NBA Salary Prediction

AUTHOR
Eric Cheung

```
library(tidyverse)
library(scales)
```

```
nba_sal <- read_csv("nba_2022-23_all_stats_with_salary.csv")
```

```
New names:
Rows: 467 Columns: 52
── Column specification
──────────────────────────────────────────────────────── Delimiter: "," chr
(3): Player Name, Position, Team dbl (49): ...1, Salary, Age, GP, GS, MP, FG,
FGA, FG%, 3P, 3PA, 3P%, 2P, 2PA...
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.
• `` -> `...1`
```

```
nba_dat <- nba_sal |> select(`Player Name`:Age, GP, MP, TRB:PTS)

nba_dat |>
  select(-`Player Name`) |>
  summarise(
    avg_salary = mean(Salary),
    avg_age = mean(Age),
    avg_GP = mean(GP),
    avg_MP = mean(MP),
    avg_TRB = mean(TRB),
    avg_AST = mean(AST),
    avg_STL = mean(STL),
    avg_BLK = mean(BLK),
    avg_TOV = mean(TOV),
    avg_PF = mean(PF),
    avg_PTS = mean(PTS)
  ) |>
  pivot_longer(
    cols = avg_salary:avg_PTS,
    names_to = "Averages",
    values_to = "Values"
  )
```

```
# A tibble: 11 × 2
   Averages        Values
   <chr>            <dbl>
 1 avg_salary 8416599.
 2 avg_age         25.8
```

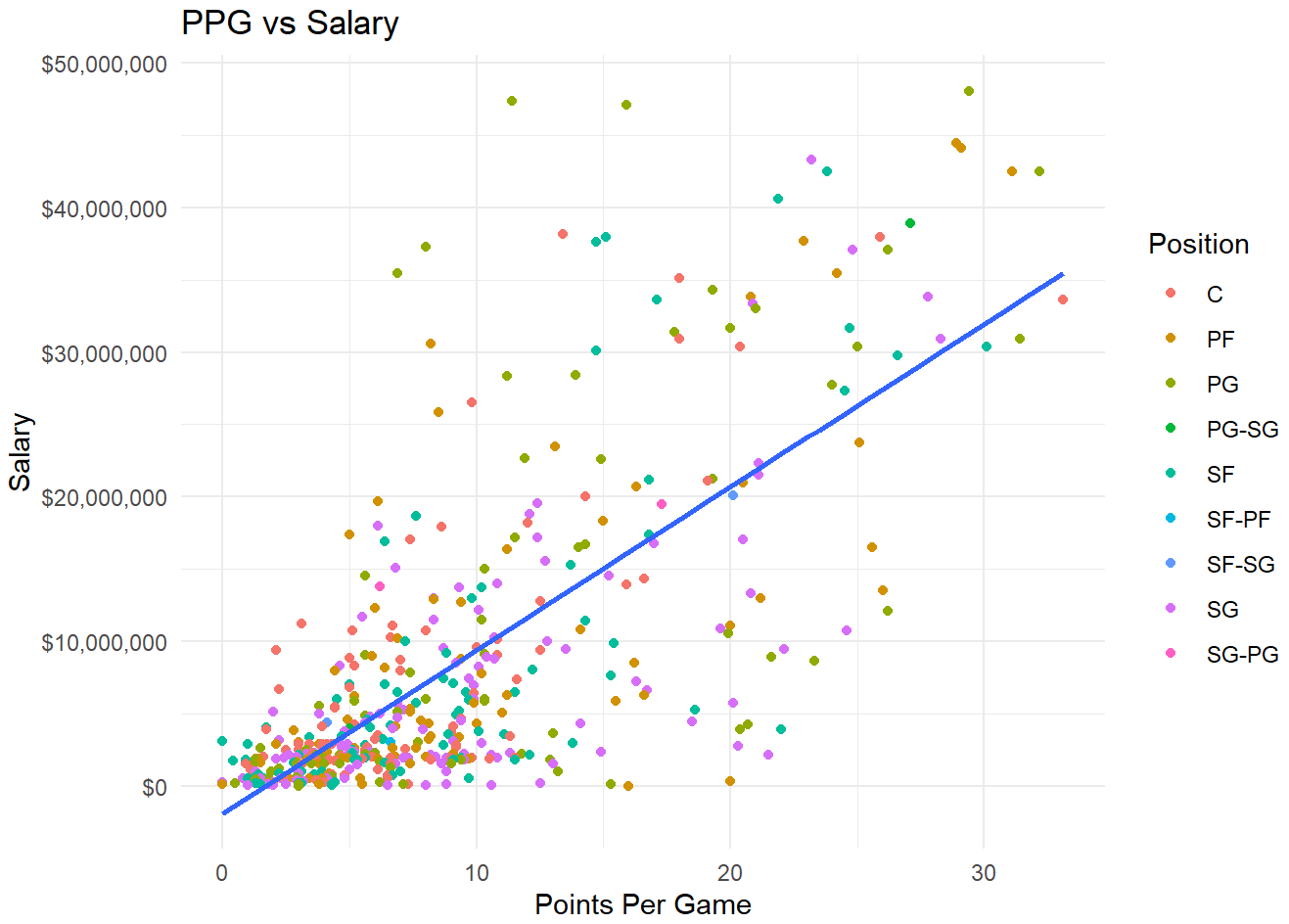
```
 3 avg_GP          48.2
 4 avg_MP          19.9
 5 avg_TRB          3.53
 6 avg_AST          2.11
 7 avg_STL          0.610
 8 avg_BLK          0.379
 9 avg_TOV          1.11
10 avg_PF           1.70
11 avg_PTS          9.13
```

