IST-718 Homework/Lab-1 (Week3)

Prof. Jonathan M Fox
Student: Sharat Sripada (vssripad)

Introduction

This case study provides an opportunity to demonstrate the ability to combine datasets and produce meaningful analysis learned through the first three-weeks of the course.

Specifically, we are looking to answer the following questions:

- What would his salary be if we were still in the Big East? What if we went to the Big Ten?
- What schools did we drop from our data, and why?
- What effect does graduation rate have on the projected salary?
- How good is our model?
- What is the single biggest impact on salary size?

Dataset

The dataset initially related to Coaches data (an excerpt is seen below) was loaded into a data-frame **df_coaches**. This would be the master data-frame:

```
import pandas as pd


# Read the coaches data using pandas
df_coaches = pd.read_csv('Coaches9.csv', sep = ',')
print ('Shape of data: \n', df_coaches.shape)
df_coaches.head(50)
```

```
Shape of data:
 (129, 9)
```

	School	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout
0	Air Force	Mt. West	Troy Calhoun	885000	885000	247000	--	$0	--
1	Akron	MAC	Terry Bowden	$411,000	$412,500	$225,000	$50,000	$0	$688,500
2	Alabama	SEC	Nick Saban	$8,307,000	$8,307,000	$1,100,000	$500,000	$0	$33,600,000
3	Alabama at Birmingham	C-USA	Bill Clark	$900,000	$900,000	$950,000	$165,471	$0	$3,847,500
4	Appalachian State	Sun Belt	Scott Satterfield	$712,500	$712,500	$295,000	$145,000	$0	$2,160,417
5	Arizona	Pac-12	Kevin Sumlin	$1,600,000	$2,000,000	$2,025,000	--	$0	$10,000,000

NOTE:
Two additional datasets related to Graduation success Rate (GSR)/Federal Graduation Rate (FGR) and Stadium capacities were merged into the master data-frame.

First, data related to GSR/FGR was retrieved from https://web3.ncaa.org/aprsearch/gsrsearch and loaded into a data-frame **df_gsr_fgr**:

```python
# Read data of Graduation success Rate (GSR) and Federal Graduation Rate (FGR)
# Definitions:
#   |- FGR - Indicates the percentage of freshmen who entered and received athletics aid during a given
#      academic year who graduated within six years
#   |- GSR - The GSR adds to the first-time freshmen, those students who entered midyear as well as
#      student-athletes who transferred into an institution and received athletics aid.

df_gsr_fgr = pd.read_csv('Football_GSR_FGR.csv', sep = '\t',
                         names=['Cohort Year', 'School', 'Conference', 'Sport', 'State',
                                'GSR', 'FGR'])
df_gsr_fgr.head()
```

	Cohort Year	School	Conference	Sport	State	GSR	FGR
0	2007	Abilene Christian University	Southland Conference	Football	TX	51	48.0
1	2007	University of Akron	Mid-American Conference	Football	OH	60	55.0
2	2007	Alabama A&M University	Southwestern Athletic Conf.	Football	AL	39	50.0
3	2007	Alabama State University	Southwestern Athletic Conf.	Football	AL	64	47.0
4	2007	University of Alabama	Southeastern Conference	Football	AL	80	60.0

Finally, data related to stadium sizing was retrieved from source https://www.collegegridirons.com/comparisons-by-capacity/ and loaded into data-frame **df_stadiums:**

```python
# Get stadium-size from mapping to school/college from college-stadiums.csv
# Data-source:
#     |- https://www.collegegridirons.com/comparisons-by-capacity/

df_stadiums = pd.read_csv('college-stadiums.csv', sep='\t')
df_stadiums.head()
```

	Stadium	College	Conference	Capacity	Opened
0	Michigan Stadium	Michigan	Big Ten	107,601	1927
1	Beaver Stadium	Penn State	Big Ten	106,572	1960
2	Ohio Stadium	Ohio State	Big Ten	104,944	1922
3	Kyle Field	Texas A&M	SEC	102,733	1904
4	Neyland Stadium	Tennessee	SEC	102,521	1921

Data cleaning/munging

Data was cleaned and munged in steps.

