MLB

24 July 2020

Salaries and Performance in Major League Baseball

We might expect that the salary performance relationship in baseball will be more like the NBA than the EPL, given that the organizational structure has many similarities with the NBA.

We follow the same steps as we did for both those leagues.

```
# As usual, we begin by loading the packages we will need

options(warn = -1)
library("readxl",quietly = TRUE)
library("tidyverse",quietly = TRUE)
```

```
# Now we load the data

MLB = read_excel("MLB pay and performance.xlsx")
```

```
MLB %>% summary()
```

```
##      season          Team               lgID              salaries
##  Min.   :1985   Length:918         Length:918         Min.   :  880000
##  1st Qu.:1993   Class :character   Class :character   1st Qu.: 25435708
##  Median :2001   Mode  :character   Mode  :character   Median : 50537324
##  Mean   :2001                                         Mean   : 60042633
##  3rd Qu.:2009                                         3rd Qu.: 84416083
##  Max.   :2016                                         Max.   :231978886
##       wpc               G               W
##  Min.   :0.2654   Min.   :112.0   Min.   : 43.00
##  1st Qu.:0.4506   1st Qu.:162.0   1st Qu.: 71.25
##  Median :0.5000   Median :162.0   Median : 80.00
##  Mean   :0.4998   Mean   :159.9   Mean   : 79.94
##  3rd Qu.:0.5494   3rd Qu.:162.0   3rd Qu.: 89.00
##  Max.   :0.7160   Max.   :164.0   Max.   :116.00
```

```
MLB %>% str()
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    918 obs. of  7 variables:
##  $ season  : num  1997 1998 1999 2000 2001 ...
##  $ Team    : chr  "ANA" "ANA" "ANA" "ANA" ...
##  $ lgID    : chr  "AL" "AL" "AL" "AL" ...
##  $ salaries: num  31135472 41281000 55388166 51464167 47535167 ...
```

```
## $ wpc      : num  0.519 0.525 0.432 0.506 0.463 ...
## $ G        : num  162 162 162 162 162 162 162 162 162 162 ...
## $ W        : num  84 85 70 82 75 99 77 92 65 100 ...
```

We can see that we have 918 observations in total covering the seasons 1985 to 2016. This data covers even more years than our NBA or EPL data, and therefore we would expect the effect of salary inflation to be even greater. We can see that when we measure the total expenditure on salaries by season:

```r
Sumsal <- MLB %>%
  group_by(season)%>%
   dplyr::summarise(salaries = sum(salaries))%>%rename(allsal = salaries)
Sumsal
```

```
## # A tibble: 32 x 2
##    season    allsal
##     <dbl>     <dbl>
##  1   1985 261964696
##  2   1986 307854518
##  3   1987 272575375
##  4   1988 300452424
##  5   1989 359995711
##  6   1990 443881193
##  7   1991 613048418
##  8   1992 805543323
##  9   1993 901740134
## 10   1994 927836287
## # ... with 22 more rows
```

In 1985, the total salaries paid out by MLB teams amounted to $262 million and by 2016 this had risen to $3750 million. As with the NBA and EPL, this does not reflect improvements in player quality, but rather the growth of revenues of MLB and the capacity of players to bargain for a significant share of these revenues.