Introduction

The average NBA player in the during the 2022-23 season made a salary of 8.4 million dollars while averaging 9.1 points, 3.5 rebounds, and 2.1 assists per game. I want to create some models that will attempt to predict an NBA players salary. I will based these models solely on statistics from the player's basic box score. Of course salaries in real life are based on many more factors than included in this analysis, But it will be interesting to see what salaries would be like based only on box score statistics. The data I have is of 467 NBA players' salaries and their basic statistics including points per game, assists per game, total rebound per game, etc.

```
nba_dat |> ggplot(aes(x = PTS, y = Salary)) +
  geom_point(aes(color = Position)) +
  geom_smooth(method = lm, se = FALSE) +
  scale_y_continuous(labels = label_dollar()) +
  labs(x = "Points Per Game",
       title = "PPG vs Salary") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

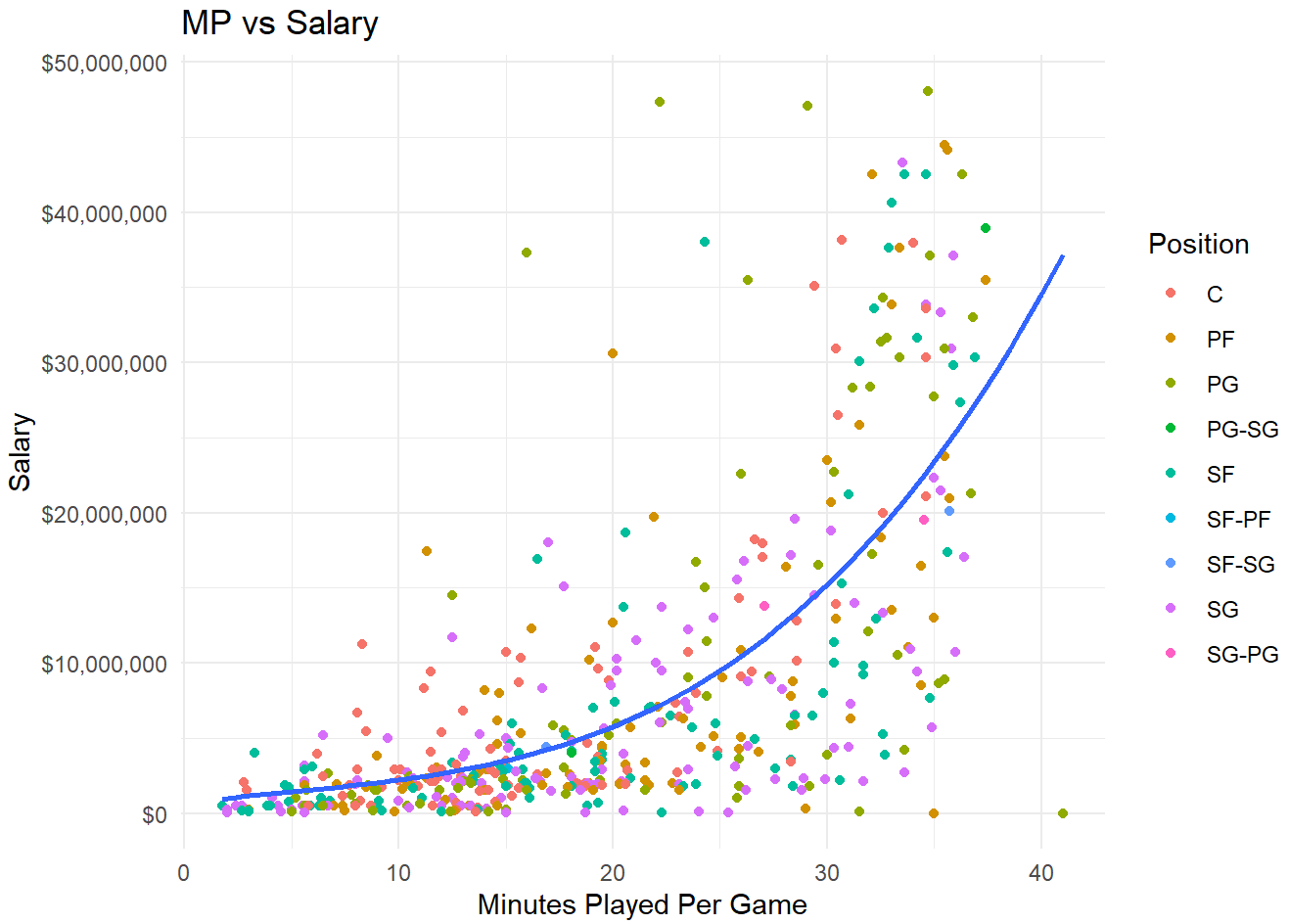



```
cor(nba_dat$Salary, nba_dat$PTS)
```

```
[1] 0.7275967
```

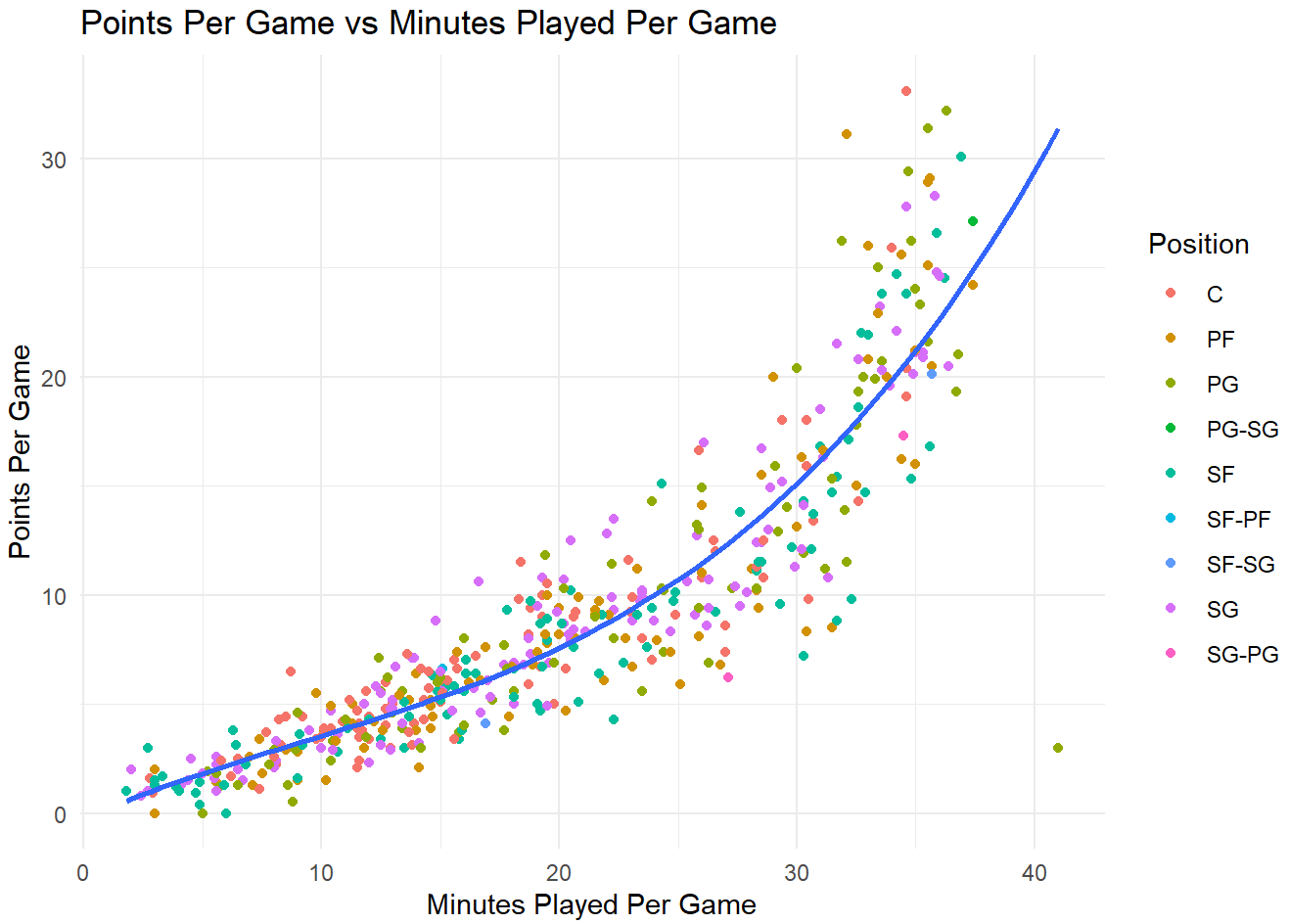
There seems to be a general trend that players who average more points per game make a higher salary. The correlation between those two variables is quite high at 0.72 which indicates a strong linear relationship. However, which position a player plays does not seem to have an effect on the salary as there are highly paid player from all five positions. Therefore I will not include the position variable in any of my models so it won't introduce any unneeded variance. I believe that points is one of the biggest on-court factor that contributes to a players salary therefore I will include it in all of my models.

