Does the Big Business of College Sports Impact Student Outcomes?

Predicting post-graduate earnings using University spending behaviors

Overview and Business Understanding

College sports is big business. This biggest schools, like Alabama, Texas, Clemson, and Ohio State generate tens of millions, even hundreds of millions of dollars in revenue across their football and basketball teams. This money has been growing significantly over the past 15 years. From 2006 to 2016, NCAA D1 Football Subdivision (FBS) schools saw overall revenues from increase to 8.5 billion dollars from 4.4 billion dollars (https://www.ipr.northwestern.edu/documents/working-papers/2020/wp-20-42.pdf). Although these FBS schools (130 schools across 10 major conferences) field 20 or more sports, 58% of total athletic department revenue is derived from men's football and basketball. (https://www.ipr.northwestern.edu/documents/working-papers/2020/wp-20-42.pdf) What makes these sports even more profitable are the multi-billion-dollar media deals that the major conferences make with networks like ESPN, CBS, and NBC. Conferences like the Big Ten and the SEC even have their own media networks.

The topic of paying athletes has been the subject of hundreds of articles and numerous economics papers. Instead, I want to explore the connection, if any exists, between what a school spends on its sports programs and the outcomes of its students. Do schools that spend a lot on sports spend less on academics? Do students perform better or worse at major sporting institutions than those that are known more for academic rigor?

Additionally, we'll explore inequality between male and female students. We know about the gender pay gap, but we're not as familiar with how unsupported women's sports and programs are in general at many universities, big and small. Girls and women's sports have long been underfunded and under-supported for decades, and only started to improve with the introduction of Title IX in 1972. [The title prohibits sex-based discrimination at all schools and programs that receive federal funding. (https://en.wikipedia.org/wiki/Title_IX)](https://en.wikipedia.org/wiki/Title_IX) It requires schools (colleges in this case) to offer proportional athletic opportunities to men and women. However, compliance falls short despite these regulations. [According to the Women's Sports Foundation (https://www.womenssportsfoundation.org/wp-content/uploads/2020/01/Chasing-Equity-Full-Report-Web.pdf)](https://www.womenssportsfoundation.org/wp-content/uploads/2020/01/Chasing-Equity-Full-Report-Web.pdf), 87% of the 1,084 NCAA-participant institutions offered disproportionately higher rates of athletic opportunities to male athletes compared with their enrollment. Division 1 schools fared even worse, with only 8.6% of these institutions offering athletic opportunities to female athletes proportional to their enrollment.

The regression analysis in this notebook will analyze how much of an effect sports spending has on average earnings of post-grads 6 years from when they enrolled. In addition, the data has over 100 features, and we'll use the analysis to quantify to what affect these features have on 6 year average earnings. We may find that more traditional metrics, like admission rate, SAT scores, and family income, have larger effects on earnings than sports spending, but we can still find value in what schools spend on its sports programs. If schools with high sports budgets have underperforming students, it may be time to shift priorities to more academic-driven spending.

Data Understanding

We have data from 3 sources:

- [College Athletics Financial Information (CAFI) Database (http://cafidatabase.knightcommission.org/nfs)](http://cafidatabase.knightcommission.org/nfs) This dataset, created by the Knight Commission on Intercollegiate Athletics, captures the expenses and revenues of ahtletic departments across the country. Not all data is accounted for because Private schools are not required to share this information, but the data is complete for all public schools. This dataset has features like total football spending and coaching salaries, student fees collected, ticket sales, and medical expenses. The data consists of all Division 1 schools in the Football Bowl Subdivision (FBS).
- [Equity in Athletics (https://ope.ed.gov/athletics/#/)](https://ope.ed.gov/athletics/#/) This dataset, created by the Office of Postsecondary Education of the U.S. Department of Education. From the website: "The data are drawn from the OPE Equity in Athletics Disclosure Website database. This database consists of athletics data that are submitted annually as required by the Equity in Athletics Disclosure Act (EADA), by all co-educational postsecondary institutions that receive Title IV funding (i.e., those that participate in federal student aid programs) and that have an intercollegiate athletics program. This data allows us to look at how schools are supporting men's and women's sports at very granular detail. We can see how much a school spends on each coach for each t, weamhat the operating expenses are per male and female athlete for each sport, and how many students participate in sports. This will help answer any inequality questions we may have.
- [College Scorecard (https://collegescorecard.ed.gov/)](https://collegescorecard.ed.gov/) This dataset is also a product of the Department of Education. It contains many thousands of columns, capturing things like demographics, loan repayment rates, average and median earnings of post-grads, and even average family incomes at each school. Our target - 6 year average earnings, is from this

dataset. That feature, captured in 2014-15, contains the average earnings of that school's post-graduates, 6 years after they enrolled at that school. So these are students that started back in 2008-09 and 2009-10.

In [1]:
```
#importing libraries
import pandas as pd
import numpy as np
from numpy import mean
import itertools
import warnings
import json
warnings.filterwarnings('ignore')

import matplotlib
import matplotlib.pyplot as plt
from matplotlib.pylab import rcParams
plt.style.use('ggplot')
import seaborn as sns

import plotly as py
import plotly.express as px
import plotly.graph_objects as go
from yellowbrick.regressor import ResidualsPlot

import statsmodels.api as sm
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.graphics.tsaplots import plot_pacf
from statsmodels.tsa.seasonal import seasonal_decompose
from statsmodels.stats.outliers_influence import variance_inflation_factor

import sklearn

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, MinMaxScaler, RobustScaler
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
from sklearn.impute import SimpleImputer
from sklearn.linear_model import Lasso, Ridge, LinearRegression
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import ElasticNetCV, ElasticNet
from sklearn.model_selection import RepeatedKFold, KFold
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error
from sklearn.impute import KNNImputer
from sklearn import metrics

from statsmodels.stats.diagnostic import het_breuschpagan
from statsmodels.stats.diagnostic import het_white
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols

import pmdarima as pm
from pmdarima.arima import auto_arima
from pmdarima import model_selection
from pmdarima.utils import decomposed_plot
from pmdarima.arima import decompose

import folium
```

In [2]:
```
#importing data

#sports expenditures
ncaa_rev_exp_df = pd.read_excel('NCAA_rev_exp/ncaa_revenue_expenses_2009_2015.xls

#equity data
eada_df = pd.read_excel('EADA_Files/eada_all_2009_2015.xlsx')

#outcomes data
outcomes_df = pd.read_excel('Outcomes_Data_Files/outcomes_data_2009_2014.xlsx',
                            na_values = ['PrivacySuppressed', 'Null', 'NULL'])
```

In [3]:
```
#combining equity data with ncaa data
eada_merge = pd.merge(ncaa_rev_exp_df, eada_df,
                        how='left',
                      left_on=['IPEDS_ID','INSTNM', 'Year'],
                      right_on = ['IPEDS_ID','INSTNM', 'Year'])

#creating main combined dataset
ncaa_df = pd.merge(eada_merge, outcomes_df,
                   how='left',
                   left_on=['IPEDS_ID','INSTNM', 'Year'],
                   right_on = ['IPEDS_ID','INSTNM', 'Year'])
ncaa_df.head()
```

Out[3]:

	IPEDS_ID	INSTNM	Year	NCAA Subdivision	FBS Conference	Total Expenses	Excess Transfers Back	Other Expenses
0	197869	Appalachian State University	2009	Football Championship Subdivision	Sun Belt Conference	1.765941e+07	0.0	3.729257e+06
1	197869	Appalachian State University	2010	Football Championship Subdivision	Sun Belt Conference	1.781469e+07	0.0	3.366068e+06
2	197869	Appalachian State University	2011	Football Championship Subdivision	Sun Belt Conference	1.648940e+07	0.0	1.609778e+06
3	197869	Appalachian State University	2012	Football Championship Subdivision	Sun Belt Conference	1.802044e+07	0.0	1.799869e+06
4	197869	Appalachian State University	2013	Football Championship Subdivision	Sun Belt Conference	2.204318e+07	0.0	1.769266e+06

5 rows × 132 columns

In [4]:
```
#dropping Total Revenues and Total Expenses (which come from Knight Commission da
#from the EADA dataset which is complete for all schools

ncaa_df.drop(columns = ['Total Expenses', 'Total Revenues', 'STABBR'], inplace = 

#creating new column called pct_revenue_from_students, which is the percentage of
#annual student fees (most of which go to athletic departments)

ncaa_df['pct_revenue_from_students'] = ncaa_df['Student Fees'] / ncaa_df['Grand T
```

In [5]:
```
#Our target variable will be 6 year mean earnings from 2015. We'll also drop 2015
prediction_df = pd.read_excel('2015_predictions.xlsx')

ncaa_df = ncaa_df[ncaa_df['Year'] != 2015]
```

In [6]:
```
prediction_df.head()
```

Out[6]:

	IPEDS_ID	INSTNM	Year	NCAA Subdivision	FBS Conference	Excess Transfers Back	Other Expenses	Medical
0	197869	Appalachian State University	2015	Football Bowl Subdivision	Sun Belt Conference	0.0	3.881606e+06	1.261922e+05
1	104151	Arizona State University	2015	Football Bowl Subdivision	Pacific-12 Conference	0.0	1.077719e+07	1.869622e+06
2	100858	Auburn University	2015	Football Bowl Subdivision	Southeastern Conference	0.0	1.430038e+07	8.916366e+05
3	150136	Ball State University	2015	Football Bowl Subdivision	Mid-American Conference	0.0	1.480496e+06	7.249489e+05
4	223232	Baylor University	2015	Football Bowl Subdivision	Big 12 Conference	NaN	NaN	NaN

5 rows × 132 columns

In [7]:
```
ncaa_df['Year'].value_counts()
```

Out[7]:
```
2014    118
2013    117
2012    116
2011    115
2010    114
2009    114
Name: Year, dtype: int64
```

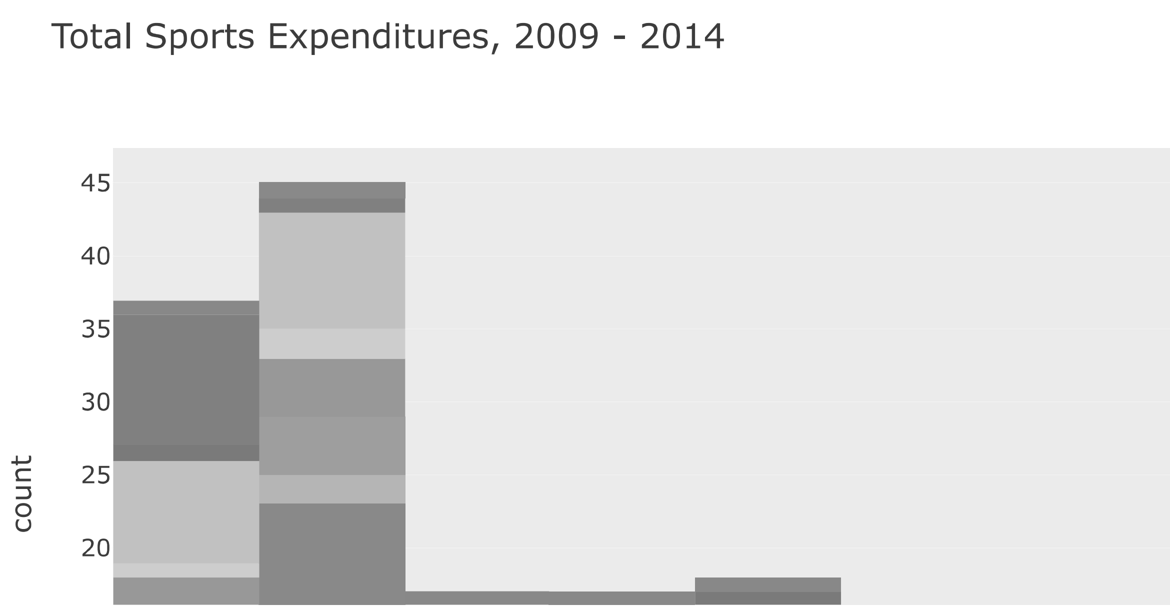
Exploring the data

In [8]:
```
len(ncaa_df['INSTNM'].unique())
```

Out[8]: 118

We have 118 total schools in our dataset. Let's look at the distribution of athletic expenses.

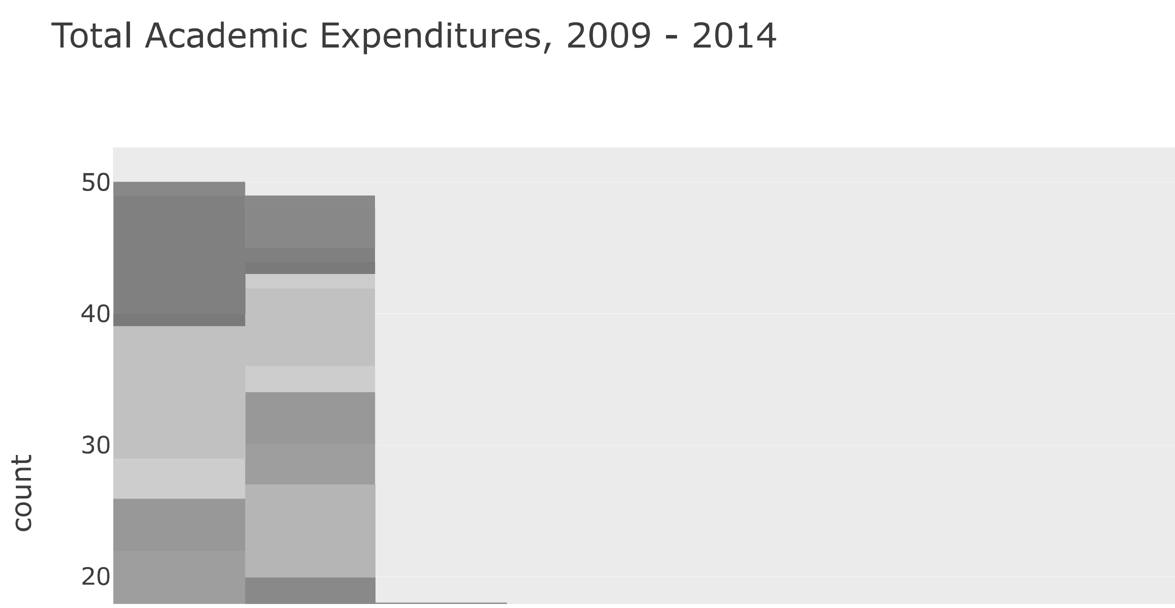
In [9]:
```
school_sports_group = pd.DataFrame(ncaa_df.groupby(['INSTNM', 'FBS Conference'])

school_sports_group.reset_index(inplace = True)
fig = px.histogram(school_sports_group, x="Grand Total Expenses", color = 'FBS Co
                   title = 'Total Sports Expenditures, 2009 - 2014')
fig.show()
```



We can see above that all of the big-time football schools at major conferences, like Auburn, Alabama, Texas, and Ohio State, are towards the higher end of expenses, with the SEC, Big Ten, and Big 12 occupying space in the $600M club.

Let's compare this now to total academic spending.

In [10]:
```
school_academic_group = pd.DataFrame(ncaa_df.groupby(['INSTNM', 'FBS Conference']
school_academic_group.reset_index(inplace = True)
fig = px.histogram(school_academic_group, x="Total Academic Spending (University-
                   title = 'Total Academic Expenditures, 2009 - 2014')
fig.show()
```

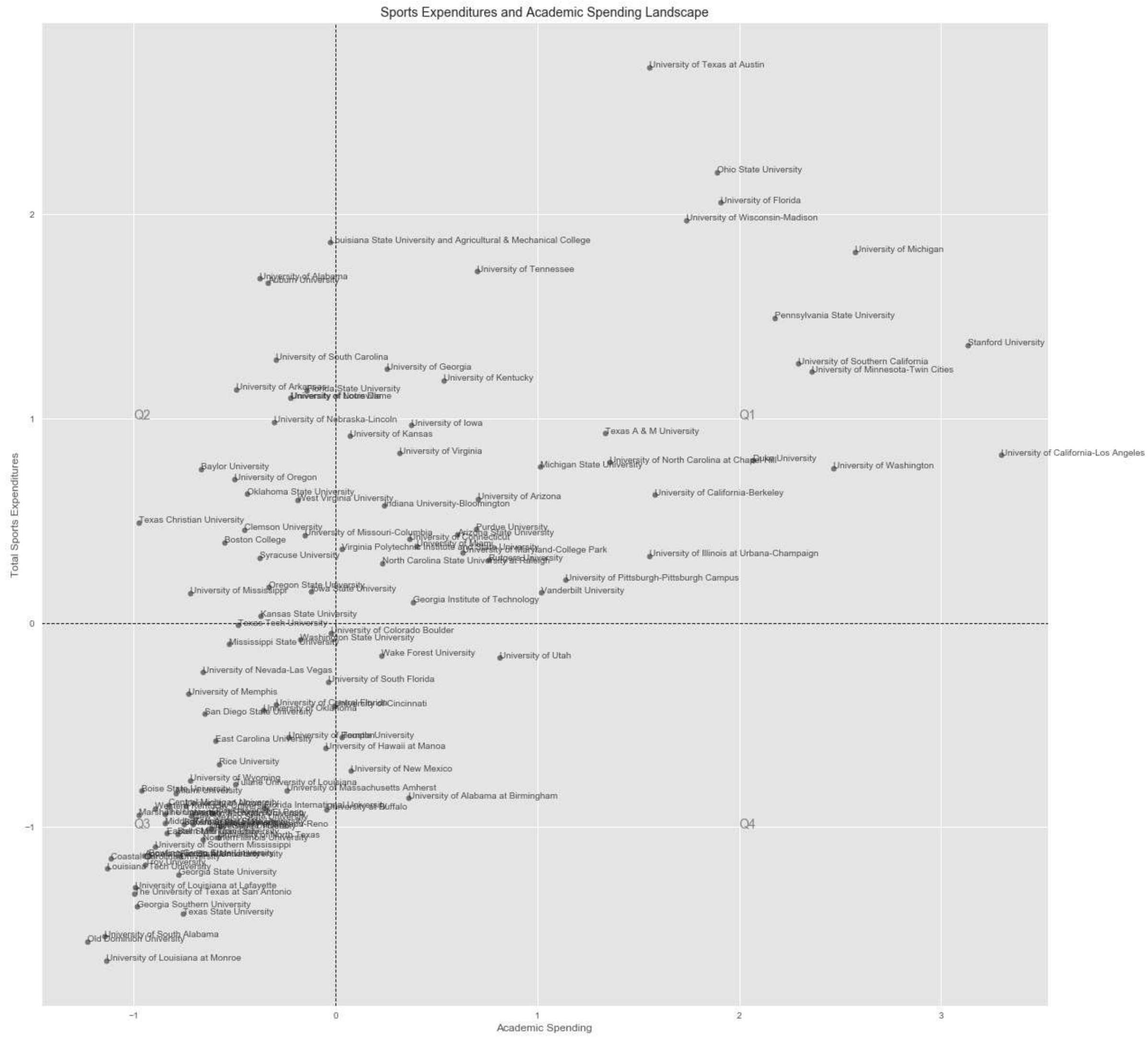


The distribution is more concentrated on the lower end, but the schools are a bit more varied overall, with the Pac-12 and ACC occupying space greater than $10B. Only one SEC school, the University of Florida, spent more than 10B over the 6 years. We also see representation on the higher end of schools traditionally known for great academics like Stanford, UCLA, Duke, and Michigan. Let's now see if there's any strong correlation between academic spending and athletic expenditures:

In [11]:
```
school_expenditures_group = pd.DataFrame(ncaa_df.groupby('INSTNM').agg({'Total Ac
                                                                       'Grand Total Exp
#as_index=False
school_expenditures_group.reset_index(inplace = True)
fig = px.scatter(school_expenditures_group, x="Total Academic Spending (Universit
                 color = 'INSTNM',
                  title = 'Total Academic Expenditures and Expenses, 2009 - 2014'
fig.show()
```

In [12]:
```
def normalize(feature):
    return (feature - feature.mean()) / feature.std()

school_expenditures_group_norm = school_expenditures_group.copy()
school_expenditures_group_norm.set_index('INSTNM', inplace = True)

school_expenditures_group_norm = school_expenditures_group_norm.apply(normalize)
```

In [13]:
```
#repurposed code from https://github.com/hashABCD/Publications/blob/main/Medium/Q

plt.figure(figsize=(20,20))
plt.style.use('ggplot')
sns.scatterplot(data=school_expenditures_group_norm, x='Total Academic Spending (

#Title and labels
plt.title('Sports Expenditures and Academic Spending Landscape')
plt.xlabel('Academic Spending')
plt.ylabel('Total Sports Expenditures')

#Country names
for i in range(school_expenditures_group_norm.shape[0]):
          plt.text(school_expenditures_group_norm['Total Academic Spending (Unive
                   y=school_expenditures_group_norm['Grand Total Expenses'][i],
                   s=school_expenditures_group_norm.index[i], alpha=0.8)

#Quadrant Marker
plt.text(x=2, y=-1, s="Q4",alpha=0.7,fontsize=14, color='b')
plt.text(x=-1, y=-1, s="Q3",alpha=0.7,fontsize=14, color='b')
plt.text(x=-1, y=1, s="Q2", alpha=0.7,fontsize=14, color='b')
plt.text(x=2, y=1, s="Q1", alpha=0.7,fontsize=14, color='b')

#Mean values
plt.axhline(y=school_expenditures_group_norm['Grand Total Expenses'].mean(), colo
plt.axvline(x=school_expenditures_group_norm['Total Academic Spending (University

plt.savefig('blog_2.png')
plt.show()
```



Using a quadrant graph above, we can see what schools are spending more or less on sports and academics as compared to their peers. 0 is the average, so schools in Q1 are spending above average in athletics and academics. Schools like Michigan, UCLA, Ohio State, and Stanford, are leading the pack in both sports and academic expenditures.

Schools that fall in Q3 are generally smaller schools with smaller budgets, like East Carolina University and Rice. Schools in Q2 spend above average in sports, but below average in academics as compared to the rest. We see some traditional football powerhouses, like Clemson, Alabama, and Baylor on this side of the graph. Moving forward, it'll be interesting to see if schools prioritizing sports expenditures over academic spending end up hurting their students' outcomes over the medium to long run.