We now merge these totals into our original dataset.

```r
MLB <- left_join(MLB, Sumsal, by="season")
head(MLB)
```

```
## # A tibble: 6 x 8
##   season Team  lgID  salaries   wpc     G     W     allsal
##    <dbl> <chr> <chr>    <dbl> <dbl> <dbl> <dbl>      <dbl>
## 1   1997 ANA   AL    31135472 0.519   162    84 1127285885
## 2   1998 ANA   AL    41281000 0.525   162    85 1278282871
## 3   1999 ANA   AL    55388166 0.432   162    70 1494228750
## 4   2000 ANA   AL    51464167 0.506   162    82 1666135102
## 5   2001 ANA   AL    47535167 0.463   162    75 1960663313
## 6   2002 ANA   AL    61721667 0.611   162    99 2024077522
```

```r
tail(MLB)
```

```
## # A tibble: 6 x 8
##   season Team  lgID   salaries   wpc     G     W     allsal
##    <dbl> <chr> <chr>     <dbl> <dbl> <dbl> <dbl>      <dbl>
## 1   2011 WAS   NL     63856928 0.497   161    80 2784505291
## 2   2012 WAS   NL     80855143 0.605   162    98 2932741192
## 3   2013 WAS   NL    113703270 0.531   162    86 3034525648
## 4   2014 WAS   NL    131983680 0.593   162    96 3192317623
## 5   2015 WAS   NL    155587472 0.512   162    83 3514142569
## 6   2016 WAS   NL    141652646 0.586   162    95 3750137392
```

```r
# Here we create the variable 'relsal' for the MLB

MLB[,'relsal']= MLB[,'salaries']/MLB[,'allsal']
head(MLB)
```

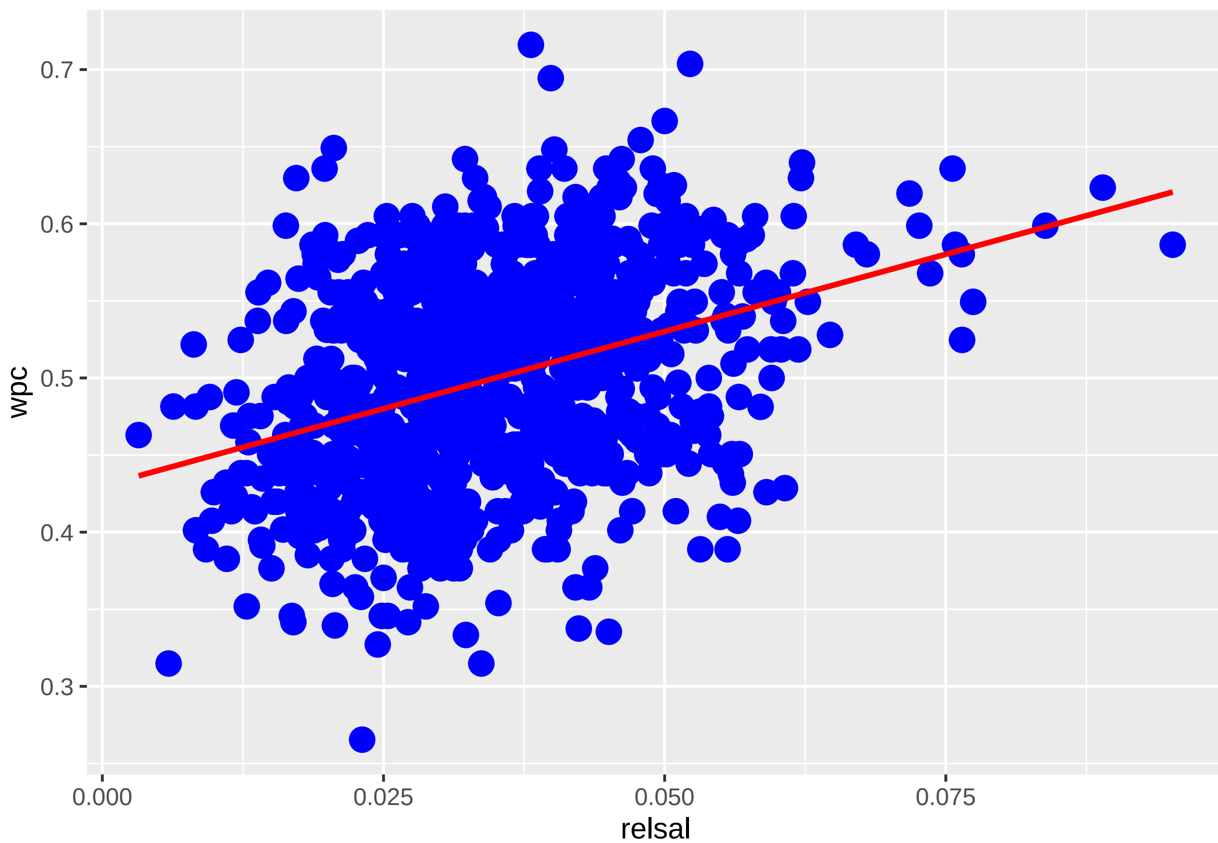
```
## # A tibble: 6 x 9
##   season Team  lgID  salaries   wpc     G     W     allsal relsal
##    <dbl> <chr> <chr>    <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>
## 1   1997 ANA   AL    31135472 0.519   162    84 1127285885 0.0276
## 2   1998 ANA   AL    41281000 0.525   162    85 1278282871 0.0323
## 3   1999 ANA   AL    55388166 0.432   162    70 1494228750 0.0371
## 4   2000 ANA   AL    51464167 0.506   162    82 1666135102 0.0309
## 5   2001 ANA   AL    47535167 0.463   162    75 1960663313 0.0242
## 6   2002 ANA   AL    61721667 0.611   162    99 2024077522 0.0305
```

```r
tail(MLB)
```

```
## # A tibble: 6 x 9
##   season Team  lgID   salaries   wpc     G     W     allsal relsal
##    <dbl> <chr> <chr>     <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>
## 1   2011 WAS   NL     63856928 0.497   161    80 2784505291 0.0229
## 2   2012 WAS   NL     80855143 0.605   162    98 2932741192 0.0276
## 3   2013 WAS   NL    113703270 0.531   162    86 3034525648 0.0375
## 4   2014 WAS   NL    131983680 0.593   162    96 3192317623 0.0413
## 5   2015 WAS   NL    155587472 0.512   162    83 3514142569 0.0443
## 6   2016 WAS   NL    141652646 0.586   162    95 3750137392 0.0378
```

