

visualisation

June 16, 2025

```
[ ]: import pandas as pd

data = pd.read_csv("law_data.csv")

mean_success_race = data.groupby("race")["first_pf"].mean()
mean_success_sex = data.groupby("sex")["first_pf"].mean()

mean_note_race = data.groupby("race")["UGPA"].mean()
mean_note_sex = data.groupby("sex")["UGPA"].mean()

display(mean_success_race, mean_success_sex)
display(mean_note_race, mean_note_sex)
```

```
race
Amerindian    0.686869
Asian         0.815385
Black         0.617785
Hispanic      0.754098
Mexican       0.755784
Other         0.836177
Puertorican   0.700000
White         0.920263
Name: first_pf, dtype: float64
```

```
sex
1    0.876271
2    0.897911
Name: first_pf, dtype: float64
```

```
race
Amerindian    2.961616
Asian         3.218107
Black         2.893760
Hispanic      3.138525
Mexican       3.031620
Other         3.207850
Puertorican   3.022727
White         3.259776
Name: UGPA, dtype: float64
```

```
sex
1    3.290133
2    3.177134
Name: UGPA, dtype: float64
```

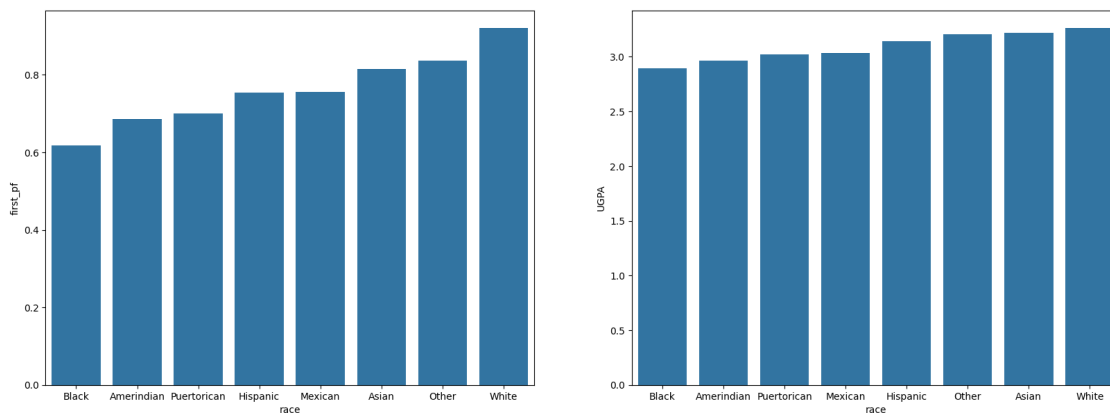
```
[ ]: import seaborn as sns
import matplotlib.pyplot as plt

plt.rcParams['font.family'] = 'sans-serif'
plt.rcParams['font.sans-serif'] = ['Tahoma', 'DejaVu Sans',
                                   'Lucida Grande', 'Verdana']

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=((20,7)))
fig.suptitle("Success rate and average score based on ethnicity", fontsize=25)
sns.barplot(ax=ax1, x = "race", y = "first_pf", data=mean_success_race.
            ↪to_frame().sort_values(by="first_pf"))
sns.barplot(ax=ax2, x = "race", y = "UGPA", data=mean_note_race.to_frame().
            ↪sort_values(by="UGPA"))

plt.show()
```

