Simple Linear Regression and Multiple Regression

You are a data analyst for a basketball team and have access to a large set of historical data that you can use to analyze performance patterns. The coach of the team and your management have requested that you come up with regression models that predict the total number of wins for a team in the regular season based on key performance metrics. Although the data set is the same that you used in the previous projects, the data set used here has been aggregated to study the total number of wins in a regular season based on performance metrics shown in the table below. These regression models will help make key decisions to improve the performance of the team. You will use the Python programming language to perform the statistical analyses and then prepare a report of your findings to present for the team's management. Since the managers are not data analysts, you will need to interpret your findings and describe their practical implications.

Variable	What does it represent
total_wins	Total number of wins in a regular season
avg_pts	Average points scored in a regular season
avg_elo_n	Average relative skill of each team in a regular season
avg_pts_differential	Average point differential between the team and their opponents in a regular season
avg_elo_differential	Average relative skill differential between the team and their opponent in a regular season

The average relative skill (represented by the variable **avg_elo_n** in the data set) is simply the average of a team's relative skill in a regular season. Relative skill is measured using the ELO rating. This measure is inferred based on the final score of a game, the game location, and the outcome of the game relative to the probability of that outcome. The higher the number, the higher the relative skill of a team.

Step 1: Data Preparation

This step uploads the data set from a CSV file and transforms the data into a form that will be used to create regression models. The data will be aggregated to calculate the number of wins for teams in a basketball regular season between the years 1995 and 2015.

In [ ]:
```
import numpy as np
import pandas as pd
import scipy.stats as st
import matplotlib.pyplot as plt
from IPython.display import display, HTML

# dataframe for this project
nba_wins_df = pd.read_csv('nba_wins_data.csv')  # read the csv file

display(HTML(nba_wins_df.head().to_html()))     # display the first five row
print("printed only the first five observations...")
print("Number of rows in the dataset =", len(nba_wins_df))
```

	year_id	fran_id	avg_pts	avg_opp_pts	avg_elo_n	avg_opp_elo_n	avg_
0	1995	Bucks	99.341463	103.707317	1368.604789	1497.311587	
1	1995	Bulls	101.524390	96.695122	1569.892129	1488.199352	
2	1995	Cavaliers	90.451220	89.829268	1542.433391	1498.848261	
3	1995	Celtics	102.780488	104.658537	1431.307532	1495.936224	
4	1995	Clippers	96.670732	105.829268	1309.053701	1517.260260	

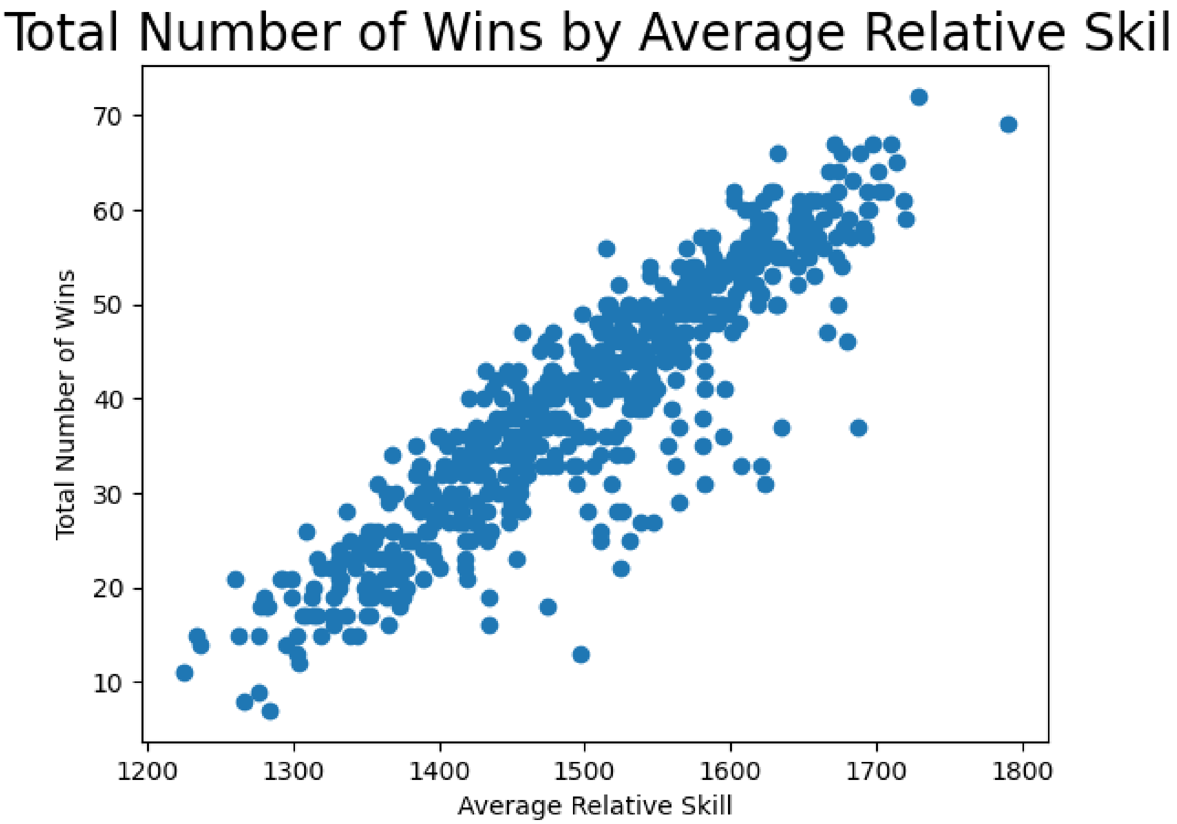
```
printed only the first five observations...
Number of rows in the dataset = 618
```

Step 2: Scatterplot and Correlation for the Total Number of Wins and Average Relative Skill

Your coach expects teams to win more games in a regular season if they have a higher average relative skill compared to their opponents. This is because the chances of winning are higher if a team can maintain high average relative skill. Therefore, it is expected that the total number of wins and the average relative skill are correlated. Calculate the Pearson correlation coefficient and its P-value.

In [ ]:
```
import scipy.stats as st

# ---- TODO: make your edits here ----
plt.plot(nba_wins_df['avg_elo_n'], nba_wins_df['total_wins'], 'o')

plt.title('Total Number of Wins by Average Relative Skill', fontsize=20)
plt.xlabel('Average Relative Skill')
plt.ylabel('Total Number of Wins')
plt.show()
```

```python
# ---- TODO: make your edits here ----
correlation_coefficient, p_value = st.pearsonr(nba_wins_df['avg_elo_n'], nba

print("Correlation between Average Relative Skill and the Total Number of Wi
print("Pearson Correlation Coefficient =",  round(correlation_coefficient,4)
print("P-value =", round(p_value,4))
```




```
Correlation between Average Relative Skill and the Total Number of Wins
Pearson Correlation Coefficient = 0.9072
P-value = 0.0
```

Step 3: Simple Linear Regression: Predicting the Total Number of Wins using Average Relative Skill

The coach of your team suggests a simple linear regression model with the total number of wins as the response variable and the average relative skill as the predictor variable. He expects a team to have more wins in a season if it maintains a high average relative skill during that season. This regression model will help your coach predict how many games your team might win in a regular season. Create this simple linear regression model.

In [ ]:
```python
import statsmodels.formula.api as smf
```

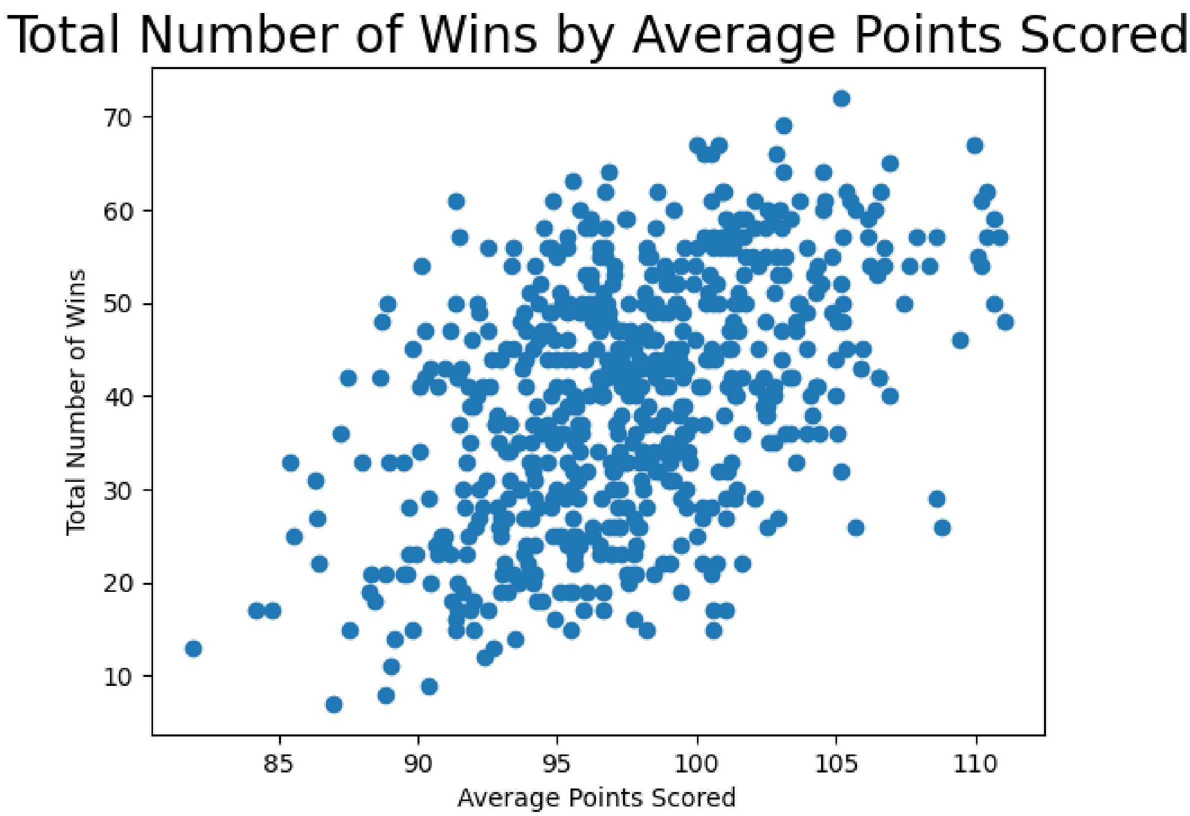
```python
# Simple Linear Regression
# ---- TODO: make your edits here ---
model1 = smf.ols('total_wins ~ avg_elo_n', nba_wins_df).fit()
print(model1.summary())
```

```
                            OLS Regression Results
==============================================================================
==
Dep. Variable:             total_wins   R-squared:                       0.8
23
Model:                            OLS   Adj. R-squared:                  0.8
23
Method:                 Least Squares   F-statistic:                     286
5.
Date:                Mon, 22 Apr 2024   Prob (F-statistic):          8.06e-2
34
Time:                        12:44:16   Log-Likelihood:                -193
0.3
No. Observations:                 618   AIC:                             386
5.
Df Residuals:                     616   BIC:                             387
3.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
==
                 coef    std err          t      P>|t|      [0.025      0.97
5]
------------------------------------------------------------------------------
--
Intercept   -128.2475      3.149    -40.731      0.000    -134.431    -122.0
64
avg_elo_n      0.1121      0.002     53.523      0.000       0.108       0.1
16
==============================================================================
==
Omnibus:                      152.822   Durbin-Watson:                   1.0
98
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              393.2
23
Skew:                          -1.247   Prob(JB):                     4.10e-
86
Kurtosis:                       6.009   Cond. No.                     2.14e+
04
==============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
[2] The condition number is large, 2.14e+04. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

Step 4: Scatterplot and Correlation for the Total Number of Wins and Average Points Scored

Your coach expects teams to win more games in a regular season if they score more points on average during the season. This is because the chances of winning are higher if a team maintains high average points scored. Therefore, it is expected that the total number of wins and the average points scored are correlated. Calculate the Pearson correlation coefficient and its P-value.

In [ ]:
```
import scipy.stats as st

# ---- TODO: make your edits here ----
plt.plot(nba_wins_df['avg_pts'], nba_wins_df['total_wins'], 'o')

plt.title('Total Number of Wins by Average Points Scored', fontsize=20)
plt.xlabel('Average Points Scored')
plt.ylabel('Total Number of Wins')
plt.show()

correlation_coefficient, p_value = st.pearsonr(nba_wins_df['avg_pts'], nba_w

print("Correlation between Average Points Scored and the Total Number of Wir
print("Pearson Correlation Coefficient =",  round(correlation_coefficient,4)
print("P-value =", round(p_value,4))

# ---------------------------------------------------------------------------
# import scipy.stats as st

# # ---- TO-DO: make your edits here ----
# plt.plot(nba_wins_df['avg_pts'], nba_wins_df['total_wins'], 'o')

# plt.title('Total Number of Wins by Average Points Scored', fontsize=20)
# plt.xlabel('Average Points Scored')
# plt.ylabel('Total Number of Wins')
# plt.show()


# # ---- TO-DO: make your edits here ----
# correlation_coefficient, p_value = st.pearsonr(['avg_pts'], nba_wins_df['t

# print("Correlation between Average Points Scored and the Total Number of W
# print("Pearson Correlation Coefficient =",  round(correlation_coefficient,
# print("P-value =", round(p_value,4))
```




```
Correlation between Average Points Scored and the Total Number of Wins
Pearson Correlation Coefficient = 0.4777
P-value = 0.0
```

Step 5: Multiple Regression: Predicting the Total Number of Wins using Average Points Scored and Average Relative Skill

Instead of presenting a simple linear regression model to the coach, you can suggest a multiple regression model with the total number of wins as the response variable and the average points scored and the average relative skill as predictor variables. This regression model will help your coach predict how many games your team might win in a regular season based on metrics like the average points scored and average relative skill. This model is more practical because you expect more than one performance metric to determine the total number of wins in a regular season. Create this multiple regression model.

In [ ]:
```
import statsmodels.formula.api as smf

# Multiple Regression
# ---- TODO: make your edits here ---
model2 = smf.ols('total_wins ~ avg_elo_n + avg_pts', nba_wins_df).fit()
print(model2.summary())
```

```
                            OLS Regression Results
==============================================================================
==
Dep. Variable:             total_wins   R-squared:                       0.8
37
Model:                            OLS   Adj. R-squared:                  0.8
37
Method:                 Least Squares   F-statistic:                     158
0.
Date:                Mon, 22 Apr 2024   Prob (F-statistic):          4.41e-2
43
Time:                        12:44:16   Log-Likelihood:                 -190
4.6
No. Observations:                 618   AIC:                             381
5.
Df Residuals:                     615   BIC:                             382
9.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
==
                 coef    std err          t      P>|t|      [0.025      0.97
5]
------------------------------------------------------------------------------
--
Intercept   -152.5736      4.500    -33.903      0.000    -161.411    -143.7
36
avg_elo_n      0.1055      0.002     47.952      0.000       0.101       0.1
10
avg_pts        0.3497      0.048      7.297      0.000       0.256       0.4
44
==============================================================================
==
Omnibus:                       89.087   Durbin-Watson:                   1.2
03
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              160.5
40
Skew:                          -0.869   Prob(JB):                     1.38e-
35
Kurtosis:                       4.793   Cond. No.                     3.19e+
04
==============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
[2] The condition number is large, 3.19e+04. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

Step 6: Multiple Regression: Predicting the Total Number of Wins using Average Points

Scored, Average Relative Skill, Average Points Differential and Average Relative Skill Differential

The coach also wants you to consider the average points differential and average relative skill differential as predictor variables in the multiple regression model. Create a multiple regression model with the total number of wins as the response variable, and average points scored, average relative skill, average points differential and average relative skill differential as predictor variables. This regression model will help your coach predict how many games your team might win in a regular season based on metrics like the average score, average relative skill, average points differential and average relative skill differential between the team and their opponents.

In [ ]:
```
import statsmodels.formula.api as smf

# Multiple Regression
# Adding 'avg_pts' as one of the predictors as per the instructions
model3 = smf.ols('total_wins ~ avg_pts + avg_elo_n + avg_pts_differential +

# Print the summary of the model
print(model3.summary())

# -----------------------------------------------------------------------------
# ------------------------------------  Version 1.- Initial version  ----
# -----------------------------------------------------------------------------
# import statsmodels.formula.api as smf

# # Multiple Regression
# model3 = smf.ols('total_wins ~ avg_elo_n + avg_pts_differential + avg_elo_
# # In order to run a multiple regression model, we need to include all the
# print(model3.summary()) # print the model 3 summary.
```

```
                            OLS Regression Results                            
==============================================================================
==
Dep. Variable:             total_wins   R-squared:                       0.8
78
Model:                            OLS   Adj. R-squared:                  0.8
77
Method:                 Least Squares   F-statistic:                     110
2.
Date:                Mon, 22 Apr 2024   Prob (F-statistic):          3.07e-2
78
Time:                        12:44:16   Log-Likelihood:                -181
5.5
No. Observations:                 618   AIC:                             364
1.
Df Residuals:                     613   BIC:                             366
3.
Df Model:                           4                                         
Covariance Type:            nonrobust                                         
==============================================================================
============
                           coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
------------
Intercept               34.5753     25.867      1.337      0.182     -16.223
85.373
avg_pts                  0.2597      0.043      6.070      0.000       0.176
0.344
avg_elo_n               -0.0134      0.017     -0.769      0.442      -0.048
0.021
avg_pts_differential     1.6206      0.135     12.024      0.000       1.356
1.885
avg_elo_differential     0.0525      0.018      2.915      0.004       0.017
0.088
==============================================================================
==
Omnibus:                      193.608   Durbin-Watson:                   0.9
79
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              598.4
16
Skew:                          -1.503   Prob(JB):                    1.14e-1
30
Kurtosis:                       6.769   Cond. No.                    2.11e+
05
==============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.
[2] The condition number is large, 2.11e+05. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

End of Project Three