Before running a regression, we use ggplot() to look at the relationship between salaries and win percentage on a chart.

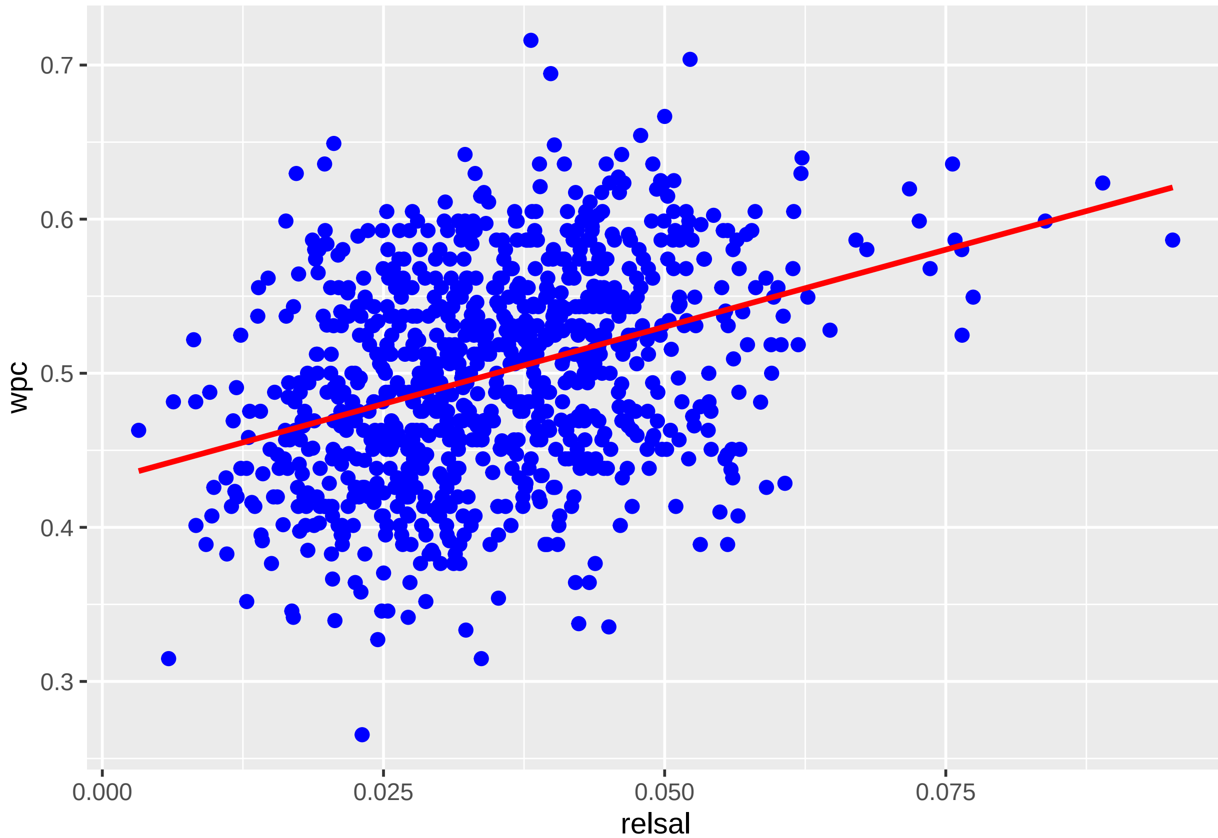
```r
ggplot(data = MLB,aes(x = relsal,y = wpc )) + geom_point(color='blue',size=4)+
  geom_smooth(method = "lm", se = FALSE,color = "red")
```