In [14]:
```
school_expenditures_group.corr()
```

Out[14]:

	Total Academic Spending (University-Wide)	Grand Total Expenses
Total Academic Spending (University-Wide)	1.000000	0.690049
Grand Total Expenses	0.690049	1.000000

Let's take a look at the top 10 sports spenders and top 10 academic spenders:

In [15]:
```
school_sports_group.sort_values(by = 'Grand Total Expenses', ascending = False).h
```

Out[15]:

	INSTNM	FBS Conference	Grand Total Expenses
134	University of Texas at Austin	Big 12 Conference	806209070.0
40	Ohio State University	Big Ten Conference	714172167.0
87	University of Florida	Southeastern Conference	687489018.0
140	University of Wisconsin-Madison	Big Ten Conference	671998381.0
27	Louisiana State University and Agricultural & ...	Southeastern Conference	652661070.0
107	University of Michigan	Big Ten Conference	644125540.0
133	University of Tennessee	Southeastern Conference	627137554.0
73	University of Alabama	Southeastern Conference	620799663.0
2	Auburn University	Southeastern Conference	616976142.0
45	Pennsylvania State University	Big Ten Conference	586066159.0

In [16]:
```
school_academic_group.sort_values(by = 'Total Academic Spending (University-Wide)
```

Out[16]:

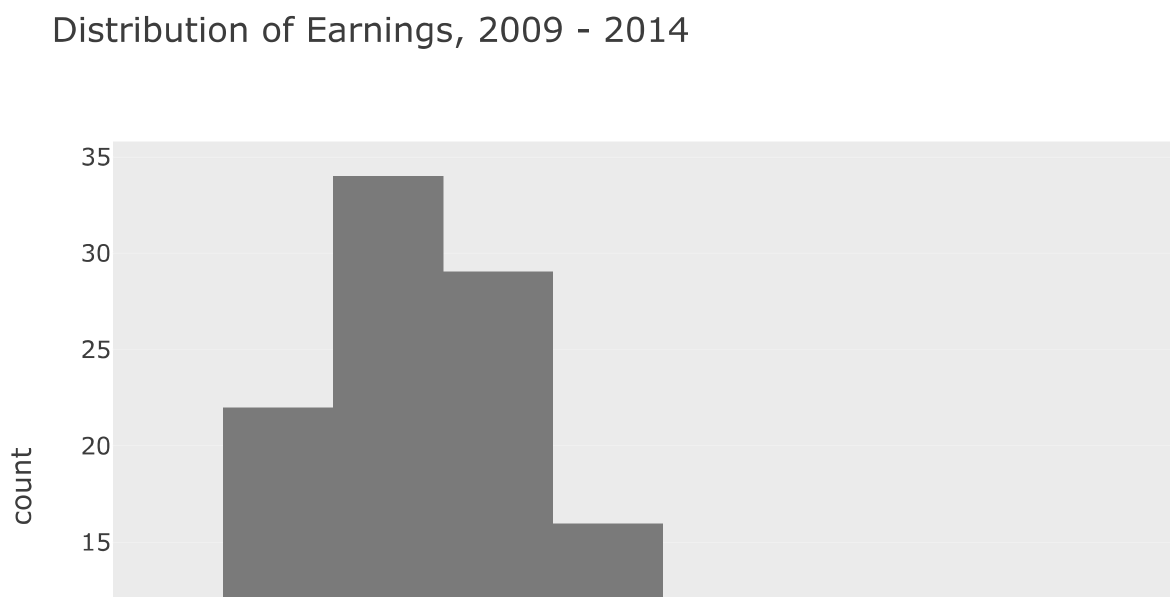
	INSTNM	FBS Conference	Total Academic Spending (University-Wide)
78	University of California-Los Angeles	Pacific-12 Conference	1.949101e+10
53	Stanford University	Pacific-12 Conference	1.878749e+10
107	University of Michigan	Big Ten Conference	1.641962e+10
139	University of Washington	Pacific-12 Conference	1.597102e+10
108	University of Minnesota-Twin Cities	Big Ten Conference	1.551876e+10
131	University of Southern California	Pacific-12 Conference	1.523084e+10
45	Pennsylvania State University	Big Ten Conference	1.474232e+10
12	Duke University	Atlantic Coast Conference	1.428856e+10
87	University of Florida	Southeastern Conference	1.359673e+10
40	Ohio State University	Big Ten Conference	1.352496e+10

Looking at the scatter plot, we definitely see a strong, positive correlation between sports expenditures and academic spending. This makes sense because schools that can afford large sports budgets likely have larger budgets university-wide. However, there are some differences between the groups of schools, which may indicate how university adminstrations prioritize their spending.

For example, LSU, Auburn, and Alabama are top 10 in sports expenditures, but spend much less proportionately on academics. Conversely, schools like Michigan, Ohio State, and Penn State, are in both top 10 lists.

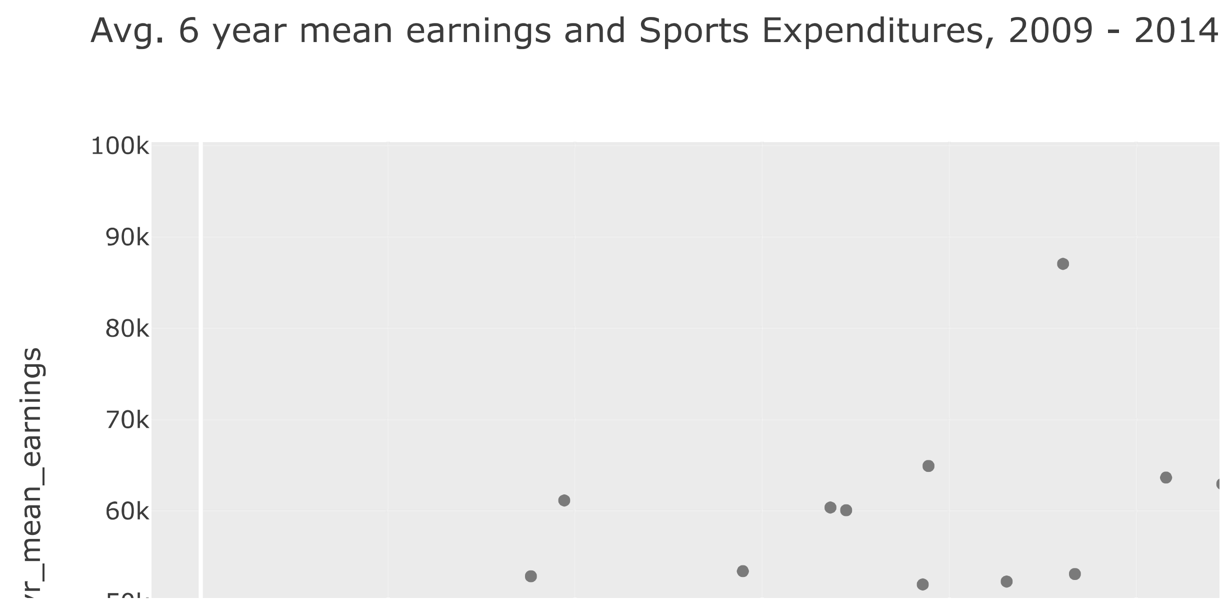
Let's take a look at what schools produce the best earnings of its graduates and if there's any correlation with sports spending or academics spending.

In [17]:
```
school_earnings_group = pd.DataFrame(ncaa_df.groupby('INSTNM').agg({'6yr_mean_ear
                                                                                   }))
#as_index=False
school_earnings_group.reset_index(inplace = True)
fig = px.histogram(school_earnings_group, x="6yr_mean_earnings",
                   title = 'Distribution of Earnings, 2009 - 2014')
fig.show()
```



So we only have two outliers, Duke and Stanford, coming in at over $80K for average 6 year earnings. The rest are concentrated between 40K and 60K. Let's create a scatter plot to look at theses earnings in comparison too sports expenditures and academic spending.

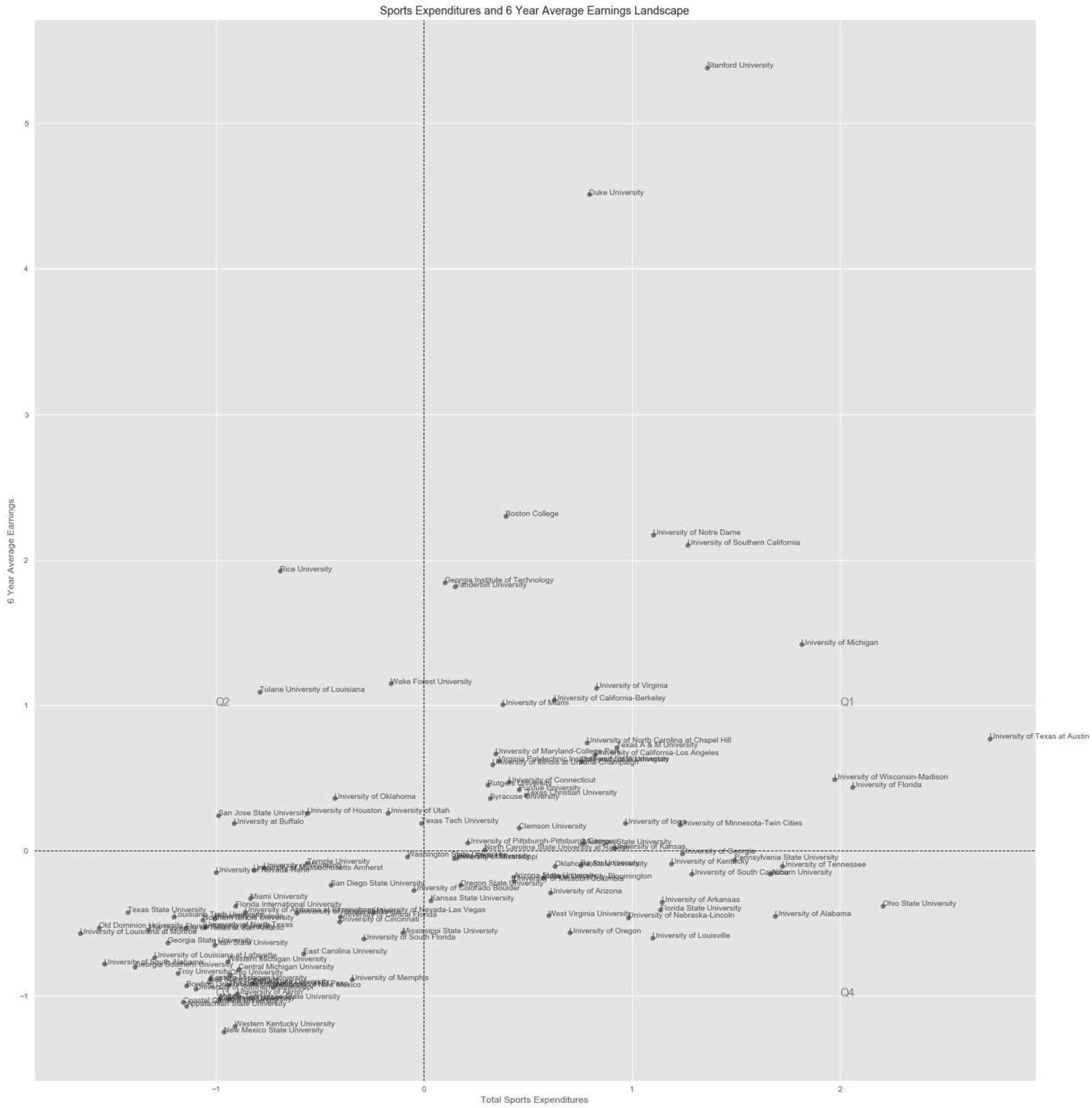
In [18]:
```
school_earnings_sports_group = pd.DataFrame(ncaa_df.groupby('INSTNM').agg({'6yr_m
                                                                   'Grand Total Exp
#as_index=False
school_earnings_sports_group.reset_index(inplace = True)
fig = px.scatter(school_earnings_sports_group, x='Grand Total Expenses', y = '6yr
                 labels={"Grand Total Expenses": "Total Sports Expenditures"},
                  title = 'Avg. 6 year mean earnings and Sports Expenditures, 200
fig.show()
```



In [19]:
```
school_earnings_sports_group_norm = school_earnings_sports_group.copy()
school_earnings_sports_group_norm.set_index('INSTNM', inplace = True)

school_earnings_sports_group_norm = school_earnings_sports_group_norm.apply(norma
```

In [20]:
```
plt.figure(figsize=(23,25))
plt.style.use('ggplot')
sns.scatterplot(data=school_earnings_sports_group_norm, x='Grand Total Expenses',

#Title and labels
plt.title('Sports Expenditures and 6 Year Average Earnings Landscape')
plt.xlabel('Total Sports Expenditures')
plt.ylabel('6 Year Average Earnings')

#Country names
for i in range(school_earnings_sports_group_norm.shape[0]):
          plt.text(school_earnings_sports_group_norm['Grand Total Expenses'][i],
                   y=school_earnings_sports_group_norm['6yr_mean_earnings'][i],
                   s=school_earnings_sports_group_norm.index[i], alpha=0.8)

#Quadrant Marker
plt.text(x=2, y=-1, s="Q4",alpha=0.7,fontsize=14, color='b')
plt.text(x=-1, y=-1, s="Q3",alpha=0.7,fontsize=14, color='b')
plt.text(x=-1, y=1, s="Q2", alpha=0.7,fontsize=14, color='b')
plt.text(x=2, y=1, s="Q1", alpha=0.7,fontsize=14, color='b')

#Mean values
plt.axhline(y=school_earnings_sports_group_norm['6yr_mean_earnings'].mean(), colo
plt.axvline(x=school_earnings_sports_group_norm['Grand Total Expenses'].mean(), c

plt.savefig('blog_2.png')
plt.show()
```

```
posx and posy should be finite values
posx and posy should be finite values
posx and posy should be finite values
posx and posy should be finite values
```



Looking at this quadrant graph, in Q2, we see some of the smaller schools, with below average spending, actually have above average 6 year earnings. This includes schools like Rice, Tulane, and San Jose State. The most interesting quadrant is Q4, where we have schools that spent above on sports expenditures, but have students with below average earnings. This quadrant includes schools like Alabama, Ohio State, Florida State, Arizona, and Penn State. Interestingly, Penn State and Ohio State are above average spenders on academics as well, but are still producing below average earnings. We'll explore to what effect, if any, does sports expenditures have on earnings.

In [21]:
```
school_earnings_sports_group.corr()
```

Out[21]:

	6yr_mean_earnings	Grand Total Expenses
6yr_mean_earnings	1.0000	0.4609
Grand Total Expenses	0.4609	1.0000

In [22]:
```
school_earnings_academics_group = pd.DataFrame(ncaa_df.groupby('INSTNM').agg({'6y
                                                                  'Total Academic
#as_index=False
school_earnings_academics_group.reset_index(inplace = True)
fig = px.scatter(school_earnings_academics_group, x='Total Academic Spending (Uni
                  title = 'Avg. 6 year mean earnings and Academic Expenditures, 2
fig.show()
```

In [23]:
```
school_earnings_academics_group.corr()
```

Out[23]:

	6yr_mean_earnings	Total Academic Spending (University-Wide)
6yr_mean_earnings	1.000000	0.614788
Total Academic Spending (University-Wide)	0.614788	1.000000

Unsurprisngly, we see a stronger correlation between academic spending and 6 year average earnings. Let's look at the top 10 average earning schools.

In [24]:
```
earnings = school_earnings_group.sort_values('6yr_mean_earnings', ascending = Fal
earnings.drop(columns = 'index', inplace = True)
display(earnings.head(10))

earnings[(earnings['INSTNM'] == 'University of Alabama') | (earnings['INSTNM'] ==
```

	INSTNM	6yr_mean_earnings
0	Stanford University	95775.0
1	Duke University	87050.0
2	Boston College	64925.0
3	University of Notre Dame	63650.0
4	University of Southern California	62950.0
5	Rice University	61150.0
6	Georgia Institute of Technology	60375.0
7	Vanderbilt University	60075.0
8	University of Michigan	56100.0
9	Wake Forest University	53400.0

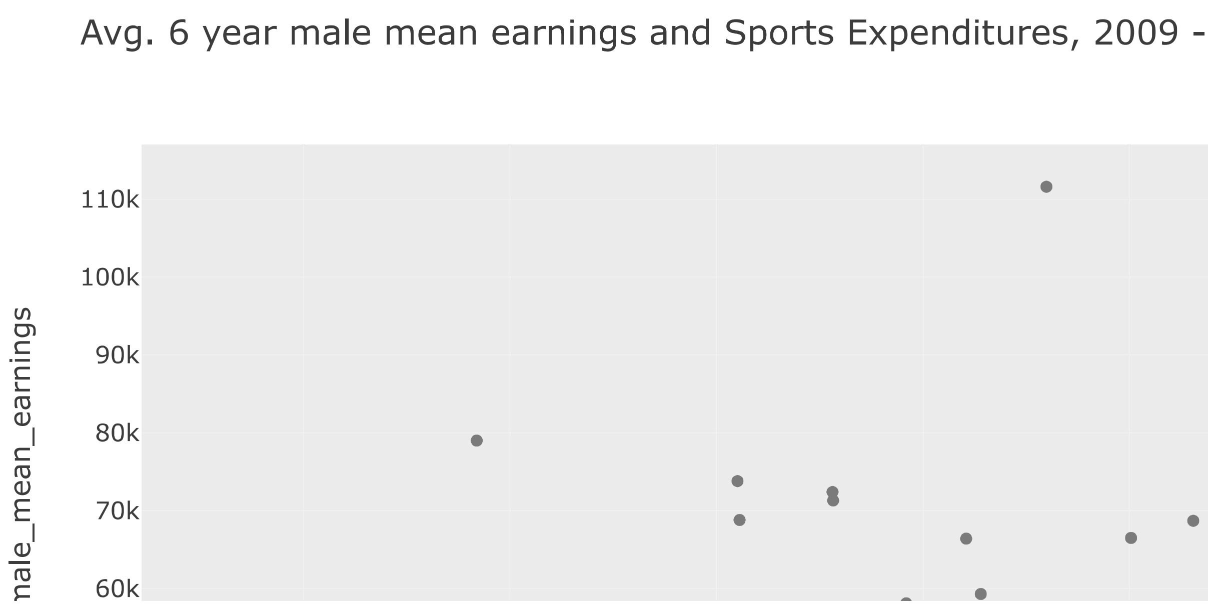
Out[24]:

	INSTNM	6yr_mean_earnings
14	University of Texas at Austin	49625.0
51	University of Tennessee	40825.0
55	Auburn University	40300.0
75	University of Alabama	37425.0

Of the schools in the top 10 of sports expenditures, only Michigan produces earnings in the top 10. Alabama, which spends the 8th most on sports, is ranked 76th in 6 year average earnings. On the other hand, UT at Austin spends the most on sports but is ranked 15th in average earnings, likely because the school is ranked 11th in academic spending. In dollar terms, this equates to a nearly $12K difference in average earnings.

Now let's look at earnings by gender. We know that women receive only percentage of what mean earn in the workplace, but does university spending have any connection to how men and women perform upon graduating? Can schools shrink the gap by spending more in places than others?

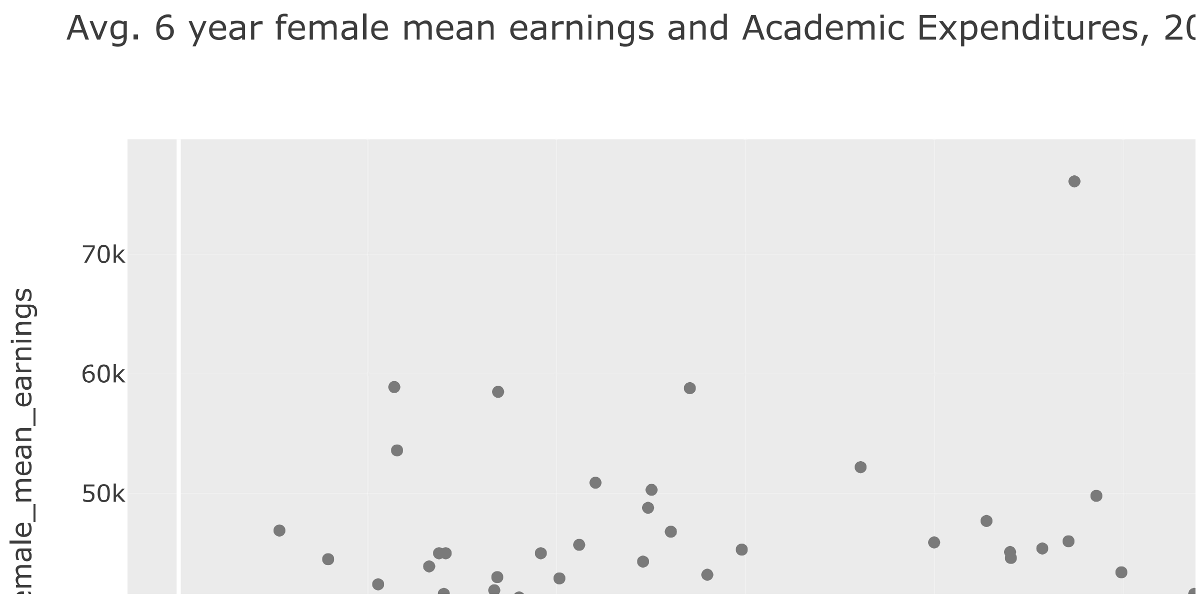
In [25]:
```
school_earnings_male_sports_group = pd.DataFrame(prediction_df.groupby('INSTNM').
                                                                  'Grand Total Exp
#as_index=False
school_earnings_male_sports_group.reset_index(inplace = True)
fig = px.scatter(school_earnings_male_sports_group, x='Grand Total Expenses', y =
                  title = 'Avg. 6 year male mean earnings and Sports Expenditures
fig.show()
```



In [26]:
```
school_earnings_female_sports_group = pd.DataFrame(prediction_df.groupby('INSTNM'
                                                                     'Grand Total Exp

school_earnings_female_sports_group.reset_index(inplace = True)

fig = px.scatter(school_earnings_female_sports_group, x='Grand Total Expenses', y
                  title = 'Avg. 6 year female mean earnings and Sports Expenditur
fig.show()
```

In [27]:
```
school_earnings_male_academic_group = pd.DataFrame(prediction_df.groupby('INSTNM'
                                                                          'Total Academic 
#as_index=False
school_earnings_male_academic_group.reset_index(inplace = True)
fig = px.scatter(school_earnings_male_academic_group, x='Total Academic Spending 
                 y = '6yr_male_mean_earnings',
                  title = 'Avg. 6 year male mean earnings and Academic Expenditur
fig.show()
```

In [28]:
```
school_earnings_female_academic_group = pd.DataFrame(prediction_df.groupby('INSTN
                                                                  'Total Academic
#as_index=False
school_earnings_female_academic_group.reset_index(inplace = True)
fig = px.scatter(school_earnings_female_academic_group, x='Total Academic Spendir
                 y = '6yr_female_mean_earnings',
                  title = 'Avg. 6 year female mean earnings and Academic Expendit
fig.show()
```



In [29]:
```
school_earnings_male_sports_group.corr()
```

Out[29]:

	6yr_male_mean_earnings	Grand Total Expenses
6yr_male_mean_earnings	1.000000	0.439589
Grand Total Expenses	0.439589	1.000000

In [30]:
```
school_earnings_female_sports_group.corr()
```

Out[30]:

	6yr_female_mean_earnings	Grand Total Expenses
6yr_female_mean_earnings	1.000000	0.474252
Grand Total Expenses	0.474252	1.000000

In [31]:
```
school_earnings_male_academic_group.corr()
```

Out[31]:

	6yr_male_mean_earnings	Total Academic Spending (University-Wide)
6yr_male_mean_earnings	1.000000	0.595898
Total Academic Spending (University-Wide)	0.595898	1.000000

In [32]:
```
school_earnings_female_academic_group.corr()
```

Out[32]:

	6yr_female_mean_earnings	Total Academic Spending (University-Wide)
6yr_female_mean_earnings	1.000000	0.614876
Total Academic Spending (University-Wide)	0.614876	1.000000

It appears as though female earnings fluctuate slighlty more with spending than male earnings do, although not by much. Female earnings are of a smaller magnitude, so they may be more elastic than male earnings. Any increase in school spending would have a greater impact on female post-graduate earnings than it would on the male students.

In [33]:
```
school_earnings_male_group = pd.DataFrame(prediction_df.groupby('INSTNM').agg({'6
                                                                  }))
school_earnings_female_group = pd.DataFrame(prediction_df.groupby('INSTNM').agg({
                                                                  }))
```

In [34]:
```
male_earnings = school_earnings_male_group.sort_values('6yr_male_mean_earnings',
male_earnings
```

Out[34]:

INSTNM	6yr_male_mean_earnings
Duke University	111600
Stanford University	109200
Rice University	79000
University of Notre Dame	78700
Wake Forest University	73800
Vanderbilt University	72400
Boston College	71300
Georgia Institute of Technology	68800
University of Southern California	68700
University of Virginia	66500

In [35]:
```
female_earnings = school_earnings_female_group.sort_values('6yr_female_mean_earni
female_earnings
```

Out[35]:

INSTNM	6yr_female_mean_earnings
Duke University	76100
Stanford University	75800
Boston College	58900
Georgia Institute of Technology	58800
University of Notre Dame	58500
University of Southern California	54900
Rice University	53600
Vanderbilt University	52200
Wake Forest University	50900
University of Miami	50300

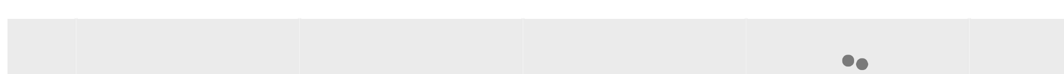
We have nearly identical lists, with just Miami and Virginia differing between mens and womens earnings. The pay gap is clear in these lists. Male earnings at Virigina are larger than female earnings at all but two schools in the female top 10 list. Further analysis will be done in this notebook to see what features have the largest impact on male and female earnings.