STEP-1:
On data-frame **df_gsr_fgr** missing values were replaced with statistical mean for the respective GSR/FGR data. This may not always be the best approach and could potentially skew data. Also, we convert the datatypes for the GSR and FGR columns to type int:

```python
# Clean/munge data related to gsr/fgr
from numpy import mean
print('--- Working on dataframe: df_gsr_fgr --- \n')

# Replace all NaN/missing values with mean for respective columns
column_mean = lambda column: mean(df_gsr_fgr.loc[:, column])
values = {'GSR': column_mean('GSR'), 'FGR': column_mean('FGR')}
df_gsr_fgr = df_gsr_fgr.fillna(values)

# Verify no NaN values
print('Missing data in dataset:', df_gsr_fgr.isna().any().any())

# Check datatypes on GSR/FGR columns in specific
print('Data-types: \n', df_gsr_fgr.dtypes)

# Modify the FGR data astype to int64 as well
df_gsr_fgr['FGR'] = df_gsr_fgr['FGR'].astype(int)

print('Data-types after conversion: \n', df_gsr_fgr.dtypes)
```

```
--- Working on dataframe: df_gsr_fgr ---

Missing data in dataset: False
Data-types:
 Cohort Year      int64
School          object
Conference      object
Sport           object
State           object
GSR              int64
FGR            float64
dtype: object
Data-types after conversion:
 Cohort Year     int64
School         object
Conference     object
Sport          object
State          object
GSR             int64
FGR             int64
dtype: object
```

1) Data-frame **df_coaches** were munged for the 'Capacity' column replacing characters to make them numbers for data-analysis:

```python
for school in df_coaches['School']:
    try:
        capacity = int(df_stadiums[(df_stadiums['College'] == school)]['Capacity'].values[0].replace(',',''))
    except IndexError:
        no_data += 1
        capacity = 0
    pop_value('Capacity', school, capacity)
```

Data from the two data-frames were then merged into the master data-frame **df_coaches** before preparing the data for visualization and modelling:

- replace missing values/zeros with mean/avg where applicable
- check data types & modify appropriately using *astype()*
- remove '$' symbol using Python regex and convert currency columns - *SchoolPay, TotalPay, Bonus, BonusPaid, AssistantPay, Buyout* to float.

A sample of the merged data-frame and columns is seen below:

	School	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	State	Capacity
9	Army	Ind.	Jeff Monken	932521.0	932521.0	0.0	0.0	0.0	0.0	0	0	USA	38000.0
35	Fresno State	Mt. West	Jeff Tedford	1550000.0	1550000.0	2765000.0	1240000.0	0.0	5440000.0	0	0	USA	41031.0
38	Georgia State	Sun Belt	Shawn Elliott	569000.0	569000.0	220000.0	60000.0	0.0	1500000.0	0	0	USA	23000.0
39	Georgia Tech	ACC	Paul Johnson	3060018.0	3060018.0	1330000.0	225000.0	0.0	4000000.0	0	0	USA	55000.0
52	Louisiana-Lafayette	Sun Belt	Billy Napier	850000.0	850000.0	435000.0	0.0	0.0	2671875.0	0	0	USA	31000.0
53	Louisiana-Monroe	Sun Belt	Matt Viator	390000.0	390000.0	50000.0	0.0	0.0	175000.0	0	0	USA	30427.0
55	LSU	SEC	Ed Orgeron	3500000.0	3500000.0	1575000.0	100000.0	0.0	5291667.0	0	0	USA	100500.0
69	Navy	AAC	Ken Niumatalolo	2163000.0	2163000.0	0.0	0.0	0.0	0.0	0	0	USA	34000.0
88	Penn State	Big Ten	James Franklin	4800000.0	4800000.0	1000000.0	300000.0	0.0	18375000.0	0	0	USA	106572.0
95	South Alabama	Sun Belt	Steve Campbell	600000.0	600000.0	295000.0	0.0	0.0	918333.0	0	0	USA	40646.0

oll output; double click to hide

Data types on all columns within data-frame **df_coaches**:

```
School           object
Conference       object
Coach            object
SchoolPay       float64
TotalPay        float64
Bonus           float64
BonusPaid       float64
AssistantPay    float64
Buyout          float64
GSR               int64
FGR               int64
State            object
Capacity        float64
```