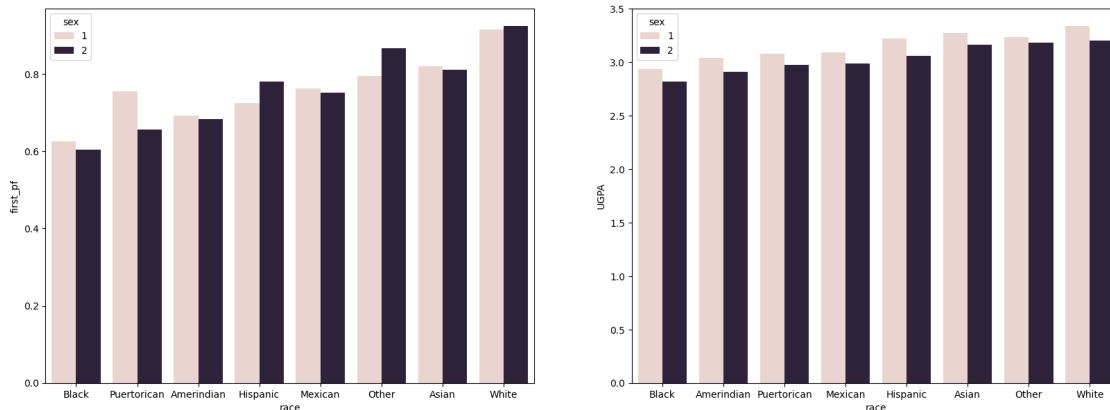
Success rate and average score based on ethnicity



```
[ ]: fig, (ax1, ax2) = plt.subplots(1, 2, figsize=((20,7)))
fig.suptitle("Success rate and average score based on ethnicity and sex",
            ↪fontsize=25)
sns.barplot(ax=ax1, x = "race", y = "first_pf", data=data.groupby(["race",
            ↪"sex"])["first_pf"].mean().to_frame().sort_values(by="first_pf"), hue="sex")
sns.barplot(ax=ax2, x = "race", y = "UGPA", data=data.groupby(["race",
            ↪"sex"])["UGPA"].mean().to_frame().sort_values(by="UGPA"), hue="sex")

plt.show()
```

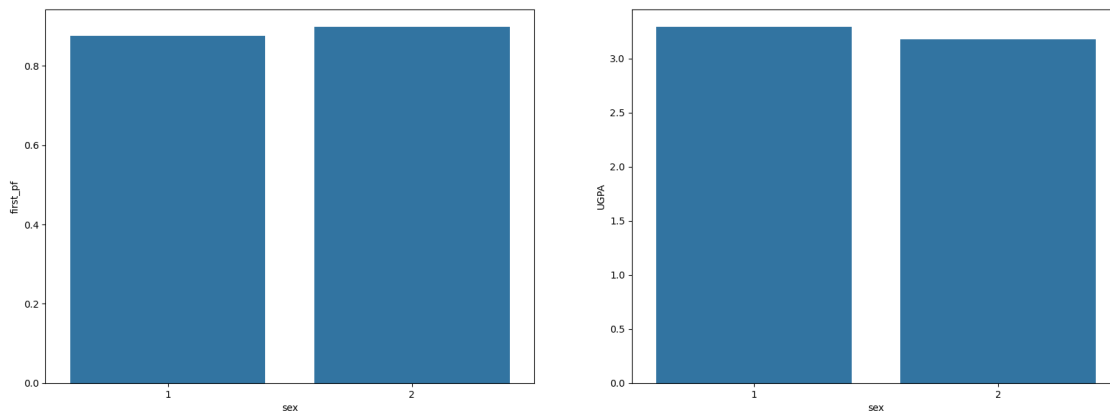
Success rate and average score based on ethnicity and sex



```
[ ]: fig, (ax1, ax2) = plt.subplots(1, 2, figsize=((20,7)))
fig.suptitle("Success rate and average score based on sex", fontsize=25)
sns.barplot(ax=ax1, x = "sex", y = "first_pf", data=mean_success_sex.to_frame().
    ↪sort_values(by="first_pf"))
sns.barplot(ax=ax2, x = "sex", y = "UGPA", data=mean_note_sex.to_frame().
    ↪sort_values(by="UGPA"))

plt.show()
```

Success rate and average score based on sex



```
[ ]: count_by_race = data.groupby("race")["first_pf"].count().to_frame().
    ↪sort_values(by="first_pf", ascending=False)
count_by_sex = data.groupby("sex")["first_pf"].count().to_frame().
    ↪sort_values(by="first_pf")
```

```

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=((15,7)))
fig.suptitle("Count based on ethnicity and sex", fontsize=25)
ax1.pie(count_by_race["first_pf"], labels=count_by_race.T.columns, colors=sns.
    color_palette("pastel"))
ax2.pie(count_by_sex["first_pf"], labels=count_by_sex.T.columns, colors=sns.
    color_palette("pastel"))

plt.tight_layout()
plt.show()

```

Count based on ethnicity and sex

