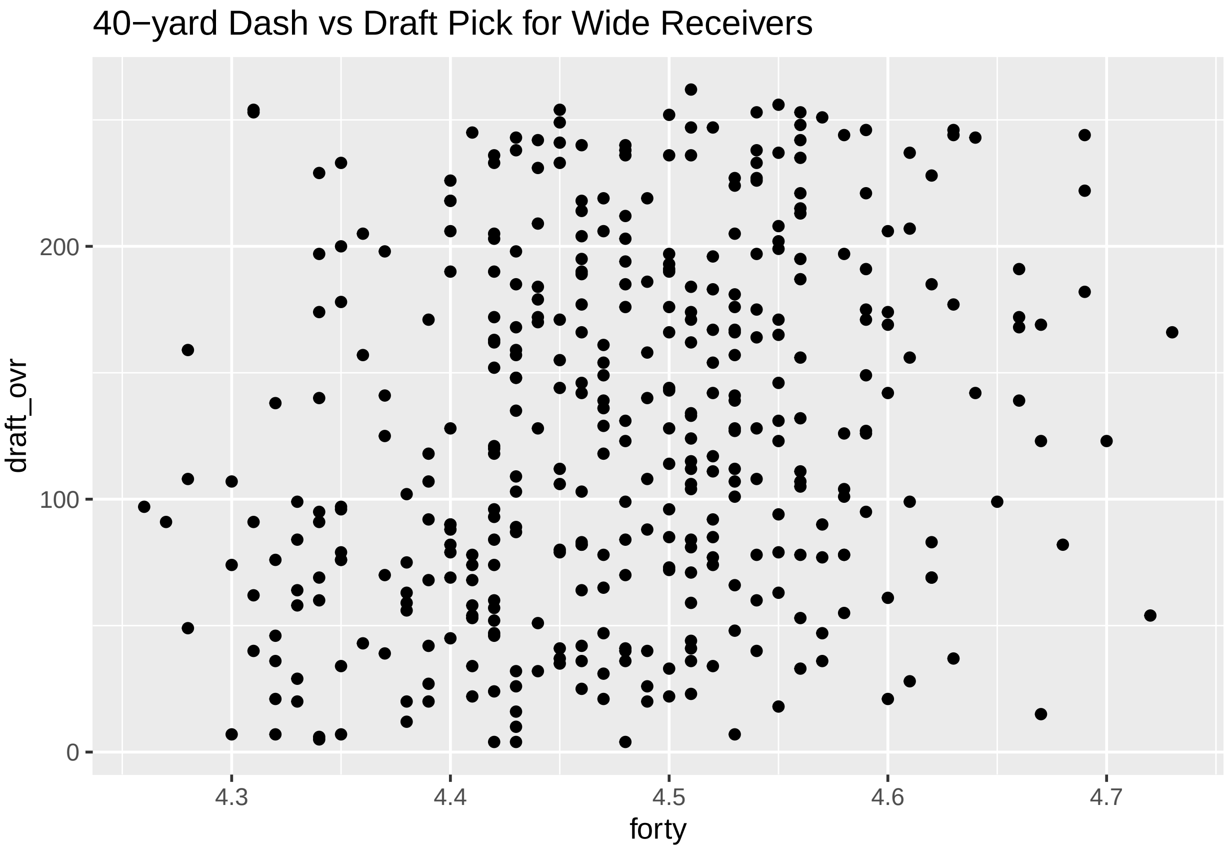
4330 Final Project

Johnny Ferrara

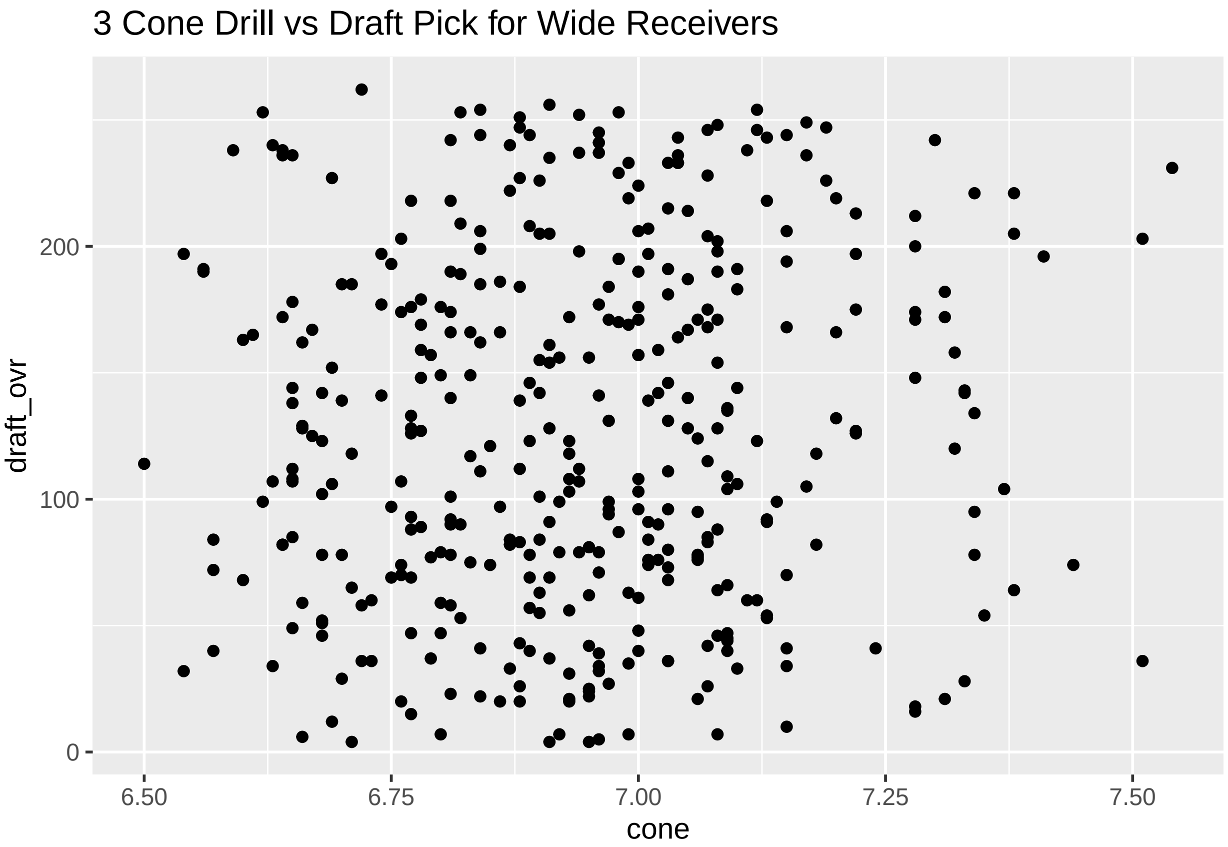
```r
# Load libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)

# Read and prepare data
nfl_data <- read.csv("nflVerseDraft.csv")
nfl_receivers <- nfl_data %>%
  filter(pos == "WR", !is.na(forty), !is.na(cone), !is.na(draft_ovr))

# Descriptive Statistics
summary_stats <- summary(nfl_receivers[c("forty", "cone", "draft_ovr")])

# Data Visualization
ggplot(nfl_receivers, aes(x = forty, y = draft_ovr)) +
  geom_point() +
  labs(title = "40-yard Dash vs Draft Pick for Wide Receivers")
```



```
ggplot(nfl_receivers, aes(x = cone, y = draft_ovr)) +
  geom_point() +
  labs(title = "3 Cone Drill vs Draft Pick for Wide Receivers")
```



```r
# Correlation Analysis
cor_forty <- cor.test(nfl_receivers$forty, nfl_receivers$draft_ovr)
print(cor_forty)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nfl_receivers$forty and nfl_receivers$draft_ovr
## t = 4.8531, df = 387, p-value = 1.766e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1435048 0.3310627
## sample estimates:
##       cor
## 0.2395171
```

```r
cor_cone <- cor.test(nfl_receivers$cone, nfl_receivers$draft_ovr)
print(cor_cone)
```

```
##
##  Pearson's product-moment correlation
##
## data:  nfl_receivers$cone and nfl_receivers$draft_ovr
## t = 1.5308, df = 387, p-value = 0.1266
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.02201768  0.17565698
