Explaining Relationships Using Regression Analysis

A regression model can be used for two different purposes. It can be used to explain how one variable (y) depends on another variable (x). Indeed, here we are often trying to identify *causal* inferences, to explain how a system works. It can also be used to make *forecasts* about future outcomes.

During this week we are going to focus on the first purpose, thinking about potential causal links in the performance of professional sports teams. We are going to focus on the performance of teams across an entire season, in terms of either win percentage or league position.

The main input to any team is, of course, the players themselves. Teams compete to hire the best players and player agents seek to find the best financial deal for their clients. It is reasonable to expect, therefore, that team expenditure on player salaries should be an important factor in determining team performance. To be clear, the logic of this is NOT that paying higher salaries will make players perform better. At the top level of professional sports, all players are highly motivated, and salary probably does not play a significant motivational role. Rather, competition for players means that salaries are likely to reflect relative abilities. Better players command higher salaries, and as a result the aggregate pay of players on the team is likely to be a good predictor of team performance.

There are a number of sources for player salary data. In the North American major leagues, salary negotiations are framed by collective bargaining agreements with player unions, which often publish individual player salary data. In European soccer leagues, aggregate salary data is to be found in audited financial statements of professional clubs, which are often available to the public (notably in England). Cricket players in the Indian Premier League have their salaries determined in a public auction.

This week we are going to examine the wage-performance relationship in four different leagues - the NBA, the English Premier League, Major League Baseball and the National Hockey League. While our focus is on the role of salaries, we will also consider other factors that might be relevant, which will help us to think about some of the issues that arise in regression analysis.

We start with the NBA.

In [1]:
```
# As usual, we begin by loading the packages we will need

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.formula.api as smf
```

In [2]:
```
# Now we load the data

NBA=pd.read_excel("../../Data/Week 5/NBA pay and performance.xlsx")
```

We use .describe() to look at the summary statistics for the data. From this we can see that we have 210 observations, for teams running from 2012 to 2018 (7 seasons). Our two variables of interest are win percentage and team salaries. We can also use .info() to summarize the dataframe.

In [3]:
```
NBA.describe()
```

Out[3]:

	season	wpc	salaries
count	210.000000	210.000000	2.100000e+02
mean	2015.000000	0.497843	7.825339e+07
std	2.004779	0.154052	2.523282e+07
min	2012.000000	0.106000	2.893890e+07
25%	2013.000000	0.378000	5.937660e+07
50%	2015.000000	0.512000	7.372819e+07
75%	2017.000000	0.606000	9.564297e+07
max	2018.000000	0.890000	1.425601e+08

In [4]:
```
NBA.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 210 entries, 0 to 209
Data columns (total 4 columns):
Team        210 non-null object
season      210 non-null int64
wpc         210 non-null float64
salaries    210 non-null int64
dtypes: float64(1), int64(2), object(1)
memory usage: 6.6+ KB
```

If salaries reflect player ability, then the the success of a team should depend on how much more or less it pays than its rivals. However, if we look at salaries paid out in different seasons, there is clearly inflation in player salaries. We can see this if we use the .groupby() command to look at total salaries over the seven seasons:

In [5]:
```
Sumsal = NBA.groupby(['season'])['salaries'].sum().reset_index().rename(columns={'salaries':'allsal'})
Sumsal
```

Out[5]:

	season	allsal
0	2012	1438650614
1	2013	1837810750
2	2014	1976549213
3	2015	2153667904
4	2016	2523411764
5	2017	3159860863
6	2018	3343261288

We can see that salaries in 2018 were more than double the level in 2012, and increased consistently from year to year. This does not imply that the players were gettign better from season to season. Rather, this is a reflection of the growing revenues of the NBA, and the ability of players to bargain for a more or less constant share of this growing revenue.

So, if we now want to account for team performance in terms of salaries, we need to make sure we compare like with like. What $1 million would buy in 2012 was not the same as what it would buy in 2018. It's easy to adjust for this. We simply divide the salary of each team in each season by the total spending of all teams in that season, so that we have a measure of salary spending relative to the competition.

To do this we first use pd.merge() to add the aggregate salaries for each season to our original dataframe:

In [6]:
```
NBA = pd.merge(NBA, Sumsal, on=['season'], how='left')
display(NBA)
```

	Team	season	wpc	salaries	allsal
0	Atlanta Hawks	2012	0.606	55503683	1438650614
1	Atlanta Hawks	2013	0.537	60437642	1837810750
2	Atlanta Hawks	2014	0.463	58841508	1976549213
3	Atlanta Hawks	2015	0.732	62487671	2153667904
4	Atlanta Hawks	2016	0.585	82337675	2523411764
5	Atlanta Hawks	2017	0.524	105882053	3159860863
6	Atlanta Hawks	2018	0.293	97118111	3343261288
7	Boston Celtics	2012	0.591	56768577	1438650614
8	Boston Celtics	2013	0.506	69572590	1837810750
9	Boston Celtics	2014	0.305	70105837	1976549213
10	Boston Celtics	2015	0.488	60436154	2153667904

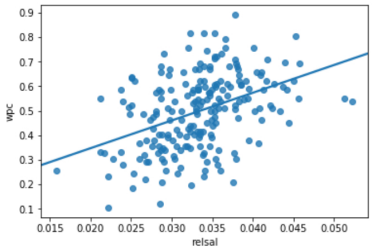
We can now create a variable which we call 'relsal', which measures the share of team's salary spend in the total spending of all teams in that season:

In [7]:
```
NBA['relsal']= NBA['salaries']/NBA['allsal']
```

Before running a regression, it makes sense to look at the relationship between salaries and win percentage on a chart. To do this we use sns.reglot(). Since our argument is that higher relative salaries mean better players which in turns leads to more wins, we put relsal on the x axis and wpc on the y axis.

In [8]:
```
sns.regplot(x="relsal", y="wpc", data = NBA, ci=False)
```

Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x7fcfe6e54198>



It's clear from the data that there is a positive correlation between relsal and wpc, as shown by the regression line which regplot adds to the scatter diagram. We now run a regression using smf.ols() in order to derive the coefficients of the regression and other diagnostic statistics.

In [9]:
```
wpcsal1_lm = smf.ols(formula = 'wpc ~ relsal', data=NBA).fit()
print(wpcsal1_lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    wpc   R-squared:                       0.172
Model:                            OLS   Adj. R-squared:                  0.168
Method:                 Least Squares   F-statistic:                     43.26
Date:                Thu, 29 Jul 2021   Prob (F-statistic):           3.81e-10
Time:                        19:43:00   Log-Likelihood:                 115.16
No. Observations:                 210   AIC:                            -226.3
Df Residuals:                     208   BIC:                            -219.6
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      0.1211      0.058      2.086      0.038       0.007       0.236
relsal        11.3009      1.718      6.577      0.000       7.914      14.688
==============================================================================
Omnibus:                        2.994   Durbin-Watson:                   1.026
Prob(Omnibus):                  0.224   Jarque-Bera (JB):                2.134
Skew:                           0.043   Prob(JB):                        0.344
Kurtosis:                       2.514   Cond. No.                         177.
==============================================================================

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

The first things to look at in any regression are the coefficients of the explanatory variables and their statistical significance. We can see here that the coefficient on relsal is 11.3009. This means that every one percentage (0.01) increase in the share of the team in total salaries leads to and 11.3 x .01 increase in win percentage - that is roughly .11 , or eleven percentage points. That is a very large increase, but not that the share of salaries, as can be seen on the x axis of the chart above, ranges from around .02 (2%) to .05 (5%). Thus going from the lowest salary to the highest share (from 2% to 5%) will produce a 3 x .11 = .33 increase in win percentage - from around roughly 33% to 66%.

This estimate is statistically significant. The coefficient estimate is more than six times larger than its standard error (this ratio is called the t- statistic (6.577). The p-value (P>|t|) tells us the probability of observing such estimate if the true value were actually zero. The p-value here is shown as "0.000" - however, it can never be exactly zero. It is just that, in this case, it is so small that is does not register up to three decmial places. The usual standard for statistical significance is a p-value below 0.05. Clearly, in this case the estimate clearly beats that standard.

How much of the variation in team performance is captured by this relsal? We can see this from the R-squared coefficient which is 0.172, or 17.2%. Clearly, there is much else to team performance than salaries alone.

Is the coefficient estimate plausible? Our regression estimate is the best estimate we have of the effect, but *only under the assumption that our regression includes all of the relevant variables.* If there are other variables which influence performance other than salary share, then our regression estimate will be biased, either upward or downward. This is the problem known as "omitted variable bias" (OVB). There is no way to be certain that OVB is not a problem, it requires good judgment and careful thought to decide on whether there are other variables that should be included.

The fact that the R-squared value was only 0.172 might give one to think that there are other factors to include, but it is always possible that the remainder is just random - the effect of luck, which no doubt plays a role in every game.

But in this case, it is possible to think of other factors that might be relevant and therefore should be included. One such is "lagged dependent variable", which here means the value of win percentage in the previous season. While the salary level should capture many aspects of team quality, salaries are not renegotiated every year, and many aspects of team quality would have been in pace in the previous season. So we can add this lagged dependent variable to our regression, and then see if this changes our estimate of the impact of salaries.

We create this variable in two stages. First, we sort the data by team, and then by season, using .sort_values()

Self test

Based on this model, what would be the win percentage of a team for whom the value of relsal was 4%?

In [10]:
```
NBA.sort_values(by=['Team','season'], ascending=True)
```

Out[10]:

	Team	season	wpc	salaries	allsal	relsal
0	Atlanta Hawks	2012	0.606	55503683	1438650614	0.038580
1	Atlanta Hawks	2013	0.537	60437642	1837810750	0.032886
2	Atlanta Hawks	2014	0.463	58841508	1976549213	0.029770
3	Atlanta Hawks	2015	0.732	62487671	2153667904	0.029015
4	Atlanta Hawks	2016	0.585	82337675	2523411764	0.032630
5	Atlanta Hawks	2017	0.524	105882053	3159860863	0.033508
6	Atlanta Hawks	2018	0.293	97118111	3343261288	0.029049
7	Boston Celtics	2012	0.591	56768577	1438650614	0.039460
8	Boston Celtics	2013	0.506	69572590	1837810750	0.037856
9	Boston Celtics	2014	0.305	70105837	1976549213	0.035469
10	Boston Celtics	2015	0.488	60436154	2153667904	0.028062

Second, we create the lagged value of wpc. This done by using .groupby() together with .shift(1). The value 1 in .shift() signifies the value is for the previous season that is being added to each row. If we used .shift(2) it would add the value from 2 seasons before, and so on.

Note that because we define the lagged value to apply to each team, there is a missing value (NaN) for each team in 2012, which is the first year in our data.

In [11]:
```
NBA['wpc_lag'] = NBA.groupby('Team')['wpc'].shift(1)
NBA
```

Out[11]:

	Team	season	wpc	salaries	allsal	relsal	wpc_lag
0	Atlanta Hawks	2012	0.606	55503683	1438650614	0.038580	NaN
1	Atlanta Hawks	2013	0.537	60437642	1837810750	0.032886	0.606
2	Atlanta Hawks	2014	0.463	58841508	1976549213	0.029770	0.537
3	Atlanta Hawks	2015	0.732	62487671	2153667904	0.029015	0.463
4	Atlanta Hawks	2016	0.585	82337675	2523411764	0.032630	0.732
5	Atlanta Hawks	2017	0.524	105882053	3159860863	0.033508	0.585
6	Atlanta Hawks	2018	0.293	97118111	3343261288	0.029049	0.524
7	Boston Celtics	2012	0.591	56768577	1438650614	0.039460	NaN
8	Boston Celtics	2013	0.506	69572590	1837810750	0.037856	0.591
9	Boston Celtics	2014	0.305	70105837	1976549213	0.035469	0.506
10	Boston Celtics	2015	0.488	60436154	2153667904	0.028062	0.305

In [12]:
```
# this command allows us to see all rows in the window

pd.set_option('display.max_rows', 250)
NBA
```

Out[12]:

	Team	season	wpc	salaries	allsal	relsal	wpc_lag
0	Atlanta Hawks	2012	0.606	55503683	1438650614	0.038580	NaN
1	Atlanta Hawks	2013	0.537	60437642	1837810750	0.032886	0.606
2	Atlanta Hawks	2014	0.463	58841508	1976549213	0.029770	0.537
3	Atlanta Hawks	2015	0.732	62487671	2153667904	0.029015	0.463
4	Atlanta Hawks	2016	0.585	82337675	2523411764	0.032630	0.732
5	Atlanta Hawks	2017	0.524	105882053	3159860863	0.033508	0.585
6	Atlanta Hawks	2018	0.293	97118111	3343261288	0.029049	0.524
7	Boston Celtics	2012	0.591	56768577	1438650614	0.039460	NaN
8	Boston Celtics	2013	0.506	69572590	1837810750	0.037856	0.591
9	Boston Celtics	2014	0.305	70105837	1976549213	0.035469	0.506
10	Boston Celtics	2015	0.488	60436154	2153667904	0.028062	0.305

We now run our regression again, but adding wpc_lag into the regression equation:

In [13]:
```
wpcsal2_lm = smf.ols(formula = 'wpc ~wpc_lag + relsal', data=NBA).fit()
print(wpcsal2_lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    wpc   R-squared:                       0.416
Model:                            OLS   Adj. R-squared:                  0.410
Method:                 Least Squares   F-statistic:                     62.78
Date:                Thu, 29 Jul 2021   Prob (F-statistic):           2.64e-21
Time:                        19:43:02   Log-Likelihood:                 129.13
No. Observations:                 179   AIC:                            -252.3
Df Residuals:                     176   BIC:                            -242.7
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      0.1293      0.054      2.374      0.019       0.022       0.237
wpc_lag        0.6031      0.066      9.125      0.000       0.473       0.734
relsal         2.0165      1.861      1.083      0.280      -1.656       5.689
==============================================================================
Omnibus:                        1.976   Durbin-Watson:                   1.994
Prob(Omnibus):                  0.372   Jarque-Bera (JB):                1.972
Skew:                          -0.251   Prob(JB):                        0.373
Kurtosis:                       2.886   Cond. No.                         235.
==============================================================================

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

The result of this change is quite dramatic. Not only has the coefficient on relsal fallen to only 2.0165, but it is not statistically insignificant (p_value is 0.28, greater than the critical value of .05). Last year's win percentage is highly significant and the R-squared of the regression has risen to 0.416.

It is quite usual for the lagged dependent variable to be significant in situations such as these- history matters. But should we now abandon our theory that wages influence performance? One should also be cautious about a theory that relies only on the lagged dependent variable - while history matters, one should also expect there to be specific, quantifiable factors that drive history.

While the omission of the lagged dependent variable in the first regression suggests that there may have been OVB that biased the estimate upwards, there is also the possibility that OVB can bias the estimate downwards.

Clearly, not all teams are identical, while our regression estimates thus far treat each team as if it were, in the sense that spending would affect the performance of each team in the same way, and that last year's win percentage would imact this year's in the same way. In our regression specification we want to find balance between treating each team as if it were identical, and treating each team as if it were completely unique. The truth is likely to be that there are common factors affecting all teams, but that there are also idiosyncrasies. This is often described as heterogeneity.

One way we can introduce heterogeneity is through fixed effects. Fixed effects are dummy variables. For each team there is a fixed effect, equal to one if the row relates to the team in question, and zero otherwise. Each team can have its own fixed effects. Estimation of fixed effects allows us to identify differences between the teams that are independent of the impact of salaries or of the lagged dependent variable.

Adding fixed effects is very easy in Python. The variable 'Team' identifies the team names, and if we add "C(Team)" to the regression formula Python will estimate a fixed effect for each team.

We now run the regression with the lagged dependent variable and fixed effects, to see what impact this has on the estimate of the salary coefficient.

Self test

Based on this model, what would be the win percentage of a team that (a) had 0.5 win percentage last season and (b) had a value of relsal equal to 3%?

In [14]:
```
wpcsal3_lm = smf.ols(formula = 'wpc ~ wpc_lag + relsal +C(Team)', data=NBA).fit()
print(wpcsal3_lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    wpc   R-squared:                       0.585
Model:                            OLS   Adj. R-squared:                  0.498
Method:                 Least Squares   F-statistic:                     6.686
Date:                Thu, 29 Jul 2021   Prob (F-statistic):           5.88e-16
Time:                        19:43:03   Log-Likelihood:                 159.66
No. Observations:                 179   AIC:                            -255.3
Df Residuals:                     147   BIC:                            -153.3
Df Model:                          31
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                        0.2333      0.074      3.140      0.002       0.086       0.380
C(Team)[T.Boston Celtics]        0.0179      0.063      0.282      0.779      -0.108       0.143
C(Team)[T.Brooklyn Nets]        -0.1431      0.069     -2.060      0.041      -0.280      -0.006
C(Team)[T.Charlotte Bobcats]    -0.0254      0.065     -0.392      0.696      -0.153       0.103
C(Team)[T.Chicago Bulls]        -0.0293      0.064     -0.461      0.645      -0.155       0.096
C(Team)[T.Cleveland Cavaliers]   0.0063      0.066      0.096      0.924      -0.124       0.137
C(Team)[T.Dallas Mavericks]     -0.0350      0.064     -0.550      0.583      -0.161       0.091
C(Team)[T.Denver Nuggets]       -0.0070      0.063     -0.111      0.912      -0.132       0.118
C(Team)[T.Detroit Pistons]      -0.0607      0.065     -0.936      0.351      -0.189       0.068
C(Team)[T.Golden State Warriors]  0.1683      0.064      2.635      0.009       0.042       0.295
C(Team)[T.Houston Rockets]       0.1155      0.063      1.827      0.070      -0.009       0.240
C(Team)[T.Indiana Pacers]       -0.0169      0.064     -0.265      0.791      -0.143       0.109
C(Team)[T.Los Angeles Clippers]  0.0642      0.064      0.996      0.321      -0.063       0.192
C(Team)[T.Los Angeles Lakers]   -0.1530      0.067     -2.271      0.025      -0.286      -0.020
C(Team)[T.Memphis Grizzlies]    -0.0114      0.064     -0.178      0.859      -0.138       0.115
C(Team)[T.Miami Heat]            0.0145      0.065      0.222      0.825      -0.115       0.144
C(Team)[T.Milwaukee Bucks]      -0.0503      0.064     -0.781      0.436      -0.178       0.077
C(Team)[T.Minnesota Timberwolves]  -0.0783      0.066     -1.192      0.235      -0.208       0.051
C(Team)[T.New Jersey Nets]      -6.498e-18   1.54e-17     -0.423      0.673   -3.69e-17    2.39e-17
C(Team)[T.New Orleans Hornets]  -0.0586      0.067     -0.880      0.380      -0.190       0.073
C(Team)[T.New York Knicks]      -0.1144      0.067     -1.716      0.088      -0.246       0.017
C(Team)[T.Oklahoma City Thunder]  0.0649      0.064      1.010      0.314      -0.062       0.192
C(Team)[T.Orlando Magic]        -0.1411      0.065     -2.163      0.032      -0.270      -0.012
C(Team)[T.Philadelphia 76ers]   -0.1193      0.066     -1.797      0.074      -0.251       0.012
C(Team)[T.Phoenix Suns]         -0.1059      0.064     -1.643      0.103      -0.233       0.021
C(Team)[T.Portland Trail Blazers]  0.0361      0.063      0.568      0.571      -0.089       0.162
C(Team)[T.Sacramento Kings]     -0.1125      0.066     -1.711      0.089      -0.243       0.017
C(Team)[T.San Antonio Spurs]     0.1328      0.064      2.063      0.041       0.006       0.260
C(Team)[T.Toronto Raptors]       0.0778      0.064      1.225      0.223      -0.048       0.203
C(Team)[T.Utah Jazz]             0.0034      0.063      0.054      0.957      -0.122       0.129
C(Team)[T.Washington Wizards]   -0.0009      0.064     -0.014      0.989      -0.128       0.126
wpc_lag                          0.2353      0.085      2.754      0.007       0.066       0.404
relsal                           4.9388      2.102      2.349      0.020       0.785       9.093
==============================================================================
Omnibus:                        2.710   Durbin-Watson:                   2.072
Prob(Omnibus):                  0.258   Jarque-Bera (JB):                2.573
Skew:                          -0.118   Prob(JB):                        0.276
Kurtosis:                       3.538   Cond. No.                     2.07e+17
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 5.39e-33. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

There are 30 fixed effects listed in the output, one for each team. A positive fixed effect means that in some way the team was able to perform above average, and a negative fixed effect implies below average performance. However, most of the fixed effects are not statistically significant. The significant fixed effects are: The Brooklyn Nets (negative), the Golden State Warriors (positive), The LA Lakers (negative), Orlando Magic (negative), San Antonio Spurs (positive).

Looking at our main variable of interest, relsal, it is clear that this variable is once again statistically significant with the addition of the fixed effects. Thus, we might conclude that the absence of the fixed effects biased the coefficient estimate downwards. The coefficient is statistically significant at the 5% level, and the value of 4.9388 implies that increasing salary share in the NBA by 1 percentage point (e.g. from 2% to 3%) will lead to an increase in win percentage of almost 5%. This is smaller than our original estimate, but also because of the presence of the lagged dependent variable, spending which increases win percentage this season will also have an effect, albeit a smaller one, on the following season. Indeed, an increase in spending today will create a ripple effect which will be discernible in performance for a number of years into the future.

Also note that the size of the lagged dependent variable is smaller once we add the fixed effects. Finally, with this third specification the R-squared of the regression has now risen to 0.585 (close to 60%), which accounts for significant fraction of the overall variation.

We should never expect to explain 100% of the variation of outcomes in sport - if we could do that then each game would be perfectly predictable - and then what would be the point of watching?

Self test

Run the regression of win percentage on relsal with fixed effects but without the lagged depedent variable. Compare your output results. Compare this to the previous three regressions. Which do you think is the best representation of the data. Why?

Conclusion

In this notebook we have explored the possibility of using regression analysis to explore the validity of a causal explanation of team success. That causal explanation was itself not derived from the data, but based on a theory that player quality will be reflected in salaries and therefore salaries will predict team success.

You might be wondering about why this works at all with the NBA, since the league operates a salary cap system which is intended to equalize resources

among the teams. If each team spent the same amount of money on players, our theory predicts that each team can expect to win 50% of its games, and team performances will vary randomly around this mean. However, the NBA cap is a "soft cap", meaning that there are many exemptions, so teams spend varying amounts in reality. Some leagues, such as the NFL, operate a hard cap, which strictly prohibit spending above the cap. The NFL also has a salary floor, which prevents teams from spending a lot less than average. When looking at NFL data, therefore, it is much harder to identifty the effect of wage spending on performance.

We next turn to look at the salary performance relationship in the English Premier League.

In [ ]: