xgb

2024-12-07

‘ ## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

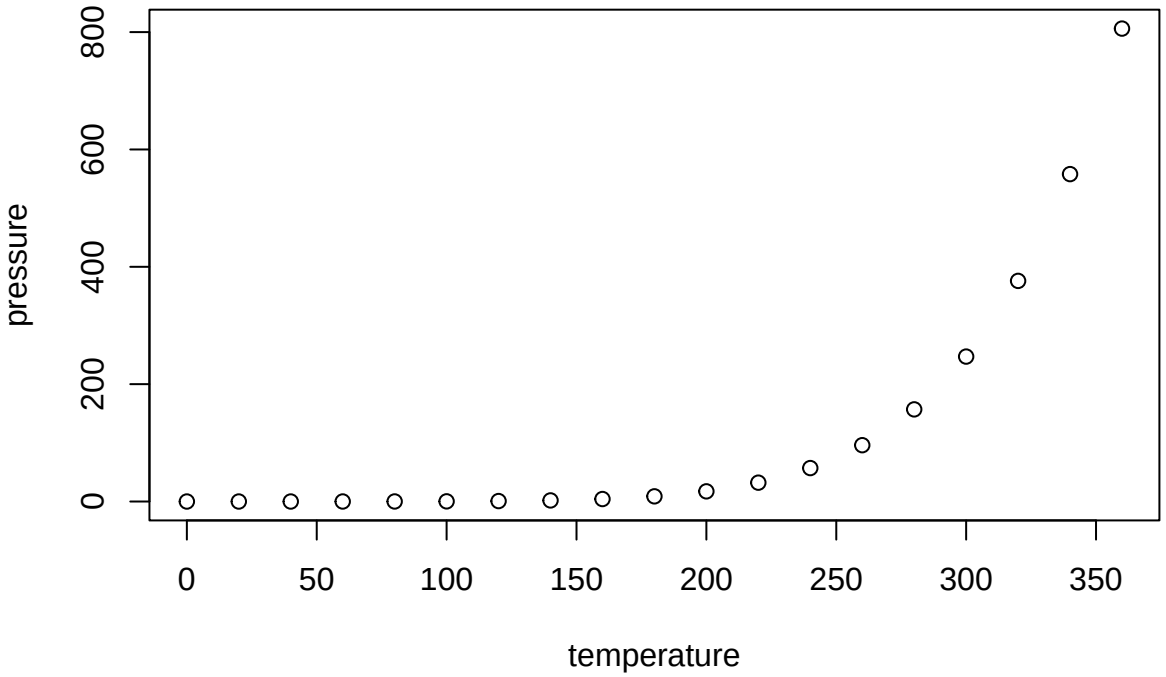
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```r
library(xgboost)
set.seed(42)
train_x <- read.csv("xgb_train_x.csv")
train_y <- read.csv("xgb_train_y.csv")
test_x <- read.csv("xgb_test_x.csv")
test_y <- read.csv("xgb_test_y.csv")


#Define final training and testing sets
xgb_train = xgb.DMatrix(data = as.matrix(train_x), label = unlist(train_y))
xgb_test = xgb.DMatrix(data = as.matrix(test_x), label = unlist(test_y))


#Define watchlist
watchlist = list(train=xgb_train, test=xgb_test)

#Fit XGBoost model and display training and testing data at each round
xgb_model = xgb.train(data = xgb_train, max.depth = 3,
                  watchlist=watchlist, nrounds = 70)
```

```
## [1]  train-rmse:16.351229    test-rmse:16.484171
## [2]  train-rmse:11.811347    test-rmse:12.128655
## [3]  train-rmse:8.581002 test-rmse:8.840908
## [4]  train-rmse:6.285137 test-rmse:6.465342
## [5]  train-rmse:4.673629 test-rmse:4.822035
## [6]  train-rmse:3.511651 test-rmse:3.435052
## [7]  train-rmse:2.675627 test-rmse:2.498768
## [8]  train-rmse:2.069472 test-rmse:1.834512
## [9]  train-rmse:1.636494 test-rmse:1.290090
## [10] train-rmse:1.314316 test-rmse:0.962761
## [11] train-rmse:1.081714 test-rmse:0.778236
## [12] train-rmse:0.899081 test-rmse:0.742409
## [13] train-rmse:0.757631 test-rmse:0.729652
## [14] train-rmse:0.641829 test-rmse:0.705980
## [15] train-rmse:0.550699 test-rmse:0.720267
## [16] train-rmse:0.475856 test-rmse:0.719716
## [17] train-rmse:0.413701 test-rmse:0.714229
## [18] train-rmse:0.362479 test-rmse:0.729419
## [19] train-rmse:0.320795 test-rmse:0.723379
## [20] train-rmse:0.280176 test-rmse:0.736308
## [21] train-rmse:0.248736 test-rmse:0.736752
## [22] train-rmse:0.224459 test-rmse:0.740297
## [23] train-rmse:0.199794 test-rmse:0.751140
## [24] train-rmse:0.178695 test-rmse:0.773267
## [25] train-rmse:0.159814 test-rmse:0.775373
## [26] train-rmse:0.140390 test-rmse:0.793187
## [27] train-rmse:0.124773 test-rmse:0.806212
## [28] train-rmse:0.112396 test-rmse:0.812152
## [29] train-rmse:0.100422 test-rmse:0.823292
## [30] train-rmse:0.085693 test-rmse:0.827910
## [31] train-rmse:0.074425 test-rmse:0.831044
## [32] train-rmse:0.069612 test-rmse:0.829186
## [33] train-rmse:0.061815 test-rmse:0.830828
## [34] train-rmse:0.054888 test-rmse:0.830283
```

```
## [35] train-rmse:0.044927 test-rmse:0.829337
## [36] train-rmse:0.037754 test-rmse:0.827525
## [37] train-rmse:0.033995 test-rmse:0.826099
## [38] train-rmse:0.027800 test-rmse:0.826097
## [39] train-rmse:0.025128 test-rmse:0.823633
## [40] train-rmse:0.021566 test-rmse:0.825214
## [41] train-rmse:0.019278 test-rmse:0.825974
## [42] train-rmse:0.016346 test-rmse:0.826906
## [43] train-rmse:0.014241 test-rmse:0.827798
## [44] train-rmse:0.012579 test-rmse:0.827602
## [45] train-rmse:0.010724 test-rmse:0.827319
## [46] train-rmse:0.009265 test-rmse:0.825807
## [47] train-rmse:0.007897 test-rmse:0.825636
## [48] train-rmse:0.007088 test-rmse:0.825305
## [49] train-rmse:0.006257 test-rmse:0.825281
## [50] train-rmse:0.005463 test-rmse:0.825470
## [51] train-rmse:0.004776 test-rmse:0.825384
## [52] train-rmse:0.004159 test-rmse:0.825666
## [53] train-rmse:0.003795 test-rmse:0.825810
## [54] train-rmse:0.003397 test-rmse:0.825761
## [55] train-rmse:0.003107 test-rmse:0.825530
## [56] train-rmse:0.002710 test-rmse:0.825682
## [57] train-rmse:0.002434 test-rmse:0.825919
## [58] train-rmse:0.002196 test-rmse:0.825908
## [59] train-rmse:0.001994 test-rmse:0.825953
## [60] train-rmse:0.001813 test-rmse:0.826013
## [61] train-rmse:0.001633 test-rmse:0.825864
## [62] train-rmse:0.001436 test-rmse:0.825625
## [63] train-rmse:0.001226 test-rmse:0.825853
## [64] train-rmse:0.001136 test-rmse:0.825832
## [65] train-rmse:0.000982 test-rmse:0.825749
## [66] train-rmse:0.000850 test-rmse:0.825914
## [67] train-rmse:0.000769 test-rmse:0.825961
## [68] train-rmse:0.000662 test-rmse:0.825972
## [69] train-rmse:0.000588 test-rmse:0.825928
## [70] train-rmse:0.000534 test-rmse:0.825980
```

```
#Avoid overfitting by stopping when rmse starts to increase
xgb_final_model = xgb.train(data = xgb_train, max.depth = 3,
                  watchlist=watchlist, nrounds = 17)
```

```
## [1]  train-rmse:16.351229    test-rmse:16.484171
## [2]  train-rmse:11.811347    test-rmse:12.128655
## [3]  train-rmse:8.581002 test-rmse:8.840908
## [4]  train-rmse:6.285137 test-rmse:6.465342
## [5]  train-rmse:4.673629 test-rmse:4.822035
## [6]  train-rmse:3.511651 test-rmse:3.435052
## [7]  train-rmse:2.675627 test-rmse:2.498768
## [8]  train-rmse:2.069472 test-rmse:1.834512
## [9]  train-rmse:1.636494 test-rmse:1.290090
## [10] train-rmse:1.314316 test-rmse:0.962761
## [11] train-rmse:1.081714 test-rmse:0.778236
## [12] train-rmse:0.899081 test-rmse:0.742409
## [13] train-rmse:0.757631 test-rmse:0.729652
## [14] train-rmse:0.641829 test-rmse:0.705980
```

```
## [15] train-rmse:0.550699 test-rmse:0.720267
## [16] train-rmse:0.475856 test-rmse:0.719716
## [17] train-rmse:0.413701 test-rmse:0.714229
```

```
#Predictions
y_pred <- predict(xgb_model, as.matrix(test_x))
#MSE
mean((unlist(test_y) - y_pred)^2)
```

```
## [1] 0.6822436
```

```
#MAE
caret::MAE(unlist(test_y), y_pred)
```

```
## [1] 0.6715505
```

```
#RMSE
caret::RMSE(unlist(test_y), y_pred)
```

```
## [1] 0.8259804
```

```
write.csv(y_pred, "xgb_pred.csv")

#Feature Importance
importance <- xgb.importance(feature_names = colnames(train_x), model = xgb_model)
print(importance)
```

```
##                                                 Feature         Gain
##                                                  <char>        <num>
##  1:                                     total.attempts 9.678584e-01
##  2:                                total.attempts.team2 2.095350e-02
##  3:                    inbehind.offers.to.receive.team2 2.783976e-03
##  4:                attempted.defensive.line.breaks.team2 1.660710e-03
##  5:                                possession.in.contest 1.311086e-03
##  6:                           completed.line.breaks.team1 5.544672e-04
##  7:                                     crosses.team1 4.089936e-04
##  8:                                  free.kicks.team2 3.926047e-04
##  9:                             on.target.attempts.team2 3.499617e-04
## 10:                    defensive.pressures.applied.team1 3.262044e-04
## 11:                                  possession.team1 3.066205e-04
## 12:                     switches.of.play.completed.team2 3.006619e-04
## 13:                           left.inside.channel.team1 2.711760e-04
## 14: receptions.between.midfield.and.defensive.lines.team2 2.294133e-04
## 15:                      infront.offers.to.receive.team1 2.112531e-04
## 16:                           right.inside.channel.team2 1.771938e-04
## 17:                              goal.preventions.team2 1.588886e-04
## 18: receptions.between.midfield.and.defensive.lines.team1 1.568871e-04
## 19:                    inbetween.offers.to.receive.team2 1.547011e-04
## 20:                           completed.line.breaks.team2 1.423262e-04
## 21:             attempts.outside.the.penalty.area..team2 1.321056e-04
## 22:                                  yellow.cards.team1 1.154738e-04
## 23:                            crosses.completed.team1 9.661294e-05
## 24:                                 left.channel.team1 8.683481e-05
## 25:               attempts.inside.the.penalty.area.team1 6.979150e-05
## 26:                           off.target.attempts.team2 6.815463e-05
## 27:                          attempted.line.breaks.team1 6.672005e-05
## 28:                                 total.attempts.team1 6.311372e-05
```

```
## 29:                                  fouls.against.team2 6.145737e-05
## 30:                attempts.inside.the.penalty.area.team2 5.697983e-05
## 31:                                forced.turnovers.team1 5.589613e-05
## 32:                                        corners.team2 4.857006e-05
## 33:                           total.offers.to.receive.team2 4.319852e-05
## 34:                                   central.channel.team1 4.281584e-05
## 35:                             attempted.line.breaks.team2 4.063803e-05
## 36:                                   central.channel.team2 3.692065e-05
## 37:                                     fouls.against.team1 3.575754e-05
## 38:                              right.inside.channel.team1 3.490957e-05
## 39:                       switches.of.play.completed.team1 1.904818e-05
## 40:                                forced.turnovers.team2 1.800367e-05
## 41:                                        corners.team1 1.548746e-05
## 42:                                passes.completed.team2 1.305868e-05
## 43:                     defensive.pressures.applied.team2 9.648503e-06
## 44:               attempts.outside.the.penalty.area..team1 8.587246e-06
## 45:                                         passes.team1 8.011818e-06
## 46:                               on.target.attempts.team1 7.509195e-06
## 47:                  attempted.defensive.line.breaks.team1 6.315991e-06
## 48:                  completed.defensive.line.breaks.team1 4.508257e-06
## 49:                               crosses.completed.team2 4.159283e-06
## 50:                                     possession.team2 3.811083e-06
## 51:                  total.attempted.defensive.line.breaks 3.627776e-06
## 52:                                   right.channel.team2 3.544055e-06
## 53:                        inbehind.offers.to.receive.team1 3.256426e-06
## 54:                              off.target.attempts.team1 2.628080e-06
## 55:                         infront.offers.to.receive.team2 9.526783e-07
## 56:                                      free.kicks.team1 8.823403e-07
## 57:                            left.inside.channel.team2 6.079539e-07
## 58:                                   right.channel.team1 5.209141e-07
## 59:                                         passes.team2 4.993013e-07
## 60:                      inbetween.offers.to.receive.team1 3.001950e-07
## 61:                                    left.channel.team2 6.874079e-08
## 62:                         total.offers.to.receive.team1 2.885974e-08
##                                                   Feature         Gain
##            Cover   Frequency
##            <num>       <num>
##  1: 0.1447316103 0.094827586
##  2: 0.0383697813 0.037356322
##  3: 0.0572564612 0.040229885
##  4: 0.0326043738 0.022988506
##  5: 0.0341948310 0.054597701
##  6: 0.0171968191 0.017241379
##  7: 0.0090457256 0.008620690
##  8: 0.0130218688 0.008620690
##  9: 0.0156063618 0.014367816
## 10: 0.0107355865 0.008620690
## 11: 0.0310139165 0.066091954
## 12: 0.0211729622 0.017241379
## 13: 0.0290258449 0.037356322
## 14: 0.0046719682 0.008620690
## 15: 0.0101391650 0.008620690
## 16: 0.0190854871 0.011494253
## 17: 0.0086481113 0.005747126
```

```
## 18: 0.0099403579 0.011494253
## 19: 0.0048707753 0.008620690
## 20: 0.0139165010 0.008620690
## 21: 0.0195825050 0.020114943
## 22: 0.0140159046 0.011494253
## 23: 0.0078528827 0.011494253
## 24: 0.0108349901 0.011494253
## 25: 0.0107355865 0.020114943
## 26: 0.0038767396 0.008620690
## 27: 0.0334990060 0.025862069
## 28: 0.0075546720 0.017241379
## 29: 0.0260437376 0.017241379
## 30: 0.0055666004 0.008620690
## 31: 0.0218687873 0.031609195
## 32: 0.0219681909 0.017241379
## 33: 0.0117296223 0.008620690
## 34: 0.0097415507 0.014367816
## 35: 0.0148111332 0.011494253
## 36: 0.0147117296 0.014367816
## 37: 0.0197813121 0.020114943
## 38: 0.0110337972 0.017241379
## 39: 0.0204771372 0.022988506
## 40: 0.0009940358 0.002873563
## 41: 0.0071570577 0.005747126
## 42: 0.0031809145 0.002873563
## 43: 0.0061630219 0.005747126
## 44: 0.0058648111 0.011494253
## 45: 0.0048707753 0.002873563
## 46: 0.0170974155 0.022988506
## 47: 0.0073558648 0.008620690
## 48: 0.0106361829 0.008620690
## 49: 0.0299204771 0.022988506
## 50: 0.0016898608 0.005747126
## 51: 0.0013916501 0.002873563
## 52: 0.0068588469 0.005747126
## 53: 0.0020874751 0.005747126
## 54: 0.0108349901 0.017241379
## 55: 0.0226640159 0.017241379
## 56: 0.0101391650 0.008620690
## 57: 0.0030815109 0.002873563
## 58: 0.0050695825 0.005747126
## 59: 0.0070576541 0.005747126
## 60: 0.0189860835 0.014367816
## 61: 0.0030815109 0.005747126
## 62: 0.0028827038 0.005747126
##            Cover   Frequency
```

```
write.csv(importance, "xgb_imp.csv")
```