stuff-plus

Bryan Mui - UID 506021334

2025-03-05

Library packages

Read the dataset

```
data_original <- read_csv("./data/UCLA2023-2024.csv")
```

```
## Rows: 31775 Columns: 198
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (40): Date, Pitcher, PitcherThrows, PitcherTeam, Batter, BatterSide, B...
## dbl  (148): PitchNo, PAofInning, PitchofPA, PitcherId, BatterId, Inning, Out...
## lgl    (4): MeasuredDuration, PitchLastMeasuredX, PitchLastMeasuredY, PitchL...
## dttm   (2): LocalDateTime, UTCDateTime
## time   (4): Time, Tilt, UTCTime, SpinAxis3dTilt
##
## 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.
```

```
# The data set that we will be mutating
data <- data_original
head(data, 25)
```

```
## # A tibble: 25 x 198
##    PitchNo Date     Time   PAofInning PitchofPA Pitcher  PitcherId PitcherThrows
##      <dbl> <chr>    <time>      <dbl>     <dbl> <chr>        <dbl> <chr>
##  1     264 3/1/2023 24:32           6         1 Harajli~    1.00e9 Right
##  2     289 3/1/2023 36:56           3         1 Harajli~    1.00e9 Right
##  3     265 3/1/2023 25:21           7         1 Harajli~    1.00e9 Right
##  4     288 3/1/2023 36:17           2         3 Harajli~    1.00e9 Right
##  5     293 3/1/2023 38:09           3         5 Harajli~    1.00e9 Right
##  6     336 3/1/2023 06:18           3         5 Harajli~    1.00e9 Right
##  7     342 3/1/2023 08:52           4         1 Harajli~    1.00e9 Right
##  8     339 3/1/2023 07:28           3         8 Harajli~    1.00e9 Right
##  9     281 3/1/2023 33:55           1         2 Harajli~    1.00e9 Right
## 10     262 3/1/2023 23:35           5         4 Harajli~    1.00e9 Right
## # i 15 more rows
## # i 190 more variables: PitcherTeam <chr>, Batter <chr>, BatterId <dbl>,
## #   BatterSide <chr>, BatterTeam <chr>, PitcherSet <chr>, Inning <dbl>,
```

```
## #   Top_Bottom <chr>, Outs <dbl>, Balls <dbl>, Strikes <dbl>,
## #   TaggedPitchType <chr>, AutoPitchType <chr>, PitchCall <chr>, KorBB <chr>,
## #   TaggedHitType <chr>, PlayResult <chr>, OutsOnPlay <dbl>, RunsScored <dbl>,
## #   Notes <chr>, RelSpeed <dbl>, VertRelAngle <dbl>, HorzRelAngle <dbl>, ...
```

