsize=14)

ax1.set_xlabel('Race', fontsize = 14)
ax1.set_ylabel('median income (USD)', fontsize = 14)
axes.plot()
```

```
[13]: []
```



0.0.1 Stacked bar chart

[14]:

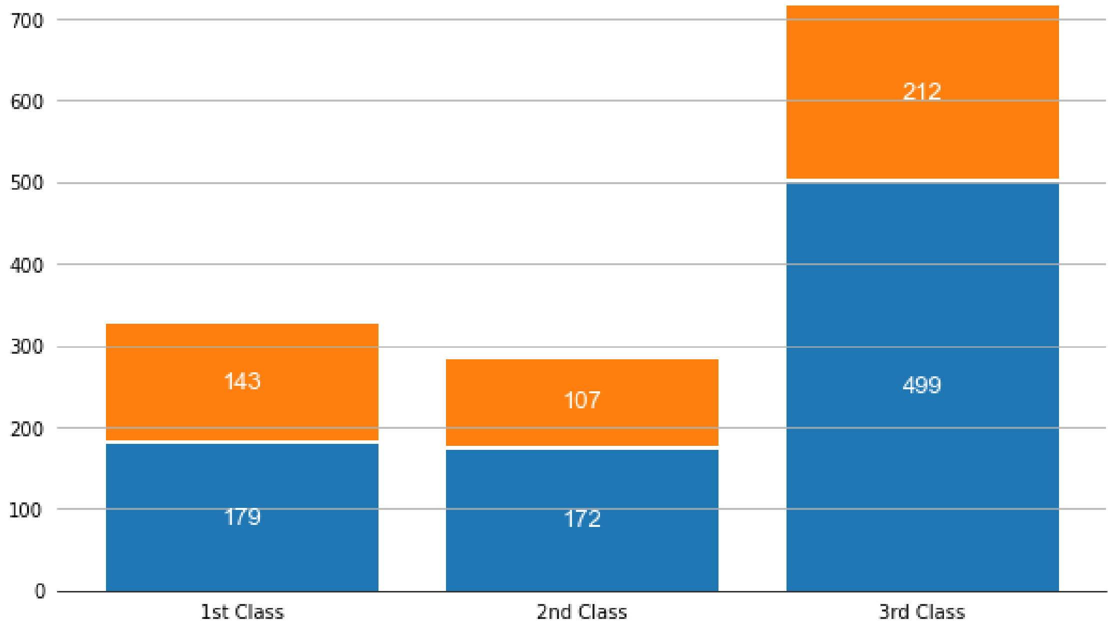
```
titanic_df = pd.read_csv(os.path.join('..','data','Titanic.csv'))
```

[15]:

```
gender_distribution = {'Male':np.array([179, 172, 499]),
                       'Female':np.array([143, 107, 212])}
classes = ['1st Class', '2nd Class', '3rd Class']
```

[16]:

```
fig, ax = plt.subplots(1,1,figsize=(10,6))
bottom = 0
for boolean, number in gender_distribution.items():
    axes = ax.bar(classes, number, label=boolean, bottom= bottom)
    bottom += number+5
    ax.bar_label(axes, label_type='center', color = 'white', family='Arial',␣
 ↪size =13)

ax.spines[:].set_visible(False)
ax.spines['bottom'].set_visible(True)

ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.yaxis.grid()
```

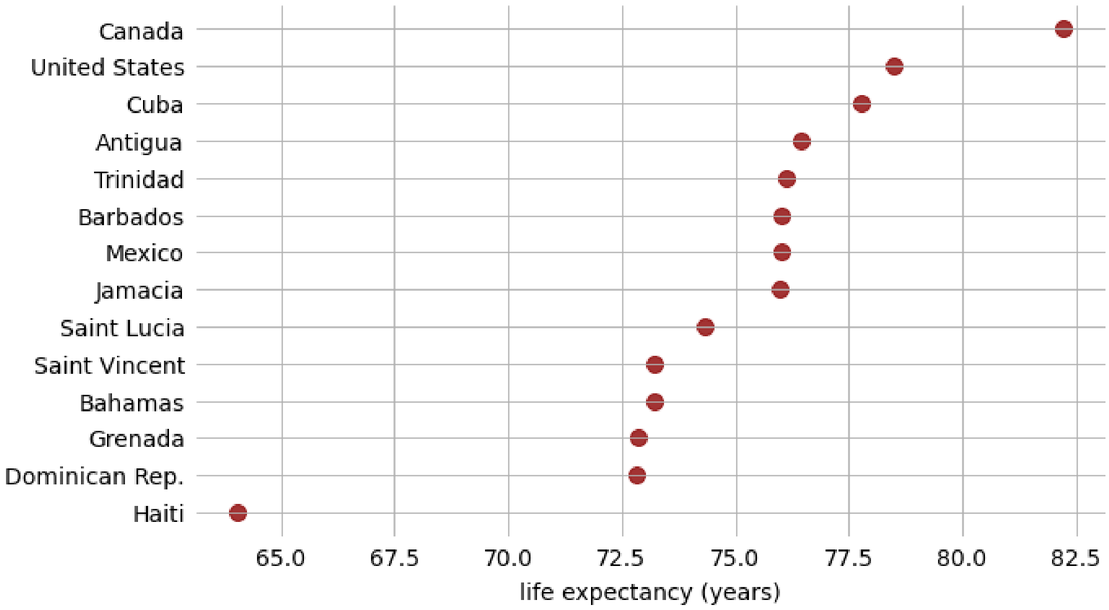


0.0.2 Dot plots and heatmaps

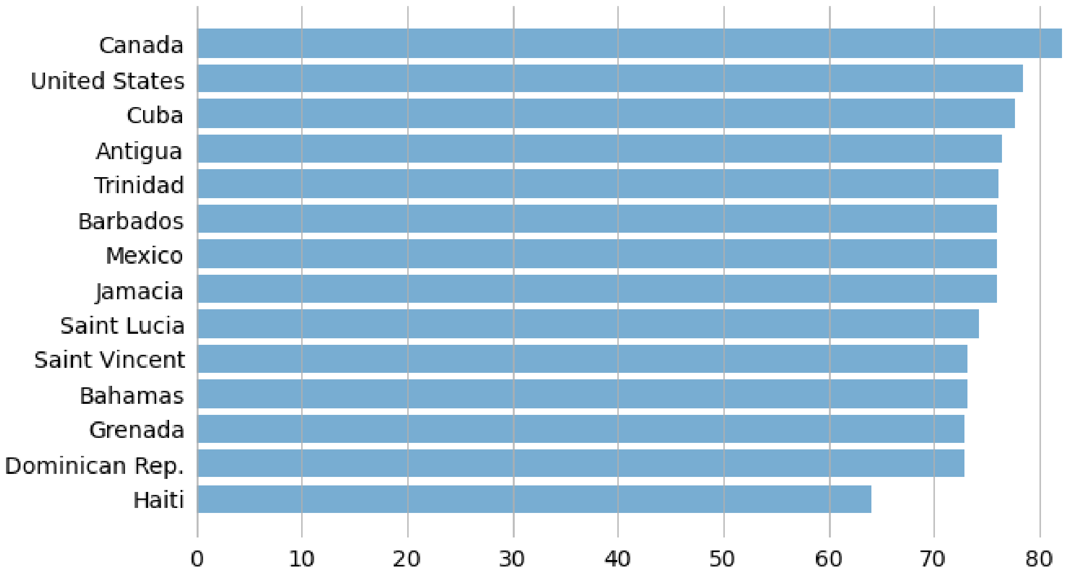
```
[17]: life_expectancy = pd.read_excel(os.path.join('..','data','life expectance.
        ↪xlsx'))
```