## sample estimates:
```

```
##        cor
## 0.07758208
```

```
# Linear Regression Analysis
lm_forty <- lm(draft_ovr ~ forty, data = nfl_receivers)
lm_cone <- lm(draft_ovr ~ cone, data = nfl_receivers)
summary(lm_forty)
```

```
##
## Call:
## lm(formula = draft_ovr ~ forty, data = nfl_receivers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -147.808  -56.232   -6.779   55.768  158.262
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -707.23     171.94  -4.113 4.77e-05 ***
## forty         186.30      38.39   4.853 1.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 68.73 on 387 degrees of freedom
## Multiple R-squared:  0.05737,    Adjusted R-squared:  0.05493
## F-statistic: 23.55 on 1 and 387 DF,  p-value: 1.766e-06
```

```
summary(lm_cone)
```

```
##
## Call:
## lm(formula = draft_ovr ~ cone, data = nfl_receivers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -123.862  -56.473   -3.695   59.084  141.139
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -65.82     126.03  -0.522    0.602
## cone           27.78      18.15   1.531    0.127
##
## Residual standard error: 70.57 on 387 degrees of freedom
## Multiple R-squared:  0.006019,   Adjusted R-squared:  0.003451
## F-statistic: 2.343 on 1 and 387 DF,  p-value: 0.1266
```

```
lm_forty_conf <- confint(lm_forty, level = 0.95)
print(lm_forty_conf)
```

```
##                  2.5 %    97.5 %
## (Intercept) -1045.2887 -369.1805
## forty         110.8282  261.7809
```

```
lm_cone_conf <- confint(lm_cone, level=0.95)
print(lm_cone_conf)
```

```
##                   2.5 %    97.5 %
## (Intercept) -313.615585 181.97898
## cone          -7.898915  63.45825
```

```r
# Team Comparison
team_comparison <- nfl_receivers %>%
  group_by(draft_team) %>%
  summarize(avg_forty = mean(forty, na.rm = TRUE),
            avg_cone = mean(cone, na.rm = TRUE),
            avg_draft_ovr = mean(draft_ovr, na.rm = TRUE))

print(team_comparison)
```

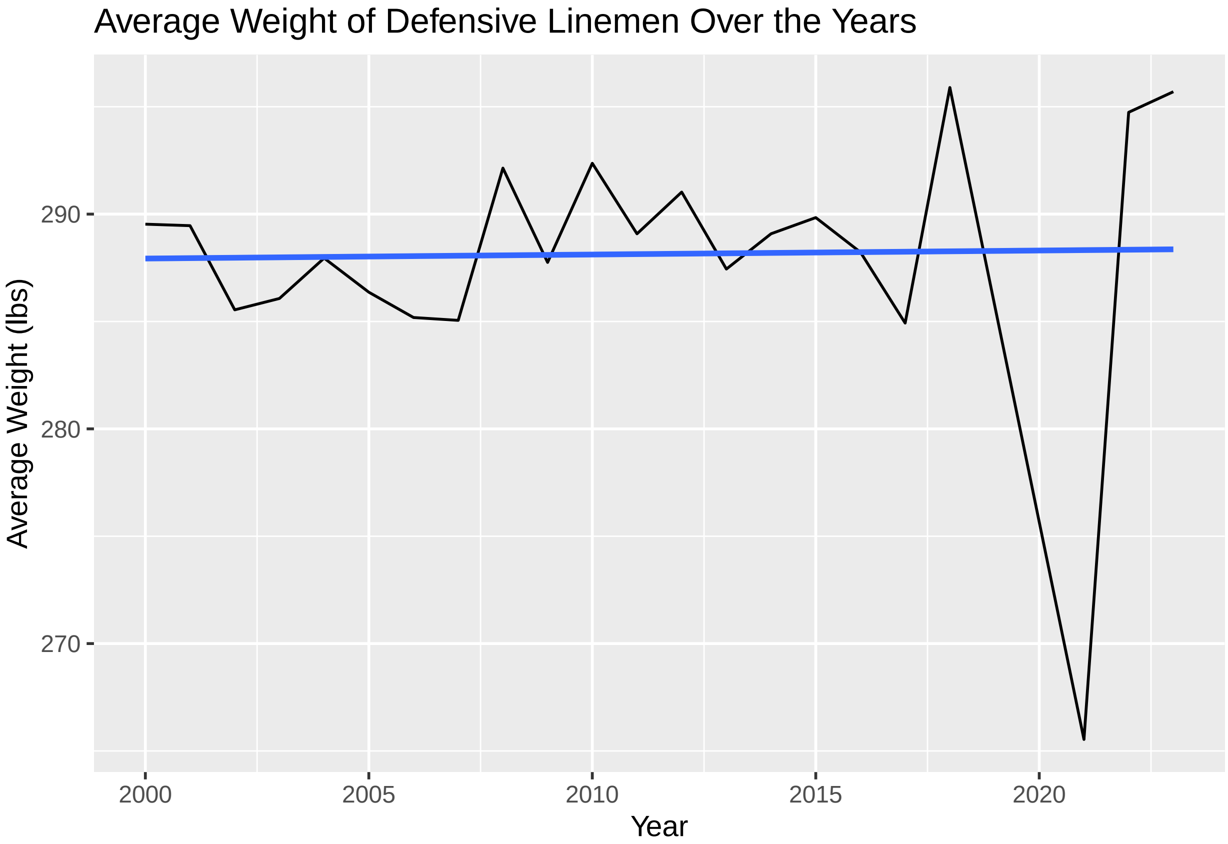
```
## # A tibble: 36 x 4
##    draft_team         avg_forty avg_cone avg_draft_ovr
##    <chr>                  <dbl>    <dbl>         <dbl>
##  1 Arizona Cardinals       4.43     6.98         128.
##  2 Atlanta Falcons         4.49     6.90         127.
##  3 Baltimore Ravens        4.49     6.96         140.
##  4 Buffalo Bills           4.47     6.98         101.
##  5 Carolina Panthers       4.48     7.02         113.
##  6 Chicago Bears           4.51     7.04         134.
##  7 Cincinnati Bengals      4.49     6.95         118
##  8 Cleveland Browns        4.49     6.92          88.6
##  9 Dallas Cowboys          4.52     6.90         148.
## 10 Denver Broncos          4.45     6.91         129.
## # i 26 more rows
```

```r
# Filter the data for defensive linemen (DL)
dl_data <- nfl_data %>%
  filter(pos %in% c("DT", "DE"), !is.na(wt), !is.na(draft_year))

# Calculate average weight per year
avg_wt_per_year <- dl_data %>%
  group_by(draft_year) %>%
  summarize(avg_weight = mean(wt, na.rm = TRUE))
print(avg_wt_per_year)
```

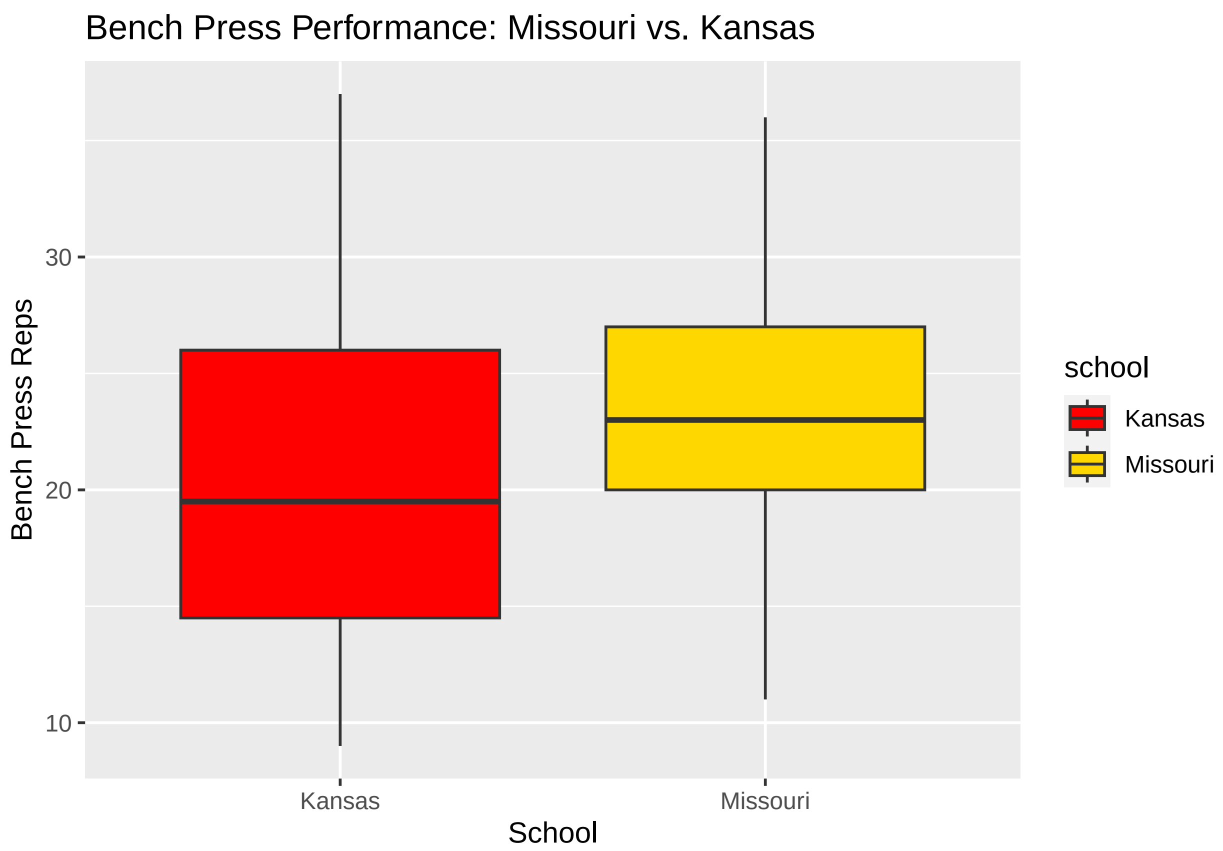
```
## # A tibble: 22 x 2
##    draft_year avg_weight
##         <int>      <dbl>
##  1       2000       290.
##  2       2001       289.
##  3       2002       286.
##  4       2003       286.
##  5       2004       288.
##  6       2005       286.
##  7       2006       285.
##  8       2007       285.
##  9       2008       292.
## 10       2009       288.
## # i 12 more rows
```

```r
# Plot the trend of average weight over the years
ggplot(avg_wt_per_year, aes(x = draft_year, y = avg_weight)) +
  geom_line() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Average Weight of Defensive Linemen Over the Years",
       x = "Year",
       y = "Average Weight (lbs)")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# Filter the data for players from Mizzou and Kansas
strength_data <- nfl_data %>%
  filter(school %in% c("Missouri", "Kansas"), !is.na(bench))

# Create boxplots to compare the bench press performance
ggplot(strength_data, aes(x = school, y = bench, fill = school)) +
  geom_boxplot() +
  labs(title = "Bench Press Performance: Missouri vs. Kansas",
       x = "School",
       y = "Bench Press Reps") +
  scale_fill_manual(values = c("Missouri" = "gold", "Kansas" = "red"))
```



```r
# Perform a t-test or non-parametric test to compare bench press performance
mizzou_strength <- strength_data %>%
  filter(school == "Missouri") %>%
  pull(bench)

kansas_strength <- strength_data %>%
  filter(school == "Kansas") %>%
  pull(bench)

# Use t-test or Wilcoxon test based on data distribution
t_test_result <- t.test(mizzou_strength, kansas_strength, alternative = "greater")
wilcox_test_result <- wilcox.test(mizzou_strength, kansas_strength, alternative = "greater")
```

```
## Warning in wilcox.test.default(mizzou_strength, kansas_strength, alternative =
## "greater"): cannot compute exact p-value with ties
```

```r
# Print results
print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  mizzou_strength and kansas_strength
## t = 1.3023, df = 21.857, p-value = 0.1032
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  -1.008117       Inf
## sample estimates:
```

```
## mean of x mean of y
##  23.51852  20.35714
```

```
print(wilcox_test_result)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  mizzou_strength and kansas_strength
## W = 236, p-value = 0.1002
## alternative hypothesis: true location shift is greater than 0
```