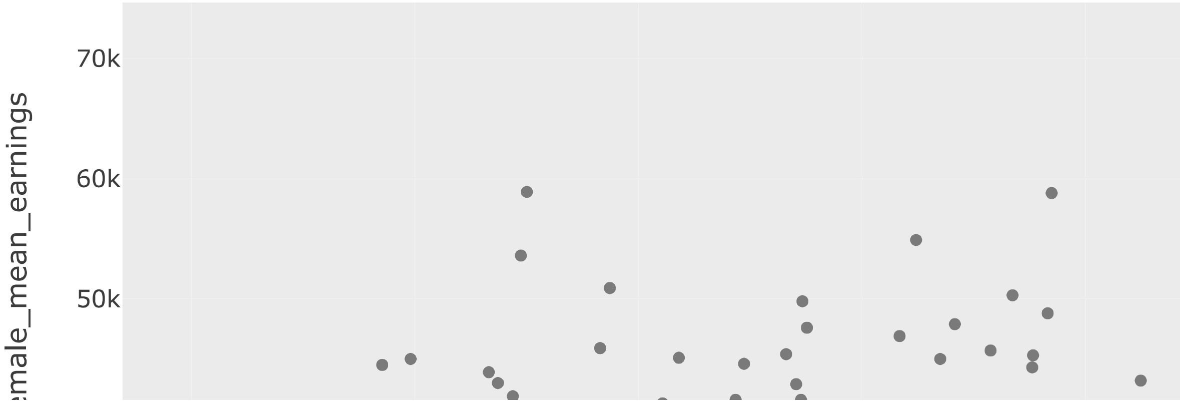
Next, let's look at if schools that spend more on women's sports have better earnings outcomes amongst its female graduates

In [36]:
```
school_earnings_female_support_group = pd.DataFrame(prediction_df.groupby('INSTNM
                                                                      " Women's Team Average Annual Ins
                                                                        "Women's Team Average Annual Ir
                                                                        "Women's Team Athletic Student
                                                                          "Total Women's Team Operating
                                                                      ' Womens_opex_per_participant ':

#as_index=False
school_earnings_female_support_group.reset_index(inplace = True)

X_cols =  [c for c in school_earnings_female_support_group.columns.to_list() if c

for col in X_cols:
    fig = px.scatter(school_earnings_female_support_group, x=col,
                 y = '6yr_female_mean_earnings',
                  title = 'Avg. 6 year female mean earnings and' + ' ' + col + '
    fig.show()
```

Avg. 6 year female mean earnings and Women's Team Average A

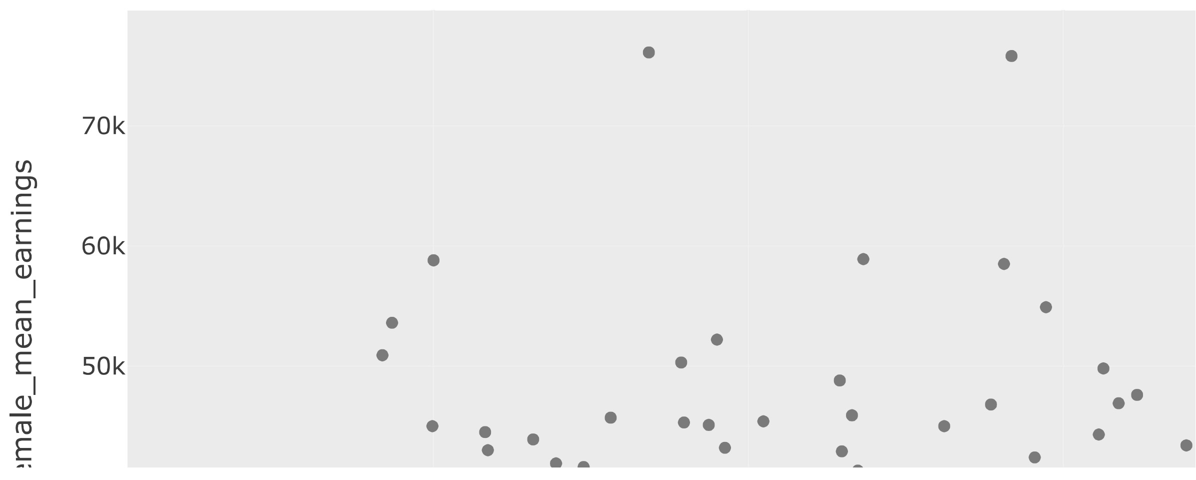
Avg. 6 year female mean earnings and Women's Team Average Ar



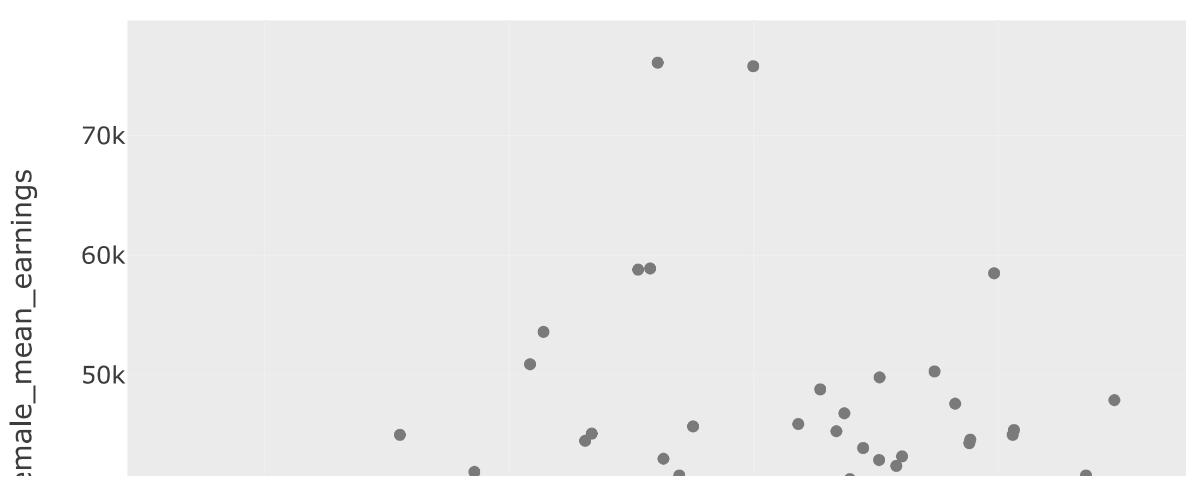


Avg. 6 year female mean earnings and Women's Team Athletic Stu

Avg. 6 year female mean earnings and Total Women's Team Opera



Avg. 6 year female mean earnings and Womens_opex_per_partic



In [37]:
```
school_earnings_female_support_group.corr()
```

Out[37]:

	6yr_female_mean_earnings	Women's Team Average Annual Institutional Salary per Head Coach	Women's Team Average Annual Institutional Salary per FTE	Women's Team Athletic Student Aid	Won - Oper Expe
6yr_female_mean_earnings	1.000000	0.438073	0.449611	0.693509	0.38
Women's Team Average Annual Institutional Salary per Head Coach	0.438073	1.000000	0.990128	0.574087	0.74
Women's Team Average Annual Institutional Salary per FTE	0.449611	0.990128	1.000000	0.585120	0.74
Women's Team Athletic Student Aid	0.693509	0.574087	0.585120	1.000000	0.66
Total Women's Team Operating Expenses	0.380033	0.741157	0.749117	0.669506	1.00
Womens_opex_per_participant	0.257935	0.722622	0.736643	0.436896	0.82

With all of the women's sports-specific features, we see a positive, linear relationship with average 6 year female earnings. The variable that has the strongest correlation is Women's team athletic student aid. What should be noted is that average female earnings are strongly correlated (.61) with total academic spending, and slightly less so with total sports expenditures (.47).

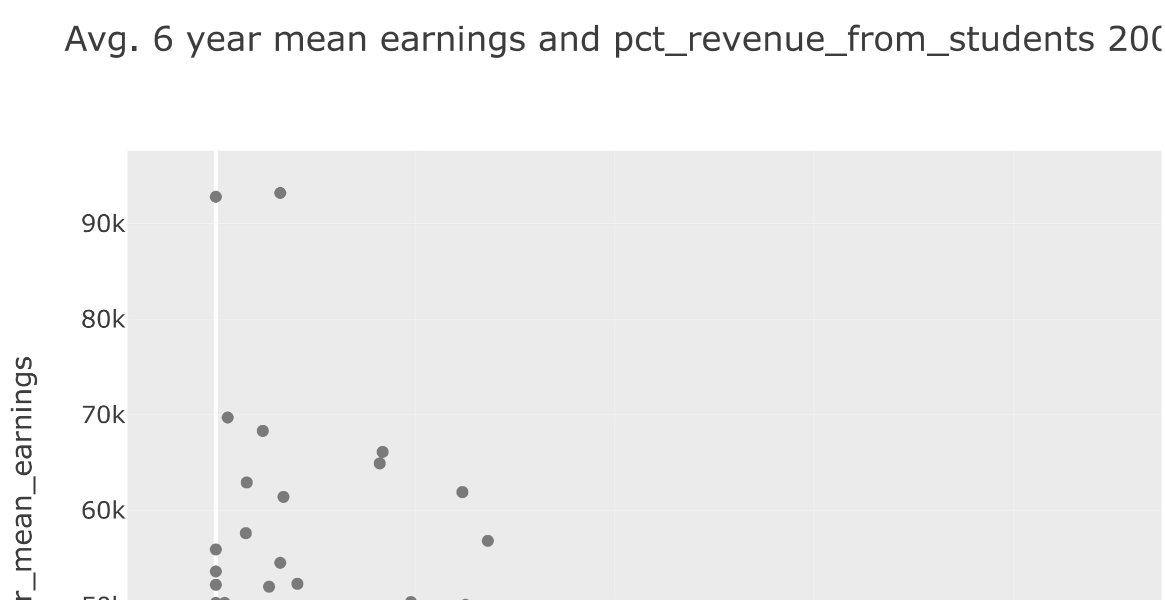
It is encouraging however, to see that schools willing to support their female students (and student-athletes) more result in female graduates earning more after they graduate. As mentioned earlier, we'll run regression analysis specifically on men's and women's earnings to try to narrow down causal relationships.

Now, let's take a look at how sports expenses relate to debt loads, and how debt loads relate to earnings

In [38]:
```
ncaa_df.columns.to_list()
```

Out[38]:
```
['IPEDS_ID',
 'INSTNM',
 'Year',
 'NCAA Subdivision',
 'FBS Conference',
 'Excess Transfers Back',
 'Other Expenses',
 'Medical',
 'Competition Guarantees',
 'Recruiting',
 'Game Expenses and Travel',
 'Facilities and Equipment',
 'Coaches Compensation',
 'Support and Admin Compensation w/Severance',
 'Athletic Student Aid',
 'Other Revenue',
 'Corporate Sponsorship, Advertising, Licensing',
 'Donor Contributions',
 'Competition Guarantees_revenue',
```

In [39]:
```
debt_to_earnings_group = pd.DataFrame(prediction_df.groupby('INSTNM').agg({'6yr_m
                                                 'pct_revenue_from_students':'mear
                                                  'tuition_rev_per_student':'mean'
                                                   'fed_loan_rate':'mean',
                                                   'median_male_debt':'mean',
                                                     'median_debt_completers':'mea
                                                 'median_loan_principal':'mean',
                                                 'avg_annual_cost_attendance':'mea
                                                 'CUML_DEBT_P90':'mean',
                                                'Grand Total Expenses':'sum'}))
#as_index=False
debt_to_earnings_group.reset_index(inplace = True)

X_cols =  [c for c in debt_to_earnings_group.columns.to_list() if c not in ['INST

#first mean earnings
for col in X_cols:
    fig = px.scatter(debt_to_earnings_group, x=col,
                 y = '6yr_mean_earnings',
                  title = 'Avg. 6 year mean earnings and' + ' ' + col + ' ' + '20
    fig.show()
```



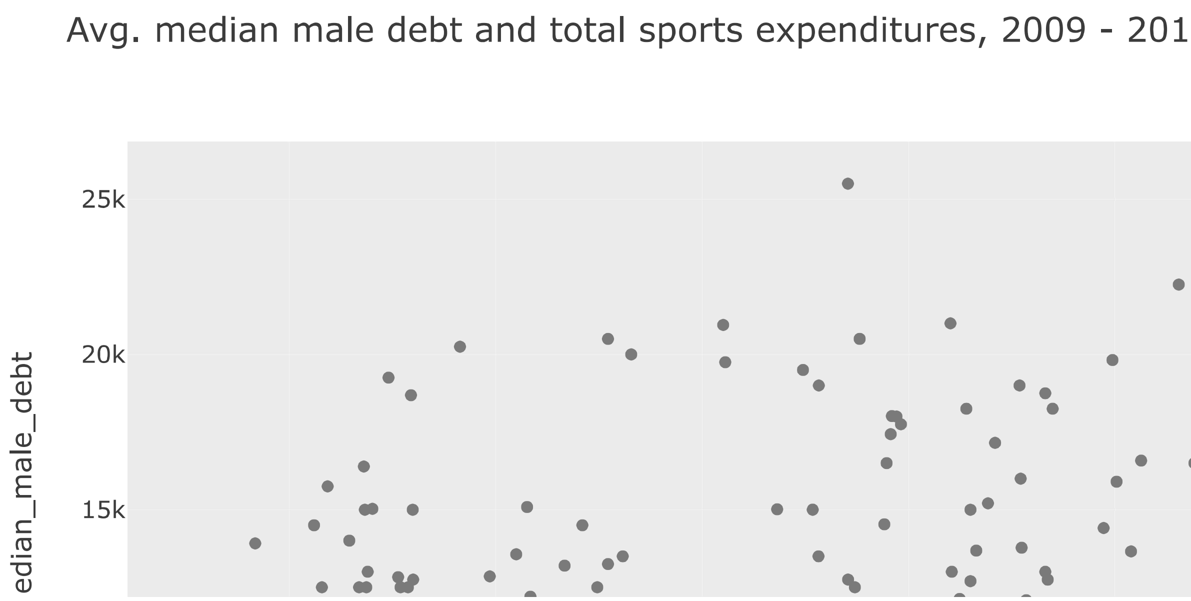
In [40]:
```
debt_to_earnings_group.corr()
```

Out[40]:

	6yr_mean_earnings	pct_revenue_from_students	tuition_rev_per_student
6yr_mean_earnings	1.000000	-0.334801	0.699340
pct_revenue_from_students	-0.334801	1.000000	-0.371392
tuition_rev_per_student	0.699340	-0.371392	1.000000
fed_loan_rate	-0.623857	0.389584	-0.391583
median_male_debt	0.219186	-0.207061	0.503258
median_debt_completers	-0.455785	0.071917	0.002935
median_loan_principal	0.172544	-0.176398	0.475545
avg_annual_cost_attendance	0.733156	-0.229312	0.878120
CUML_DEBT_P90	-0.487523	0.170094	-0.199852
Grand Total Expenses	0.460158	-0.590243	0.421565

6 year average earnings have some interesting and illustrative relationships with these cost of attendance and debt variables. Generally, earnings have negative relationships with debt features, like the federal loan rate, the median debt of graduates, and the cumulative debt in the 90th percentile. It also has somewhat strong positive relationships with the cost of attendance and total athletic expenses. These positive relationships could be the result of these variables signaling the overall resources a school has. A more expensive school likely has better resources, however we see that tuition revenue per student is positively correlated with median debt and median loan principal, possibly indicating that students who owe more in tuition are forced to take out larger loans.

In [41]:
```
sports_expeneses_debt_group = pd.DataFrame(prediction_df.groupby('INSTNM').agg({'
                                                                    'median_male_
                                                                    'median_debt_
                                                                    'median_loan_
                                                                    'avg_annual_c
                                                                    'CUML_DEBT_PS
#as_index=False
sports_expeneses_debt_group.reset_index(inplace = True)
fig = px.scatter(sports_expeneses_debt_group, x='Grand Total Expenses',
                 y = 'median_male_debt',
                  title = 'Avg. median male debt and total sports expenditures, 2
fig.show()
```



In [42]:
```
sports_expeneses_debt_group.corr()
```

Out[42]:

	Grand Total Expenses	median_male_debt	median_debt_completers	median_loan_p
Grand Total Expenses	1.000000	0.395350	-0.091294	0
median_male_debt	0.395350	1.000000	0.564977	0
median_debt_completers	-0.091294	0.564977	1.000000	0
median_loan_principal	0.361801	0.986735	0.627998	1
avg_annual_cost_attendance	0.270159	0.366298	-0.115313	0
CUML_DEBT_P90	-0.357981	0.136134	0.696863	0

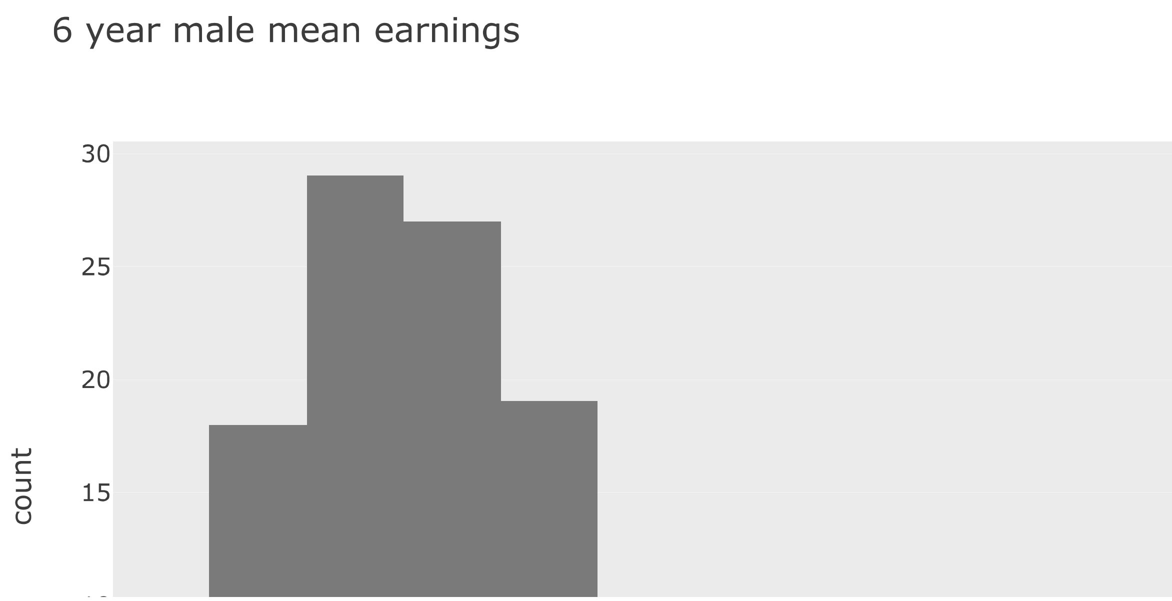
There turns out to be weaker relationships between total sports expenditures and debt loads on students.

Let's now look at the overall distributions of 6 year average earnings in our data.

In [43]:
```
fig = px.histogram(prediction_df, x="6yr_mean_earnings",
                   title = '6 year mean earnings')
fig.show()
```

In [44]:
```
fig = px.histogram(prediction_df, x="6yr_female_mean_earnings",
                   title = '6 year female mean earnings')
fig.show()
```

In [45]:
```
fig = px.histogram(prediction_df, x="6yr_male_mean_earnings",
                   title = '6 year male mean earnings')
fig.show()
```

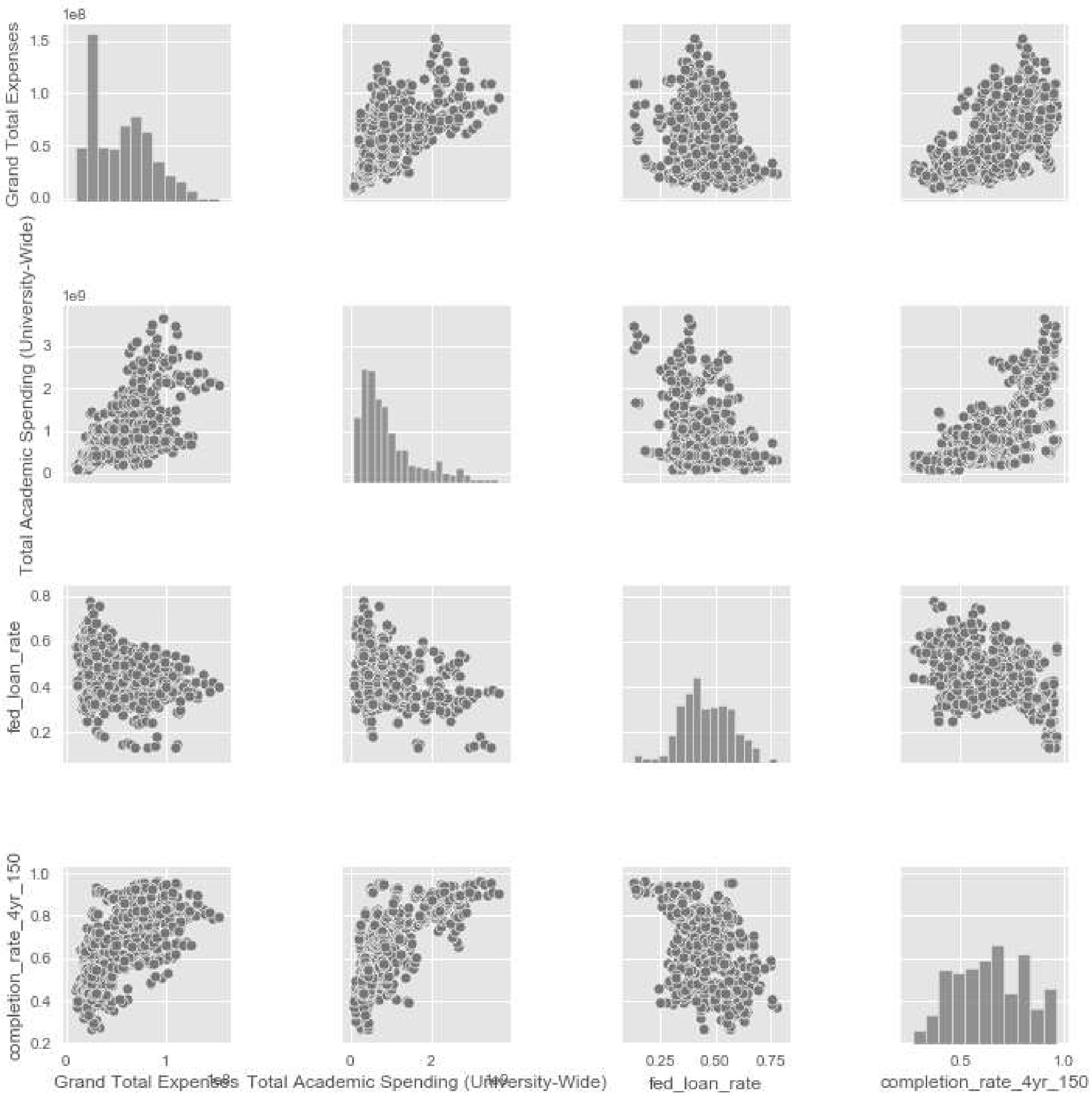


Earnings are right skewed, so logging our data will be useful when running regressions.

