FA3_Group3_Quijano_DSC1107

Julian Philip S. Quijano

2026-02-22

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3

## Warning: package 'ggplot2' was built under R version 4.4.3

## Warning: package 'tibble' was built under R version 4.4.3

## Warning: package 'tidyr' was built under R version 4.4.3

## Warning: package 'readr' was built under R version 4.4.3

## Warning: package 'purrr' was built under R version 4.4.3

## Warning: package 'dplyr' was built under R version 4.4.3

## Warning: package 'forcats' was built under R version 4.4.3

## Warning: package 'lubridate' was built under R version 4.4.3

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.5.1
## v ggplot2   4.0.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
# 1.1
load("ml_pay.rdata")
mlb_raw <- ml_pay

dim(mlb_raw)
```

```
## [1] 30 54
```

```
str(mlb_raw)
```

```
## 'data.frame':    30 obs. of  54 variables:
##  $ payroll       : num  1.12 1.38 1.16 1.97 1.46 ...
##  $ avgwin        : num  0.49 0.553 0.454 0.549 0.474 ...
##  $ Team.name.2014: Factor w/ 30 levels "Arizona Diamondbacks",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ p1998         : num  31.6 61.7 71.9 59.5 49.8 ...
##  $ p1999         : num  70.5 74.9 72.2 71.7 42.1 ...
##  $ p2000         : num  81 84.5 81.4 77.9 60.5 ...
##  $ p2001         : num  81.2 91.9 72.4 109.6 64 ...
##  $ p2002         : num  102.8 93.5 60.5 108.4 75.7 ...
##  $ p2003         : num  80.6 106.2 73.9 99.9 79.9 ...
##  $ p2004         : num  70.2 88.5 51.2 125.2 91.1 ...
##  $ p2005         : num  63 85.1 74.6 121.3 87.2 ...
##  $ p2006         : num  59.7 90.2 72.6 120.1 94.4 ...
##  $ p2007         : num  52.1 87.3 93.6 143 99.7 ...
##  $ p2008         : num  66.2 102.4 67.2 133.4 118.3 ...
##  $ p2009         : num  73.6 96.7 67.1 122.7 135.1 ...
##  $ p2010         : num  60.7 84.4 81.6 162.7 146.9 ...
##  $ p2011         : num  53.6 87 85.3 161.4 125.5 ...
##  $ p2012         : num  74.3 83.3 81.4 173.2 88.2 ...
##  $ p2013         : num  89.1 89.8 91 150.7 104.3 ...
##  $ p2014         : num  113 111 107 163 89 ...
##  $ X2014         : int  59 73 82 62 64 63 66 72 57 77 ...
##  $ X2013         : int  81 96 85 97 66 63 90 92 74 93 ...
##  $ X2012         : int  81 94 93 69 61 85 97 68 64 88 ...
##  $ X2011         : int  94 89 69 90 71 79 79 80 73 95 ...
##  $ X2010         : int  65 91 66 89 75 88 91 69 83 81 ...
##  $ X2009         : int  70 86 64 95 83 79 78 65 92 86 ...
##  $ X2008         : int  82 72 68 95 97 89 74 81 74 74 ...
##  $ X2007         : int  90 84 69 96 85 72 72 96 90 88 ...
##  $ X2006         : int  76 79 70 86 66 90 80 78 76 95 ...
##  $ X2005         : int  77 90 74 95 79 99 73 93 67 71 ...
##  $ X2004         : int  51 96 78 98 89 83 76 80 68 72 ...
##  $ X2003         : int  84 101 71 95 88 86 69 68 74 43 ...
##  $ X2002         : int  98 101 67 93 67 81 78 74 73 55 ...
##  $ X2001         : int  92 88 63 82 88 83 66 91 73 66 ...
##  $ X2000         : int  85 95 74 85 65 95 85 90 82 79 ...
##  $ X1999         : int  100 103 78 94 67 75 96 97 72 69 ...
##  $ X1998         : int  65 106 79 92 90 80 77 89 77 65 ...
##  $ X2014.pct     : num  0.415 0.514 0.577 0.437 0.451 ...
##  $ X2013.pct     : num  0.497 0.589 0.521 0.595 0.405 ...
##  $ X2012.pct     : num  0.5 0.58 0.574 0.426 0.377 ...
##  $ X2011.pct     : num  0.58 0.549 0.426 0.556 0.438 ...
##  $ X2010.pct     : num  0.401 0.562 0.407 0.549 0.463 ...
##  $ X2009.pct     : num  0.429 0.528 0.393 0.583 0.509 ...
##  $ X2008.pct     : num  0.503 0.442 0.417 0.583 0.595 ...
##  $ X2007.pct     : num  0.552 0.515 0.423 0.589 0.521 ...
##  $ X2006.pct     : num  0.469 0.488 0.432 0.531 0.407 ...
##  $ X2005.pct     : num  0.475 0.556 0.457 0.586 0.488 ...
##  $ X2004.pct     : num  0.315 0.593 0.481 0.605 0.549 ...
##  $ X2003.pct     : num  0.519 0.623 0.438 0.586 0.543 ...
##  $ X2002.pct     : num  0.605 0.623 0.414 0.574 0.414 ...
```