Removal of data-records

During the process of data-retrieval there were instances where few records had to be removed. For instance:

- when retrieving data related to GSR/FGR certain schools from the Coaches dataset could not be matched or found. This resulted in *EIGHTEEN* records being removed or dropped
- similarly, *FIFTEEN* schools could not be matched or found in stadium capacity data. These were removed too.

NOTE:
The original data-frame was preserved, and all the trimming was saved to df_coaches_trim

After all, the size of Coaches data is shown below – comprising up to *SEVENTEEN* fewer records:

```
df_coaches shape: (129, 13) vs df_coaches_trim shape: (112, 13)
Missing values in df_coaches dataframe: False
```

Exploratory Data Analysis

Since the problem we are trying to solve is related to predicting the Salary of coaches some of the exploratory data analysis or descriptive statistics and visualization are centered around the column TotalPay from the dataset.

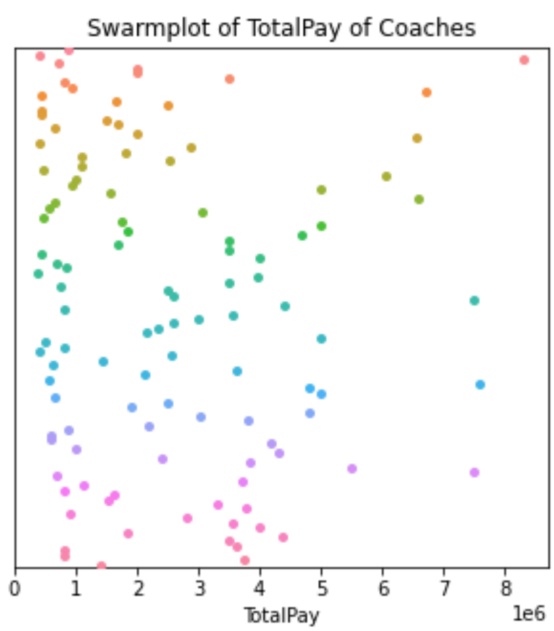
Seen below are the min, max, mean and median values related to *TotalPay* (in USD) paid to Coaches and another column that seemed to have a strong correlation to it, *Capacity* of stadiums:

```
# Exploratory data analysis
df_coaches_trim.describe()

# Of interest are the following:
# - TotalPay (USD) Min, Max, Mean and Median - 390,000, 8,307,000, 2,503,266 & 2,000,000 respectively
# - Capacity of stadiums Min, Max, Mean and Median - 15000, 107601, 51944, 50000 respectively
```

	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	Capacity
count	1.120000e+02	1.120000e+02	1.120000e+02	1.120000e+02	112.0	1.120000e+02	112.000000	112.000000	112.000000
mean	2.503266e+06	2.510810e+06	7.872615e+05	1.129687e+05	0.0	7.574159e+06	61.892857	51.482143	51944.803571
std	1.904339e+06	1.908795e+06	6.743276e+05	2.204314e+05	0.0	1.046263e+07	24.310068	20.661573	23597.660017
min	3.900000e+05	3.900000e+05	0.000000e+00	0.000000e+00	0.0	0.000000e+00	0.000000	0.000000	15000.000000
25%	8.244225e+05	8.246850e+05	2.837500e+05	0.000000e+00	0.0	8.936140e+05	57.000000	47.000000	30564.000000
50%	2.000000e+06	2.000000e+06	7.100000e+05	3.712500e+04	0.0	3.092813e+06	68.000000	56.000000	50000.000000
75%	3.640825e+06	3.640825e+06	1.106250e+06	1.062500e+05	0.0	1.032344e+07	74.500000	64.000000	64403.500000
max	8.307000e+06	8.307000e+06	3.100000e+06	1.350000e+06	0.0	6.812500e+07	99.000000	93.000000	107601.000000

This distribution and variance can also be seen in a swarm-plot. This particularly indicates the larger set of salaries distributed *<4mi USD* and just one data-point *>8mi USD*:



The min and max *TotalPay* values were used to retrieve data of Coaches who were paid the least and most salaries:

```
min_TotalPay = df_coaches_trim.describe()['TotalPay'].values[3]
max_TotalPay = df_coaches_trim.describe()['TotalPay'].values[-1]

# Coach Matt Viator from Louisiana-Monroe was paid the least Total Salary
df_coaches_trim[df_coaches_trim['TotalPay'] == min_TotalPay]
```

	School	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	State	Capacity
53	Louisiana-Monroe	Sun Belt	Matt Viator	390000.0	390000.0	50000.0	0.0	0.0	175000.0	0	0	USA	30427.0

```
# Coach Nick Saban from Alabama was paid the highest Total Salary
df_coaches_trim[df_coaches_trim['TotalPay'] == max_TotalPay]

# NOTE - Are we starting to see some correlation with Stadium Capacity(?)
```

	School	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	State	Capacity
2	Alabama	SEC	Nick Saban	8307000.0	8307000.0	1100000.0	500000.0	0.0	33600000.0	39	50	AL	101821.0

Results

Model-1

The first model will attempt to predict the salary of Coaches (here *TotalPay*) using Linear Regression/commonly known Ordinary Least squares method with variables (GSR, FGR, Capacity) from the dataset.

A summary of the model-fit and results as follows:

```
# Split the data into training and test set
np.random.seed(1000)
df_coaches_trim['runiform'] = uniform.rvs(loc = 0, scale = 1, size = len(df_coaches_trim))
df_coaches_trim_train = df_coaches_trim[df_coaches_trim['runiform'] >= 0.33]
df_coaches_trim_test = df_coaches_trim[df_coaches_trim['runiform'] < 0.33]

print('df_coaches dataframe train data (size): \n', df_coaches_trim_train.shape)
print('df_coaches dataframe test data (size): \n', df_coaches_trim_test.shape)

# Let's run a linear regression model using the ordinary least squares method (OLS)
ols_model = str('TotalPay ~ GSR + FGR + Capacity')

# Fit the model on train data
train_model_fit = smf.ols(ols_model, data = df_coaches_trim_train).fit()
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               TotalPay   R-squared:                       0.756
Model:                            OLS   Adj. R-squared:                  0.746
Method:                 Least Squares   F-statistic:                     73.28
Date:                Sat, 30 Jan 2021   Prob (F-statistic):           1.08e-21
Time:                        21:23:13   Log-Likelihood:                -1140.7
No. Observations:                  75   AIC:                             2289.
Df Residuals:                      71   BIC:                             2299.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept  -2.041e+06   5.33e+05     -3.829      0.000    -3.1e+06   -9.78e+05
GSR        -8909.0952   1.77e+04     -0.504      0.616   -4.42e+04    2.64e+04
FGR         2.686e+04   2.11e+04      1.276      0.206   -1.51e+04    6.88e+04
Capacity      72.2476      4.936     14.637      0.000      62.405      82.090
==============================================================================
Omnibus:                        8.344   Durbin-Watson:                   1.988
Prob(Omnibus):                  0.015   Jarque-Bera (JB):               10.369
Skew:                          -0.479   Prob(JB):                      0.00560
Kurtosis:                       4.549   Cond. No.                     2.71e+05
==============================================================================
```

Interpreting the results:

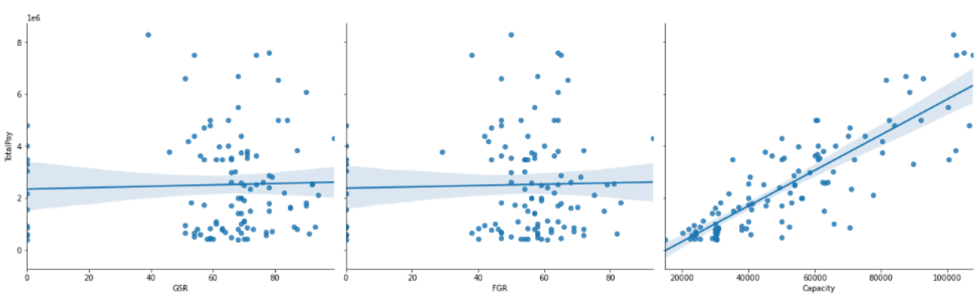
- The equation of linear regression can be written as:
  *TotalPay = 72.2476 * Capacity + 2.686e+04 * FGR - 8909.0952 * GSR - 2.041e+06*

- <u>R-square value</u>

  R-square = 0.756 is an indication of a good fit and is reflective of how close the data is to line of best-fit/regression.

- <u>P-value</u>
  - P-value for GSR/FGR >0.05 and may therefore not be statistically significant in this dataset.
  - Stadium Capacity with P-value = 0 is statistically significant.

Further, a plot of the line of least squares for each of the variables was made:



Based on the linear regression model/Model-1, the prediction of Coach Dino Babers' salary is approximated to **2,460,942USD** about 2.5% more than his current pay.

```
# Based on the linear regression model we predict Coach Dino Babers' salary at 2,460,942 USD

df_coaches_trim_test[df_coaches_trim_test['School'] == 'Syracuse']
```

	School	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	State	Capacity	runiform	Salary(Predict)
102	Syracuse	ACC	Dino Babers	2401206.0	2401206.0	0.0	0.0	0.0	0.0	78	61	NY	49250.0	0.236943	2.460942e+06

Model-2

This model will make a similar prediction as the first, by including an additional response variable *Conference*. Since this variable is categorical and non-numeric the values in the dataset will be mapped and converted to numeric data using a key-value map or Python dictionary:

```
{'Mt. West': 0, 'MAC': 1, 'SEC': 2, 'Sun Belt': 3, 'Pac-12': 4, 'Ind.': 5, 'ACC': 6, '
AAC': 7, 'C-USA': 8, 'Big Ten': 9, 'Big 12': 10}
```

A summary of the model-fit and results as follows:

```python
# Let's run a linear regression model using the ordinary least squares method (OLS)
ols_model_enc = str('TotalPay ~ GSR + FGR + Capacity + Conference_num')

# Fit the model on train data
train_model_fit_enc = smf.ols(ols_model_enc, data = df_coaches_trim_train_enc).fit()

print(train_model_fit_enc.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               TotalPay   R-squared:                       0.776
Model:                            OLS   Adj. R-squared:                  0.764
Method:                 Least Squares   F-statistic:                     60.75
Date:                Sat, 30 Jan 2021   Prob (F-statistic):           4.84e-22
Time:                        22:11:22   Log-Likelihood:                -1137.4
No. Observations:                  75   AIC:                             2285.
Df Residuals:                      70   BIC:                             2296.
Df Model:                           4
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept      -2.191e+06   5.17e+05     -4.236      0.000   -3.22e+06   -1.16e+06
GSR            -1.745e+04   1.74e+04     -1.004      0.319   -5.21e+04    1.72e+04
FGR             3.498e+04   2.05e+04      1.702      0.093   -6004.994     7.6e+04
Capacity          69.0914      4.919     14.047      0.000      59.281      78.902
Conference_num  8.721e+04   3.44e+04      2.532      0.014    1.85e+04    1.56e+05
==============================================================================
Omnibus:                        2.448   Durbin-Watson:                   2.073
Prob(Omnibus):                  0.294   Jarque-Bera (JB):                1.940
Skew:                          -0.109   Prob(JB):                        0.379
Kurtosis:                       3.757   Cond. No.                     2.73e+05
==============================================================================
```

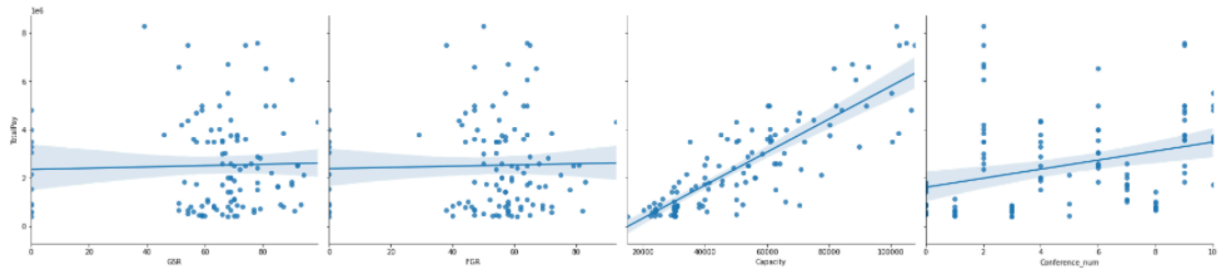
Interpreting the results:

- <u>R-square value</u>

  A marginally better R-square = 0.776

- <u>P-value</u>
  - P-value conference_num <0.05 indicates being statistically significant.

Pair plots of the line of least squares updated with Conference:



Based on the linear regression model/Model-2, the prediction of Coach Dino Babers' salary is approximated to **2,769,900USD** if Syracuse was in Big-Ten conference.

In [355]:
```
# Based on the linear regression model we predict Coach Dino Babers' salary at 2,769,900 USD if we
# transfered to Big-Ten conference

df_coaches_trim_test_enc[df_coaches_trim_test_enc['School'] == 'Syracuse']
```

Out[355]:

	Conference	Coach	SchoolPay	TotalPay	Bonus	BonusPaid	AssistantPay	Buyout	GSR	FGR	State	Capacity	runiform	Conference_num	Salary(Predict)
	ACC	Dino Babers	2401206.0	2401206.0	0.0	0.0	0.0	0.0	78	61	NY	49250.0	0.236943	9.0	2.769900e+06

NOTE:
Conference_num is changed to 9 which maps to the 'Big Ten' conference per the dictionary.

Conclusions

The conclusions are mostly presented by answering the questions we originally sought out to answer and covered to some depth throughout this document.

Questions:

- *What would his salary be if we were still in the Big East? What if we went to the Big Ten?*
  The dataset across all the data-frames did not comprise any data from Big East.

  However, taking column *Conference* into consideration when training the model, prediction has it that Coach Dino Babers' salary could be ~15% higher and approximately be **2,769, 900USD**.

- *What schools did we drop from our data, and why?*
  As stated earlier in the *Data cleaning/munging section*, had to drop 17-rows or schools owing to the following:
  - when retrieving data related to GSR/FGR certain schools from the Coaches dataset could not be matched or found. This resulted in *EIGHTEEN* records being removed or dropped
  - similarly, *FIFTEEN* schools could not be matched or found in stadium capacity data. These were removed too.

- *What effect does graduation rate have on the projected salary?*
  Graduation rate or GSR does not seem to have a significant role. This is manifested by high P-values and the line of least square plots in the *Results section*.

- *How good is our model?*
  One of the measures of a good linear regression model is an R-square value tending towards 1.0 and we have both models yield >0.75. making a moderately good model.

- *What is the single biggest impact on salary size?*
  Capacity of stadiums has the largest impact on the Salary of coaches. This is corroborated with low P-values and low variance when plotted on the line of least squares.

  Speaking in more general terms, bigger stadiums or more seating capacity would certainly put more revenue into the sport as it promotes bolstering the local economy (sale of merchandise, food/beverages, car-parking etc.) and this likely drives the stakes up for sustaining good Team performances, better Coaches and therefore higher Coach salaries.