Predicting NFL Draft Placement Using NFL Combine Performance Metrics

David L. Parks
12 December 2021
Marquette University

Introduction

Every year college football players in the United States showcase their talent for professional scouts in a series of athletic events known as the National Football League (NFL) Combine. Those who perform better than their peers at the NFL Combine get drafted by the NFL, which is the equivalent of winning the lottery. There is virtually no other career in the world where someone can leave college after their freshman year without earning a college degree and potentially walk straight into a seven or eight figure income. However, there are substantial differences in compensation between those who are drafted in the first three rounds of the NFL Draft versus those who are drafted in subsequent rounds (Figure 1). For example, Trevor Lawrence, this year's overall number one draft pick (i.e. – 1st round, 1st selection) received an initial annual contract worth approximately $36.7 million USD, not including bonuses or additional incentives.[1] Compare that to Mike Strachan, the 1st pick of the 7th round, whose initial annual contract is worth approximately $3.5 million USD, slightly less than one-tenth of Lawrence's.[2] To say that getting drafted in the first few rounds has huge financial implications is not an understatement.

With so much at stake, understanding the differences in performance between drafted and undrafted players is crucial. This project explores the relationship between NFL Combine performance metrics and NFL Draft selection by identifying statistically significant performance metrics that players and coaches can use for targeted improvement. It also implements two separate classification models to determine the probability that a player will be drafted, and in which round that player will most likely be drafted. The deliverables from this project can be used for targeted improvement of potential NFL athletes and by those in the college football player evaluation industry to help determine the appropriate round to draft a player.

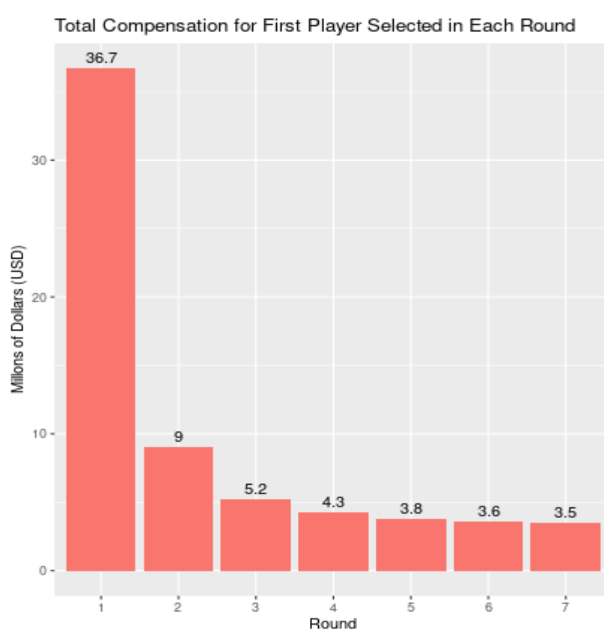


Figure 1: Total annual compensation for the first player selected in each round of the 2021 NFL Draft (in millions USD).

[1] https://overthecap.com/player/trevor-lawrence/9465/
[2] https://overthecap.com/player/mike-strachan/9693/

Business Intelligence Methods

Business Process

This project supports prospective NFL players; their trainers, coaches, and agents; and the NFL teams looking to acquire them in the following business applications: player development, contract negotiation, sponsorship, and longitudinal studies of player performance. The key deliverables are descriptive and comparative statistics for the NFL Combine performance metrics, a binomial predictive model that predicts whether a player will be drafted, and a multinomial predictive model that predicts which round a player will be drafted (Figure 2).

It is anticipated that the deliverables will improve the way that college football programs prepare their athletes for the NFL Combine and help NFL programs better evaluate prospective players. In terms of organizational readiness, teams should be willing to allocate resources to implement a relatively simple analysis that has huge financial implications for their organization. However, cultural roadblocks do exist, meaning that this project will need to consistently demonstrate value.

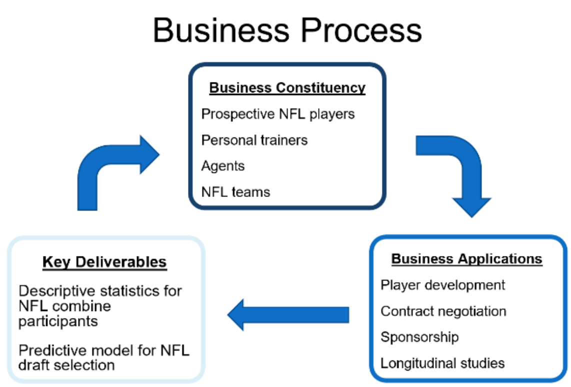


Figure 2

Data and Data Structures

The data for this project were collected from two separate publicly available sources. The 2021 NFL Combine data were gathered from NFLCombineResults.com[3] and the 2021 NFL Draft data were gathered from the 2021 NFL Draft Wikipedia page.[4] The 2021 NFL Combine data includes the following personal information:

- Name
- College
- Position
- Height
- Weight

and each participant's performance scores in the following athletic tests:

- 40-yard dash
- 225lb bench press

[3] https://nflcombineresults.com/nflcombinedata.php?year=2021&pos=&college=

[4] https://en.wikipedia.org/wiki/2021_NFL_Draft

- Vertical jump
- Broad jump
- Shuttle run
- 3-cone drill

The 2021 NFL Draft data includes the player's name, position, draft round, and selection number. The data were cleaned and formatted in Excel, and the two tables were joined using the player's name as the key column.

The data for 2021 presented unique challenges. Due to Covid-19, the NFL held separate "Pro-Days" at participating universities where prospective NFL players participated in events designed to simulate the NFL Combine. To avoid unnecessary human contact, most players only participated in the athletic tests deemed relevant to their position, resulting in a larger than usual number of missing entries (Table 1). The descriptive statistics were computed using all the available data; however, the missing entries had to be removed from the dataset to compute the comparative statistics and implement the classification models. Consequently, only 277 of the original 421 observations were available for modeling. The effects of this data loss are discussed in the next section.

Table 1: *The number of missing entries in the 2021 NFL Combine dataset.*

Metric	40-yard dash	Bench press	Vertical jump	Broad jump	Shuttle run	3-cone drill
Missing entries	63	95	54	0	74	81

Statistical Methodology

To compare the differences in NFL Combine performance metrics between drafted and undrafted players, distribution plots were created and Welch's Two Sample T-tests were conducted to compare the means for each of the athletic tests. A one-sided test was conducted for each performance metric using the appropriate direction for each respective metric. For the agility and speed drills, the alternative hypothesis is that the mean time for drafted players is less than that for undrafted players (i.e. – drafted players are faster). For the jumping drills, the alternative hypothesis is that the mean jump height/distance for drafted players is greater than that for undrafted players (i.e. – drafted players jump higher). For bench press, the alternative hypothesis is that the mean number of repetitions is greater for drafted players than for undrafted players (i.e. – drafted players are stronger). Lastly, for height and weight a two-sided test was conducted.

To help visualize the differences in performance metrics between positions for drafted and undrafted players, boxplots showing the median, 1st and 3rd interquartile range, and any outliers were created. Boxplots showing the differences in performance metrics by draft round were also generated. Tukey's Honestly Significant Differences (Tukey's HSD) tests were used to determine if there were statistically significant differences between performance metrics in those who were selected in different draft rounds. Tukey's HSD tests were also attempted using player position as the grouping variable. However, as noted in the previous section, the abundance of missing entries prevented the latter test from executing successfully. Although one-way Analysis of Variance (ANOVA) tests could have been completed prior to the Tukey HSD tests, it would have provided little value given that the goal of this project is to identify the differences between performance metrics by draft round, and not simply note that there are differences. All the statistical tests executed in this project used a preset significance level of $p = 0.05$.

Four separate classification models were also instantiated, two binomial classification models that predict whether a player will be drafted, and two multinomial classification models that predict which round a player will be drafted. For each model type, a full model using all the predictor variables was built as was a more parsimonious version of the full model. To create more parsimonious models, the number of predictor variables was reduced manually until all the predictors were statistically significant using a preset significance level of $p = 0.05$. All models were trained, tested, and evaluated using 5-fold cross-validation and compared using their respective Akaike information criterion (AIC) and accuracy scores.

Results

Distribution plots were created for each NFL Combine performance metric to help visualize the differences between drafted and undrafted players (Appendix A). As expected, the distributions for drafted players indicate better performance in all performance metrics as compared to undrafted players. These differences are supported by the results from the Welch's Two Sample T-Tests (Table 2). A significant result was observed for every test except

Table 2: *Results from the Welch's Two Sample T-Tests comparing drafted vs undrafted performance metrics.*

Metric	Height (in)	Weight (lbs)	40-yard dash (sec)	Bench press (reps)	Vertical jump (in)	Broad jump (in)	Shuttle run (sec)	3-cone drill (sec)
Drafted Mean	73.8	240.0	4.71	20.8	34.4	118.4	4.39	7.20
Undrafted Mean	73.5	236.8	4.83	18.9	32.2	115.0	4.51	7.33
P-value	0.2891	0.4701	0.0002*	0.0051*	< 0.0001*	< 0.0001*	< 0.0001*	0.0012*

** Denotes a statistically significant result.*

for height and weight. Interestingly, the distribution plot for weight showed a bimodal distribution that is almost indistinguishable between drafted and undrafted players (Figure 3). This particular distribution contradicts conventional wisdom in the strength and conditioning community that players need to "bulk up" before entering the draft, regardless of their current weight. For example, a player weighing close to 275lbs should probably consider gaining or losing weight to improve their chance of getting drafted.

Boxplots were created for each NFL Combine performance metric to show the differences in performance between positions differentiated for drafted and undrafted players

(Appendix B). Using the plot for 40-yard dash performance as a reference (Figure 5), one can see both the differences in performance between positions and within positions. For example, centers ("C") in the 2021 NFL Combine ran almost a full second slower than corner backs ("CB"), and between drafted and undrafted centers, undrafted centers ran 0.3 seconds slower than drafted centers. However, this plot also shows that for defensive tackles ("DT"), 40-yard dash performance is almost indistinguishable for drafted versus undrafted defensive tackles, meaning that once a defensive tackle is scoring within the range of drafted defensive tackles for the 40-yard dash, the player may be better off focusing on performance metrics that would distinguish them from other defensive tackles such as the bench press (Figure 6).

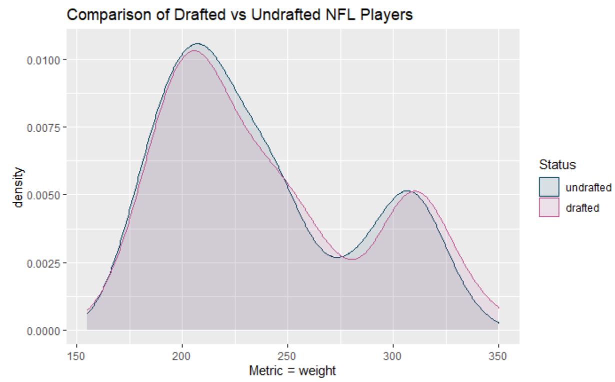


Figure 3: Distribution plot for weight between drafted and undrafted players.

For the Tukey HSD test, of the 224 possible comparisons between NFL Combine performance metrics and NFL draft round, only 4 comparisons had statistically significant results. However, with a p-value of 0.05, one should expect approximately 11 type I errors given the number of comparisons. As such, these results were discarded. To understand why there were so few significant results, boxplots for each performance metric by draft round were created (Appendix C). Using the plot for 40-yard dash as a reference, one can see that the variation between draft rounds is almost negligible, and despite there being some visual differences between round one and round two draftees, these differences were not statistically significant (Figure 4). As mentioned previously, a Tukey HSD test was attempted for each position, but the abundance of missing entries prevented this test from executing successfully.

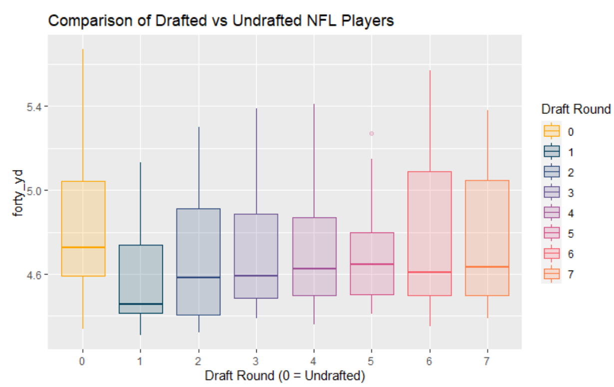


Figure 4: Boxplots for 40-yard dash performance grouped by draft round.

To predict the probability that a college football player will be drafted, a binomial classification model was fit using the 277 observations that remained after all "NA" values were removed from the dataset. All the performance metrics listed in table 2 plus a categorical column for position were provided as input to the model. A binary outcome of "undrafted" or "drafted" was used as the response. The full model was 71.5% accurate with a 312.72 AIC score. The parsimonious model, using only weight, 40-yard dash, shuttle run, and position as inputs was 70.0% accurate with a 306.7 AIC score. Since the parsimonious model has a lower AIC score, similar accuracy, and requires fewer inputs, this model should probably be used in production code. However, the coefficients for the full model are used for analysis in table 3.

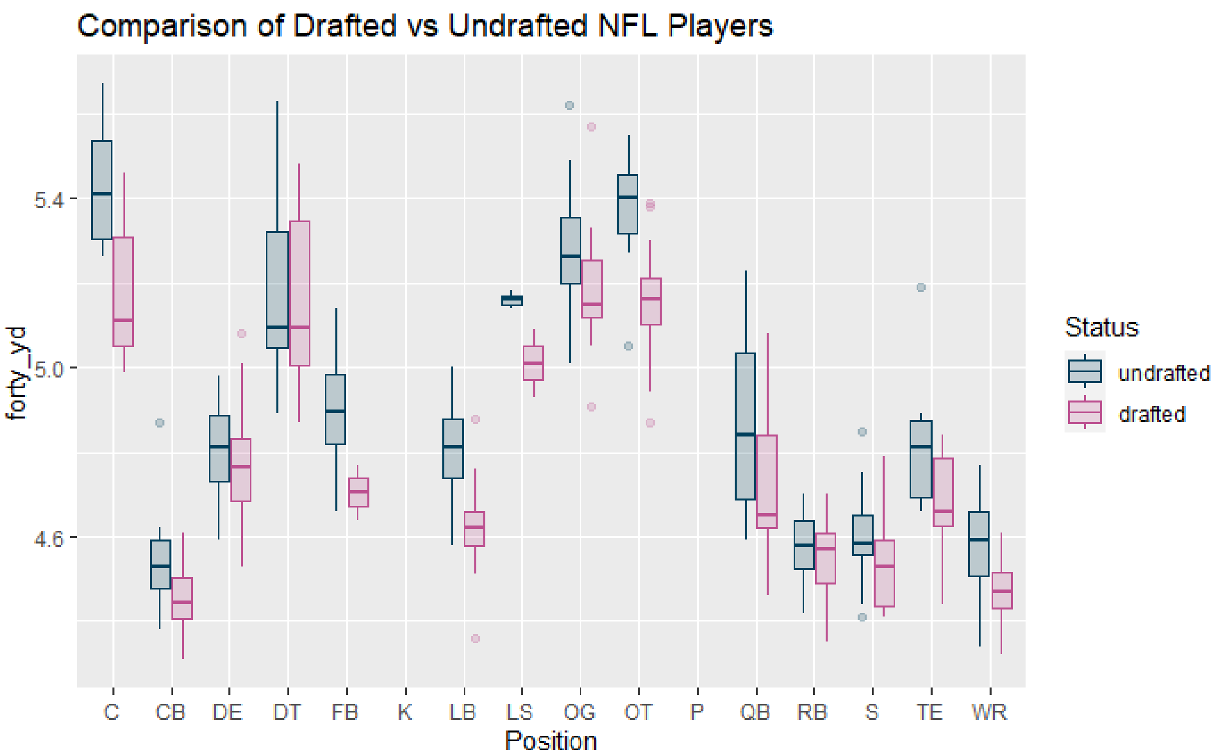


Figure 5: Comparison of 40-yard dash times for the NFL Combine displayed by position and by draft status.