Outputting the columns

```
colnames(data)
```

```
##   [1] "PitchNo"
##   [2] "Date"
##   [3] "Time"
##   [4] "PAofInning"
##   [5] "PitchofPA"
##   [6] "Pitcher"
##   [7] "PitcherId"
##   [8] "PitcherThrows"
##   [9] "PitcherTeam"
##  [10] "Batter"
##  [11] "BatterId"
##  [12] "BatterSide"
##  [13] "BatterTeam"
##  [14] "PitcherSet"
##  [15] "Inning"
##  [16] "Top_Bottom"
##  [17] "Outs"
##  [18] "Balls"
##  [19] "Strikes"
##  [20] "TaggedPitchType"
##  [21] "AutoPitchType"
##  [22] "PitchCall"
##  [23] "KorBB"
##  [24] "TaggedHitType"
##  [25] "PlayResult"
##  [26] "OutsOnPlay"
##  [27] "RunsScored"
##  [28] "Notes"
##  [29] "RelSpeed"
##  [30] "VertRelAngle"
##  [31] "HorzRelAngle"
##  [32] "SpinRate"
##  [33] "SpinAxis"
##  [34] "Tilt"
##  [35] "RelHeight"
##  [36] "RelSide"
##  [37] "Extension"
##  [38] "VertBreak"
##  [39] "InducedVertBreak"
##  [40] "HorzBreak"
##  [41] "PlateLocHeight"
##  [42] "PlateLocSide"
```

```
##  [43] "ZoneSpeed"
##  [44] "VertApprAngle"
##  [45] "HorzApprAngle"
##  [46] "ZoneTime"
##  [47] "ExitSpeed"
##  [48] "Angle"
##  [49] "Direction"
##  [50] "HitSpinRate"
##  [51] "PositionAt110X"
##  [52] "PositionAt110Y"
##  [53] "PositionAt110Z"
##  [54] "Distance"
##  [55] "LastTrackedDistance"
##  [56] "Bearing"
##  [57] "HangTime"
##  [58] "pfxx"
##  [59] "pfxz"
##  [60] "x0"
##  [61] "y0"
##  [62] "z0"
##  [63] "vx0"
##  [64] "vy0"
##  [65] "vz0"
##  [66] "ax0"
##  [67] "ay0"
##  [68] "az0"
##  [69] "HomeTeam"
##  [70] "AwayTeam"
##  [71] "Stadium"
##  [72] "Level"
##  [73] "League"
##  [74] "GameID"
##  [75] "PitchUID"
##  [76] "EffectiveVelo"
##  [77] "MaxHeight"
##  [78] "MeasuredDuration"
##  [79] "SpeedDrop"
##  [80] "PitchLastMeasuredX"
##  [81] "PitchLastMeasuredY"
##  [82] "PitchLastMeasuredZ"
##  [83] "ContactPositionX"
##  [84] "ContactPositionY"
##  [85] "ContactPositionZ"
##  [86] "GameUID"
##  [87] "UTCDate"
##  [88] "UTCTime"
##  [89] "LocalDateTime"
##  [90] "UTCDateTime"
##  [91] "AutoHitType"
##  [92] "System"
##  [93] "HomeTeamForeignID"
##  [94] "AwayTeamForeignID"
##  [95] "GameForeignID"
##  [96] "Catcher"
```

```
##  [97] "CatcherId"
##  [98] "CatcherThrows"
##  [99] "CatcherTeam"
## [100] "PlayID"
## [101] "PitchTrajectoryXc0"
## [102] "PitchTrajectoryXc1"
## [103] "PitchTrajectoryXc2"
## [104] "PitchTrajectoryYc0"
## [105] "PitchTrajectoryYc1"
## [106] "PitchTrajectoryYc2"
## [107] "PitchTrajectoryZc0"
## [108] "PitchTrajectoryZc1"
## [109] "PitchTrajectoryZc2"
## [110] "HitSpinAxis"
## [111] "HitTrajectoryXc0"
## [112] "HitTrajectoryXc1"
## [113] "HitTrajectoryXc2"
## [114] "HitTrajectoryXc3"
## [115] "HitTrajectoryXc4"
## [116] "HitTrajectoryXc5"
## [117] "HitTrajectoryXc6"
## [118] "HitTrajectoryXc7"
## [119] "HitTrajectoryXc8"
## [120] "HitTrajectoryYc0"
