Dodgers Promotion

Sam Sears

September 15, 2020

# Packages

library(ISLR)  
library(tidyverse)

## -- Attaching packages --------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.2.1 v purrr 0.3.3  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 1.0.0 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts ------------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(dplyr)

# Load Data

data = read.csv('dodgers\_data\_for\_modeling.csv')  
str(data)

## 'data.frame': 81 obs. of 20 variables:  
## $ month : Factor w/ 7 levels "APR","AUG","JUL",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ day : int 10 11 12 13 14 15 23 24 25 27 ...  
## $ attend : int 56000 29729 28328 31601 46549 38359 26376 44014 26345 44807 ...  
## $ day\_of\_week: Factor w/ 7 levels "Friday","Monday",..: 6 7 5 1 3 4 2 6 7 1 ...  
## $ opponent : Factor w/ 17 levels "Angels","Astros",..: 13 13 13 11 11 11 3 3 3 10 ...  
## $ temp : int 67 58 57 54 57 65 60 63 64 66 ...  
## $ skies : Factor w/ 2 levels "Clear ","Cloudy": 1 2 2 2 2 1 2 2 2 1 ...  
## $ day\_night : Factor w/ 2 levels "Day","Night": 1 2 2 2 2 1 2 2 2 2 ...  
## $ cap : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ shirt : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ fireworks : int 0 0 0 1 0 0 0 0 0 1 ...  
## $ bobblehead : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ promotions : int 0 0 0 1 0 0 0 0 0 1 ...  
## $ Friday : int 0 0 0 1 0 0 0 0 0 1 ...  
## $ Monday : int 0 0 0 0 0 0 1 0 0 0 ...  
## $ Saturday : int 0 0 0 0 1 0 0 0 0 0 ...  
## $ Sunday : int 0 0 0 0 0 1 0 0 0 0 ...  
## $ Thursday : int 0 0 1 0 0 0 0 0 0 0 ...  
## $ Tuesday : int 1 0 0 0 0 0 0 1 0 0 ...  
## $ Wednesday : int 0 1 0 0 0 0 0 0 1 0 ...

# Predictive Modeling

## Split Data

# Split dataset into Test and Train  
set.seed(100)  
train\_size = floor(0.80\*nrow(data))  
train\_index = sample(seq\_len(nrow(data)), size = train\_size)  
train = data[train\_index,]  
test = data[-train\_index,]  
str(train)

## 'data.frame': 64 obs. of 20 variables:  
## $ month : Factor w/ 7 levels "APR","AUG","JUL",..: 5 5 3 1 4 7 2 5 3 5 ...  
## $ day : int 26 18 13 14 28 29 22 29 1 7 ...  
## $ attend : int 36561 40906 43873 46549 49006 40724 40173 51137 55359 43713 ...  
## $ day\_of\_week: Factor w/ 7 levels "Friday","Monday",..: 3 1 1 3 5 3 7 6 4 2 ...  
## $ opponent : Factor w/ 17 levels "Angels","Astros",..: 2 5 11 11 9 15 7 4 9 7 ...  
## $ temp : int 61 64 76 57 75 84 75 74 75 67 ...  
## $ skies : Factor w/ 2 levels "Clear ","Cloudy": 2 1 2 2 1 2 1 1 1 1 ...  
## $ day\_night : Factor w/ 2 levels "Day","Night": 2 2 2 2 2 2 2 2 2 2 ...  
## $ cap : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ shirt : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ fireworks : int 0 1 1 0 0 0 0 0 0 0 ...  
## $ bobblehead : int 0 0 0 0 1 0 0 1 1 0 ...  
## $ promotions : int 0 1 1 0 1 0 0 1 1 0 ...  
## $ Friday : int 0 1 1 0 0 0 0 0 0 0 ...  
## $ Monday : int 0 0 0 0 0 0 0 0 0 1 ...  
## $ Saturday : int 1 0 0 1 0 1 0 0 0 0 ...  
## $ Sunday : int 0 0 0 0 0 0 0 0 1 0 ...  
## $ Thursday : int 0 0 0 0 1 0 0 0 0 0 ...  
## $ Tuesday : int 0 0 0 0 0 0 0 1 0 0 ...  
## $ Wednesday : int 0 0 0 0 0 0 1 0 0 0 ...

str(test)

## 'data.frame': 17 obs. of 20 variables:  
## $ month : Factor w/ 7 levels "APR","AUG","JUL",..: 1 1 1 1 5 5 5 5 4 4 ...  
## $ day : int 10 11 12 29 12 19 27 28 15 17 ...  
## $ attend : int 56000 29729 28328 48753 33735 39383 33306 38016 40432 53504 ...  
## $ day\_of\_week: Factor w/ 7 levels "Friday","Monday",..: 6 7 5 4 3 3 4 2 1 4 ...  
## $ opponent : Factor w/ 17 levels "Angels","Astros",..: 13 13 13 10 15 5 2 4 17 17 ...  
## $ temp : int 67 58 57 74 65 67 70 73 67 74 ...  
## $ skies : Factor w/ 2 levels "Clear ","Cloudy": 1 2 2 1 1 1 1 1 1 1 ...  
## $ day\_night : Factor w/ 2 levels "Day","Night": 1 2 2 1 2 2 1 2 2 1 ...  
## $ cap : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ shirt : int 0 0 0 1 0 0 0 0 0 0 ...  
## $ fireworks : int 0 0 0 0 0 0 0 0 1 0 ...  
## $ bobblehead : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ promotions : int 0 0 0 1 0 0 0 0 1 0 ...  
## $ Friday : int 0 0 0 0 0 0 0 0 1 0 ...  
## $ Monday : int 0 0 0 0 0 0 0 1 0 0 ...  
## $ Saturday : int 0 0 0 0 1 1 0 0 0 0 ...  
## $ Sunday : int 0 0 0 1 0 0 1 0 0 1 ...  
## $ Thursday : int 0 0 1 0 0 0 0 0 0 0 ...  
## $ Tuesday : int 1 0 0 0 0 0 0 0 0 0 ...  
## $ Wednesday : int 0 1 0 0 0 0 0 0 0 0 ...

## Fit Models

# Create Linear Model  
linear\_model\_promotion <- lm(attend ~ day\_of\_week + promotions, data = data)  
linear\_model\_bobblehead <- lm(attend ~ day\_of\_week + bobblehead, data = data)  
print(summary(linear\_model\_promotion))

##   
## Call:  
## lm(formula = attend ~ day\_of\_week + promotions, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -17898.2 -4090.3 50.1 3753.5 14724.3   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 29611 2748 10.774 < 2e-16 \*\*\*  
## day\_of\_weekMonday 4480 3233 1.386 0.170115   
## day\_of\_weekSaturday 11846 3106 3.813 0.000284 \*\*\*  
## day\_of\_weekSunday 10234 3020 3.388 0.001137 \*\*   
## day\_of\_weekThursday 6594 3664 1.800 0.076026 .   
## day\_of\_weekTuesday 11665 2690 4.336 4.57e-05 \*\*\*  
## day\_of\_weekWednesday 7099 3233 2.196 0.031292 \*   
## promotions 10506 2061 5.097 2.62e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6554 on 73 degrees of freedom  
## Multiple R-squared: 0.4307, Adjusted R-squared: 0.3761   
## F-statistic: 7.89 on 7 and 73 DF, p-value: 4.254e-07

print(summary(linear\_model\_bobblehead))

##   
## Call:  
## lm(formula = attend ~ day\_of\_week + bobblehead, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12076.2 -3592.2 -311.9 3050.3 15984.8   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 40117 1816 22.091 < 2e-16 \*\*\*  
## day\_of\_weekMonday -5151 2621 -1.965 0.0532 .   
## day\_of\_weekSaturday 1015 2596 0.391 0.6971   
## day\_of\_weekSunday 1181 2575 0.459 0.6478   
## day\_of\_weekThursday -4757 3584 -1.327 0.1886   
## day\_of\_weekTuesday 1800 2809 0.641 0.5236   
## day\_of\_weekWednesday -2532 2621 -0.966 0.3373   
## bobblehead 12619 2467 5.116 2.44e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6548 on 73 degrees of freedom  
## Multiple R-squared: 0.4318, Adjusted R-squared: 0.3773   
## F-statistic: 7.924 on 7 and 73 DF, p-value: 3.992e-07

# RSE of promotion model / average target variable  
6554/mean(data$attend)

## [1] 0.1596976

# RSE of bobblehead model / average target variable  
6548/mean(data$attend)

## [1] 0.1595514

This tells use that the average difference between the line of best fit and the actual attendence is 6554 or 6548. To put that into context, those are about is 16% of the average attendence. In otherwords, it’s pretty close, but not increadibly precise.

confint(linear\_model\_promotion, conf.level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 24133.2211 35087.68  
## day\_of\_weekMonday -1964.1349 10923.49  
## day\_of\_weekSaturday 5654.8610 18037.32  
## day\_of\_weekSunday 4214.3259 16253.32  
## day\_of\_weekThursday -707.9023 13896.62  
## day\_of\_weekTuesday 6303.8055 17026.71  
## day\_of\_weekWednesday 655.3651 13542.99  
## promotions 6398.4753 14614.47

confint(linear\_model\_bobblehead, conf.level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 36497.608 43736.23857  
## day\_of\_weekMonday -10375.288 72.77553  
## day\_of\_weekSaturday -4159.418 6188.70559  
## day\_of\_weekSunday -3951.191 6313.68059  
## day\_of\_weekThursday -11900.222 2386.12263  
## day\_of\_weekTuesday -3798.613 7399.09040  
## day\_of\_weekWednesday -7755.788 2692.27553  
## bobblehead 7702.701 17534.93187

From these models, if we decided we were going to run a promotion, or bobblehead promotion (holding the promotions or bobblehead variable constant), we could see the associated affect of the day of the week. If we were going to run a promotion, the day of the week that would be associated with the highest attendence would be Saturday and we could expect an 5654 to 18037 more in attendence at a 95% confidence level. For a bobblehead promotion specifically, we can’t be sure a specific day of the week would even be associated with higher attendence.

## Test Model

linear\_model\_promotion <- lm(attend ~ day\_of\_week + promotions, data = train)  
summary(linear\_model\_promotion)

##   
## Call:  
## lm(formula = attend ~ day\_of\_week + promotions, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15844.3 -3329.8 50.9 3821.0 14852.2   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 29603 3139 9.430 3.67e-13 \*\*\*  
## day\_of\_weekMonday 4078 3571 1.142 0.258244   
## day\_of\_weekSaturday 13265 3487 3.804 0.000354 \*\*\*  
## day\_of\_weekSunday 9574 3487 2.746 0.008099 \*\*   
## day\_of\_weekThursday 8283 4049 2.046 0.045476 \*   
## day\_of\_weekTuesday 9043 3141 2.879 0.005637 \*\*   
## day\_of\_weekWednesday 7689 3571 2.153 0.035600 \*   
## promotions 11082 2283 4.855 1.01e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6464 on 56 degrees of freedom  
## Multiple R-squared: 0.4512, Adjusted R-squared: 0.3826   
## F-statistic: 6.578 on 7 and 56 DF, p-value: 1.114e-05

predictions = linear\_model\_promotion %>% predict(test)  
  
RMSE(predictions, test$attend)

## [1] 7388.331

RMSE(predictions, test$attend)/mean(test$attend)

## [1] 0.1788104

R2(predictions, test$attend)

## [1] 0.2537367

RMSE of the test dataset is 7388 which gives an error rate of 18% which isn’t great. However, this is pretty close to the train data RMSE of 6464 which provides evidence that the model holds. The R-square value of the test set is 0.25 which is means there is a somewhat low correlation between the predicted attendence and the actual attendence, but it is similar to the Adjusted R-squared of the train dataset.

# Final Conclusion

**Which night would be best to run a marketing promotion?**  
In otherwords, if we decide to run a promotion (control for the promotion variable) which day\_of\_week is associated with the highest increase in attendence? The answer is Saturday. Our data tells us that we can be 95% confident that if we decide to do a promotion, and if we choose Saturday as our night to do it, we can expect 5654 to 18037 more in attendence.