```
[18]: fig, ax = plt.subplots(1,1,figsize = (10,6))
      ax.spines[:].set_visible(False)
      ax.grid()
      ax.scatter(life_expectancy['Years'], life_expectancy['Countries'],color =␣
        ↪'darkred', s = 100, alpha=.8)
      ax.invert_yaxis()
      ax.tick_params(axis='both', which='major', labelsize=14)
      ax.xaxis.set_ticks_position('none')
      ax.yaxis.set_ticks_position('none')
      ax.set_xlabel('life expectancy (years)', fontsize=14, labelpad=7)
```

```
[18]: Text(0.5, 0, 'life expectancy (years)')
```

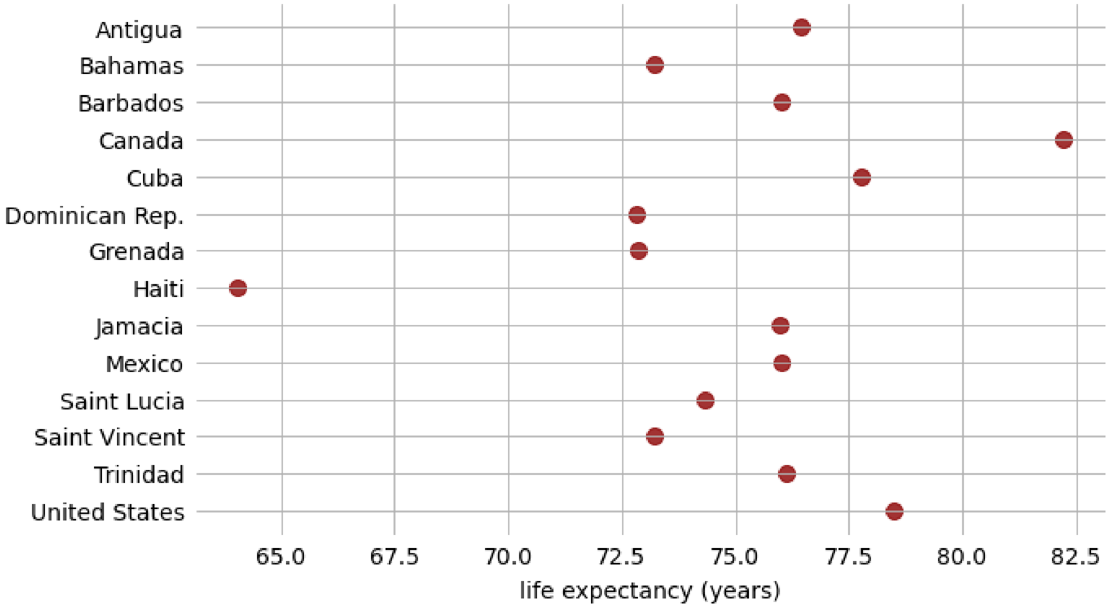


[19]:
```
fig, ax = plt.subplots(1,1,figsize = (10,6))
ax.barh(life_expectancy['Countries'], life_expectancy['Years'], alpha=0.6)
ax.invert_yaxis()

ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.xaxis.grid()
ax.tick_params(axis='both',which = 'major', labelsize = 14)
```



[20]:
```
ordered_life_expectancy = life_expectancy.sort_values(by='Countries')
fig, ax = plt.subplots(1,1,figsize = (10,6))
ax.spines[:].set_visible(False)
ax.grid()
ax.scatter(ordered_life_expectancy['Years'],␣
 ↪ordered_life_expectancy['Countries'],color = 'darkred', s = 100, alpha=.8)
ax.invert_yaxis()
ax.tick_params(axis='both', which='major', labelsize=14)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.set_xlabel('life expectancy (years)', fontsize=14, labelpad=7)
```

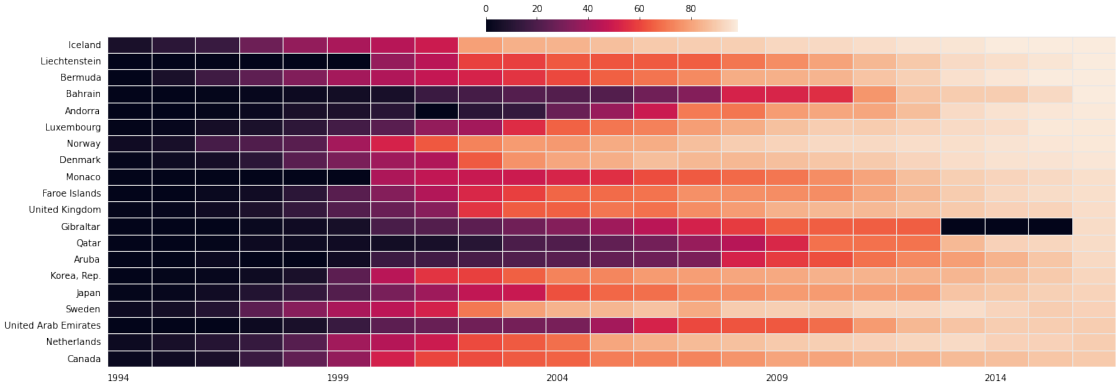
[20]:
```
Text(0.5, 0, 'life expectancy (years)')
```



[21]:
```
user_per_100 = pd.read_csv(os.path.join('..','data','internet','Internet user␣
 ↪per 100.csv'), encoding='ISO-8859-1')
user_per_100.fillna(0, inplace=True)
```

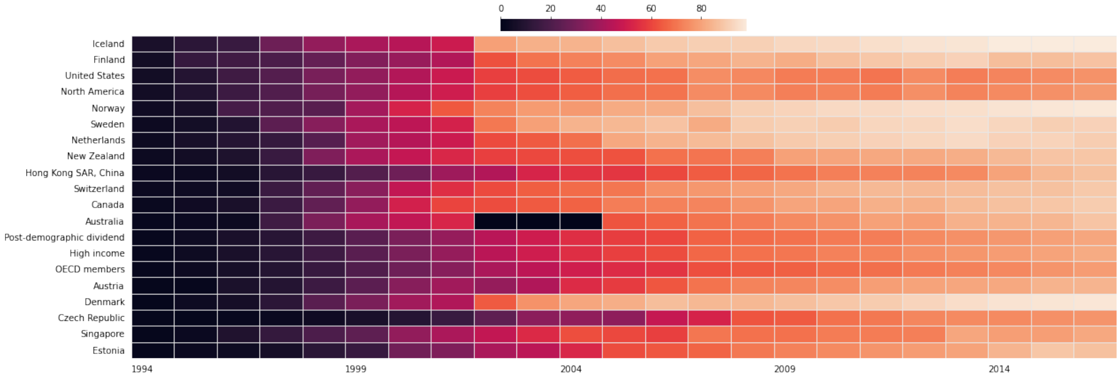
[73]:
```
use_data = user_per_100.sort_values(by='2016',ascending=False).iloc[:20,:]
fig, ax = plt.subplots(1,1, figsize = (20,8))
axes = sns.heatmap(use_data.iloc[:,8:], linewidth=.5, linecolor = '#e8e8e8',␣
 ↪cbar_kws=dict(use_gridspec=False,location="top",pad=0.01,shrink=0.25), ax=ax)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.set_xticks(np.arange(0, 23, 5), np.arange(1994, 2016, 5).tolist(), ha='left')
ax.set_yticklabels(use_data['country'],rotation=0)
ax.plot()
```