## [121] "HitTrajectoryYc1"
## [122] "HitTrajectoryYc2"
## [123] "HitTrajectoryYc3"
## [124] "HitTrajectoryYc4"
## [125] "HitTrajectoryYc5"
## [126] "HitTrajectoryYc6"
## [127] "HitTrajectoryYc7"
## [128] "HitTrajectoryYc8"
## [129] "HitTrajectoryZc0"
## [130] "HitTrajectoryZc1"
## [131] "HitTrajectoryZc2"
## [132] "HitTrajectoryZc3"
## [133] "HitTrajectoryZc4"
## [134] "HitTrajectoryZc5"
## [135] "HitTrajectoryZc6"
## [136] "HitTrajectoryZc7"
## [137] "HitTrajectoryZc8"
## [138] "ThrowSpeed"
## [139] "PopTime"
## [140] "ExchangeTime"
## [141] "TimeToBase"
## [142] "CatchPositionX"
## [143] "CatchPositionY"
## [144] "CatchPositionZ"
## [145] "ThrowPositionX"
## [146] "ThrowPositionY"
## [147] "ThrowPositionZ"
## [148] "BasePositionX"
## [149] "BasePositionY"
## [150] "BasePositionZ"
```

```
## [151] "ThrowTrajectoryXc0"
## [152] "ThrowTrajectoryXc1"
## [153] "ThrowTrajectoryXc2"
## [154] "ThrowTrajectoryYc0"
## [155] "ThrowTrajectoryYc1"
## [156] "ThrowTrajectoryYc2"
## [157] "ThrowTrajectoryZc0"
## [158] "ThrowTrajectoryZc1"
## [159] "ThrowTrajectoryZc2"
## [160] "PitchReleaseConfidence"
## [161] "PitchLocationConfidence"
## [162] "PitchMovementConfidence"
## [163] "HitLaunchConfidence"
## [164] "HitLandingConfidence"
## [165] "CatcherThrowCatchConfidence"
## [166] "CatcherThrowReleaseConfidence"
## [167] "CatcherThrowLocationConfidence"
## [168] "SpinAxis3dTransverseAngle"
## [169] "SpinAxis3dLongitudinalAngle"
## [170] "SpinAxis3dActiveSpinRate"
## [171] "SpinAxis3dSpinEfficiency"
## [172] "SpinAxis3dTilt"
## [173] "SpinAxis3dVectorX"
## [174] "SpinAxis3dVectorY"
## [175] "SpinAxis3dVectorZ"
## [176] "SpinAxis3dSeamOrientationRotationX"
## [177] "SpinAxis3dSeamOrientationRotationY"
## [178] "SpinAxis3dSeamOrientationRotationZ"
## [179] "SpinAxis3dSeamOrientationBallAngleHorizontalAmb1"
## [180] "SpinAxis3dSeamOrientationBallAngleVerticalAmb1"
## [181] "SpinAxis3dSeamOrientationBallXAmb1"
## [182] "SpinAxis3dSeamOrientationBallYAmb1"
## [183] "SpinAxis3dSeamOrientationBallZAmb1"
## [184] "SpinAxis3dSeamOrientationBallAngleHorizontalAmb2"
## [185] "SpinAxis3dSeamOrientationBallAngleVerticalAmb2"
## [186] "SpinAxis3dSeamOrientationBallXAmb2"
## [187] "SpinAxis3dSeamOrientationBallYAmb2"
## [188] "SpinAxis3dSeamOrientationBallZAmb2"
## [189] "SpinAxis3dSeamOrientationBallAngleHorizontalAmb3"
## [190] "SpinAxis3dSeamOrientationBallAngleVerticalAmb3"
## [191] "SpinAxis3dSeamOrientationBallXAmb3"
## [192] "SpinAxis3dSeamOrientationBallYAmb3"
## [193] "SpinAxis3dSeamOrientationBallZAmb3"
## [194] "SpinAxis3dSeamOrientationBallAngleHorizontalAmb4"
## [195] "SpinAxis3dSeamOrientationBallAngleVerticalAmb4"
## [196] "SpinAxis3dSeamOrientationBallXAmb4"
## [197] "SpinAxis3dSeamOrientationBallYAmb4"
## [198] "SpinAxis3dSeamOrientationBallZAmb4"
```