Now, let's look at some pairplots to get an idea of the distributions of our explanatory variables

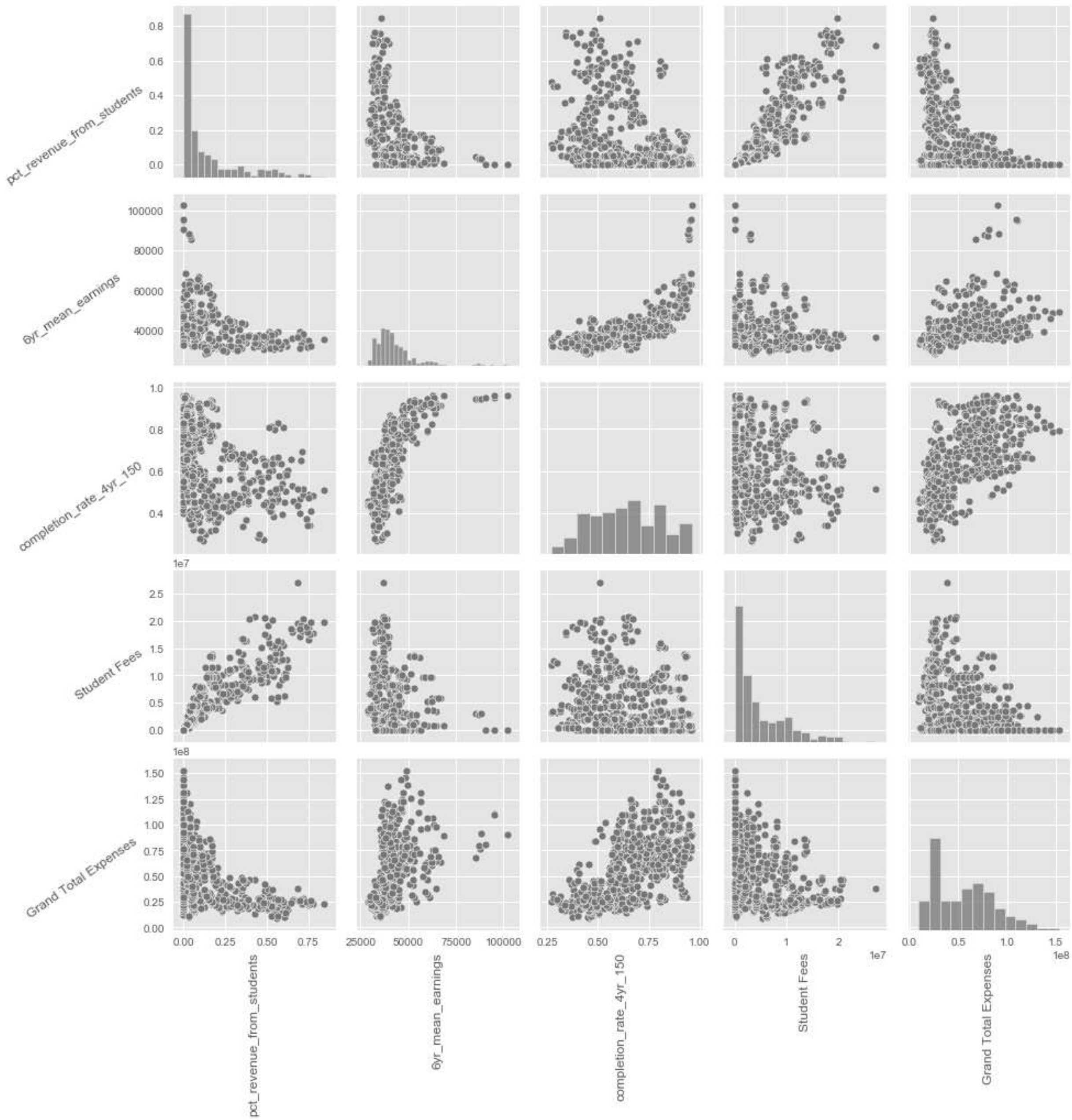
In [46]:
```
cols_to_use = ['Grand Total Expenses', 'Total Academic Spending (University-Wide)
               'fed_loan_rate','completion_rate_4yr_150']
sns.pairplot(ncaa_df[cols_to_use])
```

Out[46]: <seaborn.axisgrid.PairGrid at 0x2f196aeddd8>



Total academic spending and total expenses have a clear positive, linear relationship, and both are also skewed to the right. We also see a positive relationship between completion rate and total sports expenses and and academic spending. More spending can be inferred as more student support, so students at wealthier schools graduate at higher rates.

In [47]:
```
cols_to_use = ["pct_revenue_from_students",
               "6yr_mean_earnings",
               'completion_rate_4yr_150',"Student Fees",'Grand Total Expenses']
pairplot = sns.pairplot(ncaa_df[cols_to_use])
for ax in pairplot.axes.flatten():
    # rotate x axis labels
    ax.set_xlabel(ax.get_xlabel(), rotation = 90)
    # rotate y axis labels
    ax.set_ylabel(ax.get_ylabel(), rotation = 35)
    # set y labels alignment
    ax.yaxis.get_label().set_horizontalalignment('right')
```



Modeling

I will take at least three different approaches to modeling this data:

1. Using only sports-related data (expenses, participants, conference, etc.)
2. Using only non-sports data (family income, debt accumulated, cost of attendance, demographics, etc.)
3. Using a blend of both types of data

I'm taking this approach so we can isolate the effects that sports have on our target variable, 6 year mean earings. By comparing an all sports model to a no sports model, we can look at what better predicts future earnings. The 3rd model, which will use the best features overall, will likely be the most accurate predictor of future earnings.

Modeling will have two goals: producing accurate predictions of future earnings and diagnosing what features most impact future earnings, positive and negatively.

For this to work, I'll need to drop the columns from the NCAA dataset with excessive nulls. All private schools are missing the sports spending columns.

First crappy sports-only model

In [48]:
```
ncaa_df.columns.to_list()
```

Out[48]:
```
['IPEDS_ID',
 'INSTNM',
 'Year',
 'NCAA Subdivision',
 'FBS Conference',
 'Excess Transfers Back',
 'Other Expenses',
 'Medical',
 'Competition Guarantees',
 'Recruiting',
 'Game Expenses and Travel',
 'Facilities and Equipment',
 'Coaches Compensation',
 'Support and Admin Compensation w/Severance',
 'Athletic Student Aid',
 'Other Revenue',
 'Corporate Sponsorship, Advertising, Licensing',
 'Donor Contributions',
 'Competition Guarantees_revenue',
```

In [49]:
```
#first, going to remove columns with many nulls
cols_to_drop = ['Excess Transfers Back',
 'Other Expenses',
 'Medical',
 'Competition Guarantees',
 'Recruiting',
 'Game Expenses and Travel',
 'Facilities and Equipment',
 'Coaches Compensation',
 'Support and Admin Compensation w/Severance',
 'Athletic Student Aid',
 'Other Revenue',
 'Corporate Sponsorship, Advertising, Licensing',
 'Donor Contributions',
 'Competition Guarantees_revenue',
 'NCAA/Conference Distributions, Media Rights, and Post-Season Football',
 'Ticket Sales',
 'Institutional/Government Support',
'Total Institutional/Government Support and Student Fees',
'Total Football Spending',
 'Total Football Coaching Salaries',
 'Athletics Related Debt',
 'Annual Debt Service, Leases and Rental Fees on Athletic Facilities',]
```

In [50]:
```
ncaa_df2 = ncaa_df.drop(columns = cols_to_drop)
```

In [51]:
```
#creating my grouped dataset with all features I may want to use
ncaa_grouped_sports = ncaa_df2.groupby('INSTNM',
                    as_index=False).agg({ 'Student Fees':'sum',
                                         "Men's Team Number of Head Coaches Inclu
                                         "Women's Team Number of Head Coaches Inc
                                         'Grand Total Revenue':'sum',
                                         'Grand Total Expenses':'sum',
                                         ' Total_expenses_men_per_participant ':'
                                         'Total_expenses_women_per_participant':'
                                        })

ncaa_grouped_sports['pct_revenue_from_students'] = ncaa_grouped_sports['Student F
```

In [52]:
```
ncaa_grouped_sports.head()
```

Out[52]:

	INSTNM	Student Fees	Men's Team Number of Head Coaches Included in Average	Women's Team Number of Head Coaches Included in Average	Grand Total Revenue	Grand Total Expenses	Total_expenses_men_
0	Appalachian State University	5.586136e+07	9.0	9.000000	113592089.0	113592089.0	
1	Arizona State University	0.000000e+00	7.0	10.333333	396442258.0	395625799.0	
2	Auburn University	2.953605e+07	7.0	10.000000	652739784.0	616976142.0	
3	Ball State University	6.298556e+07	7.0	10.000000	132898085.0	132898085.0	
4	Baylor University	0.000000e+00	6.0	8.833333	453245247.0	453245247.0	

In [53]:
```
prediction_df2 = prediction_df.drop(columns = cols_to_drop)
```

In [54]:
```
#creating separate data set with prediction columns from 2015
prediction_df_reg = prediction_df2[['INSTNM','6yr_mean_earnings','6yr_female_mean
                                    '6yr_male_mean_earnings','6yr_median_earnings'
```

In [55]:
```
prediction_df_reg.head()
```

Out[55]:

	INSTNM	6yr_mean_earnings	6yr_female_mean_earnings	6yr_male_mean_earnings	6yr_mediai
0	Appalachian State University	34300	32000	36800	
1	Arizona State University	42200	38900	46000	
2	Auburn University	43100	38200	47900	
3	Ball State University	35300	33500	37400	
4	Baylor University	46600	42400	52700	

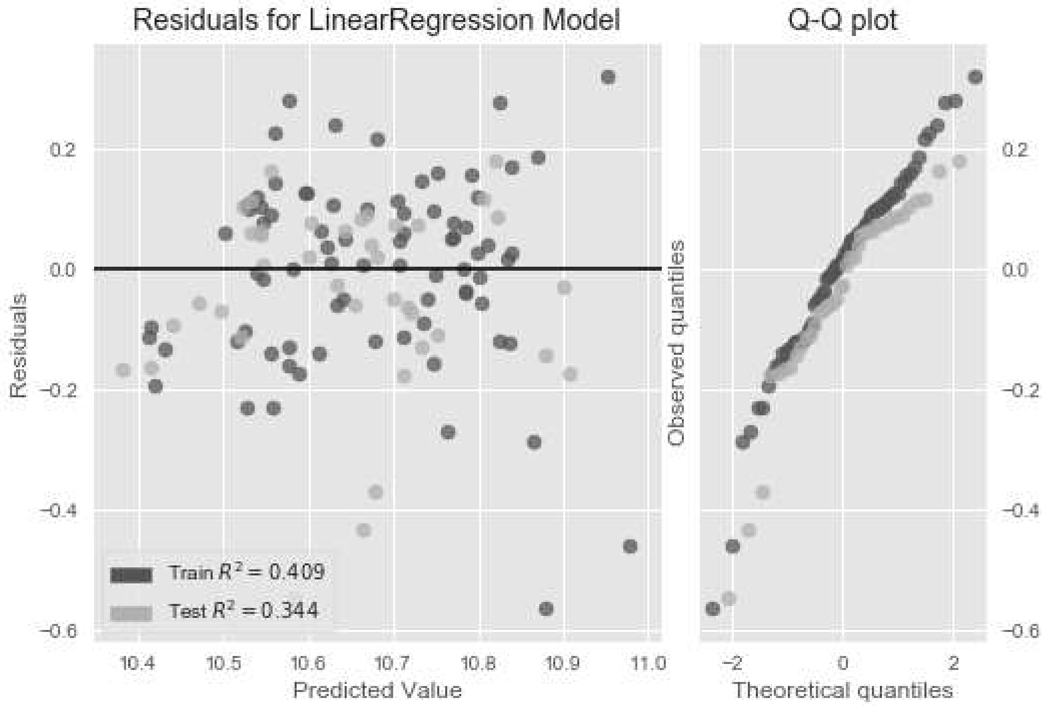
In [56]:
```
sports_merge = ncaa_grouped_sports.merge(prediction_df_reg, on='INSTNM')
```

In [57]:
```
sports_merge.columns
```

Out[57]:
```
Index(['INSTNM', 'Student Fees',
       'Men's Team Number of Head Coaches Included in Average',
       'Women's Team Number of Head Coaches Included in Average',
       'Grand Total Revenue', 'Grand Total Expenses',
       ' Total_expenses_men_per_participant ',
       'Total_expenses_women_per_participant', 'pct_revenue_from_students',
       '6yr_mean_earnings', '6yr_female_mean_earnings',
       '6yr_male_mean_earnings', '6yr_median_earnings'],
      dtype='object')
```

In [58]:
```
from project_functions_capstone_7 import *
```

In [59]:
```
ms_linear_regression(sports_merge, scale = 'yes')
```




```
Train Score: 0.40915139161624836
Test Score: 0.3444993514664233
---
Train RMSE:  0.15472179583919973
Test RMSE:  0.15954812091993262
---
Unlogged Train RMSE:  8322.728024794216
Unlogged Test RMSE:  7977.474320380442
---
```

Out[59]:

OLS Regression Results

Dep. Variable:	6yr_mean_earnings	**R-squared:**	0.409
Model:	OLS	**Adj. R-squared:**	0.341

Method:	Least Squares	**F-statistic:**	5.973
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	8.20e-06
Time:	11:29:55	**Log-Likelihood:**	34.881
No. Observations:	78	**AIC:**	-51.76
Df Residuals:	69	**BIC:**	-30.55
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.6763	0.019	573.183	0.000	10.639	10.713
Student Fees	0.0131	0.021	0.619	0.538	-0.029	0.055
Men's Team Number of Head Coaches Included in Average	0.0546	0.032	1.706	0.093	-0.009	0.118
Women's Team Number of Head Coaches Included in Average	0.0304	0.032	0.945	0.348	-0.034	0.094
Grand Total Revenue	-0.1642	0.297	-0.554	0.582	-0.756	0.427
Grand Total Expenses	0.1918	0.293	0.654	0.515	-0.393	0.777
Total_expenses_men_per_participant	0.0429	0.046	0.930	0.355	-0.049	0.135
Total_expenses_women_per_participant	0.0204	0.039	0.526	0.601	-0.057	0.098
pct_revenue_from_students	0.0030	0.021	0.140	0.889	-0.039	0.045

Omnibus:	13.782	**Durbin-Watson:**	2.221
Prob(Omnibus):	0.001	**Jarque-Bera (JB):**	16.980
Skew:	0.818	**Prob(JB):**	0.000205
Kurtosis:	4.597	**Cond. No.**	45.0

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

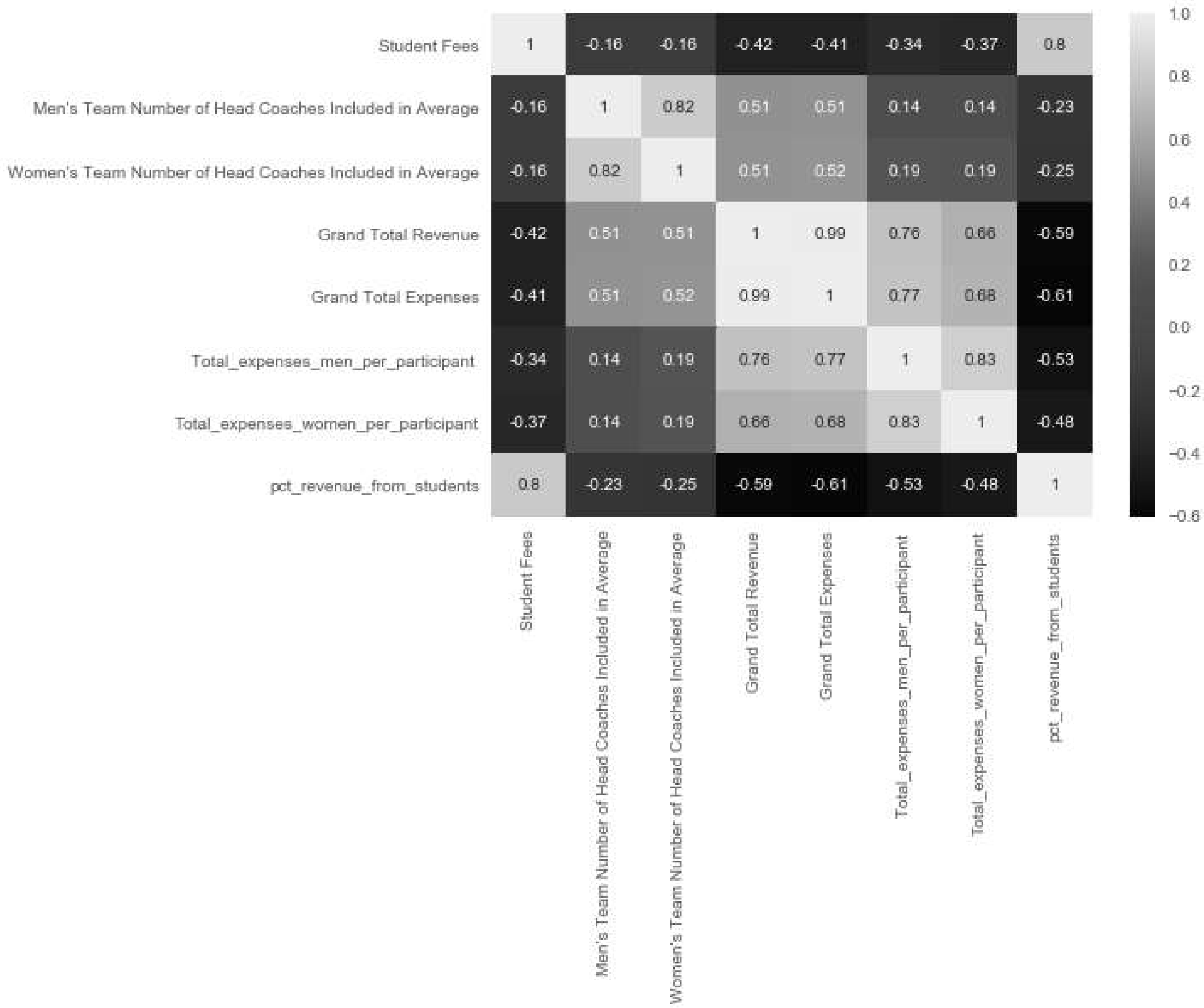
Our major issue here is that none of the x variables are statistically significant, likely because of multicollinearity. Additionally, the model overfits on the train data. Let's use a function to determine the collinearity of our x values.

In [60]:
```
ms_vif(sports_merge, nulls = 'yes')
```

Out[60]:

	VIF	features
0	6.157829	Student Fees
1	43.932129	Men's Team Number of Head Coaches Included in ...
2	51.341872	Women's Team Number of Head Coaches Included i...
3	235.013485	Grand Total Revenue
4	289.073667	Grand Total Expenses
5	30.757472	Total_expenses_men_per_participant
6	29.818787	Total_expenses_women_per_participant
7	6.245894	pct_revenue_from_students

In [61]:
```
#let's also take a look at a heatmap of correlations
X_cols = [c for c in sports_merge.columns.to_list() if c not in ['INSTNM','6yr_me
                                                                  '6yr_male_mean_earning
ax = sns.heatmap(sports_merge[X_cols].corr(), annot=True);
# need to manually set my ylim because of my version of matplotlib
ax.set_ylim(8, 0)
plt.show()
```



All variables have a VIF above 5, meaning that there is a lot of multicollinearity present in this model. Expenses are highly correlated with revenues, because, as non-profit institutions, Universities must spend all profits they make.

First crappy non-sports-only model

In [62]:
```
#creating my grouped dataset with all features I may want to use
ncaa_grouped_non_sports = ncaa_df2.groupby('INSTNM',
                    as_index=False).agg({ 'Total Academic Spending (University-Wi
                                        'ADM_RATE':'mean',
                                        'SAT_AVG':'sum',
                                         'avg_annual_cost_attendance':'sum',
                                         "instructional_expenditure_per_student":
                                         "fed_loan_rate":'mean',
                                         'FAMINC':'mean',
                                         'completion_rate_4yr_150':'mean',
                                         'median_debt_completers':'mean',
                                        })
```

In [63]:
```
non_sports_merge = ncaa_grouped_non_sports.merge(prediction_df_reg, on='INSTNM')
```

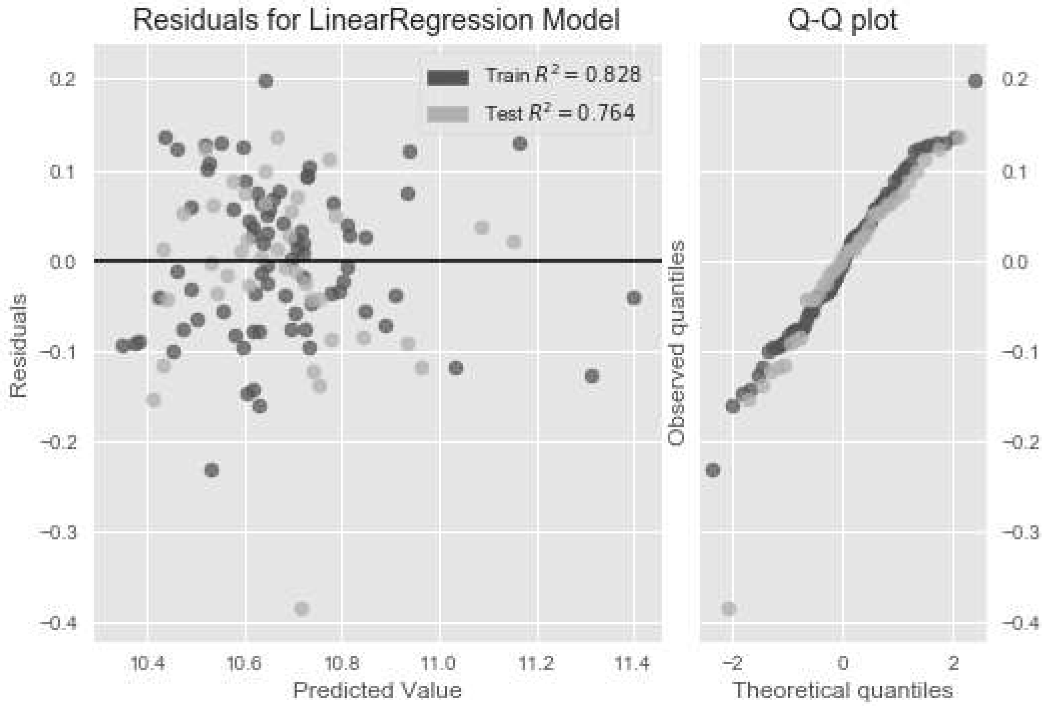
In [64]:
```
ms_vif(non_sports_merge, nulls = 'yes')
```

Out[64]:

	VIF	features
0	7.530147	Total Academic Spending (University-Wide)
1	17.677863	ADM_RATE
2	28.235212	SAT_AVG
3	22.441191	avg_annual_cost_attendance
4	9.034255	instructional_expenditure_per_student
5	61.828803	fed_loan_rate
6	64.397262	FAMINC
7	92.738592	completion_rate_4yr_150
8	126.046831	median_debt_completers

We see smaller VIF scores overall for these variables, but each score once again is over 5.

In [65]:
```
ms_linear_regression(non_sports_merge, scale = 'yes')
```




```
Train Score: 0.8282935089991963
Test Score: 0.7636477344511495
---
Train RMSE:  0.08340781525052811
Test RMSE:  0.09580430299160313
---
Unlogged Train RMSE:  3763.899231347773
Unlogged Test RMSE:  4713.1001182185755
---
```

Out[65]:

OLS Regression Results

Dep. Variable:	6yr_mean_earnings	**R-squared:**	0.828

Model:	OLS	**Adj. R-squared:**	0.806
Method:	Least Squares	**F-statistic:**	36.45
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	1.25e-22
Time:	11:29:57	**Log-Likelihood:**	83.076
No. Observations:	78	**AIC:**	-146.2
Df Residuals:	68	**BIC:**	-122.6
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.6763	0.010	1055.524	0.000	10.656	10.696
Total Academic Spending (University-Wide)	0.0262	0.020	1.319	0.191	-0.013	0.066
ADM_RATE	-0.0488	0.018	-2.665	0.010	-0.085	-0.012
SAT_AVG	-0.0268	0.014	-1.850	0.069	-0.056	0.002
avg_annual_cost_attendance	0.0225	0.018	1.242	0.219	-0.014	0.059
instructional_expenditure_per_student	0.0184	0.021	0.868	0.388	-0.024	0.061
fed_loan_rate	0.0132	0.018	0.716	0.477	-0.024	0.050
FAMINC	0.0853	0.023	3.756	0.000	0.040	0.131
completion_rate_4yr_150	0.0125	0.027	0.471	0.639	-0.040	0.065
median_debt_completers	-0.0852	0.020	-4.208	0.000	-0.126	-0.045

Omnibus:	0.418	**Durbin-Watson:**	2.140
Prob(Omnibus):	0.812	**Jarque-Bera (JB):**	0.581
Skew:	0.093	**Prob(JB):**	0.748
Kurtosis:	2.621	**Cond. No.**	8.04

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

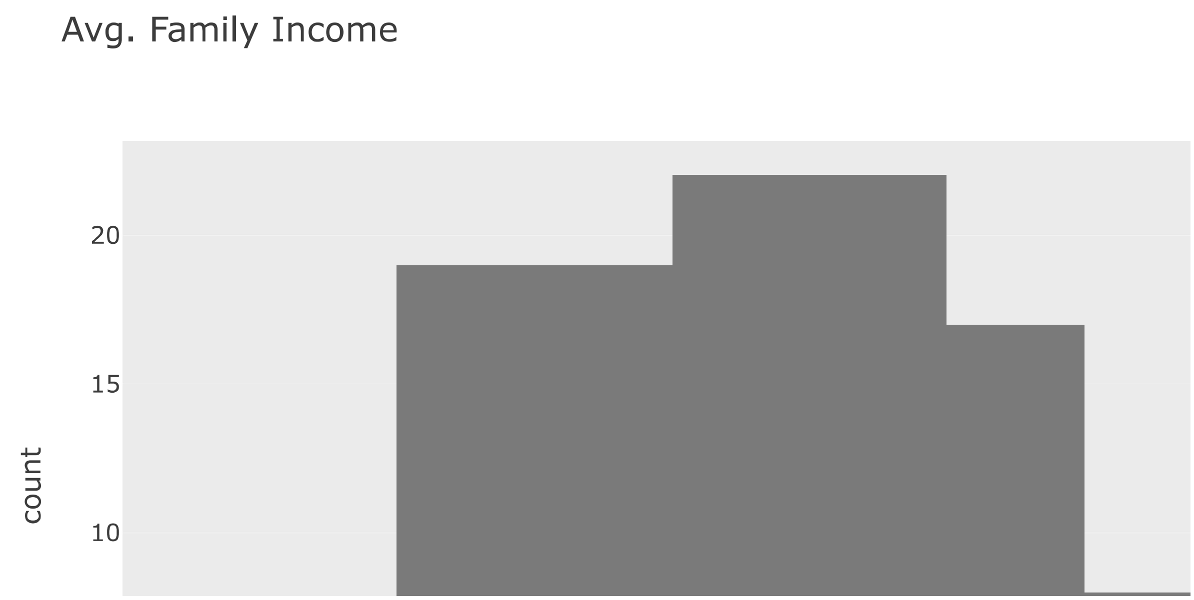
This model does better across the board, with improved RMSEs and almost double the r2. However, we do have a statistical significance problem.

We do see three statistically significant features: admission rate, median debt of graduates, and family income. Family income is positive, which makes sense given that the better off your family is, the easier time you'll have getting into more expensive schools and surround yourself with the best resources to succeed.

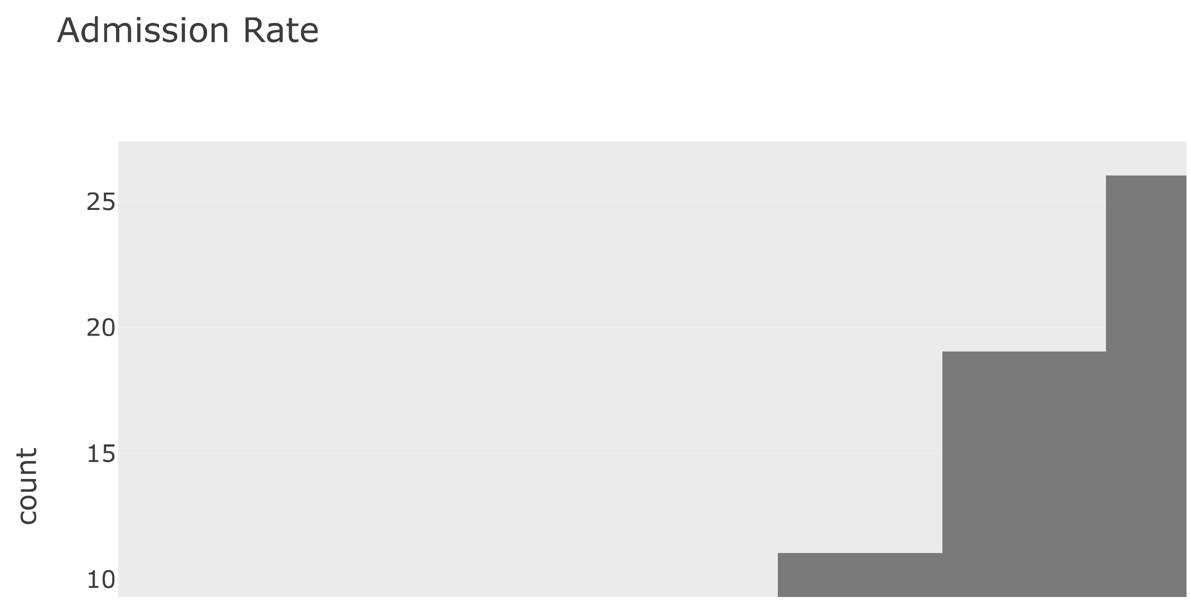
Admission rate has a negative coefficient, meaning that as admission rate increases, future earnings decrease, and vice versa. It appears as the selectivity of school goes down, it has an overall negative impact on its students. Let's see if this continues as we further refine the model.

Let's take at look at the distributions of these variables:

In [66]:
```
fig = px.histogram(ncaa_grouped_non_sports, x="FAMINC",
                   title = 'Avg. Family Income')
fig.show()
```



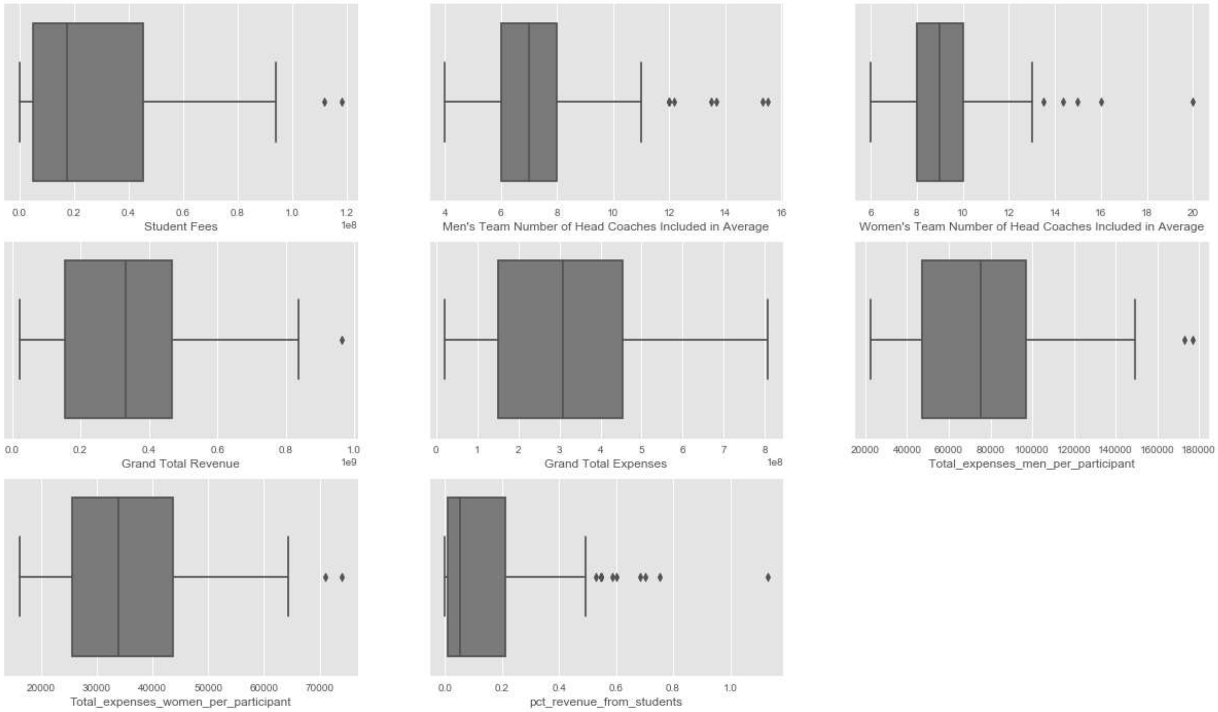
In [67]:
```
fig = px.histogram(ncaa_grouped_non_sports, x='ADM_RATE',
                   title = 'Admission Rate')
fig.show()
```



Family income follows more of a normal distribution, while admission rate is left skewed - most schools have an admission rate between 60% and 80%.

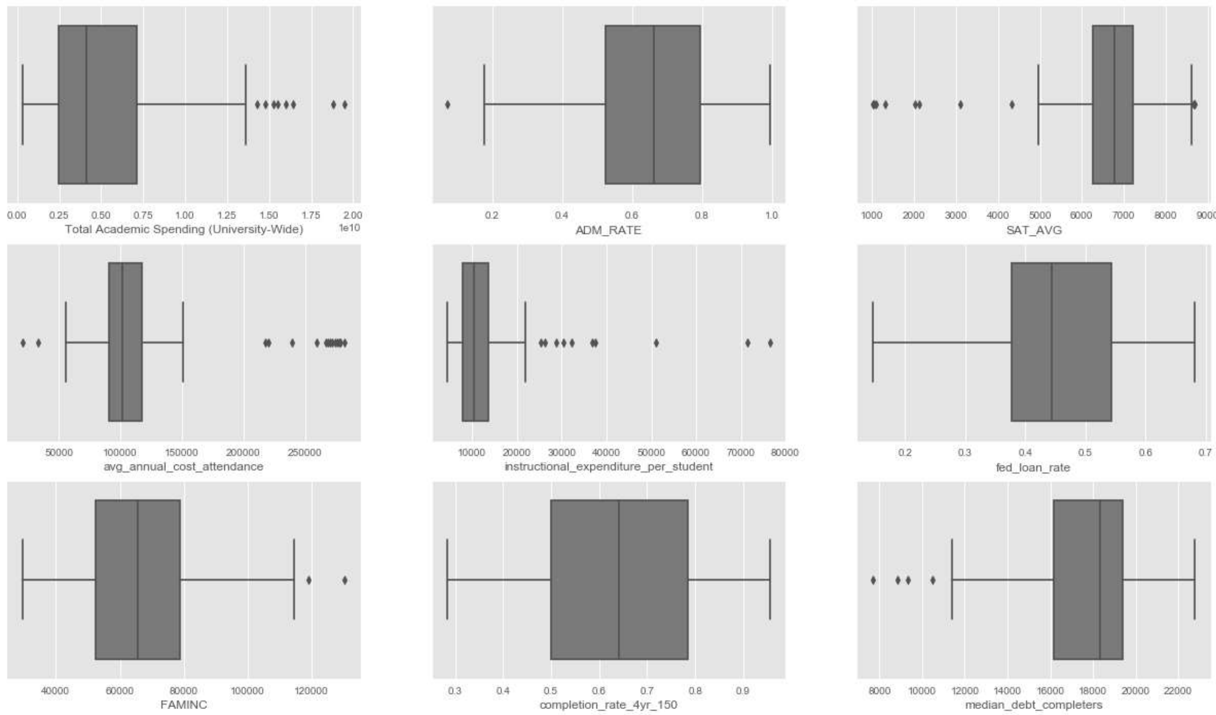
Before moving on, let's look to see what sort of outliers we're working with across sports and non-sports vars:

In [68]:
```
X_cols = [c for c in sports_merge.columns.to_list() if c not in ['INSTNM','6yr_me
                                                                   '6yr_male_mean_earning

ms_plot_univariate_panel(X_cols, sports_merge, sns.boxplot, 3)
```



We have significant outliers in athletic debt, number of head coaches, and pct revenue from students. Log transforming will help smooth these outliers, but removing some at the higher end would likely improve the model.

In [69]:
```
X_cols = [c for c in non_sports_merge.columns.to_list() if c not in ['INSTNM','6y
                                                                   '6yr_male_mean_earning
ms_plot_univariate_panel(X_cols, non_sports_merge, sns.boxplot, 3)
```



On the non-sports side, we have significant outliers in total academic spending, avg annual cost of attendance, instructional expenditure per student, and SAT average.

Fine-Tuning the Models

Next, I'll try a few different models to ascertain the most impactful and useful features. I will also now log the numeric values in addition to scaling them. The three primary models I'll be using are:

- **Lasso Regression**
- **Ridge Regression**
- **ElasticNet (combination of both Ridge and Lasso Regression)**

If needed, I'll also remove outliers which negatively affect the modeling.

Lasso Regression

Lasso regression performs L1 regularization (https://www.statisticshowto.com/lasso-regression/), which adds a penalty equal to the absolute value of the magnitude of coefficients. This method shrinks down coefficients, some to zero, which you can then remove from the model, making our model simpler and more interpretable. This will be useful with a dataset of over 100 features.

To deal with multicollinearity, the Lasso model gives one of the correlated features a large coefficient while the rest are zeroed or nearly zeroed. (https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net)

In [70]:
```
#creating a new grouped dataset with just the 2015 mean and median earnings colum
prediction_cols = ['6yr_mean_earnings', '6yr_median_earnings']

sum_cols_df = ncaa_df2[['INSTNM','Total Academic Spending (University-Wide)',
                                        'Grand Total Expenses',
                                        'Grand Total Revenue']]

sum_cols_grouped = sum_cols_df.groupby('INSTNM')[sum_cols_df.columns].sum()


X_cols = [c for c in ncaa_df2.columns.to_list() if c not in ['Total Academic Sper
                                        'Grand Total Expenses',
                                        'Grand Total Revenue']]

ncaa_grouped = ncaa_df2.groupby('INSTNM')[X_cols].mean().reset_index()
ncaa_grouped.drop(columns = prediction_cols, inplace = True)

ncaa_grouped_merge = ncaa_grouped.merge(prediction_df_reg, on='INSTNM')
ncaa_grouped_all_merge = ncaa_grouped_merge.merge(sum_cols_grouped, on='INSTNM' )
ncaa_grouped_all_merge.drop(columns = ['IPEDS_ID', 'Year'], inplace = True)

ncaa_grouped_all_merge.head()
```

Out[70]:

	INSTNM	Student Fees	Male Undergraduates	Female Undergraduates	Total Undergraduates	Football Total Participation	Tot Parti
0	Appalachian State University	9.310227e+06	7010.166667	7638.500000	14648.666667	109.500000	
1	Arizona State University	0.000000e+00	22417.000000	21461.333333	43878.333333	117.666667	
2	Auburn University	4.922676e+06	9282.333333	9203.000000	18485.333333	121.500000	
3	Ball State University	1.049759e+07	7062.833333	8764.833333	15827.666667	119.333333	
4	Baylor University	0.000000e+00	5256.333333	7341.500000	12597.833333	117.666667	

5 rows × 101 columns

In [71]:
```
lasso_model, cols = ms_lasso_regression(ncaa_grouped_all_merge, scale = 'yes')
```

```
Train R2: 0.9613143241124343
Test R2: 0.8706652122956692
*****
Train MSE: 0.0015673896227838234
Test MSE: 0.005022565580843139
*****
Train RMSE: 0.03959027182002952
Test RMSE: 0.07087006124481013
None
```

In [72]:
```
lasso_coefs = ms_eval_coefficients(lasso_model, cols)
```

```
Total number of coefficients:  94
Coefficients close to zero:  55
Intercept: 10.676291138768727

avg_annual_cost_attendance         0.068706
LOAN_EVER                          0.039499
1yr_declining_loan_balance         0.035905
pct_asian                          0.026992
median_female_debt                 0.022542
                                      ...
pct_women                         -0.022311
PELL_EVER                         -0.027005
num_male_4yr_completers           -0.031276
Men's Team Athletic Student Aid   -0.038798
median_debt_completers            -0.084023
Length: 94, dtype: float64
```

With the Lasso model, the data was slightly overfit on the training data, but the r2 values are both above .8. The model also shrunk 55 columns to zero, or close to zero, meaning we can drop these coefficients from the model. These coefficients are less important, either because they are correlated with a more powerful variable, or they themselves are only weakly related to the dependent variable, which in our case is 6 year mean earnings.

Let's now use these coefficients in a ridge regression model.

Ridge Regression

Ridge regression helps shrink large coefficients by adding a penalty term to the square of the magnitude of the coefficients. Large coefficients penalize the optimation function. The shrinkage helps reduce model complexity and multicollinearity. Ridge regression assumes the predictors are scaled, so we'll be sure to scale the X variables.

To deal with multicollinearity, [the Ridge model assigns similar coefficients to correlated predictors. (https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net)](https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net)

To begin, we'll create a dataset using the Lasso coefficients remaining, and use them to create the ridge model.

In [73]:
```
#removing coefficients that are 0 or close to 0
coef_df = pd.DataFrame(lasso_coefs)
coef_df.reset_index(inplace = True)
coef_df.rename(columns = {0:'coef', 'index':'feature'}, inplace = True)

coef_df['remove'] = coef_df.apply(lambda row: abs(row.coef) < 10**(-10), axis = 1
coef_df = coef_df[coef_df['remove'] == False]

coef_df.set_index('feature', inplace = True)
coef_df.drop(columns = 'remove', inplace = True)

coef_df.head()
```

Out[73]:

feature	coef
avg_annual_cost_attendance	0.068706
LOAN_EVER	0.039499
1yr_declining_loan_balance	0.035905
pct_asian	0.026992
median_female_debt	0.022542

In [74]:
```
#saving remaining columns as a list
coef_col_list = coef_df.index.to_list()
```

In [75]:
```
#don't overwrite this
ridge_model, cols = ms_ridge_regression(ncaa_grouped_all_merge, cols = coef_col_l
```

```
Train R2: 0.9458398454724065
Test R2: 0.9000704286141612
*****
Train MSE: 0.0021943539107766663
Test MSE: 0.0038806483132620842
*****
Train RMSE: 0.04684393141887929
Test RMSE: 0.06229484981330386
None
```

In [76]:
```
ridge_coefs = ms_eval_coefficients(ridge_model, cols)
```

```
Total number of coefficients:  39
Coefficients close to zero:  0
Intercept: 10.676291138768727

avg_annual_cost_attendance                                 0.034409
1yr_declining_loan_balance                                 0.029006
pct_asian                                                  0.025116
tuition_rev_per_student                                    0.022360
LOAN_EVER                                                  0.022276
SAT_AVG                                                    0.018244
avg_faculty_sal                                            0.017989
pct_hispanic                                               0.017693
Women's Team Average Annual Institutional Salary per FTE4  0.017341
Basketball Women's Team Revenue                            0.011370
pct_black                                                  0.010463
REGION                                                     0.009624
median_female_debt                                         0.008706
 Womens_opex_per_participant                               0.008692
Total Men's Team Participation                             0.008243
Total Women's Team Operating Expenses                      0.007749
```

After running the ridge model, we have 39 features remaining (ridge doesn't zero out coefficients). The model fits very well and produces small error scores. In the end, this might end up being the best fitting model.

Looking at our coefficients, we have some interesting findings. Average cost of attendance has the largest magnitude of all the variables left. A one standard deviation increase (because the variables are scaled using z-scores) in cost produces a 3.4% increase in 6 year average earnings. Cost of attendance is likely a proxy for other hidden variable that describe school quality (not in every case of course).

Percentage of Asian, Black, and Hispanic students also have positive coefficients, perhaps implying that more diverse schools produce better outcomes. Additionally, we see positive coefficients associated with that schools spend on women's sports. A one standard deviation increase in coaching salaries on women's teams leads to a 1.7% increase in average earnings, while the same increase in operating expenses per female athlete leads to a .87% increase in average earnings.

The coefficients that have the largest negative impact on earnings are admission rate, median debt of graduates, and the pell grant rate. The percentage of women at a school also has a negative impact, but this is likely due to the structural gender pay gap which we explored in the data understanding section. Admission rate is likely another proxy of school quality - the less selective a school is, the worse the average student is performing. A one standard deviation increase in admission rate decreases future earnings by 2.2%.

These coefficients aren't likely to all be statistically significant, so after running an ElasticNet model next, we'll input the remaining coefficients in a statsmodel regression to see the p-values associated with each coefficient.

ElasticNet

An Elastic Net model blends the penalties of ridge and lasso models to get the benefits of both. Using cross validation, we'll find the best possible parameters to run a final Elastic Net model.

In [77]:
```
#first, we'll run a cross validated Elastic Net model, which is built into the fu
e_net_cv, cols = ms_elastic_net_cv(ncaa_grouped_all_merge)
```

```
Train R2: 0.9072072159359033
Test R2: 0.9001394280336424
*****
Train MSE: 0.0037595943065333237
Test MSE: 0.0038779688013106947
*****
Train RMSE: 0.06131553071231891
Test RMSE: 0.06227333941030218
alpha: 0.100000
l1_ratio_: 0.070000
```

The model is fit nearly identically on the train and test set, while producing a small RMSE. The best parameters found in the best model were an alpha of .1, which is the Ridge penalty and an L1 ratio of .07, which is the Lasso penalty.

In [78]:
```
e_net_coefs_cv = ms_eval_coefficients(e_net_cv, cols)
```

```
Total number of coefficients:  94
Coefficients close to zero:  69
Intercept: 10.676291138768729

avg_annual_cost_attendance    0.028909
pct_asian                     0.024909
1yr_declining_loan_balance    0.021691
avg_faculty_sal               0.019566
tuition_rev_per_student       0.015847
                                ...
num_female_4yr_completers    -0.010897
pct_women                    -0.012914
ADM_RATE                     -0.027378
PELL_EVER                    -0.027620
median_debt_completers       -0.038121
Length: 94, dtype: float64
```

The Elastic Net Model moved 69 coefficients to 0 or close to. To validate this, let's use a more robust grid search model to find the best parameters.

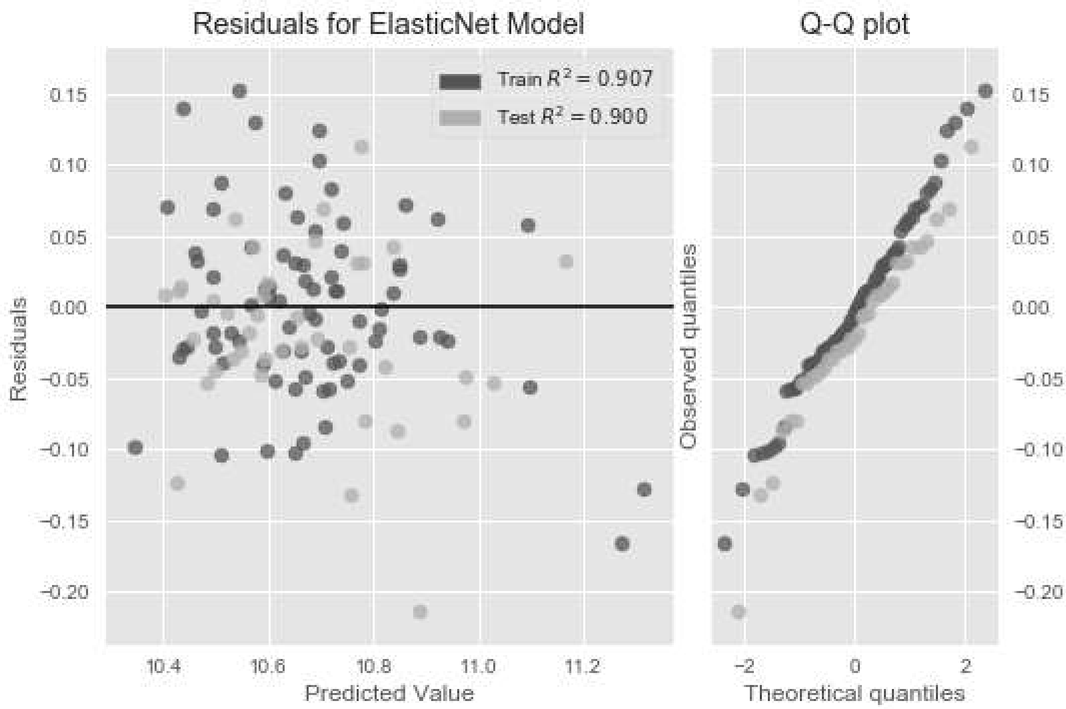
In [79]:
```
ms_repeated_kfolds(ncaa_grouped_all_merge)
```

```
est=0.746, cfg={'alpha': 1, 'l1_ratio': 0.0, 'max_iter': 100}
---
Train Score: 0.9036344644202401
Test Score: 0.7942460997063945
Train RMSE:  0.0633209062652823
Test RMSE:  0.07217176685775863
est=0.795, cfg={'alpha': 0.1, 'l1_ratio': 0.1, 'max_iter': 100}
---
Train Score: 0.9298920752567038
Test Score: 0.6774788465694221
Train RMSE:  0.05374560615606596
Test RMSE:  0.09098483335805821
est=0.792, cfg={'alpha': 0.01, 'l1_ratio': 0.5, 'max_iter': 100}
---
Train Score: 0.8979442730454816
Test Score: 0.909493138644314
Train RMSE:  0.06432840961869532
Test RMSE:  0.05514827257714502
est=0.769, cfg={'alpha': 0.1, 'l1_ratio': 0.1, 'max_iter': 100}
---
```

Using Repeated K-Folds to perform the train-test-splits rather than the standard elastic net grid search has produced slightly different best parameters. Next, we'll use two different Elastic Net models, one for each set of best parameters, and compare the two models.

In [81]:
```
#parameters from elastic net grid search
e_net_grid_sum, e_net_grid, cols_grid = ms_elastic_net(ncaa_grouped_all_merge, pa
                                                        l1_ratio = 0.07, max_iter = 1
```

```
Alpha:  0.1
L1 Ratio:  0.07
```



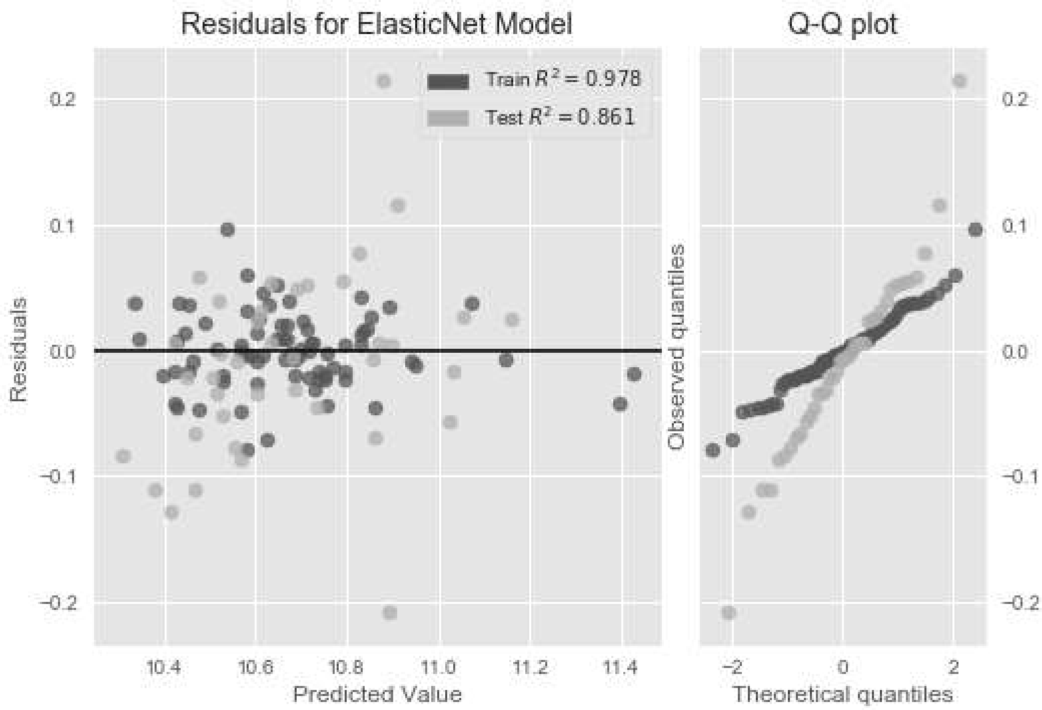

```
Train R2: 0.9072072159359033
Test R2: 0.9001394280336424
*****
Train MSE: 0.0037595943065333237
Test MSE: 0.0038779688013106947
*****
Train RMSE: 0.06131553071231891
Test RMSE: 0.06227333941030218
```

In [82]:
```
e_net_grid_coefs_ = ms_eval_coefficients(e_net_grid, cols_grid)
```

```
Total number of coefficients:  94
Coefficients close to zero:  69
Intercept: 10.676291138768729

avg_annual_cost_attendance    0.028909
pct_asian                     0.024909
1yr_declining_loan_balance    0.021691
avg_faculty_sal               0.019566
tuition_rev_per_student       0.015847
                                ...
num_female_4yr_completers    -0.010897
pct_women                    -0.012914
ADM_RATE                     -0.027378
PELL_EVER                    -0.027620
median_debt_completers       -0.038121
Length: 94, dtype: float64
```

In [84]:
```
#parameters from repeat k folds method
e_net_repeat_kfolds_sum, e_net_repeat_kfolds, cols_repeat_k = ms_elastic_net(ncaa
                                                            l1_ratio = 0.0, max_iter = 10
```

```
Alpha:  0.01
L1 Ratio:  0.0
```




```
Train R2: 0.9782901843858844
Test R2: 0.8609910524951742
*****
Train MSE: 0.0008795953263169454
Test MSE: 0.0053982502894972485
*****
Train RMSE: 0.029657972390521665
Test RMSE: 0.07347278604692521
```

In [85]:
```
e_net_repeat_coefs_ = ms_eval_coefficients(e_net_repeat_kfolds, cols_repeat_k)
```

```
Total number of coefficients:  94
Coefficients close to zero:  0
Intercept: 10.676291138768727

avg_annual_cost_attendance                0.073940
num_students                              0.059873
1yr_declining_loan_balance                0.041935
LOAN_EVER                                 0.036726
pell_grant_rate_current_academic_year     0.035218
                                            ...
Men's Team Athletic Student Aid          -0.034357
pct_women                                -0.039461
Female Undergraduates                    -0.041555
PELL_EVER                                -0.043571
median_debt_completers                   -0.072609
Length: 94, dtype: float64
```

The parameters produced by the repeated KFolds method fit and perform worse than those produced by the built in Elastic Net grid search. The model also failed to drop any coefficients to zero, making it impossible to use in a regression analysis.

Let's use the grid search parameters for the regression.

Use ElasticNet coefficients in statsmodels so we can identify statistical significance using p-values

In [86]:
```
#creating list of cols remaining after elastic net zeroed out 69 coefficients
coef_df = pd.DataFrame(e_net_grid_coefs_)
coef_df.reset_index(inplace = True)
coef_df.rename(columns = {0:'coef', 'index':'feature'}, inplace = True)

coef_df['remove'] = coef_df.apply(lambda row: abs(row.coef) < 10**(-10), axis = 1
coef_df = coef_df[coef_df['remove'] == False]

coef_df.set_index('feature', inplace = True)
coef_df.drop(columns = 'remove', inplace = True)
```

In [87]:
```
coef_col_list_enet = coef_df.index.to_list()
```

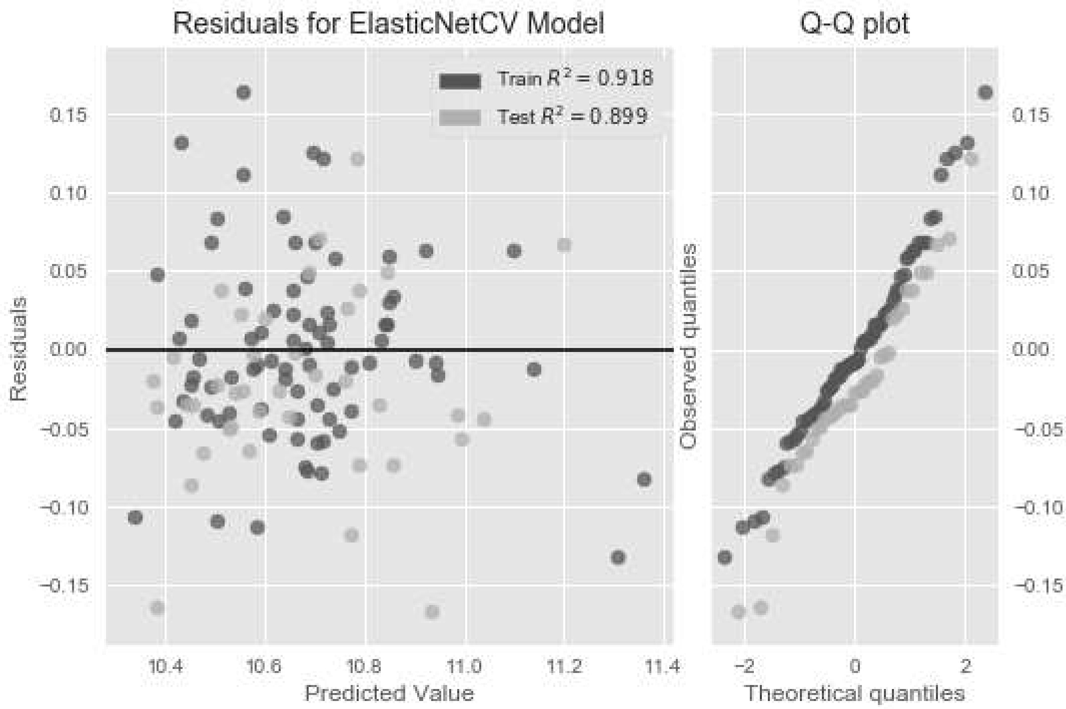
In [88]:
```
coef_col_list_enet
```

Out[88]:
```
['avg_annual_cost_attendance',
 'pct_asian',
 '1yr_declining_loan_balance',
 'avg_faculty_sal',
 'tuition_rev_per_student',
 'SAT_AVG',
 "Women's Team Average Annual Institutional Salary per FTE4",
 'pct_hispanic',
 "Total Men's Team Participation",
 '3yr_declining_loan_balance',
 'Football Total Participation',
 "Basketball Women's Team Expenses",
 'instate_tuition',
 ' Womens_opex_per_participant ',
 "Basketball Women's Team Operating Expenses",
 '5yr_declining_loan_balance',
 "Men's Team Average Annual Institutional Salary per FTE",
 'Football Total Revenue',
 'num_4yr_completers',
 'pct_mid_income_2',
 'num_female_4yr_completers',
 'pct_women',
 'ADM_RATE',
 'PELL_EVER',
 'median_debt_completers']
```

Using statsmodels elastic net on reduced column list

In [90]:
```
sum_ev_1, ev_1, ev_cols = ms_elastic_net(ncaa_grouped_all_merge, cv = 'yes',col_l
sum_ev_1
```

```
Alpha:  0.1
L1 Ratio:  0.01
```




```
Train R2: 0.9181917597664779
Test R2: 0.8993613183657649
*****
Train MSE: 0.0033145443076372017
Test MSE: 0.003908185782413542
*****
Train RMSE: 0.05757207923670294
Test RMSE: 0.06251548434118975
```

Out[90]:

OLS Regression Results

Dep. Variable:	6yr_mean_earnings	**R-squared:**	0.921
Model:	OLS	**Adj. R-squared:**	0.883
Method:	Least Squares	**F-statistic:**	24.29
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	1.30e-20
Time:	11:37:57	**Log-Likelihood:**	113.41
No. Observations:	78	**AIC:**	-174.8
Df Residuals:	52	**BIC:**	-113.5
Df Model:	25		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.6763	0.008	1361.784	0.000	10.661	10.692

avg_annual_cost_attendance	0.0528	0.038	1.405	0.166	-0.023	0.128
pct_asian	0.0384	0.016	2.416	0.019	0.007	0.070
1yr_declining_loan_balance	0	0.085	0	1.000	-0.170	0.170
avg_faculty_sal	0.0037	0.025	0.144	0.886	-0.047	0.055
tuition_rev_per_student	0.0303	0.023	1.299	0.200	-0.016	0.077
SAT_AVG	-0.0060	0.030	-0.204	0.839	-0.065	0.053
Women's Team Average Annual Institutional Salary per FTE4	0.0204	0.021	0.977	0.333	-0.022	0.062
pct_hispanic	0.0164	0.014	1.130	0.264	-0.013	0.045
Total Men's Team Participation	0.0103	0.018	0.561	0.577	-0.027	0.047
3yr_declining_loan_balance	0.0101	0.185	0.055	0.957	-0.361	0.381
Football Total Participation	0.0077	0.010	0.737	0.465	-0.013	0.029
Basketball Women's Team Expenses	0	0.041	0	1.000	-0.081	0.081
instate_tuition	-0.0312	0.037	-0.835	0.407	-0.106	0.044
Womens_opex_per_participant	0	0.022	0	1.000	-0.045	0.045
Basketball Women's Team Operating Expenses	0	0.030	0	1.000	-0.060	0.060
5yr_declining_loan_balance	0	0.119	0	1.000	-0.238	0.238
Men's Team Average Annual Institutional Salary per FTE	0	0.025	0	1.000	-0.050	0.050
Football Total Revenue	0.0006	0.025	0.022	0.982	-0.049	0.050
num_4yr_completers	-0.0251	0.219	-0.115	0.909	-0.464	0.414
pct_mid_income_2	-0.0056	0.011	-0.523	0.603	-0.027	0.016
num_female_4yr_completers	0	0.220	0	1.000	-0.441	0.441
pct_women	-0.0280	0.034	-0.821	0.415	-0.096	0.040
ADM_RATE	-0.0379	0.024	-1.600	0.116	-0.085	0.010
PELL_EVER	-0.0488	0.019	-2.539	0.014	-0.087	-0.010
median_debt_completers	-0.0451	0.014	-3.333	0.002	-0.072	-0.018

Omnibus:	2.707	**Durbin-Watson:**	1.808
Prob(Omnibus):	0.258	**Jarque-Bera (JB):**	2.175
Skew:	-0.403	**Prob(JB):**	0.337
Kurtosis:	3.144	**Cond. No.**	128.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

With a smaller feature list, the L1_ratio has fallen to .01 from .07, moving the effect closer to ridge regression, than lasso regression. This is likely because nearly all uninformative variables have already been moved to zero.

In [91]:
```
#first, going to check collinearity here - most vars are highly collinear (anythi
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet )
```

Out[91]:

	VIF	features
0	306.212240	avg_annual_cost_attendance
1	4.452324	pct_asian
2	1982.158787	1yr_declining_loan_balance
3	258.222538	avg_faculty_sal
4	49.328622	tuition_rev_per_student
5	688.359167	SAT_AVG
6	57.589947	Women's Team Average Annual Institutional Sala...
7	4.237910	pct_hispanic
8	51.737530	Total Men's Team Participation
9	17587.431633	3yr_declining_loan_balance
10	206.992887	Football Total Participation
11	70.713229	Basketball Women's Team Expenses
12	111.268006	instate_tuition
13	34.381604	Womens_opex_per_participant
14	32.259059	Basketball Women's Team Operating Expenses
15	11920.679267	5yr_declining_loan_balance
16	15.024022	Men's Team Average Annual Institutional Salary...
17	9.150885	Football Total Revenue
18	581.498612	num_4yr_completers
19	199.671292	pct_mid_income_2
20	578.006807	num_female_4yr_completers
21	358.204292	pct_women
22	35.119096	ADM_RATE
23	136.375912	PELL_EVER
24	70.976314	median_debt_completers

In [92]:
```
#removing vars that have high collinearity
coef_col_list_enet_2 = ['avg_annual_cost_attendance',
 'pct_asian',
 'pct_hispanic',
 ' Womens_opex_per_participant ',
 'Football Total Revenue',
'pct_women',
 'ADM_RATE',
 'median_debt_completers']
```

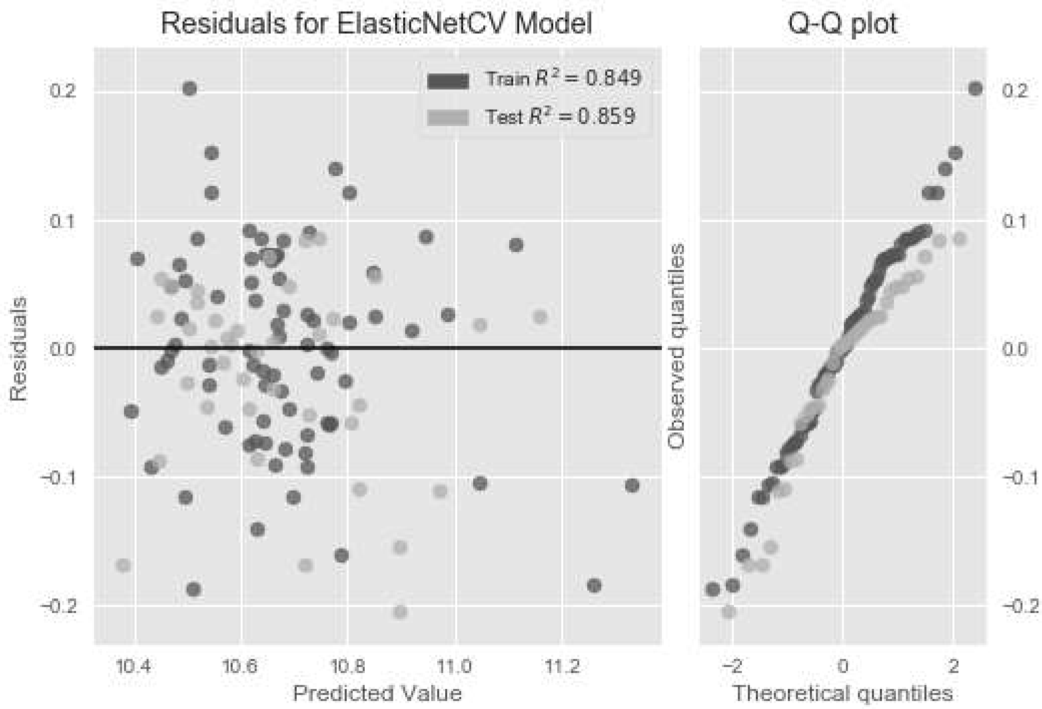
In [93]:
```
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet_2)
```

Out[93]:

	VIF	features
0	9.766123	avg_annual_cost_attendance
1	2.116188	pct_asian
2	1.904287	pct_hispanic
3	15.209906	Womens_opex_per_participant
4	5.053633	Football Total Revenue
5	46.116500	pct_women
6	21.902404	ADM_RATE
7	42.084707	median_debt_completers

In [94]:
```
sum_ev_2, ev_2, ev_cols_2 = ms_elastic_net(ncaa_grouped_all_merge, cv = 'yes', co
sum_ev_2
```

```
Alpha:  0.01
L1 Ratio:  0.9
```




```
Train R2: 0.8489213027328841
Test R2: 0.8589321316487128
*****
Train MSE: 0.006121107538831654
Test MSE: 0.005478206078350882
*****
Train RMSE: 0.07823750723809939
Test RMSE: 0.0740149044338428
```

Out[94]:

OLS Regression Results

Dep. Variable:	6yr_mean_earnings	**R-squared:**	0.855
Model:	OLS	**Adj. R-squared:**	0.838
Method:	Least Squares	**F-statistic:**	50.73
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	6.47e-26
Time:	11:40:40	**Log-Likelihood:**	89.582
No. Observations:	78	**AIC:**	-161.2
Df Residuals:	69	**BIC:**	-140.0
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.6763	0.009	1155.752	0.000	10.658	10.695

avg_annual_cost_attendance	0.0780	0.015	5.264	0.000	0.048	0.107
pct_asian	0.0390	0.012	3.388	0.001	0.016	0.062
pct_hispanic	0	0.011	0	1.000	-0.022	0.022
Womens_opex_per_participant	0	0.014	0	1.000	-0.028	0.028
Football Total Revenue	0.0395	0.015	2.698	0.009	0.010	0.069
pct_women	-0.0397	0.010	-3.868	0.000	-0.060	-0.019
ADM_RATE	-0.0449	0.017	-2.592	0.012	-0.079	-0.010
median_debt_completers	-0.0496	0.013	-3.831	0.000	-0.075	-0.024

Omnibus:	0.100	**Durbin-Watson:**	1.842
Prob(Omnibus):	0.951	**Jarque-Bera (JB):**	0.249
Skew:	0.067	**Prob(JB):**	0.883
Kurtosis:	2.758	**Cond. No.**	4.12

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

After removing variables with high multicollinearity, and re-running the elastic net model, we see the L1 ratio jumped all the up to .9 (at 1, this model is fully a Lasso model). The model reduced percent hispanic and operating expenses per female sports participant to 0, implying they do not help explain any variation in 6 year mean earnings.

In [95]:
```
for col in coef_col_list_enet_2:
    print(f"Standard Deviation of {col}: {ncaa_grouped_all_merge[col].std()}")
```

```
Standard Deviation of avg_annual_cost_attendance: 10763.611122778164
Standard Deviation of pct_asian: 0.05668133138360794
Standard Deviation of pct_hispanic: 0.08806490737272997
Standard Deviation of  Womens_opex_per_participant : 2647.5985338657956
Standard Deviation of Football Total Revenue: 21804917.101798184
Standard Deviation of pct_women: 0.04814824936663754
Standard Deviation of ADM_RATE: 0.19697873315451556
Standard Deviation of median_debt_completers: 3000.0892111368753
```

Model Summary

In this general model, we have 6 remaining non-zero coefficients.

These changes are an average when holding all else equal:

- **Average cost of attendance**: a one standard deviation increase in average cost of attendance leads to a 7.80% increase in earnings
- **Football Total Revenue**: 3.95% increase in earnings
- **Percentage of asian students**: 3.90% decrease in earnings
- **Percentage of female students**: 3.96% decrease in earnings

- **Admission rate**: 4.49% decrease
- **Median debt of 4 year (150% time) completers**: 4.96% decrease in earnings

So, our final model fits more evenly on the train and test set while reducing the test set errors. There are 6 remaining coefficients that have a statistically significant relationship with 6 year average earnings. The coefficient with the largest magnitude is the average annual cost of attendance. A 1 standard deviation increase in cost (10K) leads to a 7.80% increase in average earnings. We may be seeing the impact of added resources that a school may have. This is related to admissions rate. A one standard deviation increase in admission rate (20%) leads to a 4.49% decrease in earnings on average. As schools become less selective, earnings amongst its graduates tend to fall.

In addition, we see that a one standard deviation increase in percentage of women (4.8%) causes average earnings to drop by 3.96% on average. Women are so vastly underpaid upon graduating that just having more attend the school brings down the average earnings of the entire school.

We also see the percentage of asian students has a positive impact on earnings, perhaps speaking to how creating a more diverse enviornment can improve outcomes. It could also be the result of asian origin groups earning a much higher median income than the average American household. [Asian households in the U.S. had a median annual income of 85,800 in 2019, higher than the 61,800 among all U.S. households. (https://www.pewresearch.org/fact-tank/2021/04/29/key-facts-about-asian-origin-groups-in-the-u-s/)](https://www.pewresearch.org/fact-tank/2021/04/29/key-facts-about-asian-origin-groups-in-the-u-s/).

Surprisingly, we see a positive impact from total football revenue. A one standard deviation increase in football revenue ($2.2M) causes average earnings to rise by 3.95%. This could be a proxy for school resources in the way that average cost of attendance is, or perhaps schools are spending more on students with added football revenue. Further work in this area would look to identify how profitable football schools allocate this money towards academics.

Regressing on Male and Female Earnings

Next, we'll perform the same sort of regression analysis as above, but now using 6 year average male earnings and 6 year average female earnings to get to the bottom of what variables impact each the most.

First, we'll do female earnings.

In [99]:
```
#elastic_net grid search
e_net_female_cv, cols = ms_elastic_net_cv(ncaa_grouped_all_merge, all_earn = 'no
                                                                    dep ='6yr_femal
```

```
Train R2: 0.8940470663894027
Test R2: 0.8227895118076635
*****
Train MSE: 0.003736807916718533
Test MSE: 0.005994862753173457
*****
Train RMSE: 0.06112943576312915
Test RMSE: 0.07742649903730284
alpha: 0.100000
l1_ratio_: 0.080000
```

In [100]:
```
#elastic_net grid search
e_net_female_coefs_cv = ms_eval_coefficients(e_net_female_cv, cols)
```

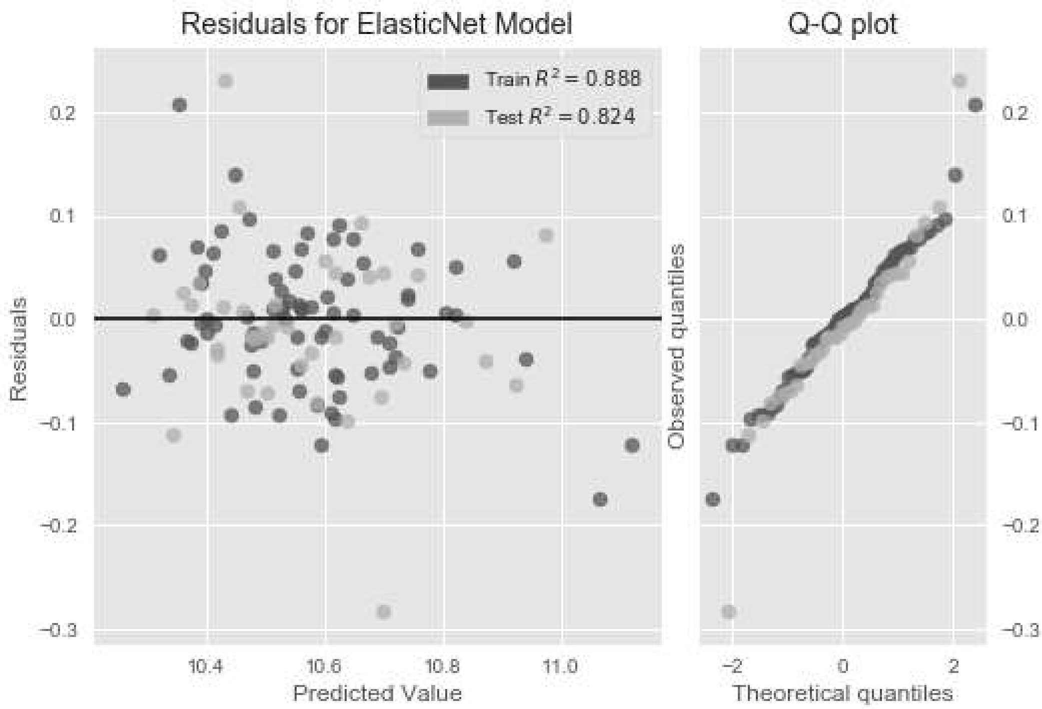
```
Total number of coefficients:  94
Coefficients close to zero:  77
Intercept: 10.574671257468763

pct_asian                          0.034007
avg_annual_cost_attendance         0.028926
avg_faculty_sal                    0.027343
1yr_declining_loan_balance         0.014726
pct_hispanic                       0.012366
                                     ...
3_yr_repayment_completers_rate     0.000000
num_female_4yr_completers         -0.001770
ADM_RATE                          -0.021393
median_debt_completers            -0.025741
PELL_EVER                         -0.032960
Length: 94, dtype: float64
```

In [101]:
```
ms_repeated_kfolds(ncaa_grouped_all_merge, all_earn = 'no', dep = '6yr_female_mea
```

```
Train Score: 0.9409185988983838
Test Score: 0.3723815407419303
Train RMSE:  0.04711956541089733
Test RMSE:  0.0732731059130594
est=0.789, cfg={'alpha': 0.1, 'l1_ratio': 0.0, 'max_iter': 1000}
---
Train Score: 0.9190423225087737
Test Score: 0.6998544036843544
Train RMSE:  0.04908108889812425
Test RMSE:  0.13283365525557406
est=0.744, cfg={'alpha': 0.01, 'l1_ratio': 0.30000000000000004, 'max_iter': 1
00}
---
Train Score: 0.8715450800598272
Test Score: 0.7804815320629157
Train RMSE:  0.06750545260958461
Test RMSE:  0.07958709876612692
est=0.706, cfg={'alpha': 0.1, 'l1_ratio': 0.1, 'max_iter': 10}
---
```

In [102]:
```
#repeated kfolds method
female_sum, e_net_female_earn, cols = ms_elastic_net(ncaa_grouped_all_merge, all_
                                         params = 'yes', alpha = .1, l1_ratio = .
```

```
Alpha:  0.1
L1 Ratio:  0.1
```




```
Train R2: 0.8880284747357394
Test R2: 0.8238347555238985
*****
Train MSE: 0.003949075007137777
Test MSE: 0.005959503149538453
*****
Train RMSE: 0.06284166617092339
Test RMSE: 0.0771978182951983
```

In [103]:
```
e_net_female_coefs_ = ms_eval_coefficients(e_net_female_earn, cols)
```

```
Total number of coefficients:  94
Coefficients close to zero:  77
Intercept: 10.574671257468763

pct_asian                          0.032461
avg_annual_cost_attendance         0.030582
avg_faculty_sal                    0.029837
1yr_declining_loan_balance         0.014927
SAT_AVG                            0.009990
                                     ...
3_yr_repayment_completers_rate     0.000000
num_female_4yr_completers         -0.000435
ADM_RATE                          -0.021113
median_debt_completers            -0.023676
PELL_EVER                         -0.031917
Length: 94, dtype: float64
```

The repeated kfolds model performs a bit better, producing nearly identical R2 scores and errors, so we'll use those parameters.

In [104]:
```
coef_df = pd.DataFrame(e_net_female_coefs_)
coef_df.reset_index(inplace = True)
coef_df.rename(columns = {0:'coef', 'index':'feature'}, inplace = True)

coef_df['remove'] = coef_df.apply(lambda row: abs(row.coef) < 10**(-10), axis = 1
coef_df = coef_df[coef_df['remove'] == False]

coef_df.set_index('feature', inplace = True)
coef_df.drop(columns = 'remove', inplace = True)
```

In [105]:
```
coef_col_list_enet_female = coef_df.index.to_list()
```

In [106]:
```
coef_col_list_enet_female
```

Out[106]:
```
['pct_asian',
 'avg_annual_cost_attendance',
 'avg_faculty_sal',
 '1yr_declining_loan_balance',
 'SAT_AVG',
 'tuition_rev_per_student',
 'pct_hispanic',
 "Women's Team Average Annual Institutional Salary per FTE4",
 'FAMINC',
 '3yr_declining_loan_balance',
 "Basketball Women's Team Revenue",
 'pct_high_income_2',
 "Basketball Women's Team Expenses",
 'num_female_4yr_completers',
 'ADM_RATE',
 'median_debt_completers',
 'PELL_EVER']
```

In [107]:
```
#ID highly collinear variables
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet_female)
```

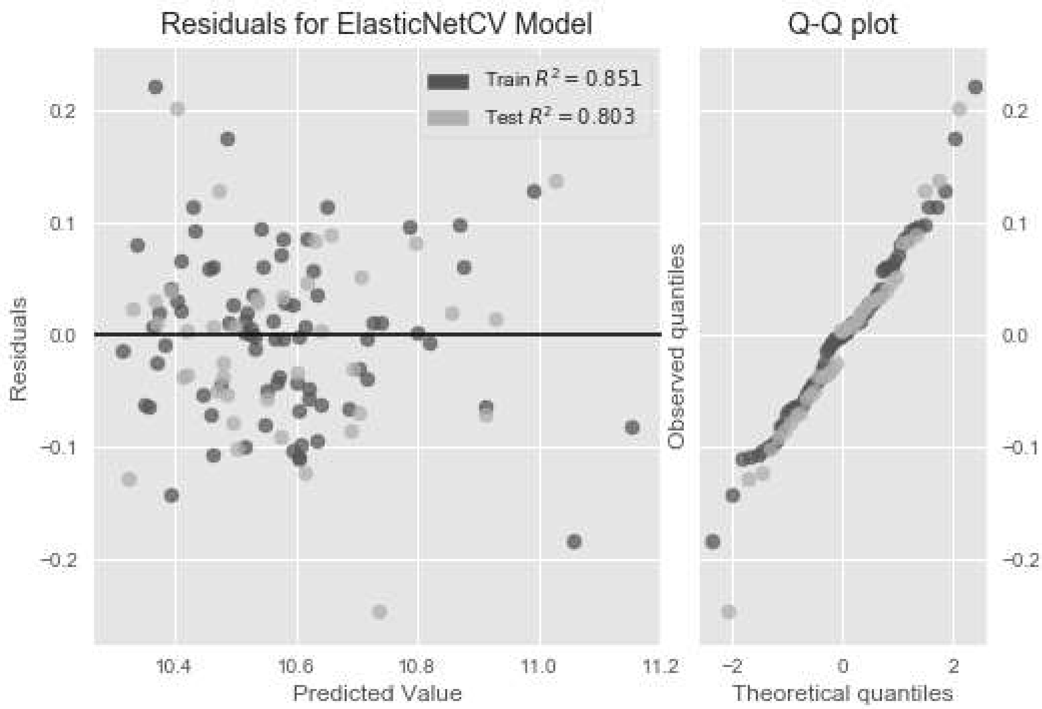
Out[107]:

	VIF	features
0	4.692733	pct_asian
1	61.685505	avg_annual_cost_attendance
2	211.660522	avg_faculty_sal
3	941.972191	1yr_declining_loan_balance
4	603.544639	SAT_AVG
5	39.812158	tuition_rev_per_student
6	2.971142	pct_hispanic
7	40.327992	Women's Team Average Annual Institutional Sala...
8	1690.549641	FAMINC
9	1514.827338	3yr_declining_loan_balance
10	4.067309	Basketball Women's Team Revenue
11	876.752032	pct_high_income_2
12	28.430236	Basketball Women's Team Expenses
13	9.622169	num_female_4yr_completers
14	30.481163	ADM_RATE
15	86.435245	median_debt_completers
16	214.787153	PELL_EVER

In [108]:
```
#remove highly collinear vars
coef_col_list_enet_female = ['pct_asian',
 'avg_annual_cost_attendance',
 'avg_faculty_sal',
 'pct_hispanic',
 'tuition_rev_per_student',
 "Women's Team Average Annual Institutional Salary per FTE4",
 "Basketball Women's Team Revenue",
 'num_female_4yr_completers',
 'ADM_RATE',
 'median_debt_completers']
```

In [110]:
```
sum_ev_1, e_net_cv_2, X_cols = ms_elastic_net(ncaa_grouped_all_merge, col_list = 
                                              cv = 'yes',all_earn = 'no', dep ='6y
sum_ev_1
```

```
Alpha:  0.1
L1 Ratio:  0.04
```




```
Train R2: 0.8506112119534488
Test R2: 0.8026170116018913
*****
Train MSE: 0.005268728168424259
Test MSE: 0.006677279303996982
*****
Train RMSE: 0.07258600532075214
Test RMSE: 0.08171462111517731
```

Out[110]:

OLS Regression Results

Dep. Variable:	6yr_female_mean_earnings	**R-squared:**	0.855
Model:	OLS	**Adj. R-squared:**	0.833
Method:	Least Squares	**F-statistic:**	39.37
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	3.37e-24
Time:	11:47:30	**Log-Likelihood:**	94.964

No. Observations:	78	**AIC:**	-167.9
Df Residuals:	67	**BIC:**	-142.0
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.5747	0.009	1208.621	0.000	10.557	10.592
pct_asian	0.0410	0.015	2.775	0.007	0.012	0.071
avg_annual_cost_attendance	0.0438	0.025	1.789	0.078	-0.005	0.093
avg_faculty_sal	0.0522	0.020	2.632	0.011	0.013	0.092
pct_hispanic	0.0071	0.011	0.641	0.523	-0.015	0.029
tuition_rev_per_student	0.0245	0.022	1.130	0.263	-0.019	0.068
Women's Team Average Annual Institutional Salary per FTE4	0.0221	0.011	2.058	0.043	0.001	0.044
Basketball Women's Team Revenue	0	0.010	0	1.000	-0.020	0.020
num_female_4yr_completers	-0.0288	0.012	-2.413	0.019	-0.053	-0.005
ADM_RATE	-0.0089	0.017	-0.515	0.608	-0.043	0.026
median_debt_completers	-0.0213	0.013	-1.606	0.113	-0.048	0.005

Omnibus:	2.928	**Durbin-Watson:**	1.943
Prob(Omnibus):	0.231	**Jarque-Bera (JB):**	2.251
Skew:	-0.393	**Prob(JB):**	0.325
Kurtosis:	3.275	**Cond. No.**	7.69

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [111]:
```
X_cols
```

Out[111]:
```
['pct_asian',
 'avg_annual_cost_attendance',
 'avg_faculty_sal',
 'pct_hispanic',
 'tuition_rev_per_student',
 "Women's Team Average Annual Institutional Salary per FTE4",
 "Basketball Women's Team Revenue",
 'num_female_4yr_completers',
 'ADM_RATE',
 'median_debt_completers']
```

In [112]:
```
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =X_cols)
```

Out[112]:

	VIF	features
0	3.365794	pct_asian
1	42.687224	avg_annual_cost_attendance
2	62.965867	avg_faculty_sal
3	2.056503	pct_hispanic
4	26.192127	tuition_rev_per_student
5	15.069893	Women's Team Average Annual Institutional Sala...
6	3.284705	Basketball Women's Team Revenue
7	6.729532	num_female_4yr_completers
8	18.708012	ADM_RATE
9	41.298388	median_debt_completers

In [113]:
```
coef_col_list_enet_female_2 = ['pct_asian',
 'pct_hispanic',
 'tuition_rev_per_student',
 "Women's Team Average Annual Institutional Salary per FTE4",
 "Basketball Women's Team Revenue",
 'num_female_4yr_completers',
 'ADM_RATE']
```

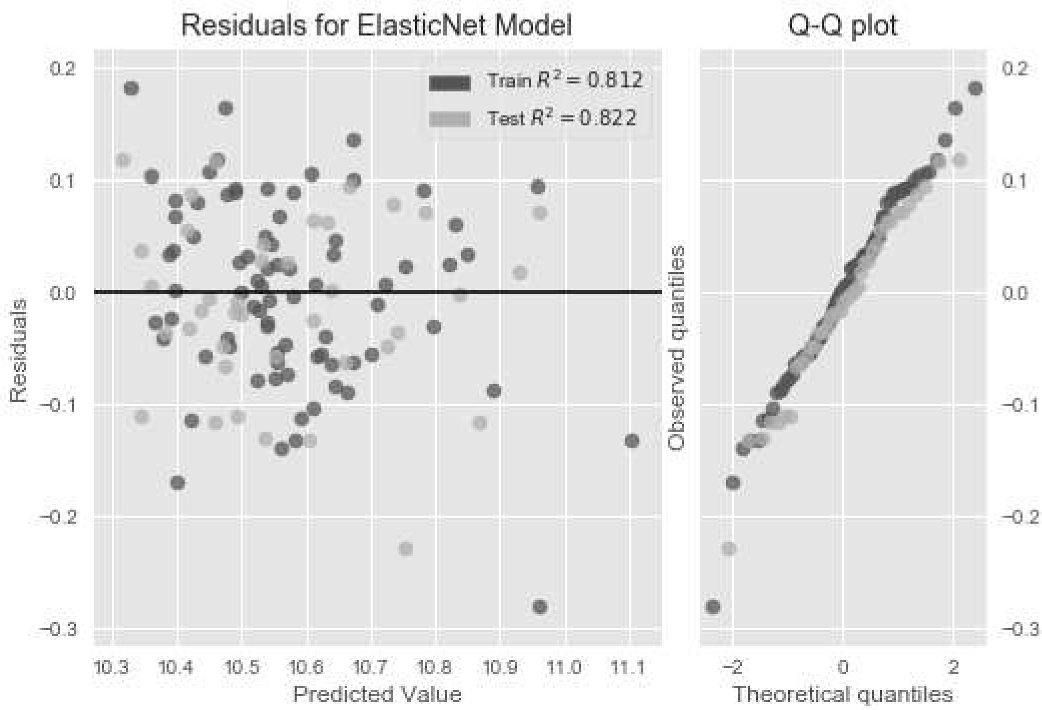
In [114]:
```
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet_female_2
```

Out[114]:

	VIF	features
0	2.093929	pct_asian
1	1.790061	pct_hispanic
2	5.134017	tuition_rev_per_student
3	10.816023	Women's Team Average Annual Institutional Sala...
4	2.659911	Basketball Women's Team Revenue
5	5.239743	num_female_4yr_completers
6	7.083166	ADM_RATE

In [115]:
```
#let's turn up the lasso effect to move some more coefficients to zero - when the
#know how to create a model, results in negative R2

sum_ev_2, e_net_cv_3, X_cols = ms_elastic_net(ncaa_grouped_all_merge, col_list =
                                                params = 'yes', alpha = .1, l1_ra
                                            all_earn = 'no', dep ='6yr_female_mea
sum_ev_2
```

```
Alpha:  0.1
L1 Ratio:  0.1
```




```
Train R2: 0.8122177736976892
Test R2: 0.8216684767518729
*****
Train MSE: 0.0066228096377628
Test MSE: 0.006032786305946845
*****
Train RMSE: 0.0813806465799013
Test RMSE: 0.07767101329290642
```

Out[115]:

OLS Regression Results

Dep. Variable:	6yr_female_mean_earnings	**R-squared:**	0.829
Model:	OLS	**Adj. R-squared:**	0.812

Method:	Least Squares	**F-statistic:**	48.53
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	2.17e-24
Time:	11:47:45	**Log-Likelihood:**	88.679
No. Observations:	78	**AIC:**	-161.4
Df Residuals:	70	**BIC:**	-142.5
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.5747	0.009	1139.746	0.000	10.556	10.593
pct_asian	0.0664	0.012	5.437	0.000	0.042	0.091
pct_hispanic	0.0202	0.010	1.931	0.058	-0.001	0.041
tuition_rev_per_student	0.0591	0.012	4.936	0.000	0.035	0.083
Women's Team Average Annual Institutional Salary per FTE4	0.0338	0.010	3.374	0.001	0.014	0.054
Basketball Women's Team Revenue	0.0054	0.010	0.537	0.593	-0.015	0.025
num_female_4yr_completers	-0.0432	0.011	-3.780	0.000	-0.066	-0.020
ADM_RATE	-0.0440	0.014	-3.157	0.002	-0.072	-0.016

Omnibus:	0.584	**Durbin-Watson:**	1.805
Prob(Omnibus):	0.747	**Jarque-Bera (JB):**	0.222
Skew:	0.104	**Prob(JB):**	0.895
Kurtosis:	3.157	**Cond. No.**	2.89

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

We're left with fewer coefficients when running the regression on just female earnings. We have 5 remaining:

- **Percentage of asian students**: a one standard deviation increase in average cost of attendance leads to a 6.64% increase in earnings
- **Tuition Revenue per student**: 5.91% increase in earnings - this is similar to average cost of attendance
- **Women's Team Average Annual Institutional Salary per FTE**: 3.38% increase in earnings
- **Number of female 4 year completers**: 4.32% decrease in earnings - likely a proxy for percentage of female students
- **Admission rate**: 4.40% decrease in earnings

We do see some overlap with the general average earnings features, but some interesting new ones pop up as well. Instead of average cost of attendance, we see tutition revenue per student, where a one standard deviation increase in tuition revenue results in a 5.91% increase in earnings.

The most intriguing estimate is from the Women's Team Average Annual Institutional Salary per FTE. We have a female-specific variable having a positive impact and female earnings. As the salaries of employees on women's teams increase, so do average female earnings. This may be an indicator of increased female student support and investment.

6 year male mean earnings regression

Now let's move on to regression average male earnings.

In [116]:
```
#grid search
e_net_male_cv, cols = ms_elastic_net_cv(ncaa_grouped_all_merge, all_earn = 'no',
```

```
Train R2: 0.8858977052465581
Test R2: 0.885501364191666
*****
Train MSE: 0.005200264851695977
Test MSE: 0.004759568538092695
*****
Train RMSE: 0.07211286190199344
Test RMSE: 0.06898962630782034
alpha: 0.100000
l1_ratio_: 0.070000
```

In [117]:
```
e_net_male_coefs_cv = ms_eval_coefficients(e_net_male_cv, cols)
```

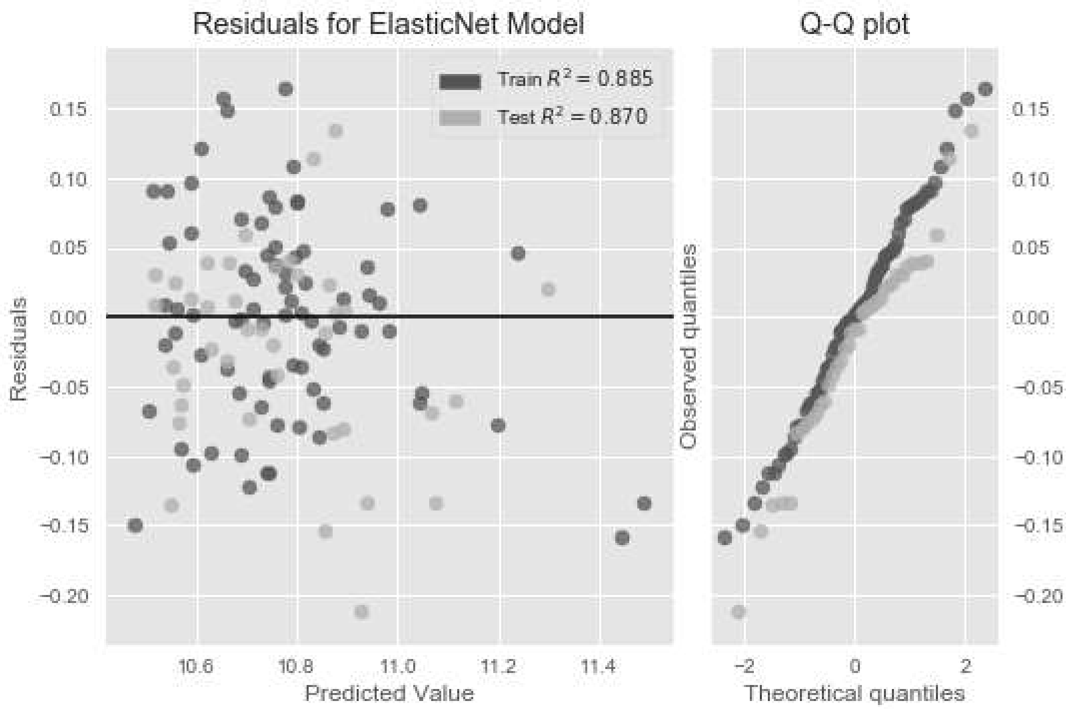
```
Total number of coefficients:  94
Coefficients close to zero:  68
Intercept: 10.774996154173587

avg_annual_cost_attendance    0.024579
1yr_declining_loan_balance    0.023809
pct_asian                     0.017385
avg_faculty_sal               0.016080
tuition_rev_per_student       0.015594
                                ...
num_female_4yr_completers    -0.011040
pct_women                    -0.011865
PELL_EVER                    -0.023188
ADM_RATE                     -0.033600
median_debt_completers       -0.047290
Length: 94, dtype: float64
```

In [118]:
```
ms_repeated_kfolds(ncaa_grouped_all_merge, all_earn = 'no', dep = '6yr_male_mean_
```

```
Train Score: 0.9389020902012168
Test Score: -1.3449735117305157
Train RMSE:  0.05407919775418733
Test RMSE:  0.11353895996745102
est=0.778, cfg={'alpha': 0.01, 'l1_ratio': 0.2, 'max_iter': 100}
---
Train Score: 0.8641826164921715
Test Score: 0.8888254081464156
Train RMSE:  0.07186877416991137
Test RMSE:  0.09501260968999306
est=0.710, cfg={'alpha': 1, 'l1_ratio': 0.0, 'max_iter': 100}
---
Train Score: 0.8818212478957853
Test Score: 0.8441681309356999
Train RMSE:  0.07294987967678652
Test RMSE:  0.07642422755491489
est=0.688, cfg={'alpha': 0.1, 'l1_ratio': 0.1, 'max_iter': 100}
---
Train Score: 0.8800667014931102
```

In [119]:
```
e_net_male_sum, e_net_male_earn_kfold, cols = ms_elastic_net(ncaa_grouped_all_mer
                                             all_earn = 'no', dep = '6yr_male_mea
                                             params = 'yes', alpha = .01, l1_rati
```

```
Alpha:  0.01
L1 Ratio:  0.9
```




```
Train R2: 0.8854796382466371
Test R2: 0.8698220381433541
*****
Train MSE: 0.005219318448558694
Test MSE: 0.005411338984362156
*****
Train RMSE: 0.07224485067157863
Test RMSE: 0.07356180384113861
```

In [120]:
```
e_net_male_coefs_kfold = ms_eval_coefficients(e_net_male_earn_kfold, cols)
```

```
Total number of coefficients:  94
Coefficients close to zero:  76
Intercept: 10.774996154173587

avg_annual_cost_attendance    0.029097
1yr_declining_loan_balance    0.027679
tuition_rev_per_student       0.018028
avg_faculty_sal               0.015866
pct_asian                     0.015149
                                ...
pct_women                    -0.009419
num_female_4yr_completers    -0.010960
PELL_EVER                    -0.027908
ADM_RATE                     -0.040809
median_debt_completers       -0.053072
Length: 94, dtype: float64
```

Elastic Net with kfolds has a stronger fit and smaller errors, so we'll use those coefficients.

In [121]:
```
coef_df = pd.DataFrame(e_net_male_coefs_kfold)
coef_df.reset_index(inplace = True)
coef_df.rename(columns = {0:'coef', 'index':'feature'}, inplace = True)

coef_df['remove'] = coef_df.apply(lambda row: abs(row.coef) < 10**(-10), axis = 1
coef_df = coef_df[coef_df['remove'] == False]

coef_df.set_index('feature', inplace = True)
coef_df.drop(columns = 'remove', inplace = True)
```

In [122]:
```
coef_col_list_enet_male = coef_df.index.to_list()
```

In [123]:
```
coef_col_list_enet_male
```

Out[123]:
```
['avg_annual_cost_attendance',
 '1yr_declining_loan_balance',
 'tuition_rev_per_student',
 'avg_faculty_sal',
 'pct_asian',
 "Women's Team Average Annual Institutional Salary per FTE4",
 "Men's Team Average Annual Institutional Salary per FTE",
 'SAT_AVG',
 'REGION',
 'Football Total Participation',
 "Total Men's Team Participation",
 'Student Fees',
 'pct_mid_income_2',
 'pct_women',
 'num_female_4yr_completers',
 'PELL_EVER',
 'ADM_RATE',
 'median_debt_completers']
```

In [124]:
```
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet_male)
```

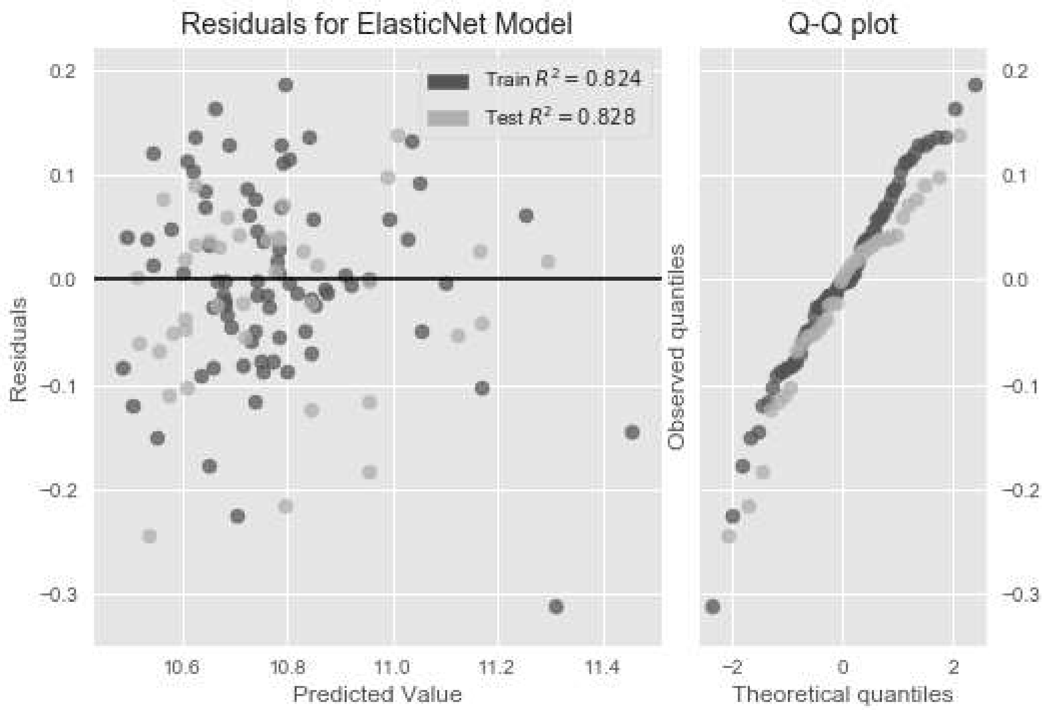
Out[124]:

	VIF	features
0	53.454039	avg_annual_cost_attendance
1	122.564607	1yr_declining_loan_balance
2	40.403964	tuition_rev_per_student
3	216.628722	avg_faculty_sal
4	4.481187	pct_asian
5	39.888409	Women's Team Average Annual Institutional Sala...
6	10.866895	Men's Team Average Annual Institutional Salary...
7	538.883877	SAT_AVG
8	21.044165	REGION
9	197.039039	Football Total Participation
10	47.431432	Total Men's Team Participation
11	3.215547	Student Fees
12	145.430784	pct_mid_income_2
13	140.764318	pct_women
14	10.233185	num_female_4yr_completers
15	98.411732	PELL_EVER
16	30.271274	ADM_RATE
17	69.642331	median_debt_completers

In [125]:
```
#dropping some correlated columns
coef_col_list_enet_male = ['avg_annual_cost_attendance',
 'tuition_rev_per_student',
 'pct_asian',
 "Men's Team Average Annual Institutional Salary per FTE",
 "Total Men's Team Participation",
 'Student Fees',
 'num_female_4yr_completers',
 'ADM_RATE']
```

In [126]:
```
e_net_male_sum_2, e_net_male_earn_2, cols = ms_elastic_net(ncaa_grouped_all_merge
                                             all_earn = 'no', dep = '6yr_male_mea
                                             params ='yes',  alpha = .01, l1_rati
                                            col_list = 'yes', cols = coef_col_lis
e_net_male_sum_2
```

```
Alpha:  0.01
L1 Ratio:  0.9
```




```
Train R2: 0.824482033819076
Test R2: 0.8279298819823292
*****
Train MSE: 0.00799931247959664
Test MSE: 0.007152744784084075
*****
Train RMSE: 0.08943887566151891
Test RMSE: 0.08457390131762917
```

Out[126]:

OLS Regression Results

Dep. Variable:	6yr_male_mean_earnings	**R-squared:**	0.831
Model:	OLS	**Adj. R-squared:**	0.812

Method:	Least Squares	**F-statistic:**	42.47
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	1.05e-23
Time:	11:51:49	**Log-Likelihood:**	79.152
No. Observations:	78	**AIC:**	-140.3
Df Residuals:	69	**BIC:**	-119.1
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	10.7750	0.011	1020.442	0.000	10.754	10.796
avg_annual_cost_attendance	0.0438	0.029	1.497	0.139	-0.015	0.102
tuition_rev_per_student	0.0006	0.025	0.023	0.981	-0.049	0.051
pct_asian	0.0593	0.013	4.425	0.000	0.033	0.086
Men's Team Average Annual Institutional Salary per FTE	0.0473	0.012	3.966	0.000	0.024	0.071
Total Men's Team Participation	0.0300	0.015	2.010	0.048	0.000	0.060
Student Fees	-0.0174	0.012	-1.498	0.139	-0.041	0.006
num_female_4yr_completers	-0.0659	0.014	-4.789	0.000	-0.093	-0.038
ADM_RATE	-0.0548	0.020	-2.739	0.008	-0.095	-0.015

Omnibus:	4.085	**Durbin-Watson:**	1.988
Prob(Omnibus):	0.130	**Jarque-Bera (JB):**	3.306
Skew:	0.399	**Prob(JB):**	0.192
Kurtosis:	3.616	**Cond. No.**	6.75

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Once again, we see some gender-specific columns, but fewer statistically significant features overall:

- **Percentage of asian students**: a one standard deviation increase in average cost of attendance leads to a 5.93% increase in earnings
- **Total Men's Team Participation**: 3.00% increase in earnings
- **Men's Team Average Annual Institutional Salary per FTE**: 4.73% increase in earnings
- **Number of female 4 year completers**: 6.59% decrease in earnings - likely a proxy for percentage of female students
- **Admission rate**: 5.48% decrease in earnings

We see demographics impacting the overall earnings of a school. We also see two gender-specific features, indicating that our model is likely valuing relevant features. Average male earnings are positively impacted by both total men's participation in sports, and the average salary of full time men's coaches. This could be a proxy for sports-related revenue and school resources. Since universities are non-profits, all profits from sports must be spent, and these profits are often spent on increased coaching salaries.

Demographics-Agnostic Regression

Lastly, we'll run a regression that excludes demographic data. We saw that the demographics of a school have large impacts on average earnings of graduates. This has more to do with structural societal trends than the actual school. No matter where a female student graduates from, she'll likely make less than her male peers. A similar statement can be made about asian graduates - they're much more likely to earn more than the average student. To attempt to isolate just the effects driven by the schools, I'll remove the demographic data and rerun the regression on 6 year average earnings.

In [127]:
```
ncaa_grouped_all_merge_no_demo = ncaa_grouped_all_merge.drop(columns = ['pct_asia
                                                                         'pct_black
                                                                         '5yr_decli
```

In [128]:
```
e_net_cv, cols = ms_elastic_net_cv(ncaa_grouped_all_merge_no_demo)
```

```
Train R2: 0.8933830693845575
Test R2: 0.8406682988951835
*****
Train MSE: 0.004319693706409302
Test MSE: 0.006187460714248668
*****
Train RMSE: 0.06572437680502799
Test RMSE: 0.07866041389573708
alpha: 0.010000
l1_ratio_: 0.790000
```

In [129]:
```
e_net_coefs_cv_no_demo = ms_eval_coefficients(e_net_cv, cols)
```

```
Total number of coefficients:  86
Coefficients close to zero:  74
Intercept: 10.676291138768727

avg_faculty_sal                                             0.045186
avg_annual_cost_attendance                                  0.040648
1yr_declining_loan_balance                                  0.035901
Women's Team Average Annual Institutional Salary per FTE4   0.013878
tuition_rev_per_student                                     0.010558
                                                              ...
3_yr_repayment_completers_rate                              0.000000
pct_mid_income_2                                           -0.002300
ADM_RATE                                                   -0.026469
PELL_EVER                                                  -0.030444
median_debt_completers                                     -0.045611
Length: 86, dtype: float64
```

In [130]:
```
coef_df = pd.DataFrame(e_net_coefs_cv_no_demo)
coef_df.reset_index(inplace = True)
coef_df.rename(columns = {0:'coef', 'index':'feature'}, inplace = True)

coef_df['remove'] = coef_df.apply(lambda row: abs(row.coef) < 10**(-10), axis = 1
coef_df = coef_df[coef_df['remove'] == False]

coef_df.set_index('feature', inplace = True)
coef_df.drop(columns = 'remove', inplace = True)
```

In [131]:
```
coef_col_list_enet_nodemo = coef_df.index.to_list()
```

In [132]:
```
coef_col_list_enet_nodemo
```

Out[132]:
```
['avg_faculty_sal',
 'avg_annual_cost_attendance',
 '1yr_declining_loan_balance',
 "Women's Team Average Annual Institutional Salary per FTE4",
 'tuition_rev_per_student',
 'Football Total Participation',
 ' Womens_opex_per_participant ',
 'REGION',
 'pct_mid_income_2',
 'ADM_RATE',
 'PELL_EVER',
 'median_debt_completers']
```

In [133]:
```
#identifying highly collinear features
ms_vif(ncaa_grouped_all_merge, id_cols = 'yes', cols =coef_col_list_enet_nodemo)
```

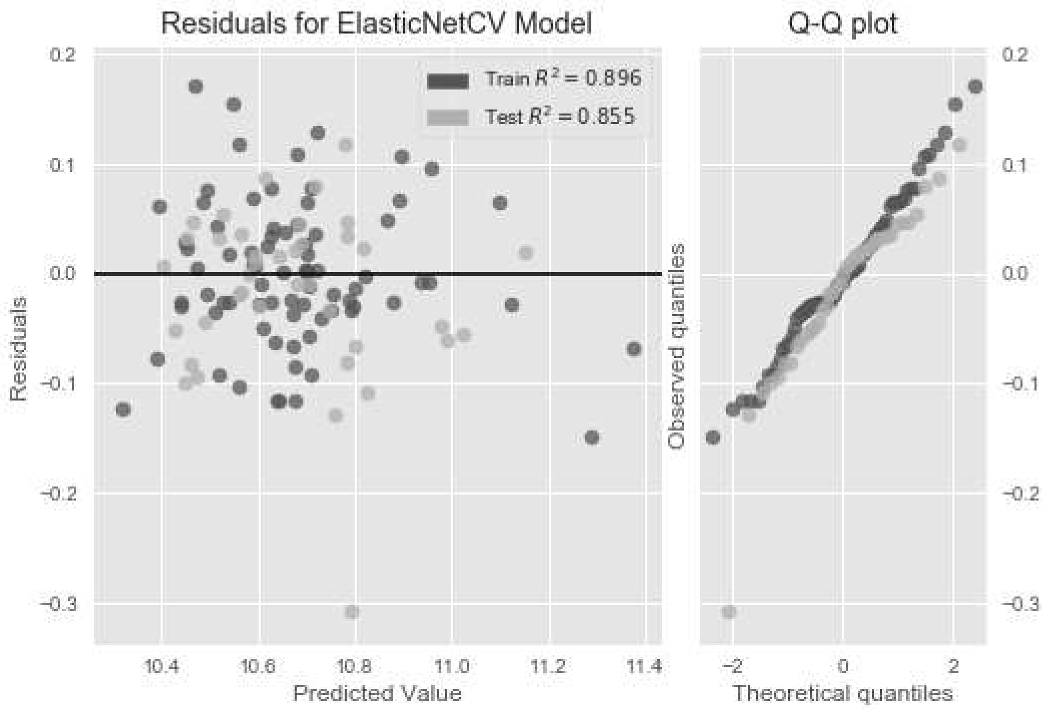
Out[133]:

	VIF	features
0	96.465729	avg_faculty_sal
1	37.012656	avg_annual_cost_attendance
2	82.441654	1yr_declining_loan_balance
3	29.994636	Women's Team Average Annual Institutional Sala...
4	33.471197	tuition_rev_per_student
5	151.040251	Football Total Participation
6	17.222587	Womens_opex_per_participant
7	15.973741	REGION
8	132.529713	pct_mid_income_2
9	24.671067	ADM_RATE
10	57.567082	PELL_EVER
11	63.915986	median_debt_completers

In [134]:
```
#removing highly collinear features
coef_col_list_enet_nodemo = ['avg_faculty_sal',
 'avg_annual_cost_attendance',
 '1yr_declining_loan_balance',
 'Football Total Participation',
 ' Womens_opex_per_participant ',
 'REGION',
 'pct_mid_income_2',
 'ADM_RATE',
 'PELL_EVER',
 'median_debt_completers']
```

In [135]:
```
e_net_nodemo_sum, e_net_nodemo, cols = ms_elastic_net(ncaa_grouped_all_merge, cv
                                                        col_list = 'yes', cols = co
                                                       params = 'yes', alpha = 0.01
e_net_nodemo_sum
```

```
Alpha:  0.01
L1 Ratio:  0.79
```




```
Train R2: 0.8962817476750817
Test R2: 0.8551650546906062
*****
Train MSE: 0.004202250798456468
Test MSE: 0.0056244961168316916
*****
Train RMSE: 0.0648247699452645
Test RMSE: 0.07499664070364546
```

Out[135]:

OLS Regression Results

Dep. Variable:	6yr_mean_earnings	**R-squared:**	0.898
Model:	OLS	**Adj. R-squared:**	0.883
Method:	Least Squares	**F-statistic:**	58.84
Date:	Sat, 31 Jul 2021	**Prob (F-statistic):**	3.04e-29
Time:	11:51:56	**Log-Likelihood:**	103.30
No. Observations:	78	**AIC:**	-184.6
Df Residuals:	67	**BIC:**	-158.7
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]

const	10.6763	0.008	1357.866	0.000	10.661	10.692
avg_faculty_sal	0.0537	0.014	3.798	0.000	0.025	0.082
avg_annual_cost_attendance	0.0500	0.015	3.271	0.002	0.019	0.081
1yr_declining_loan_balance	0.0395	0.015	2.675	0.009	0.010	0.069
Football Total Participation	0.0135	0.009	1.450	0.152	-0.005	0.032
Womens_opex_per_participant	0.0095	0.009	1.017	0.313	-0.009	0.028
REGION	0.0088	0.011	0.783	0.437	-0.014	0.031
pct_mid_income_2	-0.0089	0.009	-0.979	0.331	-0.027	0.009
ADM_RATE	-0.0218	0.016	-1.387	0.170	-0.053	0.010
PELL_EVER	-0.0422	0.014	-3.021	0.004	-0.070	-0.014
median_debt_completers	-0.0488	0.012	-3.978	0.000	-0.073	-0.024

Omnibus:	0.677	**Durbin-Watson:**	1.815
Prob(Omnibus):	0.713	**Jarque-Bera (JB):**	0.473
Skew:	-0.191	**Prob(JB):**	0.790
Kurtosis:	3.006	**Cond. No.**	5.57

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model Evaluation

The main goal of this analysis was to find what features about a school - things like demographics, sports and academic expenditures, loan rates and debt totals - have the largest impact on average earnings of graduates, 6 years after they first enrolled. This was not a typical machine learning project where we focused on how accurately we can predict earnings, but rather more of a statistical analysis to help determine what drives changes in earnings.

In this section however, I did want to call out the performance of our predictive models. The model with the best predictive power was the **Ridge Regression** model, using the remaining coefficients after running the dataset through a Lasso Regression model:

Ridge Model

- Train R2: 0.9458398454724065
- Test R2: 0.9000704286141612

- Train MSE: 0.002194353910776663
- Test MSE: 0.0038806483132620842

- Train RMSE: 0.04684393141887929
- Test RMSE: 0.06229484981330386

The Ridge model produced the smallest error term and had a very even fit on both the train and test data. This model contained 39 coefficients.

The 2nd best model was the **Elastic Net Grid Search** model, which has the benefit of combining both Ridge and Lasso penalties:

Elastic Net Grid Search

- Train R2: 0.9072072159359033
- Test R2: 0.9001394280336424

- Train MSE: 0.0037595943065333237
- Test MSE: 0.0038779688013106947

- Train RMSE: 0.06131553071231891
- Test RMSE: 0.06227333941030218

- alpha: 0.100000
- l1_ratio_: 0.070000

This model had a more even fit than the Ridge model, but the error terms were slighlty higher. What makes the Elastic Net model so useful is that you do not need to run a Lasso Regression first, and then feed the remaining coefficients into the model. It zeroes out coefficients on its own, creating a more streamlined process. Using either model would result in accurate predictions of future earnings.

Conclusion

Having removed demographics data from the final model, I've tried to isolate what school-specific effects there are on earnings. The percentage of women at a school will bring down average earnings because women earn less than men when entering the job market, for example.

In this isolated model, we have 5 remaining coefficients.

These changes are an average when holding all else equal:

- **Average faculty salary**: a one standard deviation increase in faculty salary leads to a 5.37% increase in earnings
- **Average cost of attendance**: 5.0% increase in earnings
- **Percentage of students who have a declining balance after 1 year post-grad**: 3.95% increase in earnings
- **Percentage of students who were on a pell grant**: 4.22% decrease in earnings
- **Median debt of 4 year (150% time) completers**: 4.88% decrease in earnings

These coefficients, although few in number, are valuable pieces of information. They can also be proxies for other features about a school. Average faculty salary and cost of attendance speak to the resources a school has. The higher the cost, the more money there is to pay for the best instructional talent. The percentage of students who have a declining loan balance after 1 year points to how effective graduates are at getting jobs and earning a decent salary. Students earning more after graduation are more likely to pay off more of their debt. As the debt load increases however, we see that this is associated with lower average earnings. More debt hamstrings graduates, and may force them to take a less desirable job simply so they can begin to make payments.

Taking a look at the first model which included demographic data, it was striking to see to what extent the percentage of women at a school has on average earnings 6 years from enrollment. The fact that an increase of 4.8% in the number of women at a school can reduce average earnings by over 3% speaks to how women are facing a constant uphill battle when it comes to pay.

As we saw at the top of the notebook, Duke graduates produced the highest average male and female earnings. However, the average male at Duke earned nearly 112K 6 years from enrollment, while the average female earned only 76K. This is structural inequality that cannot be solved by any one school's actions.

In the end, we found that college sports expenditures do have a statistically significant relationship with earnings in a limited capacity. In our general model, which explains 90% of the variance in average earnings, we found that Football revenue has a positive impact on earnings. This could be a proxy for school resources, or maybe even the effect of sending more athletes to the NFL for example. Either way, schools that make more from football tend to have increased earnings and better outcomes.

The results were even more interesting when estimating earnings by gender. For both men and women, the average salaries of full time employees who worked for the men's and women's teams had positive effects on earnings. The positive effects mean different things due to the large gender pay gap. A one standard deviation increase in average women's teams salaries leads to a 3.38% increase in average female earnings. This is likely an indicator of school resources, but also the amount of resources a school is willing to invest in its female students and student-athletes. Women who are better supported at these schools are better off after graduating.

On the other side, men's sports are far and away more supported that women's sports - in college and professional levels. Therefore, the earnings increase gained from increased full time employee salaries is likley more of a proxy for school resources. Men's teams, especially football and basketball, already get the best resources, coaches, and facilities that money can buy. Team employee salaries are tied to on-field success (more profits have to go somewhere), which continue to grow as schools earn more money.

Additionally, we were able to identify some schools, like Alabama, Ohio State, Penn State, and Florida State, which spend far more on sports than their peers, but produce worse student outcomes. There is clearly a balance that a school can strike between funding a profitable and lucrative sports program, and investing enough in academics to ensure the success of its student body.

Future Work

Further analysis would focus on schools that spend higher than average amounts on sports but produce below average earnings outcomes. How does their academic spending compare to their peers who produce better earnings outcomes? What areas of performance are they falling behind? The datasets used here were very detailed when it came to sports expenditures, but less so on academic spending. I'd like to find more detailed academic data to see which areas have the largest effects on earnings.

Additionally, I want access to more current data. Because they latest earnings were from 2014-15, this meant the students whose earnings we have started college back in 08-09 or 09-10. Much about the economy has changed over the last 10 years, and it would be illustrative to look at these same features but with newer data points.

Lastly, I want to figure out how to tune the model even further in a way that might preserve more features. However, it appears as if many features about a school just don't have big effects on earnings. As we saw with female earnings, so much of what you make at your job comes down to luck and societal problems that aren't easily fixable.