```
[73]: []
```



```
[75]: use_data = user_per_100.sort_values(by='1994', ascending=False).iloc[:20,:]
fig, ax = plt.subplots(1,1, figsize = (20,8))
axes = sns.heatmap(use_data.iloc[:,8:], linewidth=.5, linecolor = '#e8e8e8',␣
  ↪cbar_kws=dict(use_gridspec=False,location="top",pad=0.01,shrink=0.25), ax=ax)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')
ax.set_xticks(np.arange(0, 23, 5), np.arange(1994, 2016, 5).tolist(), ha='left')
ax.set_yticklabels(use_data['country'],rotation=0)
ax.plot()
```

```
[75]: []
```



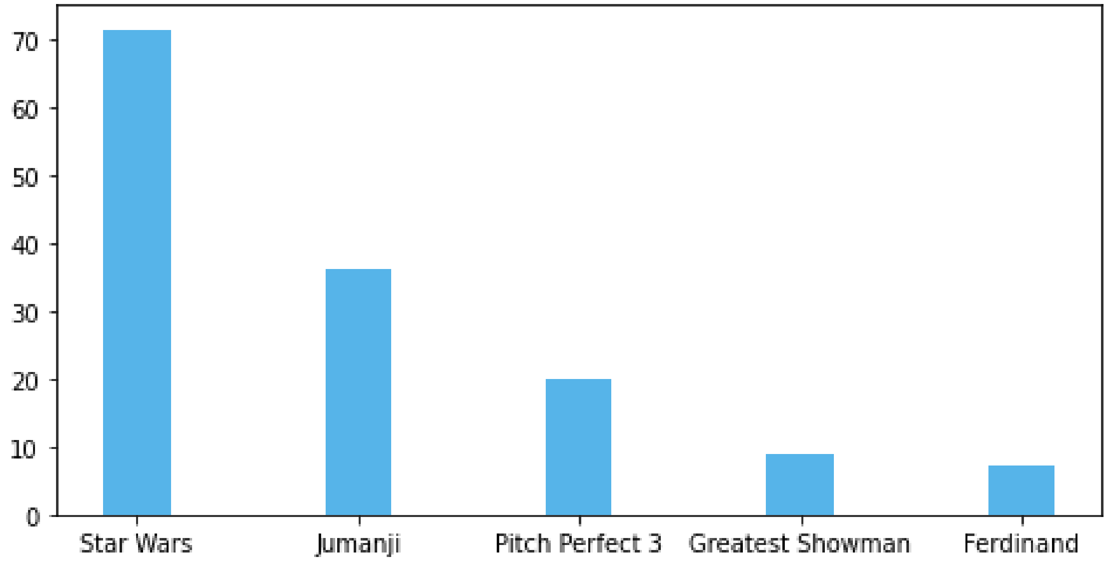
```
[ ]:
```

[ ]:

```
```

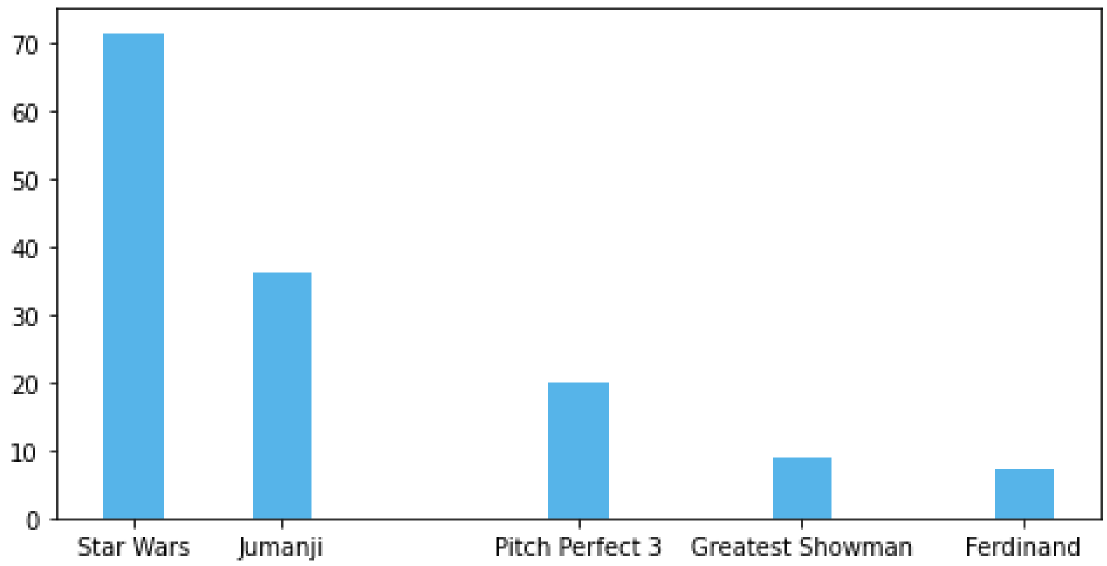
[24]:

```
fig, ax = plt.subplots(1,1,figsize=(8,4))
amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()
axes = ax.bar(movies, amount, width =0.3, color = '#56B4E9',)
```

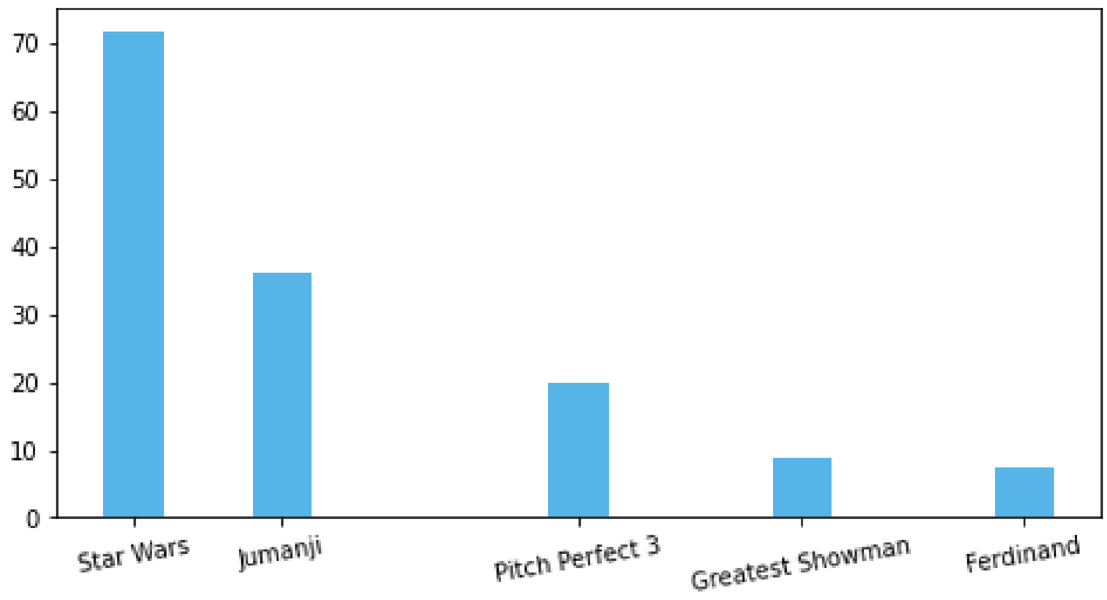


[40]:

```
fig, ax = plt.subplots(1,1,figsize=(8,4))
amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()

x_pos = [1,3,7,10,13]
ax.xaxis.set_major_locator(ticker.FixedLocator(x_pos))
ax.xaxis.set_major_formatter(ticker.FixedFormatter(movies))
axes = ax.bar(x_pos, amount, width=0.8, color = '#56B4E9',)
# ax.xaxis.set_ticklabels(movies,rotation=10)
# ax.tick_params(axis='x', rotation=10)
```



[26]:
```
fig, ax = plt.subplots(1,1,figsize=(8,4))
amount = (boxoffice['Amount']/(1000000)).tolist()
movies = boxoffice['Short Title'].tolist()

x_pos = [1,3,7,10,13]
ax.xaxis.set_major_locator(ticker.FixedLocator(x_pos))
ax.xaxis.set_major_formatter(ticker.FixedFormatter(movies))
axes = ax.bar(x_pos, amount, width=0.8, color = '#56B4E9')
ax.tick_params(axis='x', rotation=10)
```



[ ]:

```

```

[ ]:

```

```

[27]:

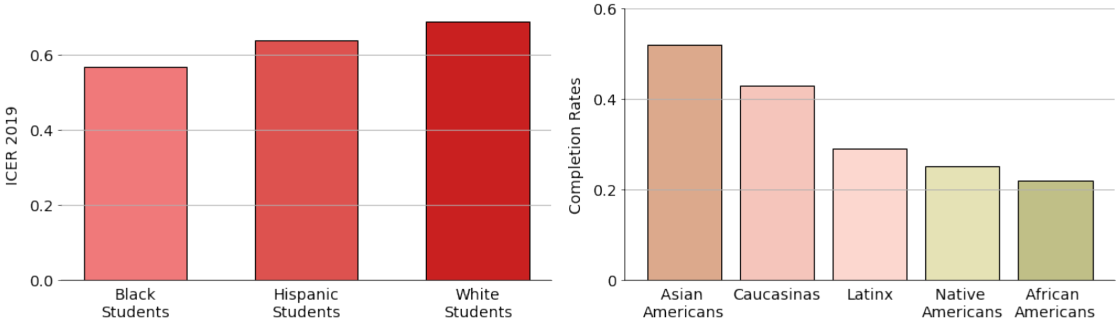
```
students = [f'Black\nStudents',f'Hispanic\nStudents',f'White\nStudents']
data = [0.57, .64, 0.69]
```

[28]:

```
fig,(ax1,ax2) = plt.subplots(1,2,figsize=(14, 4))

plt.tight_layout()
plt.subplots_adjust(wspace=0.15)
x_pos = [0.5, 0.75, 1]
ax1.xaxis.set_major_locator(ticker.FixedLocator(x_pos))
ax1.xaxis.set_major_formatter(ticker.FixedFormatter(students))
axes=ax1.bar(x_pos, data, width =0.15, color=['#f0797a','#dd524f','#c92020'],␣
 ↪edgecolor='k')
ax1.spines[:].set_visible(False)
ax1.spines['bottom'].set_visible(True)
ax1.xaxis.set_ticks_position('none')
ax1.set_yticks([0, 0.2, 0.4, 0.6])
ax1.yaxis.grid()
ax1.tick_params(axis='both', labelsize=14)
ax1.set_ylabel('ICER 2019',fontsize=14, labelpad=7)

studs = [f'Asian \nAmericans', f"Caucasinas", f'Latinx', f'Native \nAmericans',␣
 ↪f'African \nAmericans']
nums = [0.52, 0.43, 0.29, 0.25, 0.22]
ax2.bar(studs, nums, color=['#dca98c','#f5c5bb','#fcd7cf','#e5e2b5','#c0bf87'],␣
 ↪edgecolor='k')
ax2.tick_params(axis='both', which='major', labelsize=14)
ax2.yaxis.grid()
ax2.spines[:].set_visible(False)
ax2.spines['left'].set_visible(True)
ax2.spines['bottom'].set_visible(True)
ax2.xaxis.set_ticks_position('none')
ax2.set_yticks([0, 0.2, 0.4, 0.6])
ax2.set_yticklabels(['0','0.2', '0.4', '0.6'])
ax2.set_ylabel('Completion Rates', fontsize=14, labelpad=7)
```

[28]:

```
Text(525.1090909090908, 0.5, 'Completion Rates')
```


ICER 2019
0.0
0.2
0.4
0.6
Black Students
Hispanic Students
White Students
Completion Rates
0
0.2
0.4
0.6
Asian Americans
Caucasinas
Latinx
Native Americans
African Americans

[ ]: