Baseball3_TY

Thomas Young

2025-04-03

Load Hitters DATA

Hitters Data Set from Kaggle: https://www.kaggle.com/datasets/floser/hitters

```
hitters <- read_csv("Hitters.csv")
```

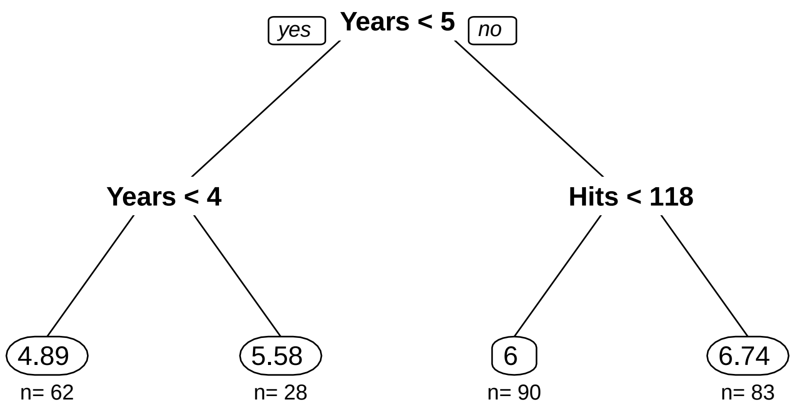
```
## Rows: 322 Columns: 20
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): League, Division, NewLeague
## dbl (17): AtBat, Hits, HmRun, Runs, RBI, Walks, Years, CAtBat, CHits, CHmRun...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# clean
hitters <- hitters |>  drop_na()

# Create variable for log salary
hitters <- hitters |>  mutate(LogSalary = log(Salary))
```

Regression Example

```
tree_model <- rpart(LogSalary ~ Years + Hits, data = hitters,
                    control = rpart.control(maxdepth = 2, cp = 0.01))
visualize_model(tree_model)
```



Compare to another model

```
mod1 <- lm(LogSalary ~ Years + Hits, data = hitters)
summary(mod1)
```

```
##
## Call:
## lm(formula = LogSalary ~ Years + Hits, data = hitters)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1840 -0.4909  0.0124  0.4376  3.1825
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.2751287  0.1183953  36.109   <2e-16 ***
## Years       0.0981627  0.0082805  11.855   <2e-16 ***
## Hits        0.0086651  0.0008796   9.851   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6424 on 260 degrees of freedom
## Multiple R-squared:  0.4821, Adjusted R-squared:  0.4781
## F-statistic:   121 on 2 and 260 DF,  p-value: < 2.2e-16
```

Compare predictions

```
predictions <- predict(tree_model, newdata = hitters)

# Add predictions to df:
hitters$PredictedLogSalary <- predictions

# Compute R2
R2 <- 1 - (sum((hitters$LogSalary - predictions)^2) / sum((hitters$LogSalary - mean(hitters$LogSalary))
R2
```

```
## [1] 0.6035802
```

Interpretation

Root Node (Years < 5): This is the starting point of the tree. It splits the data based on whether the number of years played (Years) is less than 5 or not.

- Yes Branch (Years < 5): If a player has played less than 5 years, the tree follows this branch.
- No Branch (Years >= 5): If a player has played 5 or more years, the tree follows this branch

Left Branch of Root Node (Years < 5):

Node (Years < 4): This node further splits the players who have played less than 5 years based on whether they have played less than 4 years

- Left Leaf (Circle with 4.89, n=62): This represents a terminal node (leaf) where the prediction is made. The value 4.89 is the mean log salary for the 62 players who have played less than 4 years.
- Right Leaf (Circle with 5.58, n=28): This represents another terminal node where the prediction is made. The value 5.58 is the mean log salary for the 28 players who have played between 4 and 5 years (Years >= 4 and Years < 5).

Right Branch of Root Node (Years >= 5):

Node (Hits < 118): This node splits the players who have played 5 or more years based on whether they had less than 118 hits in the previous year.

- Left Leaf (Circle with 8, n=90): This represents a terminal node where the prediction is made. The value 8 is the mean log salary for the 90 players who have played 5 or more years and had less than 118 hits. This could potentially represent outlier data (very high salary for a relatively low number of hits.) It would be worth investigating these data points.
- Right Leaf (Circle with 6.47, n=83): This represents another terminal node where the prediction is made. The value 6.47 is the mean log salary for the 83 players who have played 5 or more years and had 118 or more hits.

Overall Performance of this model was 60%, which was expected as we used only two vairbales.