```
##  $ X2001.pct     : num  0.568 0.543 0.389 0.506 0.543 ...
##  $ X2000.pct     : num  0.525 0.586 0.457 0.525 0.401 ...
##  $ X1999.pct     : num  0.613 0.632 0.479 0.577 0.411 ...
##  $ X1998.pct     : num  0.399 0.65 0.485 0.564 0.552 ...
```

```
# 1.2
mlb_aggregate <- mlb_raw %>%
  transmute(
    team = Team.name.2014,
    payroll_aggregate = rowMeans(select(., starts_with("p")), na.rm = TRUE),
    pct_wins_aggregate = rowMeans(select(., ends_with(".pct")), na.rm = TRUE)
  )

dim(mlb_aggregate)
```

```
## [1] 30  3
```

```
head(mlb_aggregate)
```

```
##                   team payroll_aggregate pct_wins_aggregate
## 1 Arizona Diamondbacks          68.00583          0.4921264
## 2       Atlanta Braves          84.42731          0.5631539
## 3    Baltimore Orioles          72.57957          0.4570920
## 4       Boston Red Sox         116.97517          0.5512860
## 5         Chicago Cubs          86.28808          0.4745898
## 6    Chicago White Sox          76.46785          0.5069221
```

```
# Payroll
payroll_long <- mlb_raw %>%
  select(Team.name.2014, starts_with("p19"), starts_with("p20")) %>%
  pivot_longer(
    cols = -Team.name.2014,
    names_to = "year",
    values_to = "payroll"
  ) %>%
  mutate(year = as.numeric(sub("p", "", year)))

# Wins
wins_long <- mlb_raw %>%
  select(Team.name.2014, starts_with("X19"), starts_with("X20")) %>%
  select(-ends_with(".pct")) %>%
  pivot_longer(
    cols = -Team.name.2014,
    names_to = "year",
    values_to = "num_wins"
  ) %>%
  mutate(year = as.numeric(sub("X", "", year)))

# Win Percentage
pct_long <- mlb_raw %>%
  select(Team.name.2014, ends_with(".pct")) %>%
  pivot_longer(
```

```
    cols = -Team.name.2014,
    names_to = "year",
    values_to = "pct_wins"
  ) %>%
  mutate(year = as.numeric(sub("\\.pct", "", sub("X", "", year))))

# Merge
mlb_yearly <- payroll_long %>%
  rename(team = Team.name.2014) %>%
  left_join(wins_long %>%
              rename(team = Team.name.2014),
            by = c("team", "year")) %>%
  left_join(pct_long %>%
              rename(team = Team.name.2014),
            by = c("team", "year"))

dim(mlb_yearly)
```

```
## [1] 510   5
```

```
head(mlb_yearly)
```

```
## # A tibble: 6 x 5
##   team                   year payroll num_wins pct_wins
##   <fct>                 <dbl>   <dbl>    <int>    <dbl>
## 1 Arizona Diamondbacks  1998    31.6        65    0.399
## 2 Arizona Diamondbacks  1999    70.5       100    0.613
## 3 Arizona Diamondbacks  2000    81.0        85    0.525
## 4 Arizona Diamondbacks  2001    81.2        92    0.568
## 5 Arizona Diamondbacks  2002   103.         98    0.605
## 6 Arizona Diamondbacks  2003    80.6        84    0.519
```

Dataset contains 30 rows, 54 columns The dataset is a data frame containing numeric variables, integer variables, and one factor variable which identifies team name. mlb_yearly contains one observation per team per year. Since there are 30 teams and 17 seasons (1998–2014), it should contain $30 \times 17 = 510$ rows.

```
computed_aggregate <- mlb_yearly %>%
  group_by(team) %>%
  summarise(
    payroll_computed = mean(payroll, na.rm = TRUE),
    pct_wins_computed = mean(pct_wins, na.rm = TRUE)
  )

comparison <- computed_aggregate %>%
  left_join(mlb_aggregate, by = "team")

# Check differences
comparison <- comparison %>%
  mutate(
    payroll_diff = payroll_computed - payroll_aggregate,
    pct_diff = pct_wins_computed - pct_wins_aggregate
  )
```

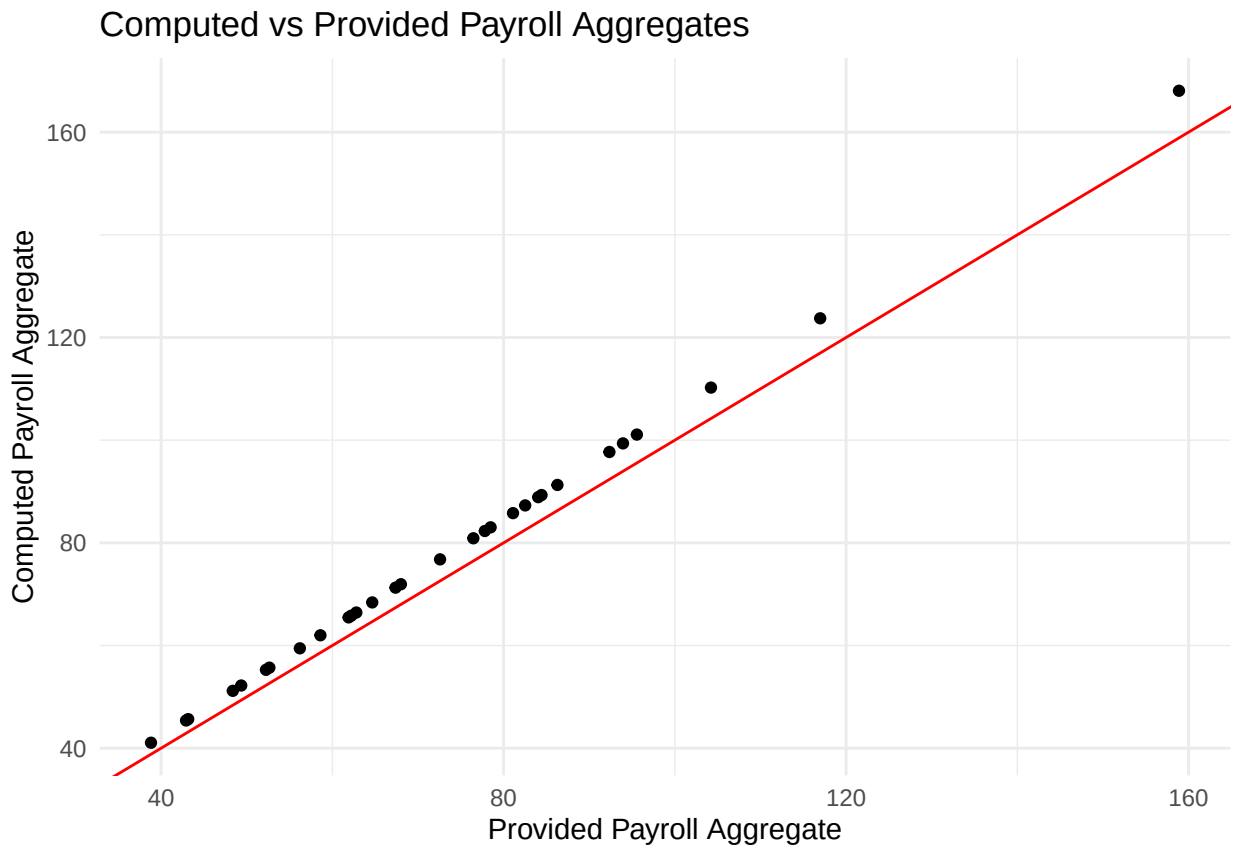
```r
summary(comparison$payroll_diff)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.244   3.284   4.068   4.270   4.879   9.187
```

```r
summary(comparison$pct_diff)
```

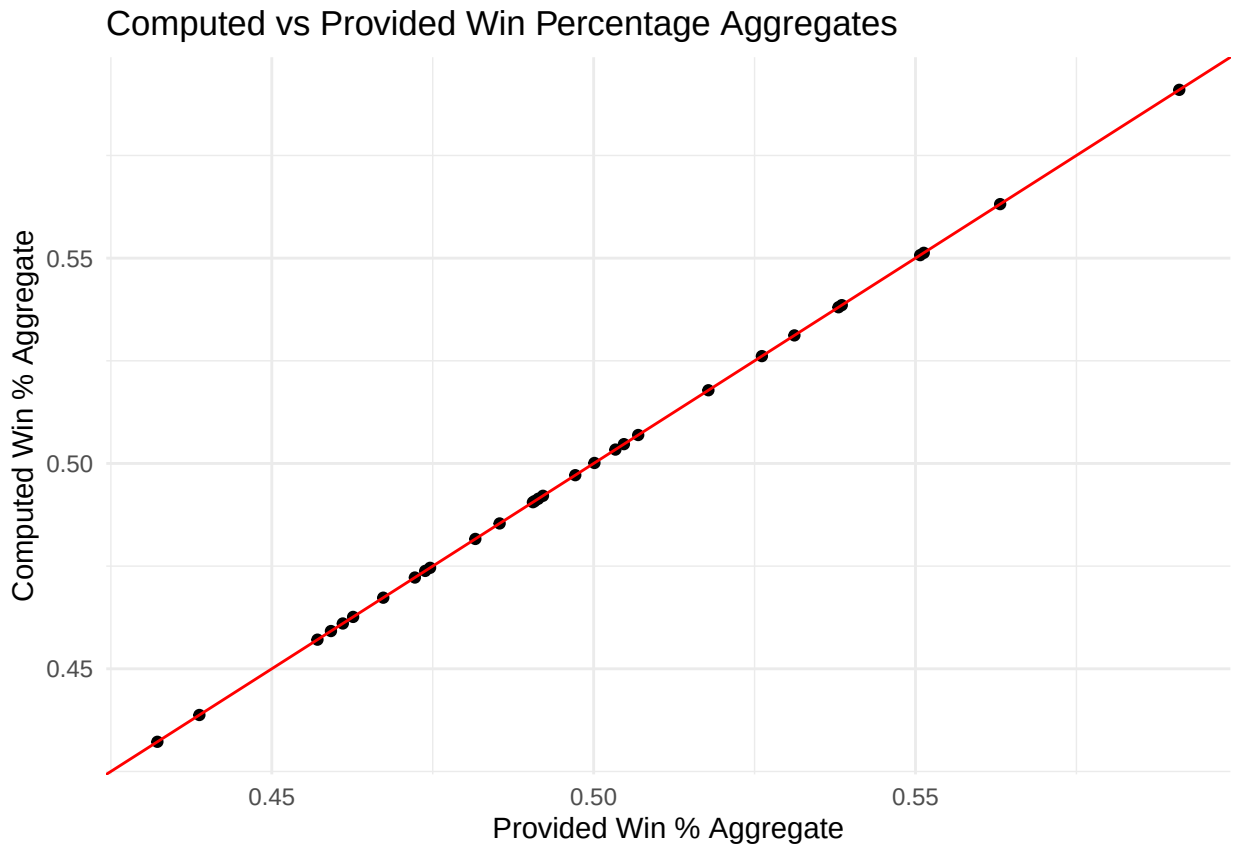
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0       0       0       0
```

```r
ggplot(comparison, aes(x = payroll_aggregate, y = payroll_computed)) +
  geom_point() +
  geom_abline(slope = 1, intercept = 0, color = "red") +
  labs(
    x = "Provided Payroll Aggregate",
    y = "Computed Payroll Aggregate",
    title = "Computed vs Provided Payroll Aggregates"
  ) +
  theme_minimal()
```




```r
ggplot(comparison, aes(x = pct_wins_aggregate, y = pct_wins_computed)) +
  geom_point() +
  geom_abline(slope = 1, intercept = 0, color = "red") +
```

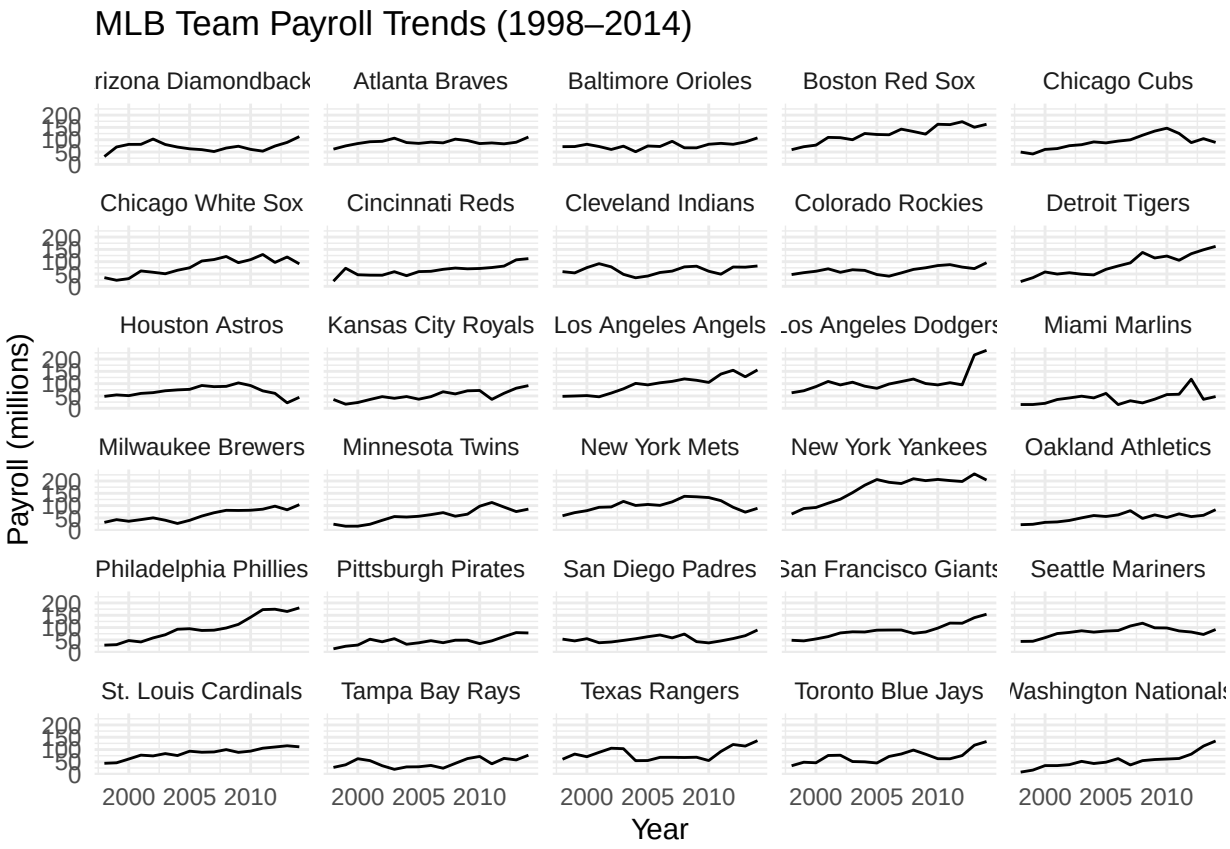
```
  labs(
    x = "Provided Win % Aggregate",
    y = "Computed Win % Aggregate",
    title = "Computed vs Provided Win Percentage Aggregates"
  ) +
  theme_minimal()
```



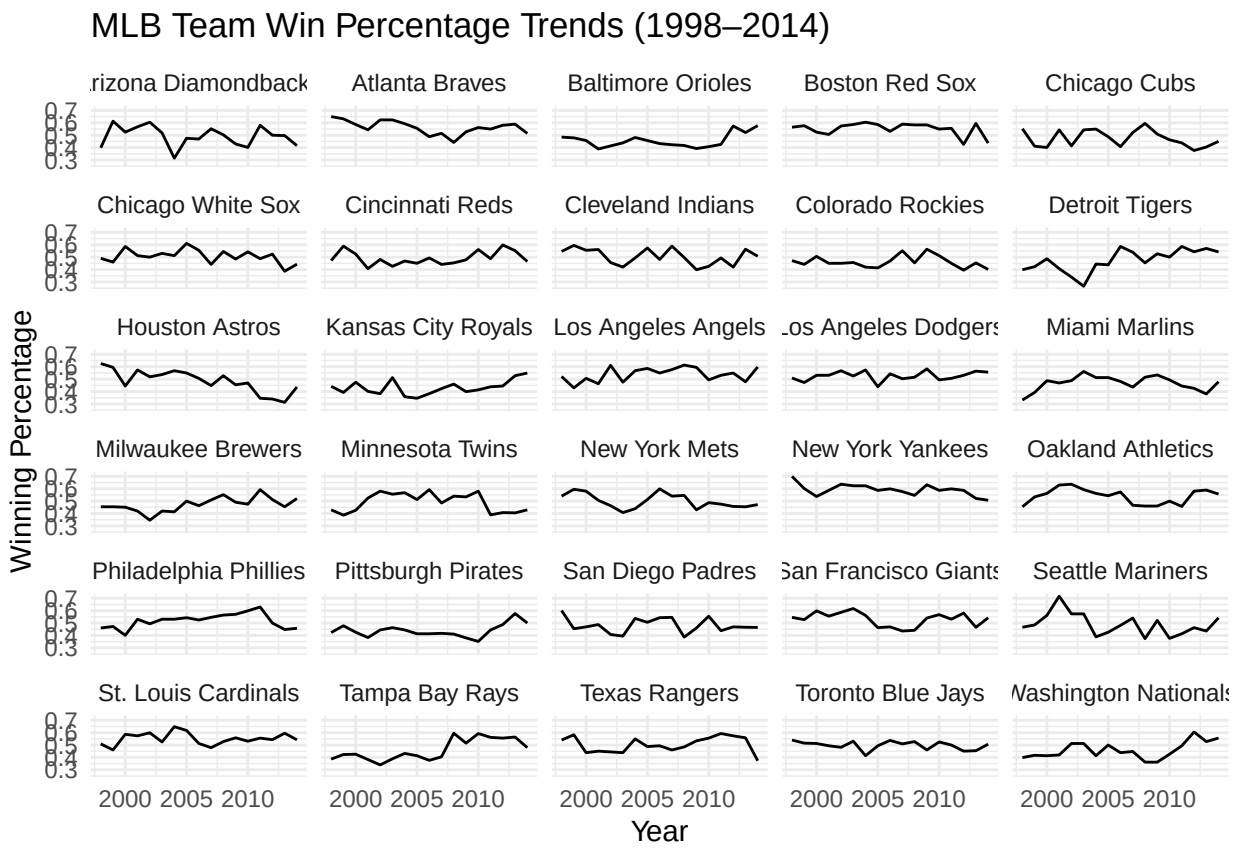
For payroll, the computed aggregates are slightly higher than the provided values (differences of 2–9), likely due to rounding or minor calculation differences. Overall, the datasets are consistent.

Differences are negligible and likely due to rounding. The scatter plots show points lying almost perfectly on the 45° line, confirming consistency.

```
ggplot(mlb_yearly, aes(x = year, y = payroll)) +
  geom_line() +
  facet_wrap(~ team, ncol = 5) +
  labs(
    title = "MLB Team Payroll Trends (1998-2014)",
    x = "Year",
    y = "Payroll (millions)"
  ) +
  theme_minimal()
```




```r
ggplot(mlb_yearly, aes(x = year, y = pct_wins)) +
  geom_line() +
  facet_wrap(~ team, ncol = 5) +
  labs(
    title = "MLB Team Win Percentage Trends (1998-2014)",
    x = "Year",
    y = "Winning Percentage"
  ) +
  theme_minimal()
```




```
mlb_aggregate %>%
  arrange(desc(payroll_aggregate)) %>%
  slice(1:3) %>%
  select(team, payroll_aggregate)
```

```
##                  team payroll_aggregate
## 1    New York Yankees          158.8775
## 2      Boston Red Sox          116.9752
## 3 Los Angeles Dodgers          104.2186
```

```
mlb_aggregate %>%
  arrange(desc(pct_wins_aggregate)) %>%
  slice(1:3) %>%
  select(team, pct_wins_aggregate)
```

```
##              team pct_wins_aggregate
## 1 New York Yankees          0.5909819
## 2   Atlanta Braves          0.5631539
## 3   Boston Red Sox          0.5512860
```

```
cor_test <- cor.test(
  mlb_aggregate$payroll_aggregate,
  mlb_aggregate$pct_wins_aggregate
)
```

```r
cor_test$estimate
```

```
##      cor
## 0.738008
```

```r
cor_test$p.value
```

```
## [1] 3.2492e-06
```

Payrolls vary widely across teams, with some consistently spending more over the 1998–2014 period. Win percentages fluctuate year-to-year, but some teams maintain higher performance over time.

The top 3 payroll teams are the Yankees, Dodgers, and Red Sox.

However the top 3 teams with the highest percent wins are the Yankees, Braves, and Red Sox.

While it is indicated that there is a general positive relationship with pay and winning, it is not a guarantee.

There is a strong positive relationship between aggregate payroll and aggregate win percentage. Teams with higher payrolls generally achieve higher winning percentages. The correlation is statistically significant, indicating that this relationship is unlikely to be due to chance. While payroll explains a large portion of performance variation, other factors may still influence team success.

```r
lm_model <- lm(pct_wins_aggregate ~ payroll_aggregate, data = mlb_aggregate)
summary(lm_model)
```

```
##
## Call:
## lm(formula = pct_wins_aggregate ~ payroll_aggregate, data = mlb_aggregate)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.039884 -0.014782 -0.001018  0.007697  0.067376
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.416334   0.014941  27.864  < 2e-16 ***
## payroll_aggregate 0.001111   0.000192   5.787 3.25e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02588 on 28 degrees of freedom
## Multiple R-squared:  0.5447, Adjusted R-squared:  0.5284
## F-statistic: 33.49 on 1 and 28 DF,  p-value: 3.249e-06
```

```r
# Estimated slope
coef(lm_model)["payroll_aggregate"]
```

```
## payroll_aggregate
##       0.001111093
```

```r
# p-value for slope
summary(lm_model)$coefficients["payroll_aggregate", "Pr(>|t|)"]
```

```
## [1] 3.2492e-06
```

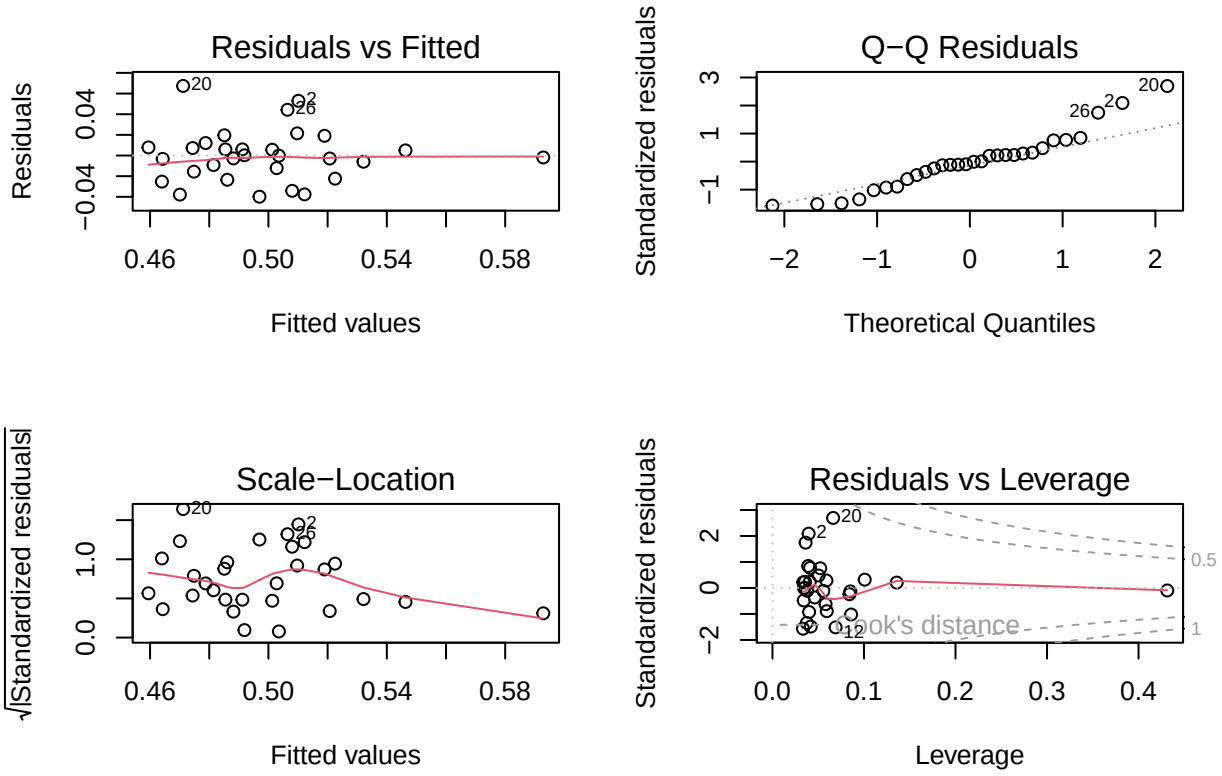
```r
# R-squared
summary(lm_model)$r.squared
```

```
## [1] 0.5446559
```

```r
# 95% Confidence interval for slope
confint(lm_model, "payroll_aggregate", level = 0.95)
```

```
##                              2.5 %      97.5 %
## payroll_aggregate 0.0007178179 0.001504368
```

```r
# Base R diagnostic plots
par(mfrow = c(2, 2))
plot(lm_model)
```




```r
n <- nrow(mlb_aggregate)
cooksd <- cooks.distance(lm_model)
influential <- which(cooksd > 4/n)

mlb_aggregate$team[influential]
```

```
## [1] Oakland Athletics
## 30 Levels: Arizona Diamondbacks Atlanta Braves ... Washington Nationals
```

```
cooksd[influential]
```

```
##        20
## 0.2577406
```

```
lm_model2 <- lm(
  pct_wins_aggregate ~ payroll_aggregate,
  data = mlb_aggregate[-influential[1], ]
)

summary(lm_model2)$coefficients
```

```
##                       Estimate   Std. Error  t value     Pr(>|t|)
## (Intercept)       0.406752088 0.0134613775 30.21623 2.366904e-22
## payroll_aggregate 0.001208294 0.0001712158  7.05714 1.378141e-07
```

```
summary(lm_model2)$r.squared
```

```
## [1] 0.6484522
```

For every additional $1 million in payroll, the team's aggregate winning percentage increases by 0.0011, or roughly 0.11%. For example, a $10 million increase in payroll corresponds to a 1.1 percentage point increase in winning percentage.

Residuals vs Fitted: Residuals scattered randomly around 0 → linearity assumption reasonable.

Normal Q-Q plot: Residuals lie close to the 45° line → residuals approximately normal.

Scale-Location plot: Residual spread roughly constant across fitted values → homoscedasticity reasonable.

Cook's Distance: Oakland Athletics has high Cook's D (0.258) → potentially influential.

Removing Oakland slightly increases the slope and R^2. This indicates the model is moderately sensitive to influential points, but the positive relationship between payroll and winning percentage remains strong. Overall, the model is stable, and conclusions about payroll impact are robust.

```
lm_log <- lm(pct_wins_aggregate ~ log(payroll_aggregate), data = mlb_aggregate)
summary(lm_log)
```

```
##
## Call:
## lm(formula = pct_wins_aggregate ~ log(payroll_aggregate), data = mlb_aggregate)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.044099 -0.016057 -0.001114  0.015034  0.070426
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.13376    0.06611   2.023   0.0527 .
```

```
## log(payroll_aggregate)  0.08575    0.01551   5.530 6.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02651 on 28 degrees of freedom
## Multiple R-squared:  0.5221, Adjusted R-squared:  0.505
## F-statistic: 30.58 on 1 and 28 DF,  p-value: 6.525e-06
```

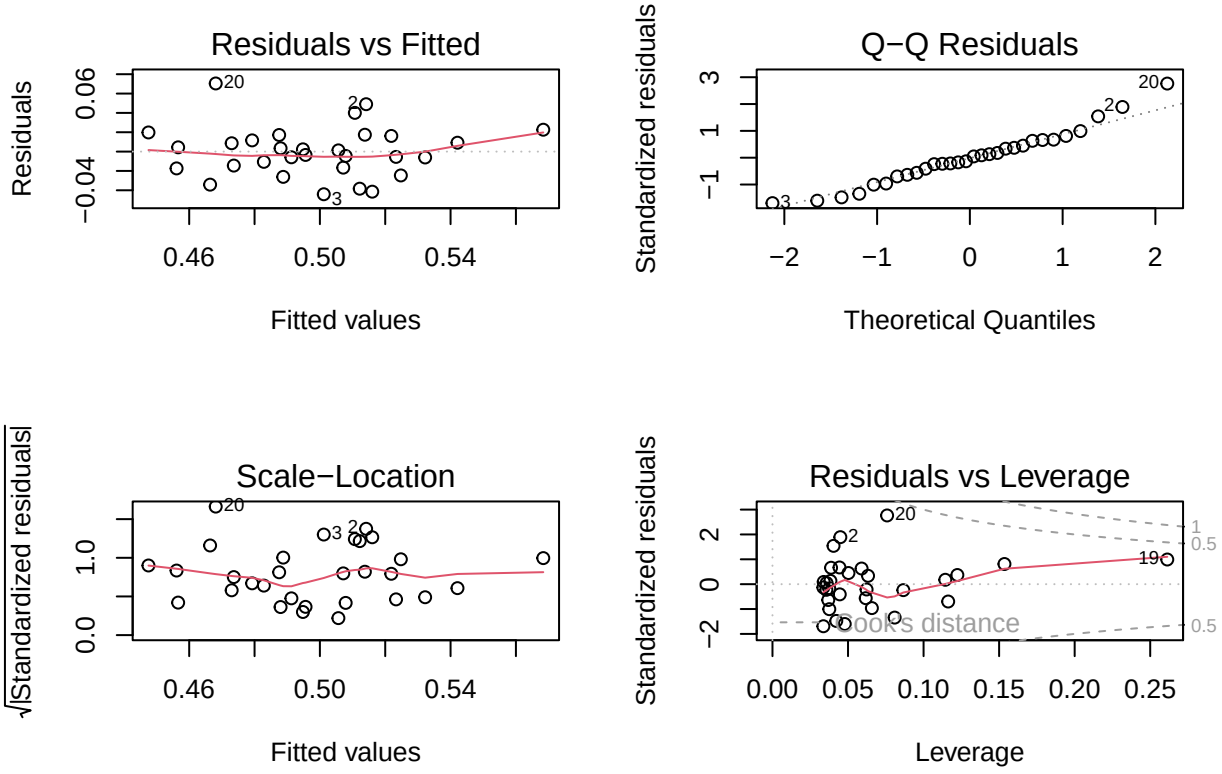
```
# Original model
summary(lm_model)$r.squared
```

```
## [1] 0.5446559
```

```
# Log-transformed model
summary(lm_log)$r.squared
```

```
## [1] 0.522051
```

```
# Diagnostic plots
par(mfrow = c(2,2))
plot(lm_log)
```



```r
mlb_aggregate <- mlb_aggregate %>%
  mutate(efficiency = pct_wins_aggregate / payroll_aggregate)

# Top 3 most efficient teams
top_efficiency <- mlb_aggregate %>%
  arrange(desc(efficiency)) %>%
  slice(1:3) %>%
  select(team, pct_wins_aggregate, payroll_aggregate, efficiency)

top_efficiency
```

```
##                team pct_wins_aggregate payroll_aggregate efficiency
## 1     Miami Marlins          0.4673161          38.82004 0.01203801
## 2 Oakland Athletics          0.5385489          49.35632 0.01091145
## 3    Tampa Bay Rays          0.4610341          43.16098 0.01068173
```

The diagnostic plots for the log-transformed model indicate a good fit to the linear regression assumptions. The Residuals vs Fitted plot shows residuals scattered evenly around zero with no clear pattern, supporting linearity. The Q-Q plot demonstrates that residuals are approximately normally distributed, with most points closely following the reference line. The Scale-Location plot reveals a fairly constant spread of residuals across fitted values, suggesting homoscedasticity is satisfied. Finally, the Residuals vs Leverage plot identifies no extreme influential points, indicating no observations unduly impact the model. Overall, these diagnostics suggest the log transformation improved model assumptions and fit compared to the original model.

The most efficient teams achieve high win percentages relative to low payroll. These teams may outperform expectations compared to regression predictions based solely on payroll. This aligns with the Moneyball concept, which emphasizes maximizing performance per dollar spent rather than simply having the highest payroll. Efficiency highlights teams that get more "bang for the buck," which is not always captured by payroll alone.

Github Link: https://github.com/SylTana/DSC1107---Data-Mining-and-Wrangling/tree/main/FA3