```
nba_dat |> ggplot(aes(x = MP, y = Salary)) +
  geom_point(aes(color = Position)) +
  geom_smooth(method = "lm", formula = y ~ poly(x, 3), se = FALSE) +
  scale_y_continuous(labels = label_dollar()) +
  labs(x = "Minutes Played Per Game",
       title = "MP vs Salary") +
  theme_minimal()
```



There appears to be a polynomial relationship between minutes played and salary as most highly paid players do not play past 35 minutes per game with a few of the highest paid players are playing between 20-30 minutes a night. I will model this relationship in my model with a polynomial regression in one of the models

```
nba_dat |>
  ggplot(aes(x = MP, y = PTS)) +
  geom_point(aes(color = Position)) +
  geom_smooth(method = lm, formula = y ~ poly(x, 3), se = FALSE) +
  labs(title = "Points Per Game vs Minutes Played Per Game",
       x = "Minutes Played Per Game",
       y = "Points Per Game") +
  theme_minimal()
```



There is a strong polynomial relationship between points per game and minutes played per game. There has been a trend going on in the NBA where star players are not playing over 35 minutes per game compared to the early 2000s where stars would regularly play over 40 minutes. But perhaps this trend will not play much of an effect on the salaries of the players as they have not stopped growing ever since. I will model this relationship through an interaction term between minutes played and points per game.

The Models

Baseline

$$\log(\text{Salary}) = \beta_0 + \beta_1 \cdot \text{PTS} + \beta_2 \cdot \text{TRB} + \beta_3 \cdot \text{AST} + \epsilon$$

Linear

$$\log(\text{Salary}) = \beta_0 + \beta_1 \cdot \text{PTS} + \beta_2 \cdot \text{TRB} + \beta_3 \cdot \text{AST} + \beta_4 \cdot \text{Age} + \beta_5 \cdot MP + \epsilon$$

Polynomial

$$\log(\text{Salary}) = \beta_0 + \beta_1 \cdot \text{PTS} + \beta_2 \cdot \text{TRB} + \beta_3 \cdot \text{AST} + \beta_4 \cdot \text{Age} + \beta_5 \cdot \text{MP} + \beta_6 \cdot \text{MP}^2 + \beta_7 \cdot \text{MP}^3 + \epsilon$$

Interaction

$$\log(\text{Salary}) = \beta_0 + \beta_1 \cdot \text{PTS} + \beta_2 \cdot \text{TRB} + \beta_3 \cdot \text{AST} + \beta_4 \cdot \text{Age} + \beta_5 \cdot \text{MP} + \beta_6 \cdot \text{PTS} \times \text{MP} + \epsilon$$

I have a baseline model with just the three very basic box score statistics points, rebounds and assists per game. A second model with age and minutes played factored in, models three which includes a polynomial term and model four with an interaction term. I will use MSPE to measure accuracy.

```
get_MSPE <- function(Y, Y_hat) {
  residuals <- Y - Y_hat
  resid_sq <- residuals^2
  SSPE <- sum(resid_sq)
  MSPE <- SSPE / length(Y)
  return(MSPE)
}
```

```
set.seed(2003125)
n <- nrow(nba_dat)
n_fold <- ceiling(n/10)
order_ids <- rep(1:10, times = n_fold)
order_ids <- order_ids[1:n]
shuffle <- sample.int(n)
shuffled_ids <- order_ids[shuffle]

dat_cv <- nba_dat
dat_cv$Salary <- log(dat_cv$Salary)
dat_cv$fold <- shuffled_ids

CV_MSPEs <- array(0, dim = c(10, 4))
colnames(CV_MSPEs) <- c("Baseline", "Linear", "Poly", "Interaction")

for (i in 1:10) {
  data_train <- filter(dat_cv, fold != i)
  data_valid <- filter(dat_cv, fold == i)

  base_mod <- lm(Salary ~ PTS + TRB + AST, data = data_train)
  lin_mod <- lm(Salary ~ PTS + TRB + AST + Age + MP, data = data_train)
  poly_mod <- lm(Salary ~ PTS + TRB + AST + Age + poly(MP,3), data = data_train)
  intera_mod <- lm(Salary ~ PTS + TRB + AST + Age + MP + PTS:MP, data = data_train)

  base_pred <- predict(base_mod, data_valid)
  lin_pred <- predict(lin_mod, data_valid)
  poly_pred <- predict(poly_mod, data_valid)
  intera_pred <- predict(intera_mod, data_valid)

  Y_valid <- data_valid$Salary
  MSPE_base <- get_MSPE(Y_valid, base_pred)
  MSPE_lin <- get_MSPE(Y_valid, lin_pred)
  MSPE_poly <- get_MSPE(Y_valid, poly_pred)
  MSPE_intera <- get_MSPE(Y_valid, intera_pred)
```

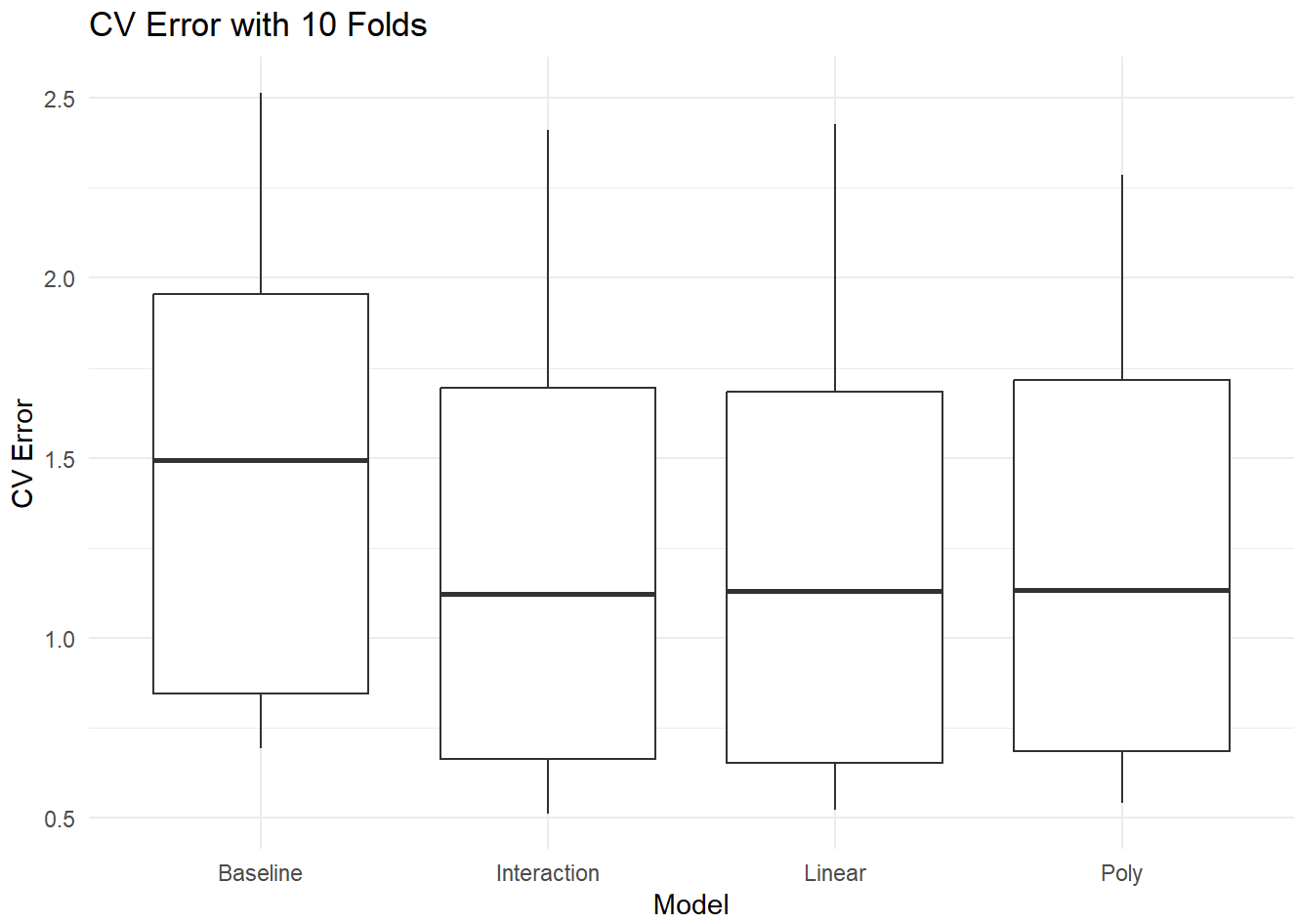
```
  CV_MSPEs[i, 1] <- MSPE_base
  CV_MSPEs[i, 2] <- MSPE_lin
  CV_MSPEs[i, 3] <- MSPE_poly
  CV_MSPEs[i, 4] <- MSPE_intera

}
```

I used 10 fold cross validation to judge the performance of my three models plus baseline. This will produce 10 calculation of MSPE for each of the 3 models. I will use the average of the 10 MSPE to use as a measure of the predictive power of each of the models.

Conclusion

```
CV_dataframe <-  as_tibble(CV_MSPEs)
CV_dataframe |>
pivot_longer(cols = everything(),
names_to = "Model",
values_to = "CV_Error") |>
ggplot(aes(Model, CV_Error)) +
geom_boxplot() +
theme_bw() +
labs(y = "CV Error",
title = "CV Error with 10 Folds") +
  theme_minimal()
```




```
CV_dataframe |> colMeans()
```

```
   Baseline      Linear        Poly Interaction
   1.490181    1.235302    1.224824    1.231477
```

The models I chose generally performed about the same as they have a similar MSPE averaged out from the 10 fold cross validation. They all have a 20% decrease in MSPE compared to the baseline. Seeing as how they all perform about the same I will be using the linear model as it is the most simple which will provide a good balance between model bias and variance.

```
imaginary_players <- tibble(PTS = c(15.6, 35.6, 15.1),
                            TRB = c(7.3, 8.7, 10.4),
                            AST = c(3.6, 12.2, 2.1),
                            Age = c(26, 28, 32),
                            MP = c(30.2, 33.6, 30.3))
imaginary_players
```

```
# A tibble: 3 × 5
    PTS   TRB   AST   Age    MP
  <dbl> <dbl> <dbl> <dbl> <dbl>
1  15.6   7.3   3.6    26  30.2
```

```
2  35.6   8.7  12.2    28  33.6
3  15.1  10.4   2.1    32  30.3
```

```
predicted_log_salaries <- predict(lin_mod, newdata = imaginary_players)

predicted_salaries <- exp(predicted_log_salaries)
predicted_salaries
```

```
       1        2        3
11379414 28666647 27334004
```

So according to my model a solid starter would make 11.4 million, an MVP winner would make 28.7 million and an all star caliber player would make 27.3 million. Its interesting to see what salaries would be like if they were only based off of on-course production. But of course there are many factors that are not represented here such as marketability, future potential, or past accomplishments.