NBA

Gao

2023-10-09

Import libraries

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- 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
```

```
library(tidymodels)
```

```
## -- Attaching packages --------------------------------------- tidymodels 1.1.1 --
## v broom        1.0.5     v rsample      1.2.0
## v dials        1.2.0     v tune         1.1.2
## v infer        1.0.5     v workflows    1.1.3
## v modeldata    1.2.0     v workflowsets 1.0.1
## v parsnip      1.1.1     v yardstick    1.2.0
## v recipes      1.0.8
## -- Conflicts ------------------------------------------ tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

Import the data

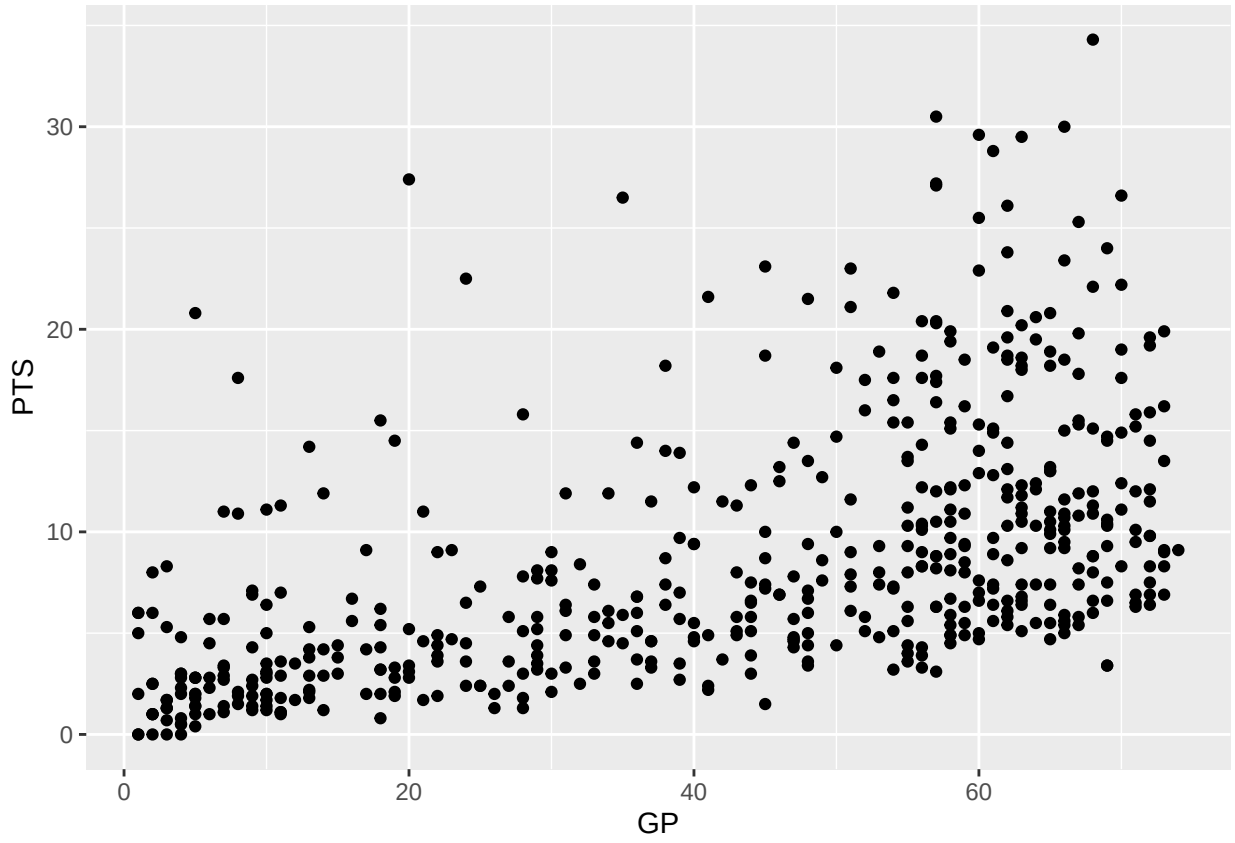
```
## [1] "/Users/andrewgao/Documents/GitHub/Advanced-Data-Science/Gao/Unit 3"
```

```
## Rows: 529 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
```

```
## chr (2): PLAYER, TEAM
## dbl (2): GP, PTS
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Create a plot

```
ggplot(NBA) + geom_point(aes(x = GP, y = PTS))
```



Create a linear regression model

```
model <- lm(PTS ~ GP, data = NBA)
model
```

```
##
## Call:
## lm(formula = PTS ~ GP, data = NBA)
##
## Coefficients:
## (Intercept)           GP
##      2.2528       0.1529
```

points-hat $= 2.253 + 0.153(\text{GP})$

Interpretation of the slope:

For each additional game played for an NBA player, we expect that his average points per game to increase by 0.153 points.

Interpretation of the y-intercept

When a player doesn't play any games, he will have an average point per game of 2.2528. This has no practical interpretations.

```
cor(NBA$GP, NBA$PTS)
```

```
## [1] 0.5435478
```

r = 0.544

There is a moderately strong positive linear correlation between games played by an NBA player and average points per game.

```
(cor(NBA$GP, NBA$PTS))^2
```

```
## [1] 0.2954442
```

r^2 = 0.2955

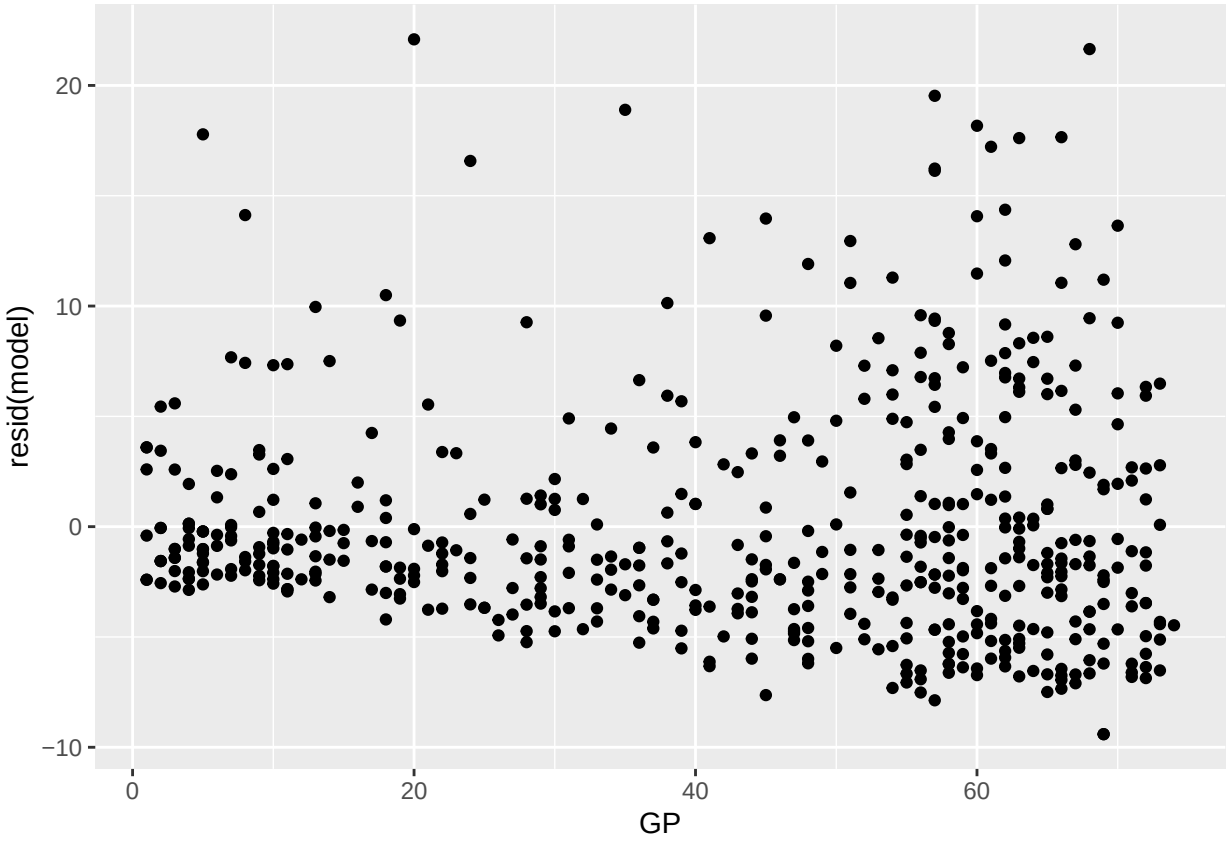
Approximately 29.55% of the variability in the mean points per game can be explained by this lienar model containing games played and mean points per game.

```
summary(model)
```

```
##
## Call:
## lm(formula = PTS ~ GP, data = NBA)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.405 -3.311 -1.412  2.377 22.089
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.25284    0.49371   4.563 6.28e-06 ***
## GP           0.15293    0.01029  14.866  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.35 on 527 degrees of freedom
## Multiple R-squared:  0.2954, Adjusted R-squared:  0.2941
## F-statistic:   221 on 1 and 527 DF,  p-value: < 2.2e-16
```

New plot containing

```
ggplot(NBA) + geom_point(aes(x = GP, y = resid(model)))
```



Interpret residual plot