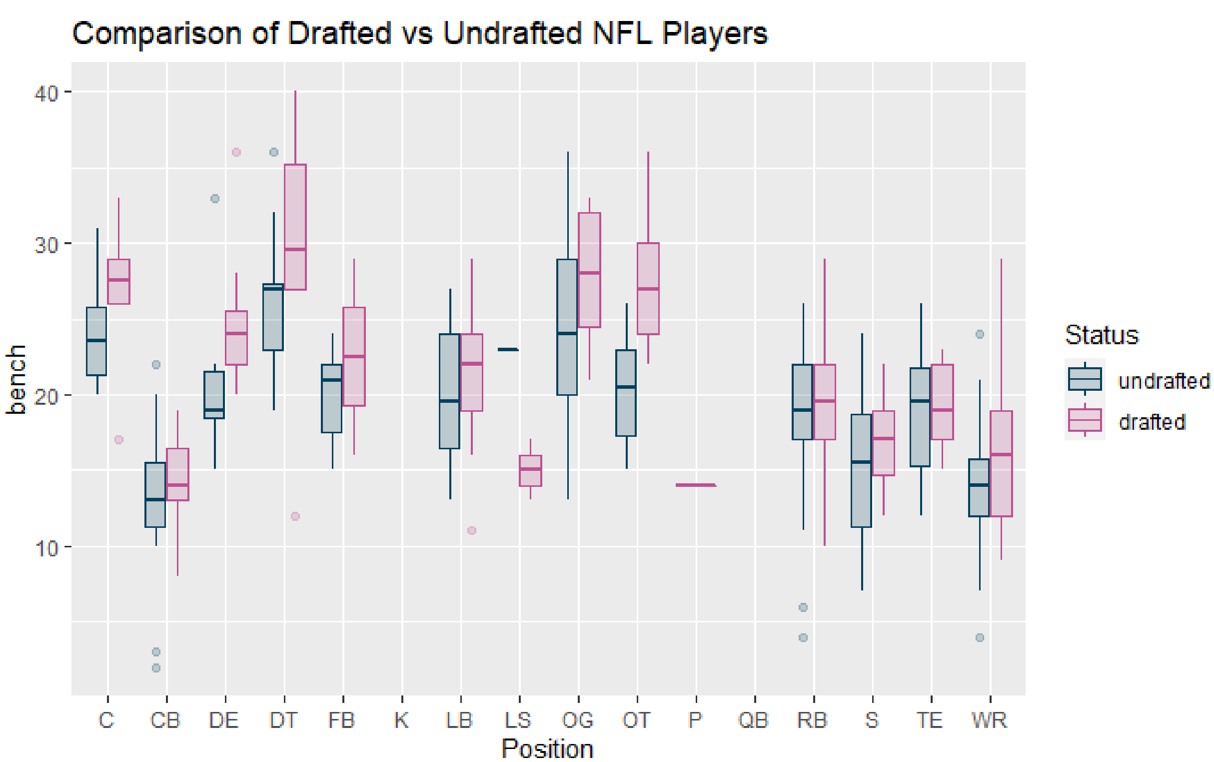


Figure 6: Comparison of bench press repetitions for the NFL Combine displayed by position and by draft status.

Exponentiating the coefficient for 40-yard dash shows that adding one-tenth of a second to a player's 40-yard dash time reduces their likelihood of getting drafted by almost 10%, whereas adding one inch to a player's vertical jump increases their likelihood of getting drafted by almost 11% (Table 3).

Table 3: *Analysis of the coefficients from the full binomial classification model.*

Coefficient for	Height	Weight	40-yard dash	Bench press	Vertical jump	Broad jump	Shuttle run	3-cone drill
Change by	+1 in	+1 lb	+1/10 sec	+1 rep	+1 in	+1 in	+1/10 sec	+1/10 sec
Effect on getting drafted	-2.6%	+5.5%	-10%	+1%	+11%	Negligible	-9.7%	-6.4%

Finally, a multinomial classification model was fit using the 277 observations that remained after all "NA" values were removed from the dataset. All the performance metrics listed in table 2 plus a categorical column for position were provided as input to the model. The outcomes were provided as factors where 1-7 represent the draft rounds and 0 represents undrafted. The full model was less than 50% accurate with a 1007.89 AIC score. The parsimonious model, using only weight, 40-yard dash, shuttle run, and position as inputs was also approximately 50% accurate with a 1006.14 AIC score. Because neither model was more than 50% accurate, no additional analysis was completed. Please see appendix D to view the full model summaries from R.

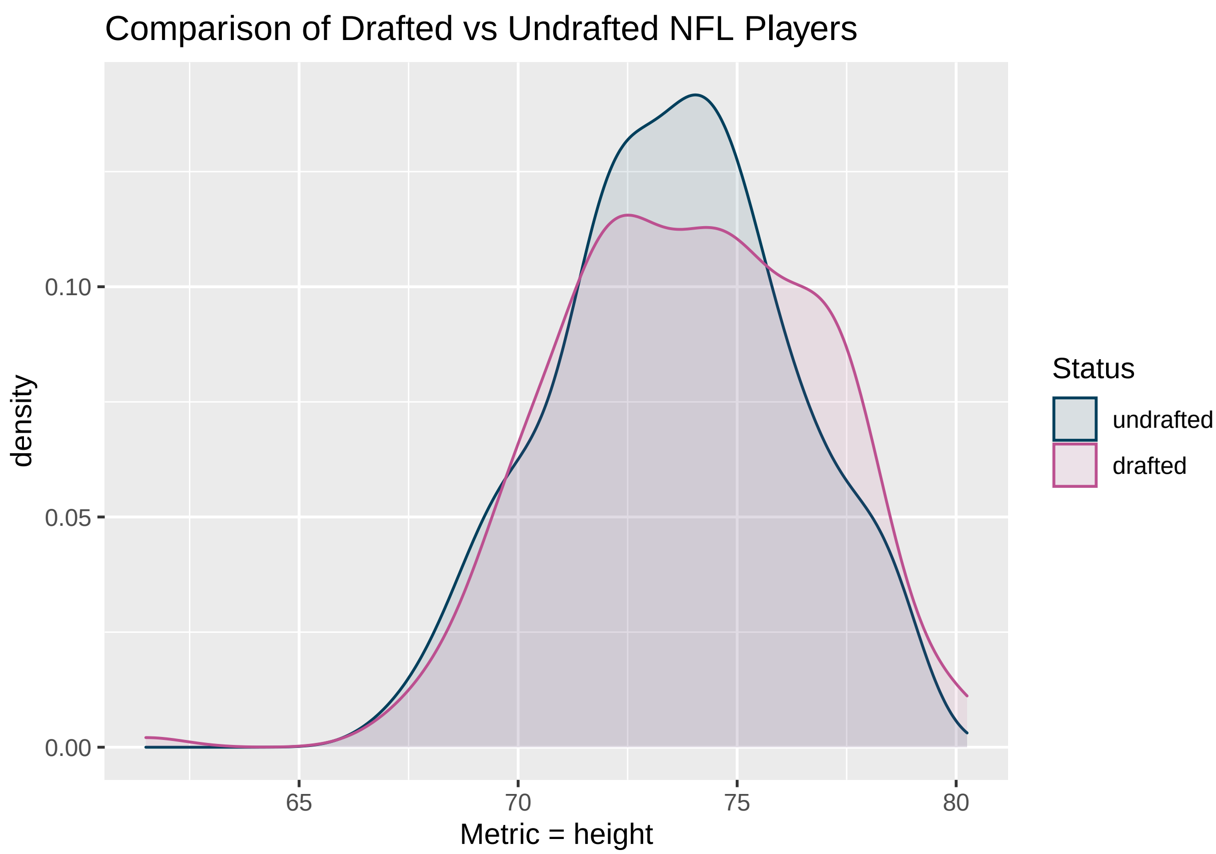
Conclusions

Overall, this project found that there are statistically significant differences between the NFL Combine performance metrics for players who get drafted by the NFL versus players who do not get drafted. It also found that these metrics can be used in modeling to predict if a player will be drafted with a ~70% accuracy rate. Strength and conditioning coaches can use these findings to create targeted training programs to improve the desired aspects of their players performance before they compete at the NFL Combine. Players can also use this information to make informed decisions about their performance and whether reclassifying to a different position might improve their chances of getting drafted.

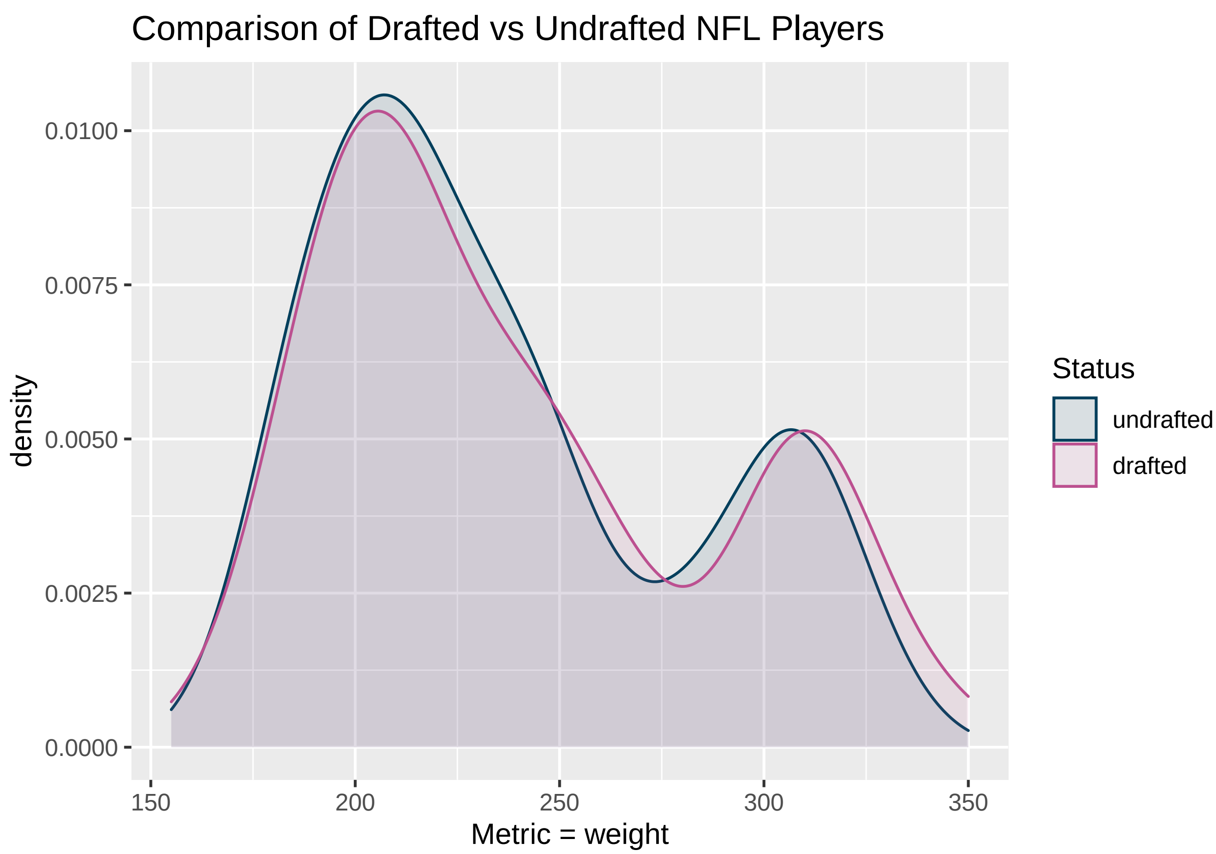
Unfortunately, neither multinomial classification model had an accuracy rate above 50% nor was there enough data to perform a Tukey HSD comparison between positions. It is possible that adding more data from previous NFL Combine events could alleviate these issues. However, data from the 2020 NFL Combine has issues similar to the dataset used for this project. Given the uncertainty surrounding Covid-19, it may be some time before three to five years' worth of "normal" data can be obtained without a detrimental number of missing values. Future projects may also want to explore the possibility of using multi-level models to reduce the influence of so many position categories in each model. Gradient boosting models may also provide a viable solution.

This project provides a tangible example of the application of data science to sports performance data. As the margins for improvement decrease and the expectations for sustained superior performance increase, projects like this will differentiate those who make it to the highest levels of professional sport and those who do not. This project also showed how important data collection is to the modeling process. Procedures used by the NFL to prevent the spread of Covid-19 reduced the usable data for modeling by almost half – a great example of unintended consequences. However, sports scientists must make use of the data they have, not the data they wished they had. Using completely free, publicly available data, this project provides a framework for improving athletic performance across multiple domains.

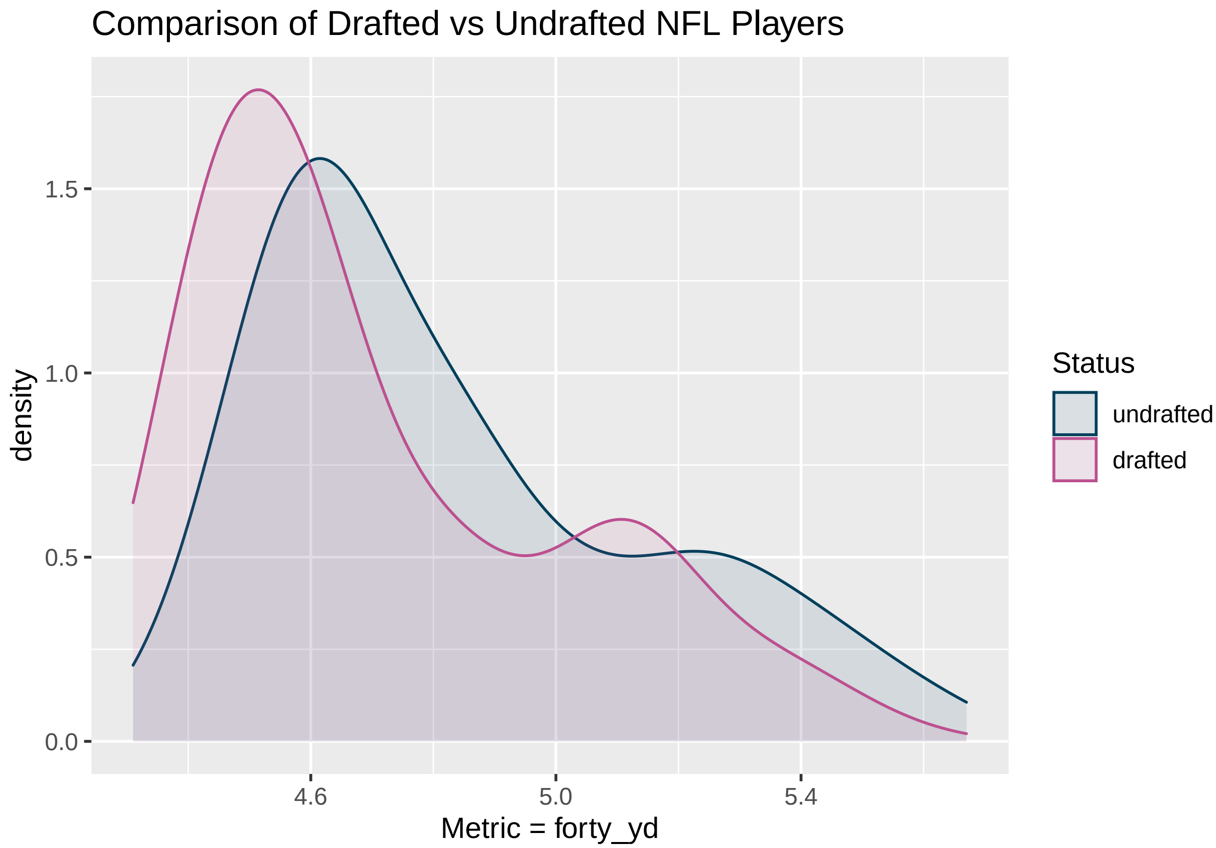
Appendix A



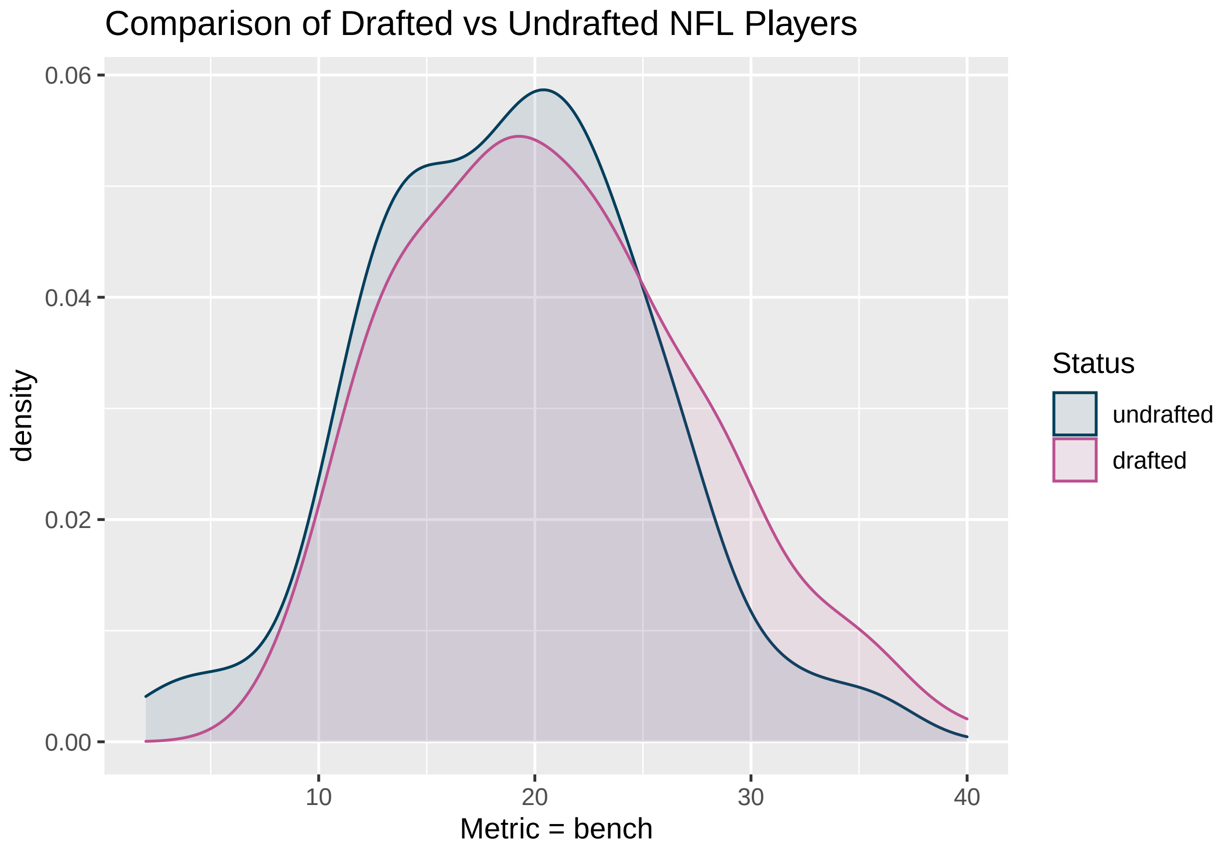
Comparison of Drafted vs Undrafted NFL Players
density
0.10
0.05
0.00
65
70
75
80
Metric = height
Status
undrafted
drafted



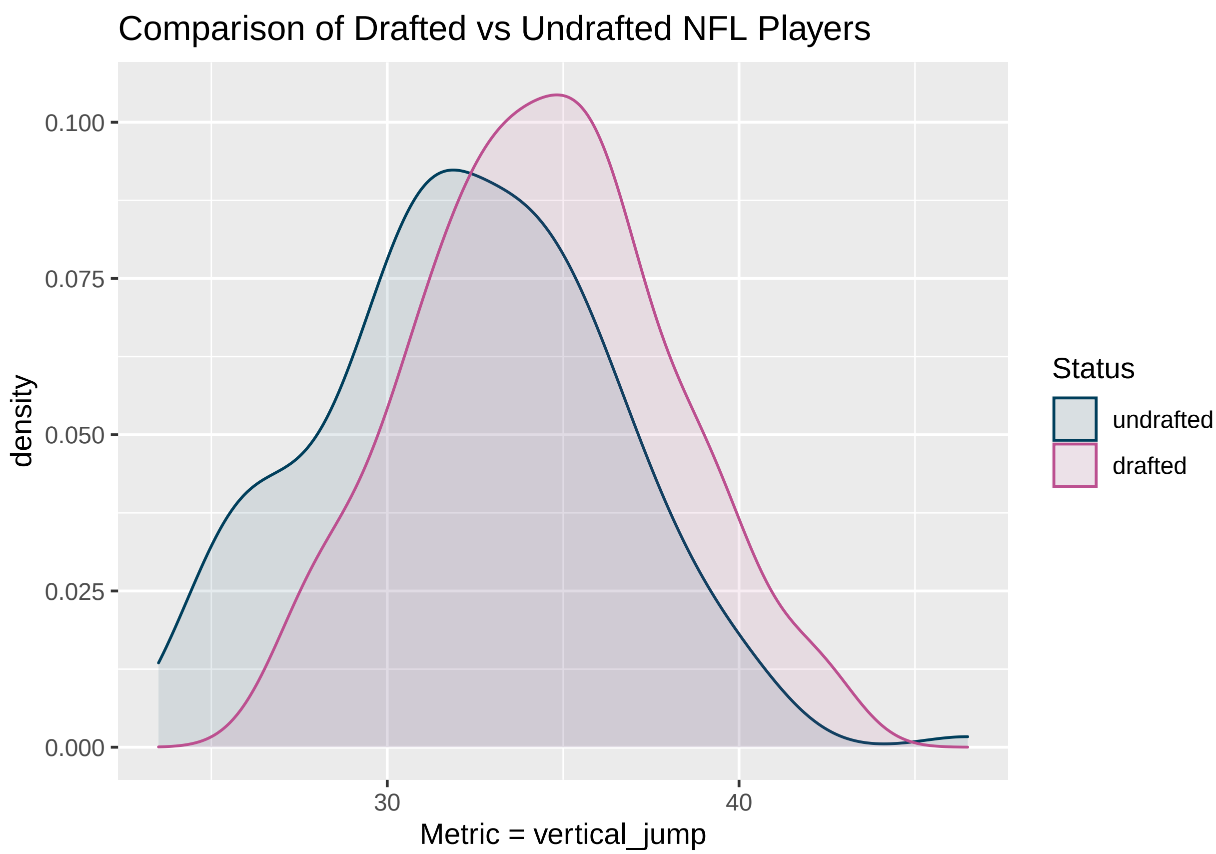

```
## Warning: Removed 63 rows containing non-finite values (stat_density).
```

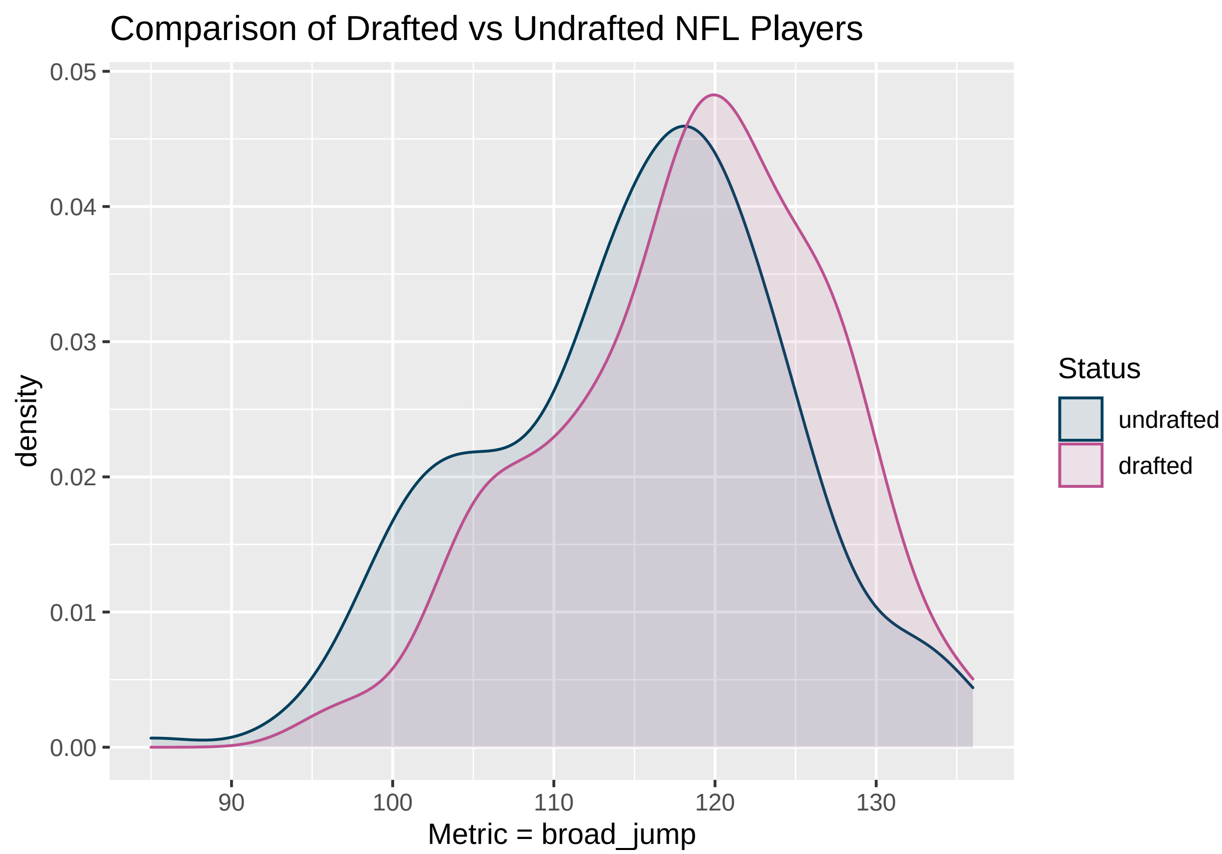



```
## Warning: Removed 95 rows containing non-finite values (stat_density).
```

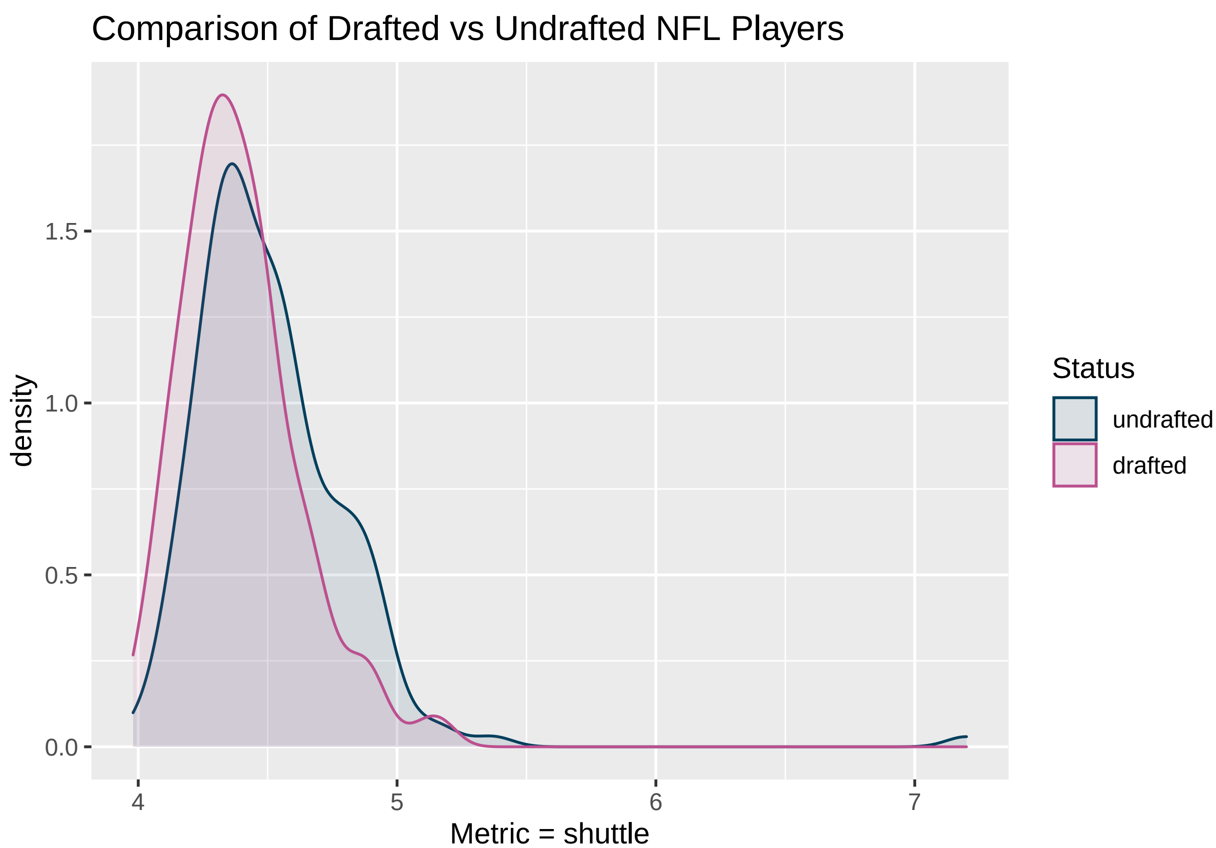



```
## Warning: Removed 54 rows containing non-finite values (stat_density).
```

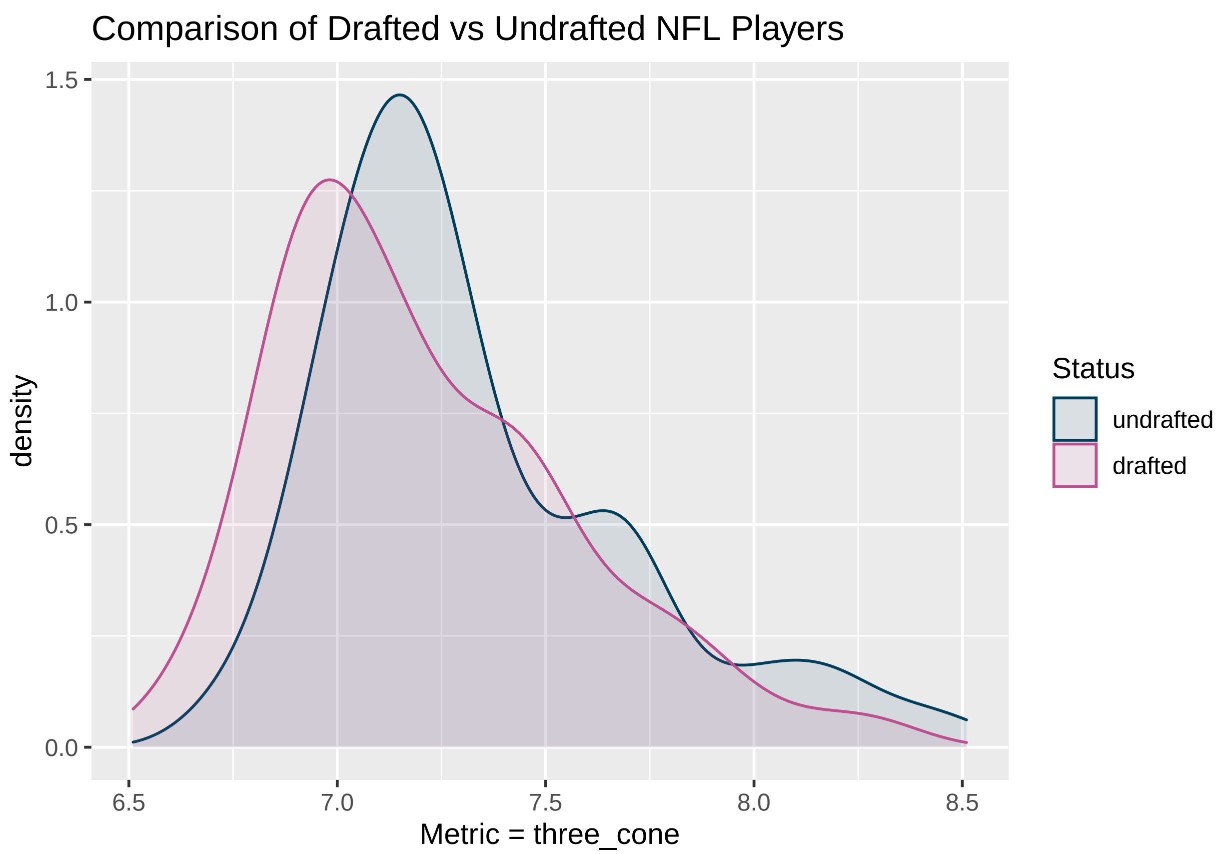

Comparison of Drafted vs Undrafted NFL Players
density
0.100
0.075
0.050
0.025
0.000
30
40
Metric = vertical_jump
Status
undrafted
drafted



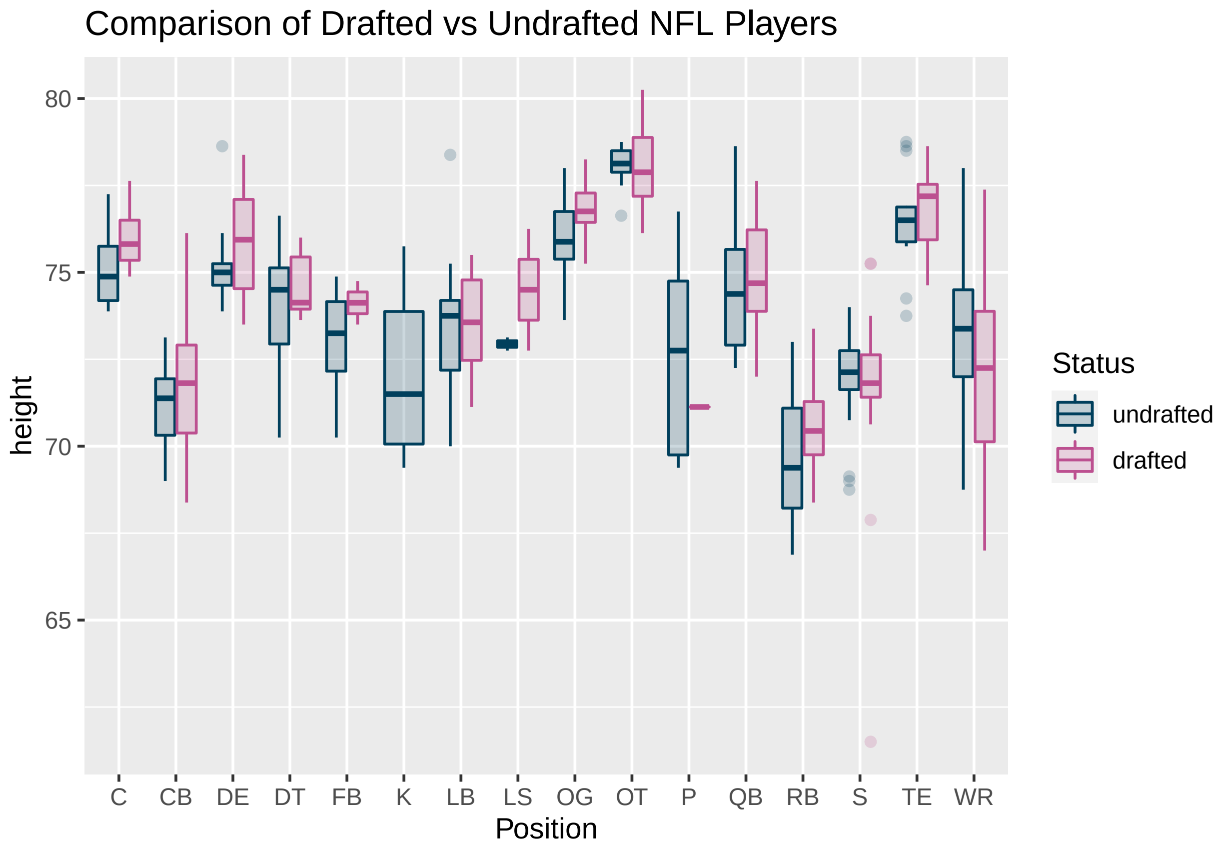

```
## Warning: Removed 74 rows containing non-finite values (stat_density).
```

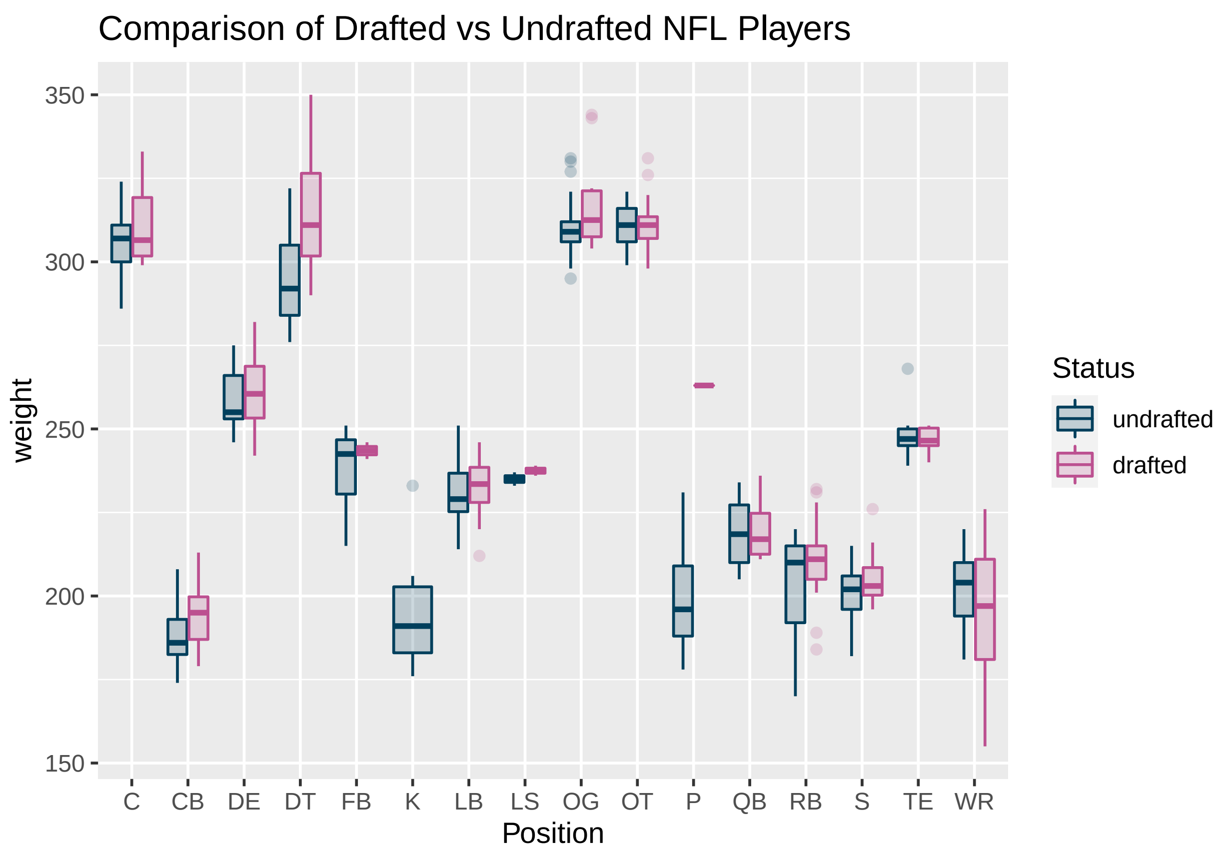



```
## Warning: Removed 81 rows containing non-finite values (stat_density).
```

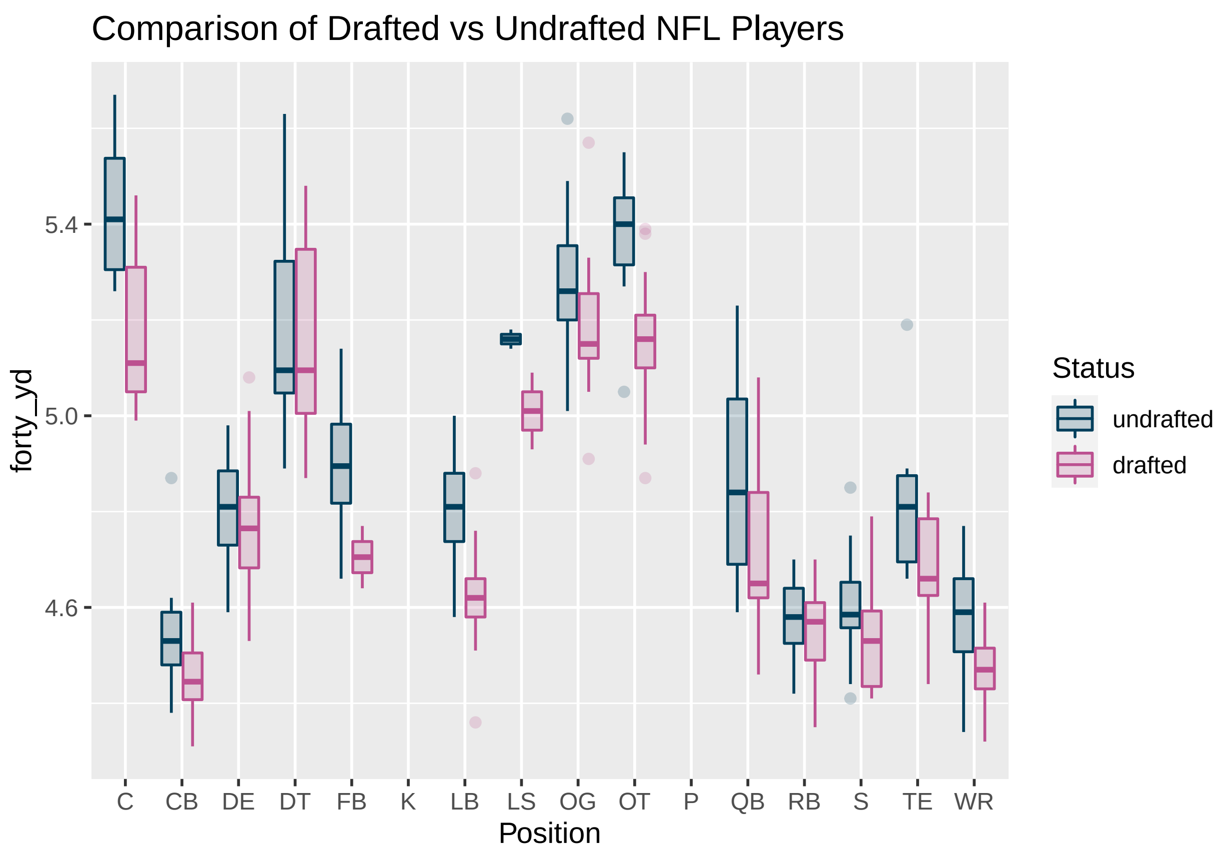

Comparison of Drafted vs Undrafted NFL Players
density
1.5
1.0
0.5
0.0
6.5
7.0
7.5
8.0
8.5
Metric = three_cone
Status
undrafted
drafted

Appendix B

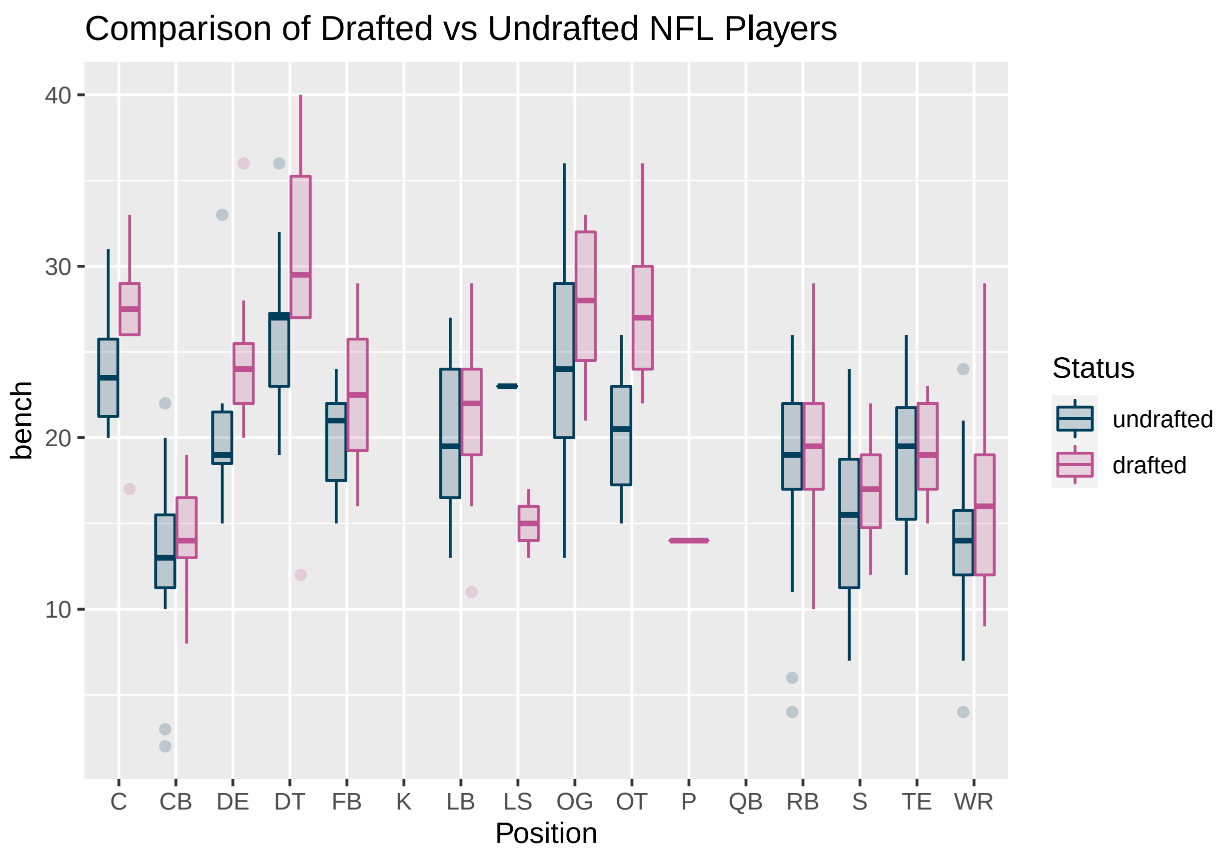

Comparison of Drafted vs Undrafted NFL Players
80
75
70
65
height
C
CB
DE
DT
FB
K
LB
LS
OG
OT
P
QB
RB
S
TE
WR
Position
Status
undrafted
drafted



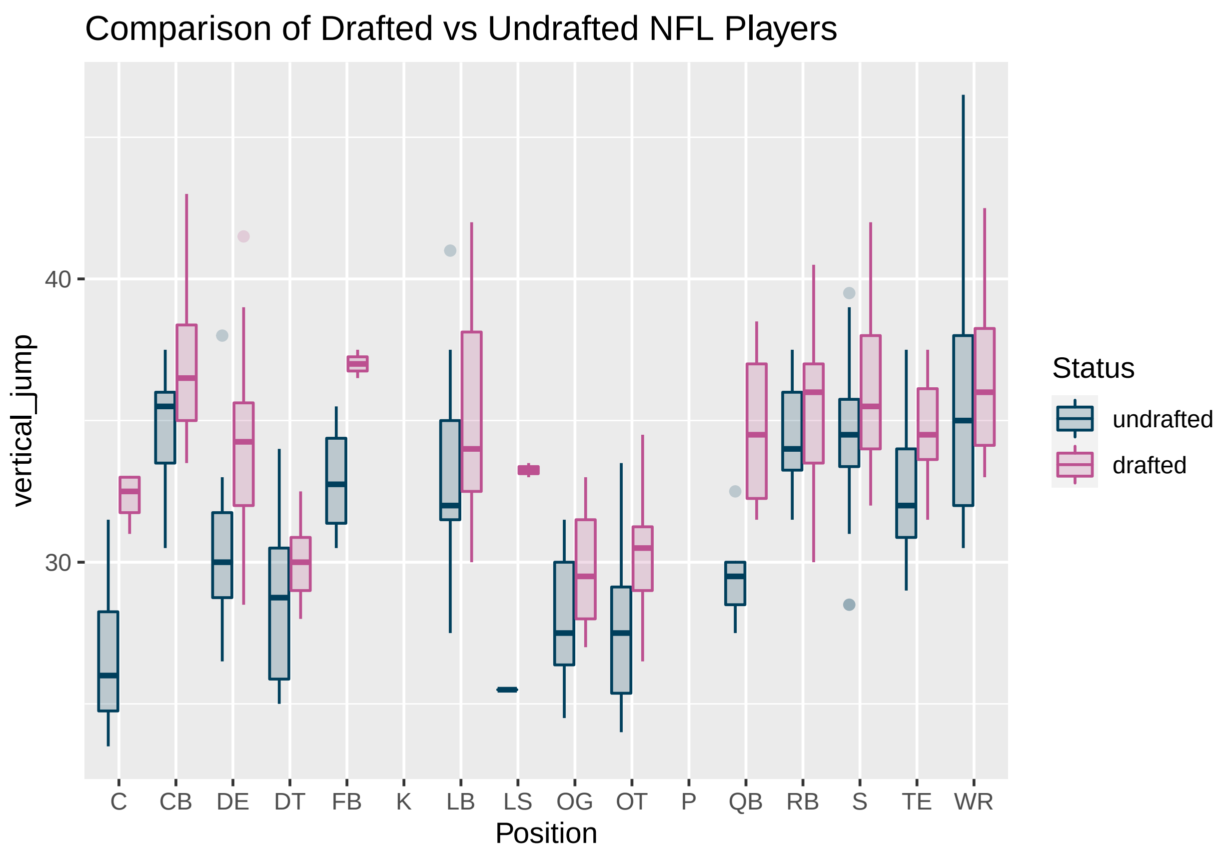

```
## Warning: Removed 63 rows containing non-finite values (stat_boxplot).
```

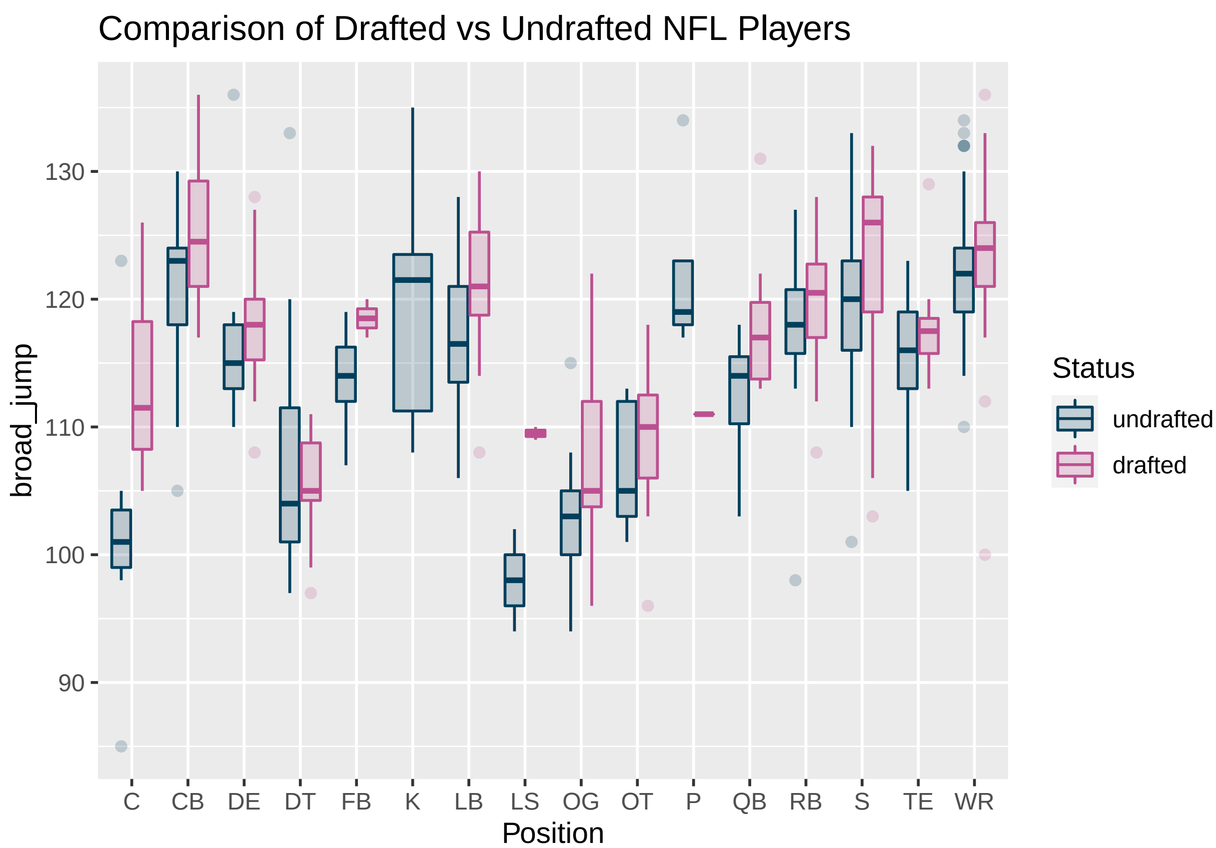



```
## Warning: Removed 95 rows containing non-finite values (stat_boxplot).
```

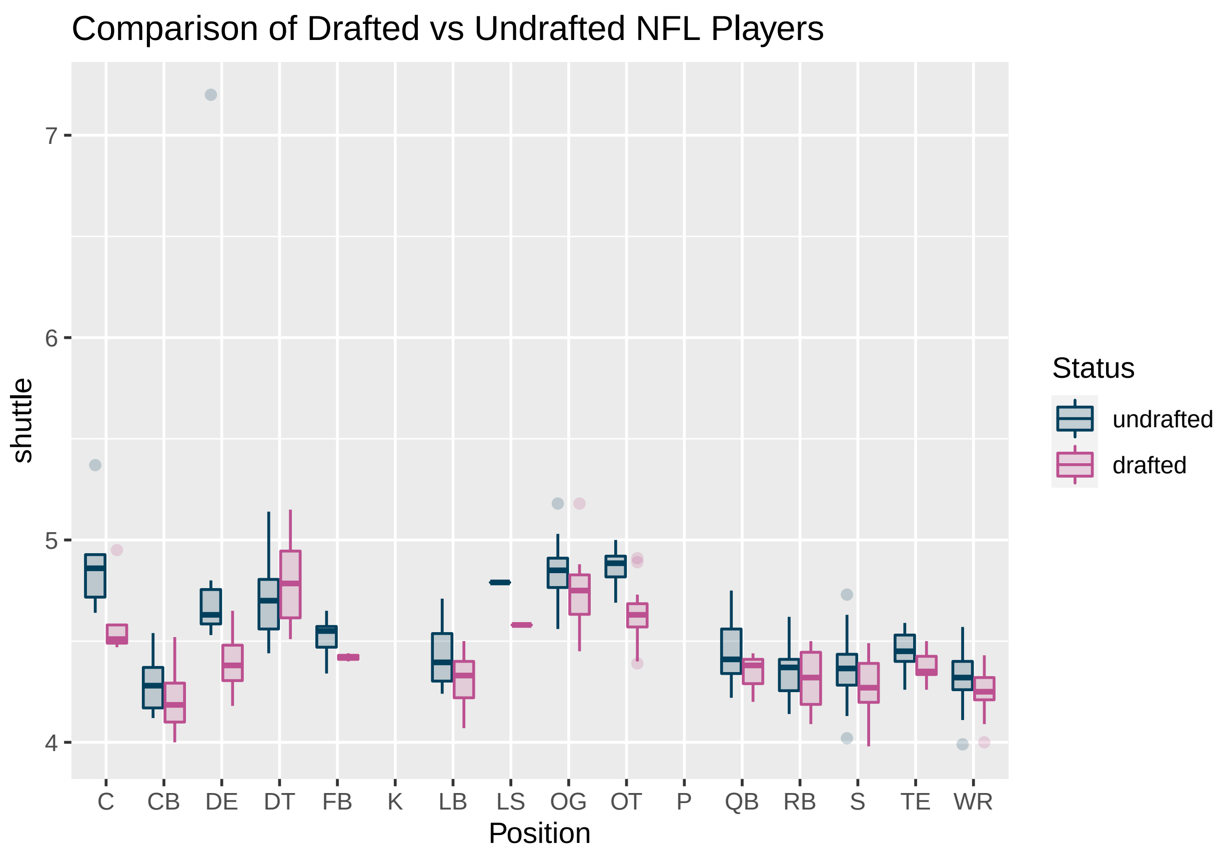



```
## Warning: Removed 54 rows containing non-finite values (stat_boxplot).
```

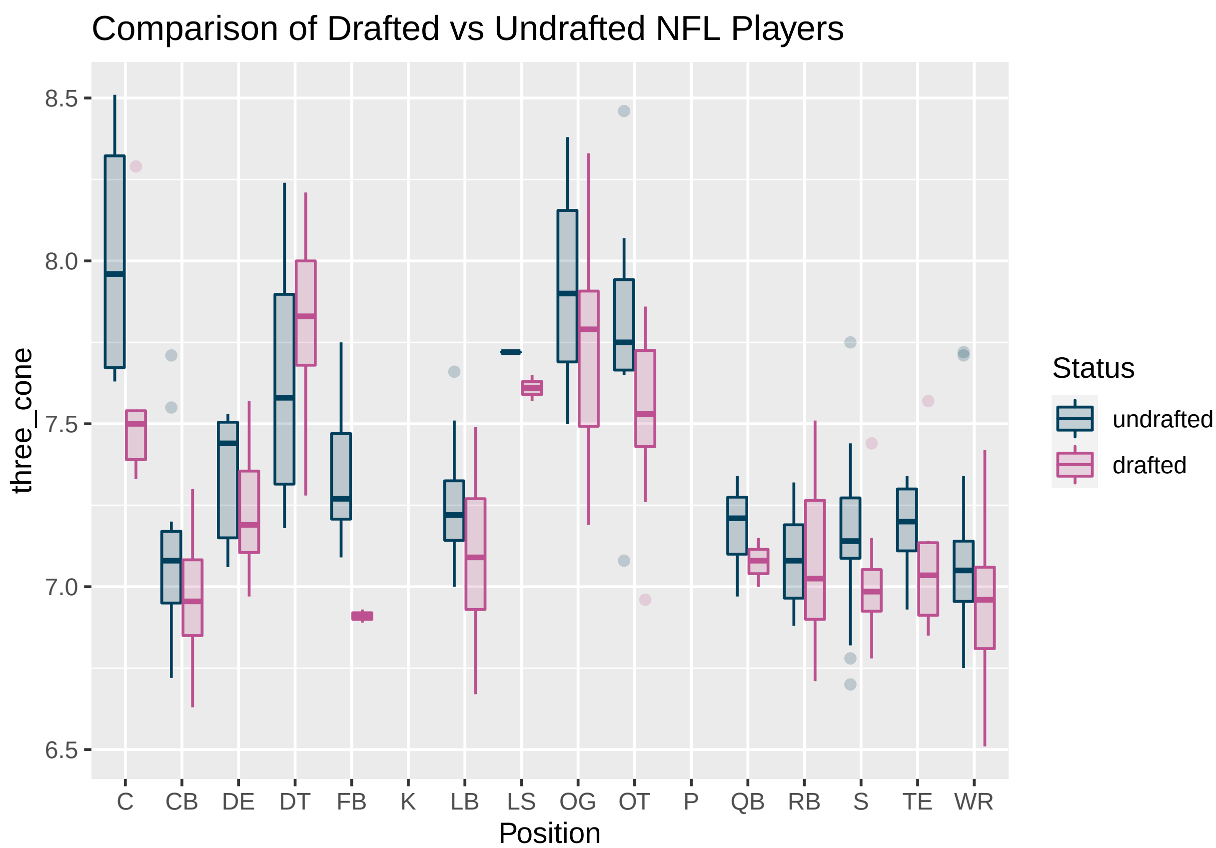

Comparison of Drafted vs Undrafted NFL Players
vertical_jump
40
30
C
CB
DE
DT
FB
K
LB
LS
OG
OT
P
QB
RB
S
TE
WR
Position
Status
undrafted
drafted




```
## Warning: Removed 74 rows containing non-finite values (stat_boxplot).
```

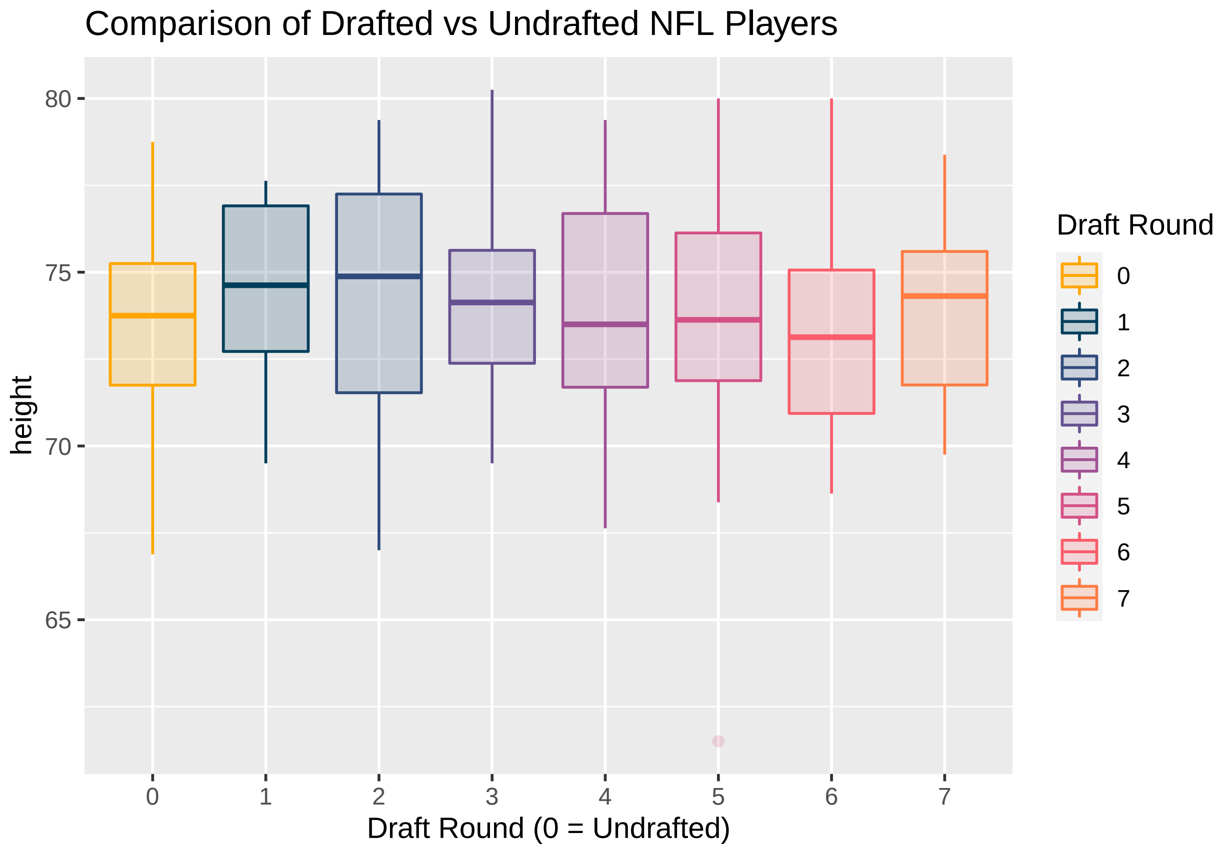



```
## Warning: Removed 81 rows containing non-finite values (stat_boxplot).
```

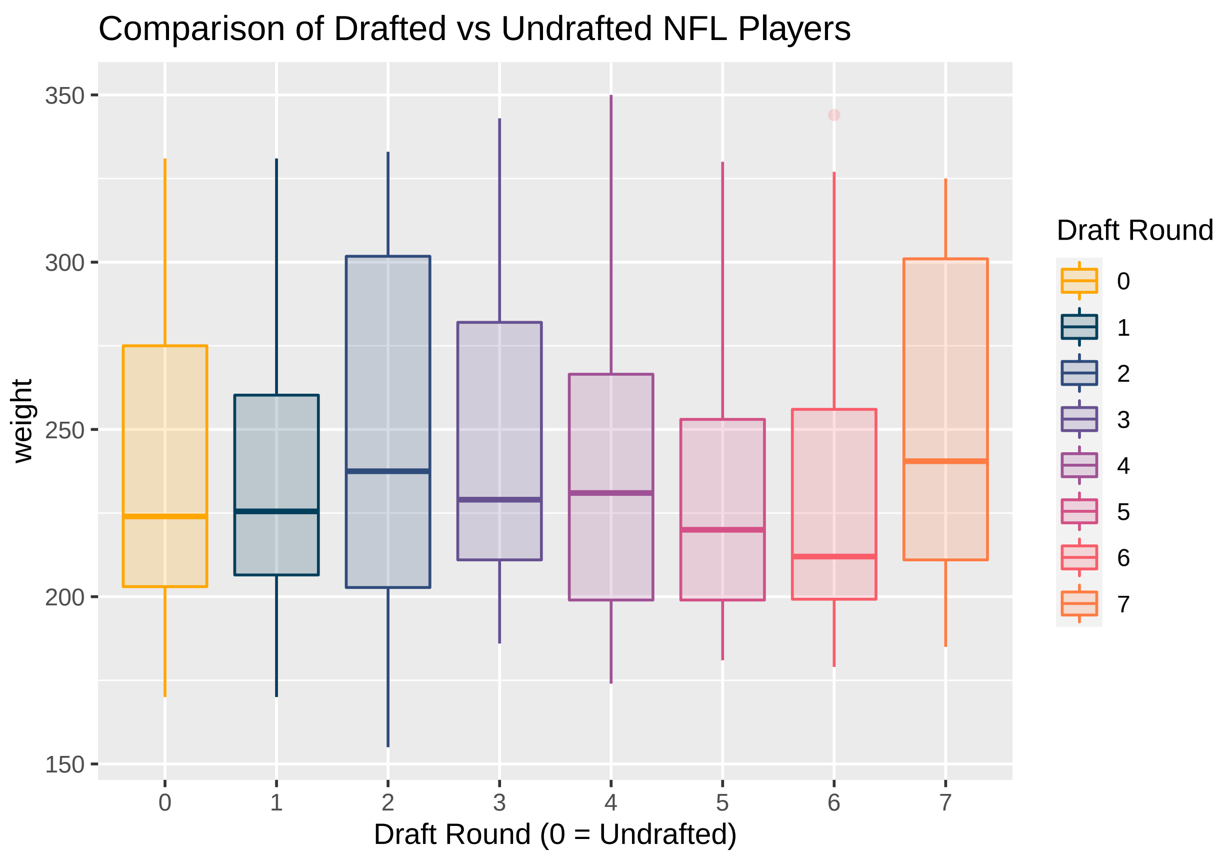


Comparison of Drafted vs Undrafted NFL Players
three_cone
8.5
8.0
7.5
7.0
6.5
C
CB
DE
DT
FB
K
LB
LS
OG
OT
P
QB
RB
S
TE
WR
Position
Status
undrafted
drafted

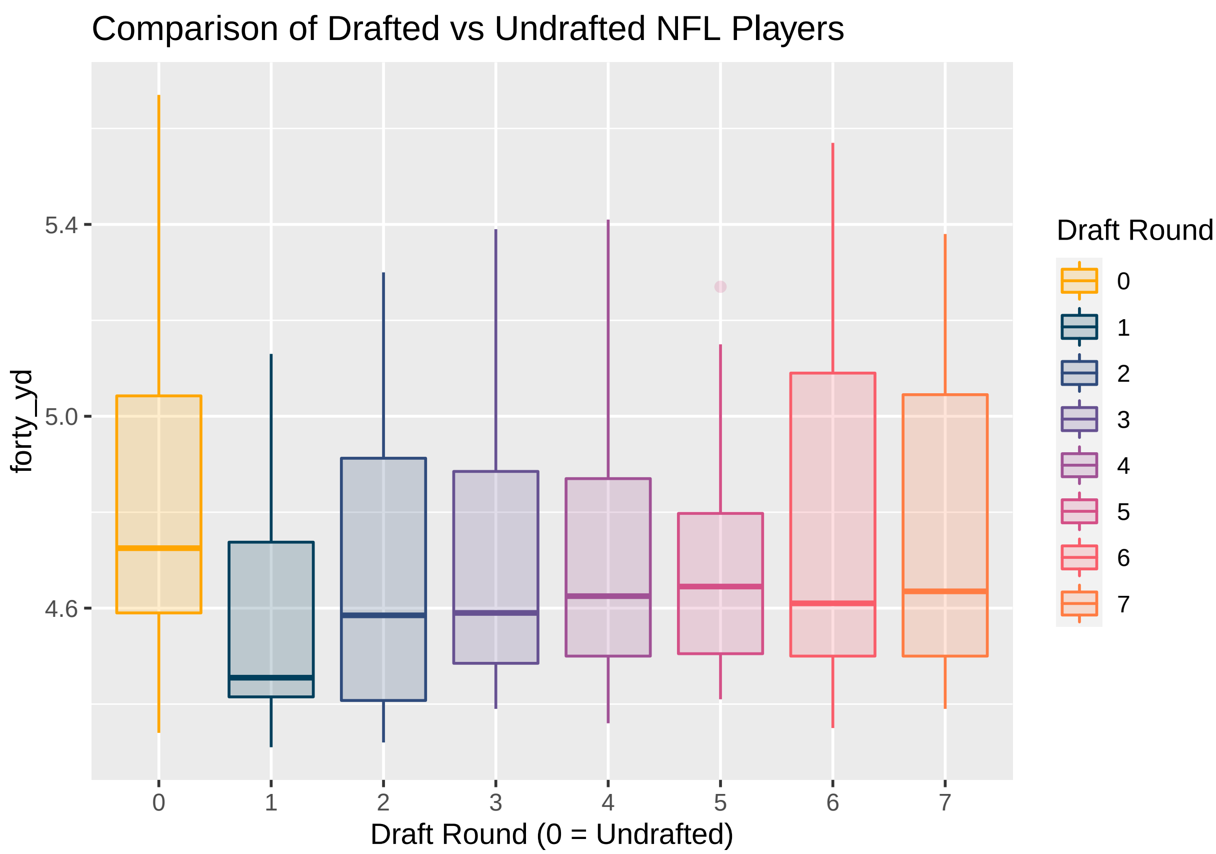
Appendix C



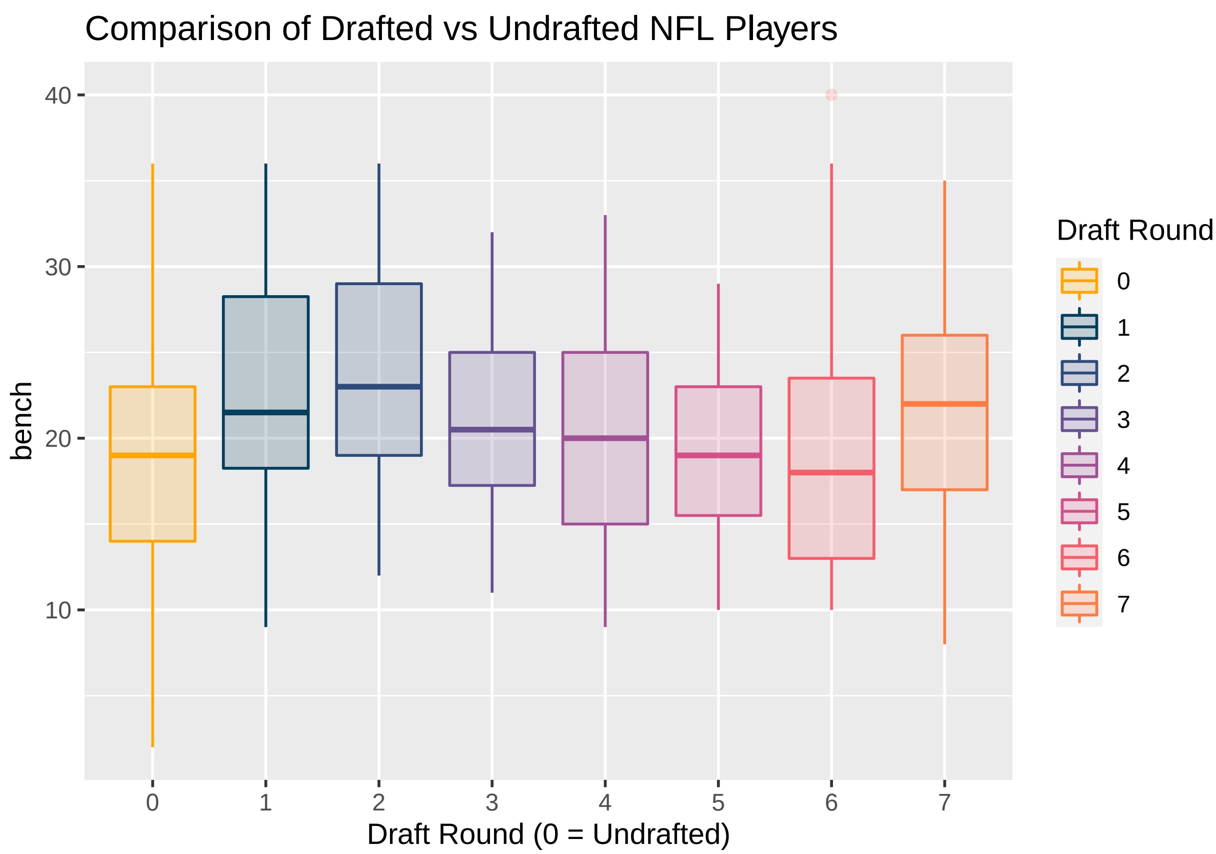
Comparison of Drafted vs Undrafted NFL Players
height
80
75
70
65
0
1
2
3
4
5
6
7
Draft Round (0 = Undrafted)
Draft Round
0
1
2
3
4
5
6
7



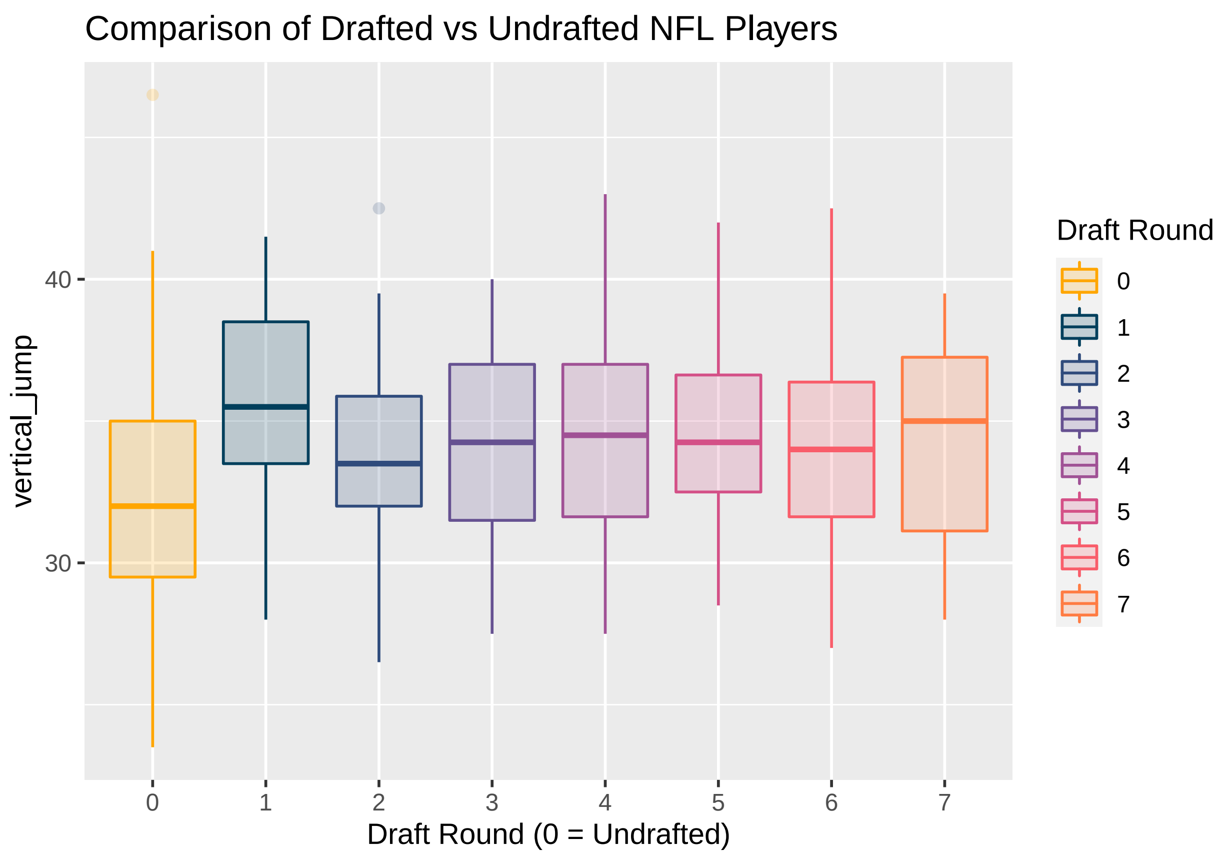

```
## Warning: Removed 63 rows containing non-finite values (stat_boxplot).
```

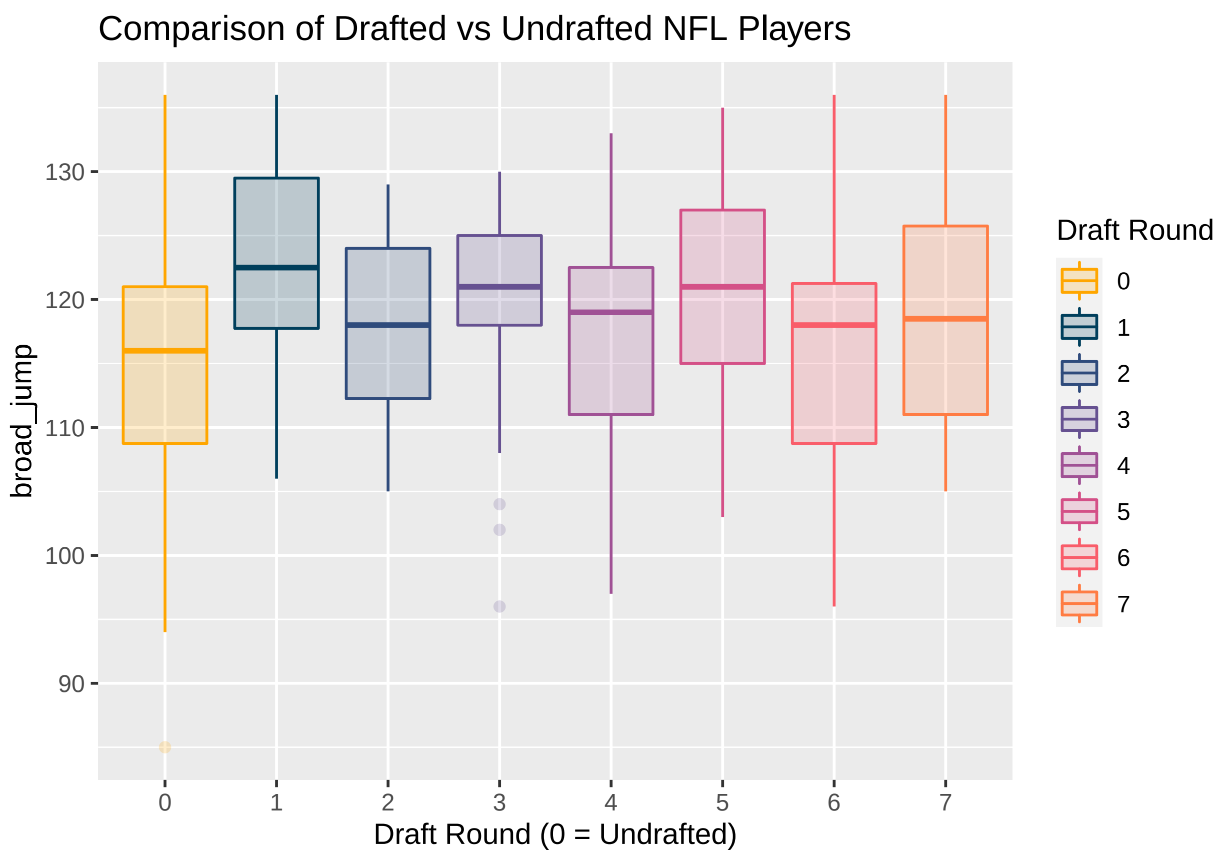



```
## Warning: Removed 95 rows containing non-finite values (stat_boxplot).
```

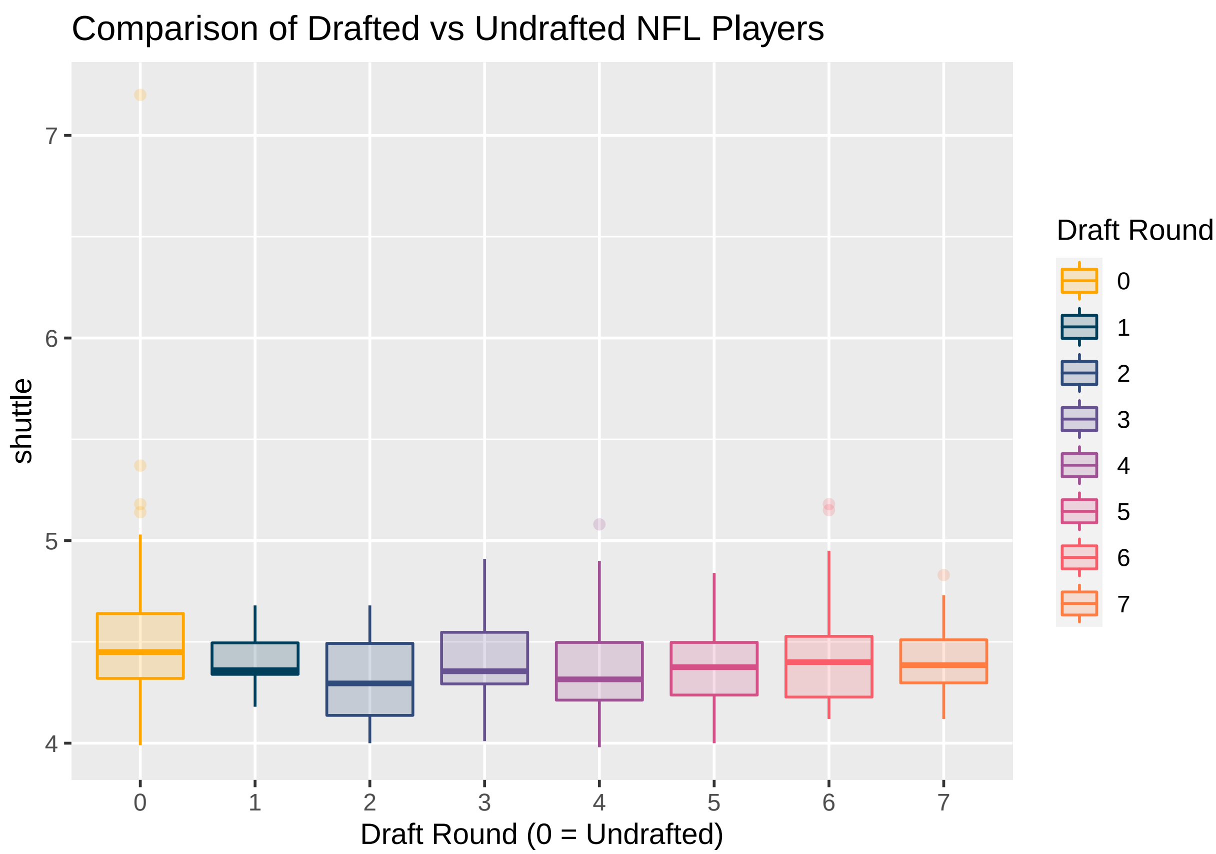



```
## Warning: Removed 54 rows containing non-finite values (stat_boxplot).
```

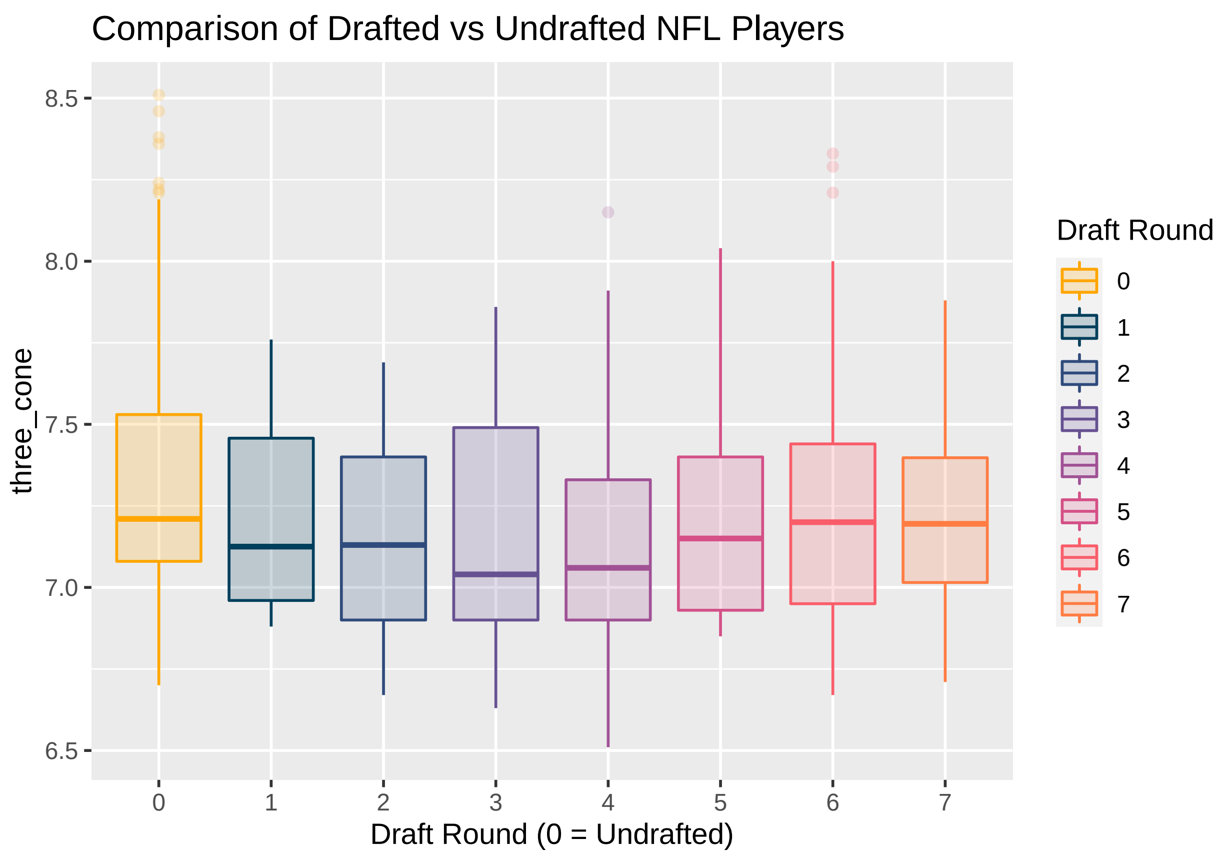

Comparison of Drafted vs Undrafted NFL Players
vertical_jump
40
30
0
1
2
3
4
5
6
7
Draft Round (0 = Undrafted)
Draft Round
0
1
2
3
4
5
6
7




```
## Warning: Removed 74 rows containing non-finite values (stat_boxplot).
```




```
## Warning: Removed 81 rows containing non-finite values (stat_boxplot).
```


Comparison of Drafted vs Undrafted NFL Players
three_cone
8.5
8.0
7.5
7.0
6.5
0
1
2
3
4
5
6
7
Draft Round (0 = Undrafted)
Draft Round
0
1
2
3
4
5
6
7

Appendix D

```r
# full binomial model

# use 5-fold cross-validation to train and test a binomial logistic regression model
# define training control
train_control <- trainControl(method = "cv", number = 5)

# train the model
model_bi_full <- train(drafted ~ ., data = data_bi, trControl = train_control,
                       method = "glm", family=binomial())
```

```r
# view results
summary(model_bi_full)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1252  -0.7727  -0.1645   0.8297   2.1489
##
## Coefficients: (3 not defined because of singularities)
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)         43.862147  13.320892   3.293 0.000992 ***
## height              -0.026685   0.113613  -0.235 0.814304
## weight               0.054108   0.018920   2.860 0.004238 **
## forty_yd            -7.088455   1.777083  -3.989 6.64e-05 ***
## bench                0.008088   0.038766   0.209 0.834729
## vertical_jump        0.104610   0.076018   1.376 0.168784
## broad_jump          -0.001740   0.039604  -0.044 0.964949
## shuttle             -3.604526   1.509699  -2.388 0.016960 *
## three_cone          -1.023239   0.958716  -1.067 0.285836
## college_positionCB -1.280090   2.268209  -0.564 0.572508
## college_positionDE -0.394396   1.502144  -0.263 0.792894
## college_positionDT -1.495026   1.204150  -1.242 0.214398
## college_positionFB -2.235552   1.798768  -1.243 0.213933
## college_positionK         NA         NA      NA       NA
## college_positionLB -1.801516   1.716709  -1.049 0.293994
## college_positionLS  3.645427   2.177581   1.674 0.094116 .
## college_positionOG  0.353334   1.167638   0.303 0.762190
## college_positionOT  1.002497   1.184494   0.846 0.397357
## college_positionP         NA         NA      NA       NA
## college_positionQB        NA         NA      NA       NA
## college_positionRB -1.402809   2.015762  -0.696 0.486479
## college_positionS  -2.074422   2.099470  -0.988 0.323119
## college_positionTE -2.328295   1.698007  -1.371 0.170315
## college_positionWR -1.846500   2.156797  -0.856 0.391926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 383.57  on 276  degrees of freedom
## Residual deviance: 270.72  on 256  degrees of freedom
## AIC: 312.72
##
## Number of Fisher Scoring iterations: 5
```

```r
model_bi_full$results
```

```
##   parameter  Accuracy     Kappa AccuracySD   KappaSD
## 1      none 0.7146104 0.4272496 0.05703391 0.1157437
```

```r
round((exp(coef(model_bi_full$finalModel)) - 1) * 100, 2)
```

```
##        (Intercept)              height              weight            forty_yd
##       1.119666e+21       -2.630000e+00        5.560000e+00       -9.992000e+01
##              bench        vertical_jump          broad_jump             shuttle
##       8.100000e-01        1.103000e+01       -1.700000e-01       -9.728000e+01
##         three_cone college_positionCB college_positionDE college_positionDT
##      -6.406000e+01       -7.220000e+01       -3.259000e+01       -7.758000e+01
## college_positionFB  college_positionK college_positionLB college_positionLS
##      -8.931000e+01                  NA       -8.350000e+01        3.729910e+03
## college_positionOG college_positionOT  college_positionP college_positionQB
##       4.238000e+01        1.725100e+02                  NA                  NA
## college_positionRB  college_positionS college_positionTE college_positionWR
##      -7.541000e+01       -8.744000e+01       -9.025000e+01       -8.422000e+01
```

```r
# parsimonious binomial model

# based on the results of the full model, manually remove insignificant independent
# variables to determine if a more parsimonious model improves the AIC and
# accuracy scores

# use 5-fold cross-validation to train and test a binomial logistic regression model
# define training control
train_control <- trainControl(method = "cv", number = 5)

# train the model
model_bi_par <- train(drafted ~ . - height - bench - broad_jump - three_cone - vertical_jump,
                      data = data_bi, trControl = train_control, method = "glm",
                      family=binomial())
```

```
## prediction from a rank-deficient fit may be misleading
```

```
# view results
summary(model_bi_par)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2594  -0.8156  -0.1850   0.7974   2.0254
##
## Coefficients: (3 not defined because of singularities)
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)          50.70780    9.46435   5.358 8.43e-08 ***
## weight                0.05484    0.01496   3.666 0.000247 ***
## forty_yd             -8.47320    1.60492  -5.280 1.30e-07 ***
## shuttle              -4.99620    1.24337  -4.018 5.86e-05 ***
## college_positionCB -1.55436    2.15474  -0.721 0.470684
## college_positionDE -0.55903    1.40583  -0.398 0.690886
## college_positionDT -1.56063    1.16977  -1.334 0.182161
## college_positionFB -1.96265    1.72649  -1.137 0.255626
## college_positionK        NA         NA      NA       NA
## college_positionLB -2.01414    1.62650  -1.238 0.215593
## college_positionLS  3.47288    2.05932   1.686 0.091715 .
## college_positionOG  0.16892    1.13013   0.149 0.881184
## college_positionOT  0.98626    1.12823   0.874 0.382028
## college_positionP        NA         NA      NA       NA
## college_positionQB       NA         NA      NA       NA
## college_positionRB -1.61107    1.95470  -0.824 0.409824
## college_positionS  -2.24647    2.00780  -1.119 0.263196
## college_positionTE -2.52448    1.57993  -1.598 0.110079
## college_positionWR -2.06269    2.02407  -1.019 0.308166
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 383.57  on 276  degrees of freedom
## Residual deviance: 274.70  on 261  degrees of freedom
## AIC: 306.7
##
## Number of Fisher Scoring iterations: 5
```

```
model_bi_par$results
```

```
##   parameter  Accuracy     Kappa AccuracySD   KappaSD
## 1      none 0.7004545 0.3996376 0.06032482 0.1214077
```

```
round((exp(coef(model_bi_par$finalModel)) - 1) * 100, 2)
```

```
##        (Intercept)             weight           forty_yd            shuttle
##       1.052249e+24       5.640000e+00      -9.998000e+01      -9.932000e+01
## college_positionCB college_positionDE college_positionDT college_positionFB
##      -7.887000e+01      -4.282000e+01      -7.900000e+01      -8.595000e+01
```

```
##  college_positionK college_positionLB college_positionLS college_positionOG
##                 NA      -8.666000e+01       3.122930e+03       1.840000e+01
## college_positionOT  college_positionP college_positionQB college_positionRB
##       1.681200e+02                 NA                 NA      -8.003000e+01
##  college_positionS college_positionTE college_positionWR
##      -8.942000e+01      -9.199000e+01      -8.729000e+01
```

```r
# full multinomial model

# use 5-fold cross-validation to train and test a multinomial regression model
# define training control
train_control <- trainControl(method = "cv", number = 5)

# train the model
model_multi_full <- train(selection_fac ~ ., data = data_multi, trControl = train_control,
                          method = "multinom")
```

```r
# view results
summary(model_multi_full)
```

```
## Warning in sqrt(diag(vc)): NaNs produced
```

```
## Call:
## nnet::multinom(formula = .outcome ~ ., data = dat, decay = param$decay)
##
## Coefficients:
##   (Intercept)       height      weight   forty_yd       bench vertical_jump
## 1 -0.24774067 -0.134407706 0.086413823 -4.8231573 -0.02129695    0.04209937
## 2  0.34164215 -0.214504907 0.055052168 -0.6826478  0.07193440    0.07083185
## 3 -0.22600820  0.122756591 0.038659221 -2.6804946  0.12753964    0.02137838
## 4  0.29174889 -0.009323163 0.037160726 -0.7712170 -0.04872810    0.29247677
## 5 -0.25450422 -0.102764961 0.026462713 -0.5336928  0.02025407    0.09130907
## 6  0.23452318  0.076257704 0.003813485 -1.3135487  0.07202777    0.16894724
## 7 -0.07193266 -0.031441363 0.061259041 -0.9107663 -0.02762316    0.26826636
##     broad_jump     shuttle three_cone college_positionCB college_positionDE
## 1  0.207011097 -2.02091120 -1.0324179          0.6250105          0.8501245
## 2  0.158711797 -2.03495970 -1.4675570          0.2410319          0.1860355
## 3  0.098313800 -1.34252773 -2.3562375          2.0200705         -0.1846736
## 4 -0.014088820 -1.09093400 -1.2428058          0.3121243          1.0935163
## 5  0.087466600 -1.16914110 -0.8717880          0.7467576          1.2658111
## 6 -0.023754391 -0.04170966 -0.8900498          0.7883504          0.4651262
## 7  0.004586228 -2.36646811 -1.2645229         -0.8518866          0.7405107
##   college_positionDT college_positionFB college_positionK college_positionLB
## 1        -1.29067688         -0.4925527                 0        -0.33743742
## 2        -1.66083904         -0.8370656                 0         0.13049480
## 3         0.01711176         -0.9061794                 0         0.45279079
## 4        -0.63506775         -0.1706817                 0        -2.08130693
## 5        -0.22408976          0.1316036                 0        -0.09586392
## 6        -0.18471537         -1.4294674                 0        -0.50238925
## 7        -0.33789300         -0.9558029                 0         0.39796361
##   college_positionLS college_positionOG college_positionOT college_positionP
## 1       -0.003835935          1.9812109          0.4628892                 0
## 2       -0.015579314         -1.0401638          1.4885659                 0
## 3       -0.012235917          0.4742029         -0.2342694                 0
## 4       -0.286097228          0.5188339          0.2236511                 0
## 5       -0.135733297         -1.0725590          0.8960355                 0
## 6        2.881189547          0.1311724          1.2188263                 0
## 7       -0.081992049          0.2382341         -0.1406385                 0
##   college_positionQB college_positionRB college_positionS college_positionTE
## 1                  0        -0.83260877       -1.14406723          0.3167713
## 2                  0         0.01499849       -0.09805443         -1.0253157
## 3                  0         0.51651977        0.18571883         -1.5090133
## 4                  0         0.53281668       -0.31280829         -0.6839419
## 5                  0        -0.72206563        0.22324278          1.0663896
## 6                  0         0.91760950       -0.39080176         -1.7021887
## 7                  0         1.75232216       -1.04165080         -1.3351348
##   college_positionWR
## 1         0.03869901
## 2        -0.14690925
## 3         0.42187151
## 4         0.12823581
```

```
## 5        -0.61157365
## 6         0.03730659
## 7         1.27863171
##
## Std. Errors:
##   (Intercept)    height     weight forty_yd      bench vertical_jump broad_jump
## 1   0.2460025 0.2506174 0.04189012 3.417881 0.10062543     0.1855732 0.10488946
## 2   0.1531867 0.2252736 0.03058223 2.654162 0.08536342     0.1571926 0.09028492
## 3   0.1658986 0.1900067 0.02868608 2.512714 0.08118948     0.1426716 0.07829617
## 4   0.2041017 0.1460822 0.02169159 2.050276 0.05805527     0.1142401 0.05833790
## 5   0.4361679 0.1501970 0.02396133 1.931481 0.05930977     0.1048765 0.05762567
## 6   0.1720102 0.1394713 0.02159124 1.925110 0.05499662     0.1108151 0.05557062
## 7   0.1330934 0.1722195 0.02324890 2.191011 0.06455688     0.1355477 0.07032784
##    shuttle three_cone college_positionCB college_positionDE college_positionDT
## 1 3.837915   2.234296          1.8719592          1.2523186           3.029382
## 2 2.945024   1.828212          1.2992400          1.3015291           2.386968
## 3 2.763402   1.750706          1.1158770          1.2093622           1.887769
## 4 2.287619   1.443164          1.1077632          0.9668157           1.514360
## 5 2.104837   1.340377          1.5313486          1.2083484           1.713759
## 6 2.086272   1.337549          0.9321415          0.9619625           1.548719
## 7 2.693845   1.606114          2.1190936          1.0879938           1.383216
##   college_positionFB college_positionK college_positionLB college_positionLS
## 1           3.050377               NaN          1.1598554        0.006284354
## 2           2.392635      1.820707e-14          0.8512256        0.028039242
## 3           2.321272               NaN          0.7498907        0.020293769
## 4           1.220397      2.152539e-15          1.5522111        3.748181036
## 5           1.449208      1.235155e-15          1.1890147        4.597874832
## 6           1.826819               NaN          0.7520628        1.312340889
## 7           2.273558      0.000000e+00          0.9233107        0.323953420
##   college_positionOG college_positionOT college_positionP college_positionQB
## 1           2.515418           2.635383               NaN                NaN
## 2           2.742921           1.756404               NaN                NaN
## 3           1.967368           1.936301      8.375064e-15       7.225450e-15
## 4           1.421987           1.419727      3.692619e-15                NaN
## 5           2.563030           1.629179      1.435597e-15                NaN
## 6           1.616246           1.399567               NaN       3.338016e-28
## 7           1.395238           1.406483      0.000000e+00       0.000000e+00
##   college_positionRB college_positionS college_positionTE college_positionWR
## 1          2.4355573          2.234611           1.433204          1.5855389
## 2          1.1629061          1.191881           2.197858          1.1573251
## 3          1.1982531          1.195267           1.801646          1.0434377
## 4          0.9368327          1.035458           1.188766          0.9902899
## 5          1.4880940          1.406344           1.236965          1.5277146
## 6          0.7198766          0.890399           1.668981          0.8530822
## 7          0.9676892          1.982981           1.954877          0.9875882
##
## Residual Deviance: 713.895
## AIC: 1007.895
```

```
model_multi_full$results
```

```
##   decay  Accuracy     Kappa AccuracySD    KappaSD
## 1 0e+00 0.4689572 0.1636648 0.03940646 0.04822757
## 2 1e-04 0.4725287 0.1601990 0.03544090 0.03892771
## 3 1e-01 0.4909102 0.0985806 0.01422279 0.03257854
```

```r
# parsimonious multinomial model

# use the same independent variables used in the parsimounious binomial model
# to determine if a more parsimonious model improves the AIC and accuracy scores

# use 5-fold cross-validation to train and test a multinomial regression model
# define training control
train_control <- trainControl(method = "cv", number = 5)

# train the model
model_multi_par <- train(selection_fac ~ . - height - bench - broad_jump - three_cone
                         - vertical_jump, data = data_multi, trControl = train_control,
                         method = "multinom")
```

```r
# view results
summary(model_multi_par)
```

```
## Warning in sqrt(diag(vc)): NaNs produced
```

```
## Call:
## nnet::multinom(formula = .outcome ~ ., data = dat, decay = param$decay)
##
## Coefficients:
##   (Intercept)     weight   forty_yd     shuttle college_positionCB
## 1   1.2904409 0.04787176 -3.6127288  0.16581806          1.8739466
## 2   1.2590075 0.03026216 -0.8753429 -1.63845646          0.9122094
## 3   1.0146975 0.03991393 -2.2269790 -0.69261304          2.8114577
## 4   1.1473463 0.01782835 -0.5661758 -1.07562979          0.8930400
## 5   0.7427009 0.01164650 -0.4466757 -0.81322570          1.5381507
## 6   0.8371576 0.00968170 -1.1055180  0.00882611          1.1278489
## 7   0.6339117 0.04021990 -0.7283467 -2.20138341         -0.5999667
##   college_positionDE college_positionDT college_positionFB college_positionK
## 1          1.4205361         -1.6754133         -0.4320214                 0
## 2          0.4181734         -1.6823445         -0.7114187                 0
```

```
## 3          0.3870286          -0.7148515          -0.6311372                  0
## 4          1.2083166          -0.7613538           0.2619667                  0
## 5          1.4695697          -0.3859965           0.3672744                  0
## 6          0.6754535          -0.3641632          -1.1491207                  0
## 7          0.9156189          -0.4056222          -0.6981372                  0
##   college_positionLB college_positionLS college_positionOG college_positionOT
## 1          0.6294666        -0.04635688          0.3259554        -0.26074307
## 2          0.6470513        -0.14732408         -1.6284404         0.83982153
## 3          1.2276771        -0.10280940         -0.5899281        -0.57970239
## 4         -1.8718714        -0.36340936         -0.1415097        -0.06071031
## 5          0.3144136        -0.35217304         -1.5710010         0.36907339
## 6         -0.1814432         2.58162568         -0.3140201         0.84266767
## 7          0.7031829        -0.13124979         -0.3325267        -0.39275852
##   college_positionP college_positionQB college_positionRB college_positionS
## 1                 0                  0         -0.5051395      -0.502755325
## 2                 0                  0          0.7117926       0.618756748
## 3                 0                  0          0.9443269       0.908748298
## 4                 0                  0          0.7966392       0.236530211
## 5                 0                  0         -0.1939592       0.892035462
## 6                 0                  0          1.2701523       0.008685164
## 7                 0                  0          2.0801869      -0.778183809
##   college_positionTE college_positionWR
## 1          0.9207567        1.219444379
## 2         -1.0383453        0.428847378
## 3         -0.9945074        1.389550615
## 4         -0.1986114        0.683318795
## 5          1.2836917        0.003724875
## 6         -1.4201152        0.433316460
## 7         -1.0618638        1.823674305
##
## Std. Errors:
##   (Intercept)     weight forty_yd  shuttle college_positionCB
## 1    2.436438 0.02946400 2.630238 2.734008           2.181026
## 2    2.228958 0.02425531 2.233270 2.367076           1.875304
## 3    1.843700 0.02230519 1.937642 2.121723           1.643047
## 4    1.855921 0.01869763 1.683606 1.686413           1.415262
## 5    2.098536 0.01831801 1.612074 1.605876           1.340932
## 6    2.066906 0.01713856 1.630994 1.610618           1.323273
## 7    2.049452 0.02098175 1.802385 1.992494           3.083321
##   college_positionDE college_positionDT college_positionFB college_positionK
## 1           1.530406           2.451011           3.571854      1.737439e-08
## 2           1.512319           2.175452           2.870775      2.153082e-09
## 3           1.537738           1.645505           3.042912               NaN
## 4           1.106547           1.459332           1.508578               NaN
## 5           1.206470           1.613150           1.562079               NaN
## 6           1.333671           1.643686           2.363063               NaN
## 7           1.216239           1.295116           2.865036      0.000000e+00
##   college_positionLB college_positionLS college_positionOG college_positionOT
## 1           1.700495          0.7597399           1.858290           2.012580
## 2           1.372218          5.2425654           2.203773           1.469319
## 3           1.335724          1.2208640           1.655902           1.673138
## 4           1.892433          3.9325827           1.355986           1.371367
## 5           1.221038          3.9877387           2.232588           1.542731
## 6           1.287439          1.7648671           1.696204           1.523806
```

```
## 7           1.285543          5.5064145           1.302263           1.320755
##   college_positionP college_positionQB college_positionRB college_positionS
## 1      1.376040e-09       1.977101e-09           3.357293          3.387946
## 2      7.668178e-12       8.394662e-15           1.724470          1.789151
## 3      8.596677e-15       1.847196e-14           1.764706          1.817064
## 4      1.809993e-15                NaN           1.306328          1.410474
## 5      1.595651e-15                NaN           1.512754          1.326297
## 6      8.519601e-20                NaN           1.247065          1.389559
## 7      0.000000e+00       0.000000e+00           1.406922          2.773109
##   college_positionTE college_positionWR
## 1           1.701823           1.991894
## 2           2.409306           1.774749
## 3           2.476928           1.630317
## 4           1.368614           1.323108
## 5           1.172821           1.399616
## 6           2.125913           1.294733
## 7           2.348457           1.479067
##
## Residual Deviance: 782.1441
## AIC: 1006.144
```

```
model_multi_par$results
```

```
##   decay  Accuracy     Kappa AccuracySD    KappaSD
## 1 0e+00 0.4911004 0.1621816 0.03049675 0.04297771
## 2 1e-04 0.4944768 0.1484348 0.02914670 0.05669059
## 3 1e-01 0.5200059 0.0194436 0.01217398 0.02006890
```