Predictor Ideas:

Velocity:

```
* RelSpeed (Release Speed)
```

* ZoneSpeed (Speed at the plate)
* EffectiveVelo (Velocity adjusted for approach angle)

Movement:

* VertBreak (Vertical movement due to spin)
* InducedVertBreak (More refined vertical movement measurement)
* HorzBreak (Horizontal movement due to spin)
* pfxx (Horizontal movement component)
* pfxz (Vertical movement component)

Spin:

* SpinRate (Total revolutions per minute)
* SpinAxis (2D spin direction)
* SpinAxis3dTransverseAngle (3D spin components)
* SpinAxis3dLongitudinalAngle
* SpinAxis3dActiveSpinRate
* SpinAxis3dSpinEfficiency

Release & Extension:

* RelHeight (Height of release)
* RelSide (Side angle of release)
* Extension (How far forward the pitcher releases the ball)

Pitch Type & Classification:

* TaggedPitchType (Human-classified pitch type)
* AutoPitchType (Algorithm-classified pitch type)

Location & Trajectory (Optional, but can improve Stuff+ models):

* PlateLocHeight (Height of the pitch as it crosses the plate)
* PlateLocSide (Side location at home plate)
* VertApprAngle (Vertical approach angle)
* HorzApprAngle (Horizontal approach angle)

For now, focusing on these variables:

* Pitch Velocity
* Vertical Break
* Horizontal Break
* Arm Angle
* Release Extension

Stuff+

Part 1: Exploring Pitch Types and Sectioning Data Based off Pitches

```r
data %>%
  group_by(TaggedPitchType) %>%
  summarize(Count = n())
```

```
## # A tibble: 12 x 2
##    TaggedPitchType  Count
##    <chr>            <int>
##  1 ChangeUp          4247
##  2 Curveball         2790
##  3 Cutter             421
##  4 Fastball         14056
##  5 FourSeamFastBall    15
##  6 OneSeamFastBall      1
##  7 Other              108
##  8 Sinker            3588
##  9 Slider            6444
## 10 Splitter            71
## 11 TwoSeamFastBall     31
## 12 Undefined            3
```

We can see that we have ample data to produce a model for 1) Fastball, 2) Curve Ball, 3) Change Up, 4) Slider, 5) Sinker. The rest of the pitches have limited observations

```r
# Section the Data based off pitch type(Run After we've transformed variables)
# data_fastball <- data %>%
#   filter(TaggedPitchType == "Fastball")
# data_curveball <- data %>%
#   filter(TaggedPitchType == "Curveball")
# data_changeup <- data %>%
#   filter(TaggedPitchType == "ChangeUp")
# data_slider <- data %>%
#   filter(TaggedPitchType == "Slider")
# data_sinker <- data %>%
#   filter(TaggedPitchType == "Sinker")
```

Part 2: Target Variable(set hit = 0/not a hit = 1)

```r
# Calculate hit/no-hit on pitch
data <- data %>%
  mutate(hit_response = ifelse(PlayResult != "Undefined", 1, 0)) %>%
  relocate(hit_response, .after = PitchCall)
```

Part 3: Calculating Stuff+

```r
# Select the variables we need, refer the beginning to see which variables are being selected
vars <- c(
    "Pitcher",
    "PitcherId",
```

```
    "TaggedPitchType",
    "RelSpeed",
    "ZoneSpeed",
    "EffectiveVelo",
    "VertBreak",
    "InducedVertBreak",
    "HorzBreak",
    "SpinRate",
    "SpinAxis",
    "Tilt",
    "RelHeight",
    "RelSide",
    "Extension",
    "VertApprAngle",
    "HorzApprAngle",
    "hit_response"
)

data <- data %>%
  select(all_of(vars))

# Now create separate datasets for all the pitch types
# Section the Data based off pitch type(Run After we've transformed variables)
data_fastball <- data %>%
  filter(TaggedPitchType == "Fastball")
data_curveball <- data %>%
  filter(TaggedPitchType == "Curveball")
data_changeup <- data %>%
  filter(TaggedPitchType == "ChangeUp")
data_slider <- data %>%
  filter(TaggedPitchType == "Slider")
data_sinker <- data %>%
  filter(TaggedPitchType == "Sinker")
```

Misc Code Chunks:

```
# Calculate team-wide SD and mean for metrics
# means <- stuff_data %>%
#   group_by("TaggedPitchType") %>%
#   summarise_all(funs(mean), na.rm = TRUE)
# means
#
# sds <- stuff_data %>%
#   group_by("TaggedPitchType") %>%
#   summarise_all(funs(sd), na.rm = TRUE)
# means
```

```
# stuff_data_standardized <- stuff_data %>%
#   group_by("TaggedPitchType") %>%
#   mutate_at(vars(RelSpeed:swstr_percent), scale) %>%
#   drop_na()
#
# stuff_data_standardized
```

Part 3a: Calculating Coefficients Using LM

Fastball Model

```
model_vars <- c(
    "RelSpeed",
    "ZoneSpeed",
    "EffectiveVelo",
    "VertBreak",
    "InducedVertBreak",
    "HorzBreak",
    "SpinRate",
    "SpinAxis",
    "Tilt",
    "RelHeight",
    "RelSide",
    "Extension",
    "VertApprAngle",
    "HorzApprAngle"
)

# get the LM equation formatted
equation <- paste("hit_response ~ ", paste(model_vars, collapse = " + "))
print(equation)
```

```
## [1] "hit_response ~  RelSpeed + ZoneSpeed + EffectiveVelo + VertBreak + InducedVertBreak + HorzBreak
```

```
# train
lm_fb <- lm(formula(equation), data = data_fastball)

# summarize
summary(lm_fb)
```

```
##
## Call:
## lm(formula = formula(equation), data = data_fastball)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78786 -0.23046 -0.17421 -0.06673  1.00530
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.007e+01  2.590e+00   7.748 1.00e-14 ***
## RelSpeed         -4.245e-01  2.643e-02 -16.060  < 2e-16 ***
## ZoneSpeed        -5.291e-01  3.067e-02 -17.252  < 2e-16 ***
## EffectiveVelo     8.365e-01  5.254e-02  15.921  < 2e-16 ***
## VertBreak         1.118e-01  2.265e-02   4.936 8.09e-07 ***
## InducedVertBreak -1.221e-01  2.270e-02  -5.379 7.62e-08 ***
## HorzBreak         6.196e-03  1.941e-03   3.193 0.001414 **
## SpinRate          2.015e-05  2.217e-05   0.909 0.363539
## SpinAxis         -1.996e-03  6.624e-04  -3.013 0.002593 **
## Tilt             -1.568e-07  2.609e-07  -0.601 0.547900
```

```
## RelHeight         3.572e-02  9.776e-03   3.654 0.000259 ***
## RelSide          -6.051e-03  6.327e-03  -0.956 0.338905
## Extension        -1.545e+00  9.028e-02 -17.112  < 2e-16 ***
## VertApprAngle    -1.173e-01  6.753e-03 -17.371  < 2e-16 ***
## HorzApprAngle    -1.249e-02  3.915e-03  -3.190 0.001424 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3912 on 13796 degrees of freedom
##   (245 observations deleted due to missingness)
## Multiple R-squared:  0.03819,    Adjusted R-squared:  0.03722
## F-statistic: 39.13 on 14 and 13796 DF,  p-value: < 2.2e-16
```

```
# standardize stuff to 100
stuff_fb <- data_fastball %>%
  mutate(
    raw_stuff = predict(lm_fb, newdata = .),
    StuffPlus = 100 * raw_stuff / mean(raw_stuff, na.rm = TRUE)
  )
```

Ranking Top 50 Pitches Given by Stuff Plus

```
top_50 <- stuff_fb %>%
  top_n(50, StuffPlus)

head(top_50, 50)
```

```
## # A tibble: 50 x 20
##    Pitcher  PitcherId TaggedPitchType RelSpeed ZoneSpeed EffectiveVelo VertBreak
##    <chr>        <dbl> <chr>              <dbl>     <dbl>         <dbl>     <dbl>
##  1 Chiment~    1.00e9 Fastball            86.5      78.6          84.0     -36.1
##  2 Chiment~    1.00e9 Fastball            85.5      77.6          83.0     -37.6
##  3 Chiment~    1.00e9 Fastball            86.2      78.4          83.8     -37.3
##  4 Chiment~    1.00e9 Fastball            86.3      78.5          84.0     -37.4
##  5 Chiment~    1.00e9 Fastball            86.4      79.1          84.2     -39.0
##  6 Grimm, ~    1.00e9 Fastball            80.6      73.3          77.2     -30.9
##  7 Taylor,~    1.00e9 Fastball            81.4      75.5          79.8     -36.9
##  8 Taylor,~    1.00e9 Fastball            86.9      78.7          84.8     -30.1
##  9 Shinn, ~    1.00e9 Fastball            86.5      79.0          82.1     -30.0
## 10 Shinn, ~    1.00e9 Fastball            87.9      79.9          83.6     -27.0
## # i 40 more rows
## # i 13 more variables: InducedVertBreak <dbl>, HorzBreak <dbl>, SpinRate <dbl>,
## #   SpinAxis <dbl>, Tilt <time>, RelHeight <dbl>, RelSide <dbl>,
## #   Extension <dbl>, VertApprAngle <dbl>, HorzApprAngle <dbl>,
## #   hit_response <dbl>, raw_stuff <dbl>, StuffPlus <dbl>
```

Misc Code Chunks

```
# xgb_fb <- xgboost(
#   data = X,
#   label = y,
#   objective = "binary:logistic",
#   nrounds = 100,
```

```
#   verbose = 0
# )
#
# importance <- xgb.importance(model = xgb_model)
# xgb.plot.importance(importance)
# model <- lm(swstr_percent ~ RelSpeed + ZoneSpeed + EffectiveVelo +
#             VertBreak + InducedVertBreak + HorzBreak +
#             pfxx + pfxz + SpinRate + SpinAxis +
#             RelHeight + RelSide + Extension +
#             PlateLocHeight + PlateLocSide +
#             VertApprAngle + HorzApprAngle,
#             data = stuff_data_standardized)
#
# summary(model)
# model$coefficients
```

```
# # Calculate Stuff+
# coefficients <- coef(model)
#
#
#
# stuff_plus_calc <- stuff_data_standardized %>%
#     rowwise() %>%
#     mutate(stuff_plus = coefficients["(Intercept)"] +
#                         coefficients["RelSpeed"] * RelSpeed +
#                         coefficients["ZoneSpeed"] * ZoneSpeed +
#                         coefficients["EffectiveVelo"] * EffectiveVelo +
#                         coefficients["VertBreak"] * VertBreak +
#                         coefficients["InducedVertBreak"] * InducedVertBreak +
#                         coefficients["HorzBreak"] * HorzBreak +
#                         coefficients["pfxx"] * pfxx +
#                         coefficients["pfxz"] * pfxz +
#                         coefficients["SpinRate"] * SpinRate +
#                         coefficients["SpinAxis"] * SpinAxis +
#                         coefficients["RelHeight"] * RelHeight +
#                         coefficients["RelSide"] * RelSide +
#                         coefficients["Extension"] * Extension +
#                         coefficients["PlateLocHeight"] * PlateLocHeight +
#                         coefficients["PlateLocSide"] * PlateLocSide +
#                         coefficients["VertApprAngle"] * VertApprAngle +
#                         coefficients["HorzApprAngle"] * HorzApprAngle)
#
# stuff_plus_calc
```

Misc

Stuff++

We could use a non-linear model to calculate Stuff+ but it would be a black-box model, meaning it gives us a score with no interpretable coefficients. The model might have better performance but low interpretability, hence we only know what the stuff is but we don't know what actually affects stuff