The chart shows a positive relationship between win percentage and relsal.

The size of the dots, which each represent a single team in a single season, is too large for the scatter to be clearly visible. We can change the size of the dots in regplot using the command "size = 2".

```
ggplot(data = MLB,aes(x = relsal,y = wpc )) + geom_point(color='blue',size = 2)+
  geom_smooth(method = "lm", se = FALSE,color = "red")
```



While there are some outliers, the relsal variable on the x axis for most teams lies between 0.01 (1%) and a little over .06 (6%). Win percentage on the y axis for most teams lies between 0.33 and 0.66.

We now run a regression using lm() in order to derive the coefficients of the regression and other diagnostic statistics.

```r
wpcsal1_lm = lm(formula = 'wpc ~ relsal', data = MLB)
wpcsal1_lm %>% summary()
```

```
##
## Call:
## lm(formula = "wpc ~ relsal", data = MLB)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.210875 -0.046377  0.001088  0.045653  0.209695
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.430053   0.006236   68.97   <2e-16 ***
## relsal      2.002137   0.168332   11.89   <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06395 on 916 degrees of freedom
## Multiple R-squared:  0.1338, Adjusted R-squared:  0.1328
## F-statistic: 141.5 on 1 and 916 DF,  p-value: < 2.2e-16
```

As with the NBA, we find that the coefficient on relsal is highly significant, but the size of our initial estimate is much smaller- recall that for the NBA the value was 11.3 - nearly six times larger than the coefficient for relsal in MLB. As an initial evaluation we can conclude that the amount of money to outperform your rivals is higher for MLB than than the NBA. Note also that the R-squared (0.134) is a little smaller than the one we found for the NBA (0.172), but not by that much. This suggests that win percentage can buy you success as reliably as it can in the NBA, it's just that you need to spend a mot more (relative to your rivals).

Self Test

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

Recall that we asked the same question when looking at the NBA. Compare you two answers. What do you think explains the difference?

Let's now see if the addition of the lagged dependent variable changes our relsal estimate.

```r
MLB <-  MLB %>% arrange(Team,season)
head(MLB)
```

```
## # A tibble: 6 x 9
##   season Team  lgID  salaries   wpc     G     W     allsal relsal
##    <dbl> <chr> <chr>    <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>
## 1   1997 ANA   AL    31135472 0.519   162    84 1127285885 0.0276
## 2   1998 ANA   AL    41281000 0.525   162    85 1278282871 0.0323
## 3   1999 ANA   AL    55388166 0.432   162    70 1494228750 0.0371
## 4   2000 ANA   AL    51464167 0.506   162    82 1666135102 0.0309
## 5   2001 ANA   AL    47535167 0.463   162    75 1960663313 0.0242
## 6   2002 ANA   AL    61721667 0.611   162    99 2024077522 0.0305
```

```r
tail(MLB)
```

```
## # A tibble: 6 x 9
##   season Team  lgID   salaries   wpc     G     W     allsal relsal
##    <dbl> <chr> <chr>     <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>
## 1   2011 WAS   NL     63856928 0.497   161    80 2784505291 0.0229
## 2   2012 WAS   NL     80855143 0.605   162    98 2932741192 0.0276
## 3   2013 WAS   NL    113703270 0.531   162    86 3034525648 0.0375
## 4   2014 WAS   NL    131983680 0.593   162    96 3192317623 0.0413
## 5   2015 WAS   NL    155587472 0.512   162    83 3514142569 0.0443
## 6   2016 WAS   NL    141652646 0.586   162    95 3750137392 0.0378
```

```r
MLB <- MLB %>%
       group_by(Team)%>%
       mutate(wpc_lag = dplyr::lag(wpc))%>%
       ungroup()
head(MLB)
```

```
## # A tibble: 6 x 10
##   season Team  lgID  salaries   wpc     G     W     allsal relsal wpc_lag
##    <dbl> <chr> <chr>    <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>   <dbl>
## 1   1997 ANA   AL    31135472 0.519   162    84 1127285885 0.0276  NA
## 2   1998 ANA   AL    41281000 0.525   162    85 1278282871 0.0323   0.519
## 3   1999 ANA   AL    55388166 0.432   162    70 1494228750 0.0371   0.525
## 4   2000 ANA   AL    51464167 0.506   162    82 1666135102 0.0309   0.432
## 5   2001 ANA   AL    47535167 0.463   162    75 1960663313 0.0242   0.506
## 6   2002 ANA   AL    61721667 0.611   162    99 2024077522 0.0305   0.463
```

```r
tail(MLB)
```

```
## # A tibble: 6 x 10
##   season Team  lgID   salaries   wpc     G     W     allsal relsal wpc_lag
##    <dbl> <chr> <chr>     <dbl> <dbl> <dbl> <dbl>      <dbl>  <dbl>   <dbl>
## 1   2011 WAS   NL     63856928 0.497   161    80 2784505291 0.0229   0.426
## 2   2012 WAS   NL     80855143 0.605   162    98 2932741192 0.0276   0.497
## 3   2013 WAS   NL    113703270 0.531   162    86 3034525648 0.0375   0.605
## 4   2014 WAS   NL    131983680 0.593   162    96 3192317623 0.0413   0.531
## 5   2015 WAS   NL    155587472 0.512   162    83 3514142569 0.0443   0.593
## 6   2016 WAS   NL    141652646 0.586   162    95 3750137392 0.0378   0.512
```

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

```r
wpcsal2_lm = lm(formula = 'wpc ~ wpc_lag + relsal', data = MLB)
wpcsal2_lm %>% summary()
```

```
##
## Call:
## lm(formula = "wpc ~ wpc_lag + relsal", data = MLB)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.191024 -0.042268 -0.000104  0.042634  0.190071
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.28390    0.01487  19.093  < 2e-16 ***
## wpc_lag      0.36136    0.03333  10.840  < 2e-16 ***
## relsal       1.02591    0.18187   5.641 2.28e-08 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05985 on 880 degrees of freedom
##   (35 observations deleted due to missingness)
## Multiple R-squared:  0.2343, Adjusted R-squared:  0.2326
## F-statistic: 134.6 on 2 and 880 DF,  p-value: < 2.2e-16
```

The lagged dependent variable here is much smaller than it was in the case of the NBA (0.6), which implies that last year's performance matters much less in determining this year's performance. There could be several reasons for this, e,g, greater player turnover in MLB, or a lower probability that player's from last year will be repeated in the current year.

As was the case with the NBA, the addition of the lagged dependent variable has reduced the size of the coefficient for relsal, halving it, but still this is not as dramatic as the reduction in the NBA case, where the variable also became statistically insignificant, which is not the case here. The R-squared has not risen as much either.

Overall, however, we can conclude that adding the lagged dependent variable has reduced the possibility of omitted variable bias.

Self test

The model implies that win percentage of a team in year t, wpc(t) = 0.2839 +0.3614 x wpc_lag + 1.0259 x relsal

Suppose relsal is 4% (0.04), calculate the value of wpc(t) if wpc(t-1) equals (a) 0.6 and (b) 0.4. How do you account for your answer?

```r
wpcsal3_lm <- lm(wpc ~ wpc_lag + relsal + factor(Team),
                    data = MLB)

wpcsal3_lm %>% summary()
```

```
##
## Call:
## lm(formula = wpc ~ wpc_lag + relsal + factor(Team), data = MLB)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.189247 -0.042986  0.000548  0.041379  0.201102
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.3223905  0.0277035  11.637  < 2e-16 ***
## wpc_lag           0.3141247  0.0349786   8.980  < 2e-16 ***
## relsal            0.8907513  0.2467980   3.609 0.000325 ***
## factor(Team)ARI -0.0112267  0.0266827  -0.421 0.674047
```

```
## factor(Team)ATL  0.0107583  0.0250981   0.429 0.668287
## factor(Team)BAL -0.0281992  0.0250922  -1.124 0.261407
## factor(Team)BOS  0.0034349  0.0252738   0.136 0.891928
## factor(Team)CAL -0.0304309  0.0289969  -1.049 0.294269
## factor(Team)CHA -0.0063547  0.0250566  -0.254 0.799854
## factor(Team)CHN -0.0223823  0.0251146  -0.891 0.373073
## factor(Team)CIN -0.0116272  0.0250625  -0.464 0.642817
## factor(Team)CLE  0.0006897  0.0250754   0.028 0.978062
## factor(Team)COL -0.0274872  0.0258770  -1.062 0.288436
## factor(Team)DET -0.0275080  0.0250940  -1.096 0.273304
## factor(Team)FLO -0.0103879  0.0268463  -0.387 0.698899
## factor(Team)HOU -0.0086682  0.0250673  -0.346 0.729582
## factor(Team)KCA -0.0314132  0.0250910  -1.252 0.210926
## factor(Team)LAA  0.0094642  0.0290214   0.326 0.744420
## factor(Team)LAN -0.0064022  0.0252512  -0.254 0.799913
## factor(Team)MIA -0.0251818  0.0377577  -0.667 0.504998
## factor(Team)MIL -0.0234125  0.0267449  -0.875 0.381605
## factor(Team)MIN -0.0151564  0.0251005  -0.604 0.546119
## factor(Team)ML4 -0.0086159  0.0284973  -0.302 0.762467
## factor(Team)MON -0.0062719  0.0266416  -0.235 0.813940
## factor(Team)NYA  0.0059161  0.0258890   0.229 0.819298
## factor(Team)NYN -0.0112707  0.0251377  -0.448 0.654008
## factor(Team)OAK  0.0099963  0.0251233   0.398 0.690812
## factor(Team)PHI -0.0183061  0.0250745  -0.730 0.465551
## factor(Team)PIT -0.0201697  0.0251800  -0.801 0.423346
## factor(Team)SDN -0.0208589  0.0250984  -0.831 0.406159
## factor(Team)SEA -0.0178425  0.0250648  -0.712 0.476752
## factor(Team)SFN  0.0041105  0.0250747   0.164 0.869827
## factor(Team)SLN  0.0083839  0.0250776   0.334 0.738224
## factor(Team)TBA -0.0197665  0.0268814  -0.735 0.462347
## factor(Team)TEX -0.0032015  0.0250570  -0.128 0.898361
## factor(Team)TOR -0.0038084  0.0250590  -0.152 0.879240
## factor(Team)WAS -0.0109524  0.0289762  -0.378 0.705540
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05988 on 846 degrees of freedom
##   (35 observations deleted due to missingness)
## Multiple R-squared:  0.2633, Adjusted R-squared:  0.232
## F-statistic: 8.401 on 36 and 846 DF,  p-value: < 2.2e-16
```

The result here is a very sharp contrast to the NBA model, where a number of the fixed effects were statistically significant; for MLB, none of them are.

When you add variables that are not statistically significant, it is logical that the R-squared will not go up very much, since you are not explaining very much. That is the case here, where the R-squared increases to only 0.26.

You may have noticed that under the R-squared is “Adj. R-squared” - where “adj.” is short for “adjusted”. This is useful to consider in this case. A simple fact about regression is that when you add variables, no matter if they are irrelevant, then you will increase the *unadjusted* R-squared. This is a consequence of the underlying algebra. We are trying to reproduce the relationship between a set of points, using a linear model, which is just an equation that produces another set of points. The closer the two sets of points, the better the model. But in the end, we could reproduce the original set of points by copying them - and in the algebra of regression this would mean providing a separate variable for each point. For example, in this regression we have 883 observations - and so if we had 883 variables in our regression we would fit the data exactly and the R-squared would be 1.0! Note that this would be true even if the variables had no logical connection with our data. The upshot of this is that adding variables increases R-squared, regardless of whether the variables really explain the data any better. Adjusted R-squared is an attempt to compensate for this effect, by reducing the value of R-squared as the number of variables in the regression increases. If the variables are statistically significant, then adjusted R-squared can still increase, but in this case we can see that with the addition of the fixed effects, adjusted R-squared has in fact fallen from 0.233 to 0.232. This is a strong suggestion that we should ignore the fixed effects.

The conclusion of this is that our second model, with just relsal and the lagged dependent variable, was our best model.

What is the impact of spending and performance in this model?

Our preferred regression model is wpc(t) = 0.284 + 0.361 x wpc(t-1) + 1.026 x relsal (t), where t refers to the season.

To work out the impact of relsal we need to eliminate the the lagged dependent variable from the equation, which we do by assuming a “steady state”- where wpc(t) = wpc(t-1). If this were the case then we would have

wpc = 1/(1-0.361) x (0.284 + 1.026 x relsal)

We can then work out these values of win percentage for very low relsal (0.01), average relsal (0.035) and very high relsal (0.06):

```
print(1/(1-0.361)*(0.284 + 1.026*.01))
```

```
## [1] 0.4605008
```

```
print(1/(1-0.361)*(0.284 + 1.026*.035))
```

```
## [1] 0.5006416
```

```
print(1/(1-0.361)*(0.284 + 1.026*.06))
```

```
## [1] 0.5407825
```

Self test

Suppose, as for the NBA, the value of the lagged dependent variable was 0.6. Use that value instead of 0.361 in the above equations. What difference does it make? Can you explain why?

The results suggest that while it is possible to buy success in MLB by increasing spending relative to your competitors, it is not that easy to do so. Even the very highest spending does not deliver a dominant performance. This might be a disappointment for those who think markets ought to work perfectly, but on the other hand, we would suggest, this is good news for baseball fans.

Conclusion

The case of MLB has much more in common with the NBA than the EPL because of similarities of the league systems. We ran essentially the same models as we did for the NBA, but we also identified a number of differences. Comparing with the NBA, we found that the lagged dependent variable was less important and all of the fixed effects were insignficant. Given our main focus was on relsal, we found that in MLB win percentage was notably less sensitive the relative wage spending than the NBA.

We conclude this week by looking at one more league that operates under the North American model, the National Hockey League (NHL).