EDA Check-in

Mars, Juniper, Sarah

```
library(ggplot2)
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
```

```
library(mltools)
library(data.table)
```

```
Attaching package: 'data.table'

The following objects are masked from 'package:dplyr':

    between, first, last
```

Proof of being able to import dataset

We imported the dataset below, but it does use a direct file path which we plan to change in the future.

```r
pokemon <- read.csv("/Users/sarah/Desktop/SDS291/FinalProject/pokemon.csv")
```

```r
pokemon$'capture_rate' = as.numeric(pokemon$'capture_rate')
```

```
Warning: NAs introduced by coercion
```

Dataset

We are planning to use the Complete Pokemon Dataset that has information on different Pokemon up to Gen 7. The link where we got the dataset is included below. Dataset Link

Research Question

How do different Pokemon's base stats influence capture rate?

Different Stats

- attack
- base_happiness
- base_egg_steps
- base_total
- defense
- hp
- sp_attack
- sp_defense
- speed

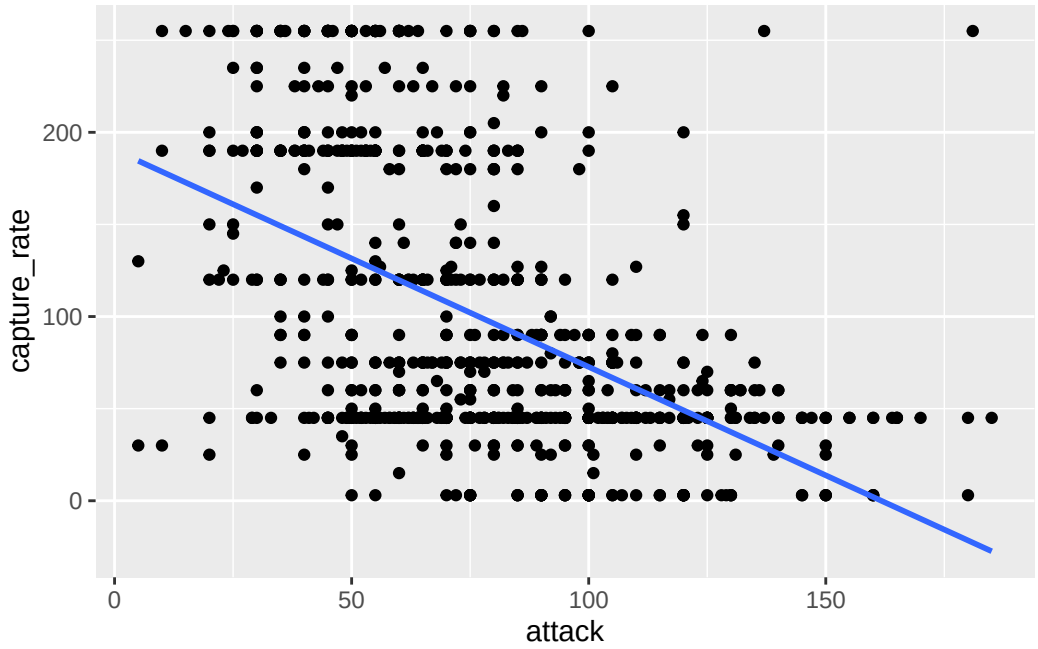
The base stats I'll focus on are attack, hp, defense, and speed. I'm not sure what are important base stats for Pokemon but I'm guessing.

Visualizations

```r
ggplot(data = pokemon, mapping = aes(x = attack, y = capture_rate)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, formula = y~x)
```

```
Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

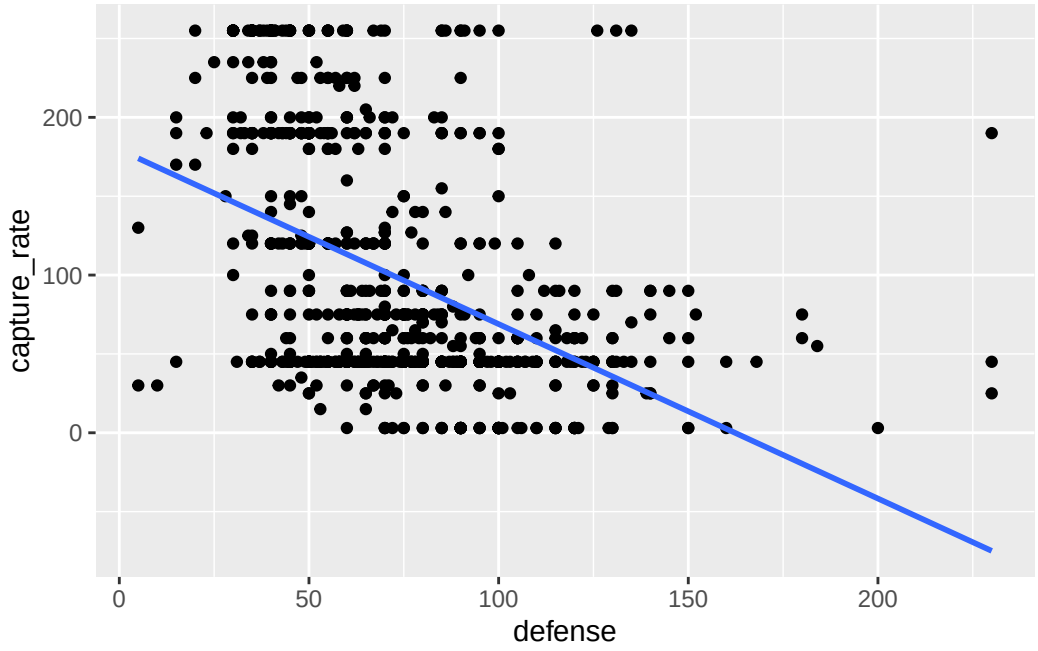
```
Warning: Removed 1 rows containing missing values (geom_point).
```




```
ggplot(data = pokemon, mapping = aes(x = defense, y = capture_rate)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, formula = y~x)
```

```
Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

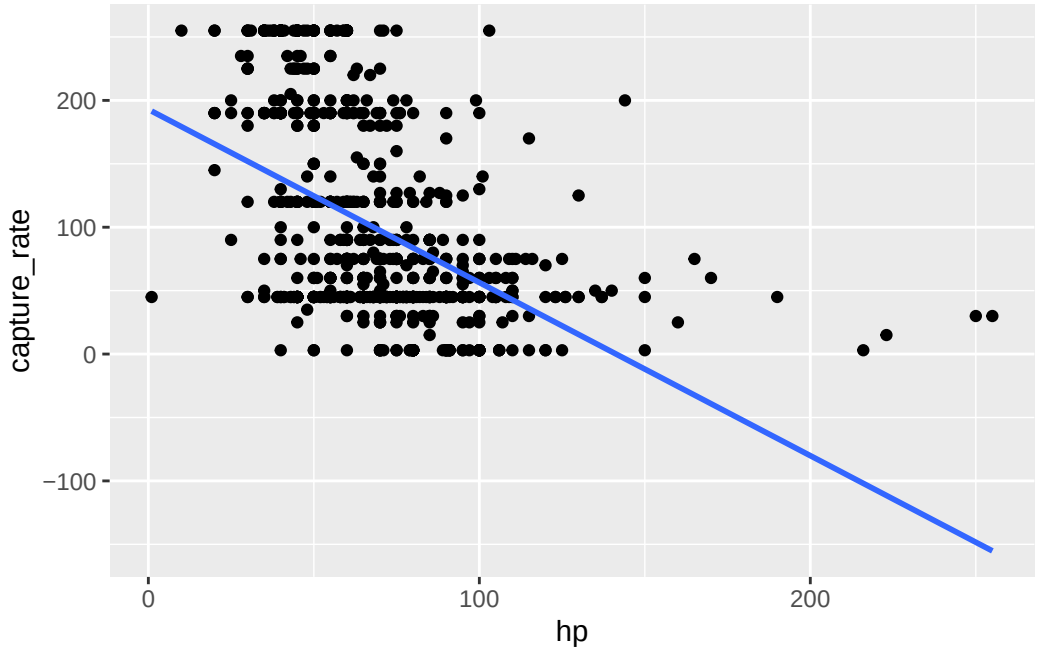
```
Warning: Removed 1 rows containing missing values (geom_point).
```




```
ggplot(data = pokemon, mapping = aes(x = hp, y = capture_rate)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, formula = y~x)
```

```
Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

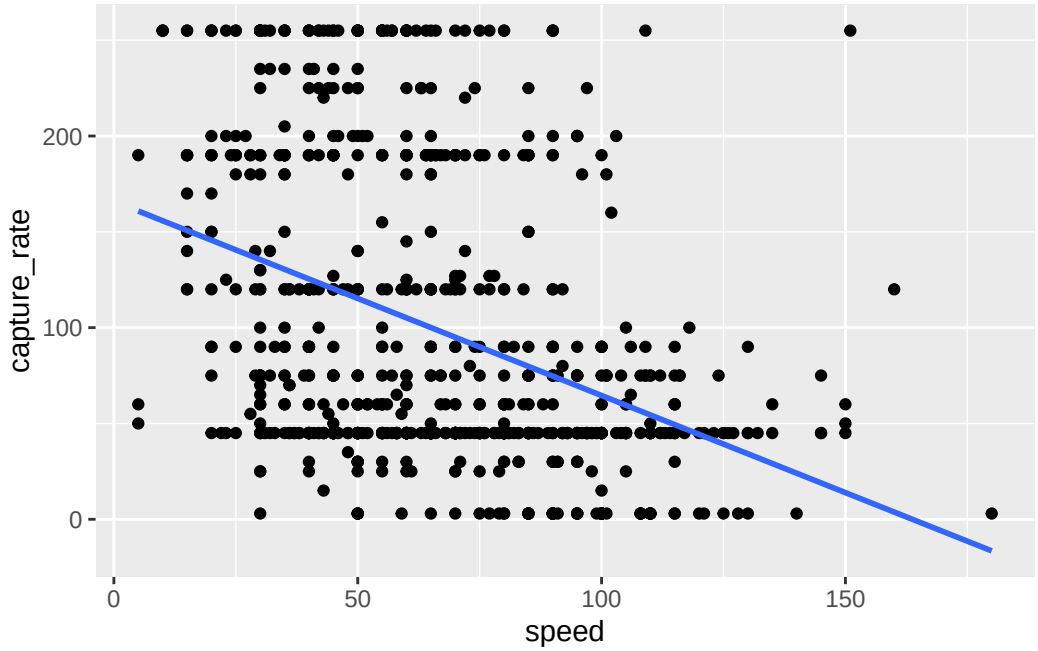
```
Warning: Removed 1 rows containing missing values (geom_point).
```




```
ggplot(data = pokemon, mapping = aes(x = speed, y = capture_rate)) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE, formula = y~x)
```

```
Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

```
Warning: Removed 1 rows containing missing values (geom_point).
```



Comparing possible models

Additive Model

```
additive_capture_model2 <- lm(capture_rate ~ attack + defense + hp + speed, data = pokemon
```

```
summary(additive_capture_model2)
```

```
Call:
lm(formula = capture_rate ~ attack + defense + hp + speed, data = pokemon)

Residuals:
     Min       1Q   Median       3Q      Max
-150.832  -36.856   -5.036   36.345  255.371

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 288.48251    7.67680  37.578  < 2e-16 ***
attack       -0.29466    0.08042  -3.664 0.000265 ***
```

```
defense      -0.77384    0.07492 -10.329  < 2e-16 ***
hp           -0.86774    0.08251 -10.517  < 2e-16 ***
speed        -0.76101    0.07525 -10.113  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 56.4 on 795 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.4558,    Adjusted R-squared:  0.453
F-statistic: 166.4 on 4 and 795 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.453

Interactive Model

```r
interact_capture_model2 <- lm(capture_rate ~ attack * defense * hp * speed, data = pokemon
```

```r
summary(interact_capture_model2)
```

```
Call:
lm(formula = capture_rate ~ attack * defense * hp * speed, data = pokemon)

Residuals:
     Min       1Q   Median       3Q      Max
-167.956  -31.949   -1.533   28.992  219.782

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)             2.615e+02  5.582e+01   4.686 3.29e-06 ***
attack                 -7.279e-02  9.924e-01  -0.073  0.94154
defense                 1.677e-01  5.320e-01   0.315  0.75270
hp                     -7.348e-02  8.772e-01  -0.084  0.93326
speed                   1.101e+00  9.999e-01   1.101  0.27102
attack:defense         -1.027e-02  1.017e-02  -1.010  0.31278
attack:hp              -1.235e-02  1.323e-02  -0.933  0.35101
defense:hp             -1.291e-02  1.015e-02  -1.273  0.20355
attack:speed           -2.156e-02  1.591e-02  -1.355  0.17572
defense:speed          -3.126e-02  1.115e-02  -2.803  0.00519 **
hp:speed               -2.085e-02  1.636e-02  -1.274  0.20303
```

```
attack:defense:hp        1.798e-04  1.415e-04   1.270  0.20436
attack:defense:speed     3.228e-04  1.606e-04   2.010  0.04482 *
attack:hp:speed          2.942e-04  2.235e-04   1.316  0.18844
defense:hp:speed         2.655e-04  1.861e-04   1.426  0.15422
attack:defense:hp:speed -3.217e-06  2.283e-06  -1.409  0.15925
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 55.09 on 784 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.488, Adjusted R-squared:  0.4782
F-statistic: 49.81 on 15 and 784 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.4782

Nested F-test

```
anova(additive_capture_model2, interact_capture_model2)
```

```
Analysis of Variance Table

Model 1: capture_rate ~ attack + defense + hp + speed
Model 2: capture_rate ~ attack * defense * hp * speed
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1    795 2528696
2    784 2379127 11    149569 4.4807 1.385e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value is below 0.05 implying that the change from additive to interactive is necessary, but I feel like the adjusted r-square doesn't justify the change.

Opinion: Would be interesting to look at but haven't tested removing some explanatory variables to see how it affects the model.

Model we plan to use

We plan to use an interaction model to predict the capture rate of pokemon (outcome variable). We plan to use the base stats of the Pokemon as our explanatory variables. We are currently

looking into attack, hp, defense, and speed, but we are thinking of adding base_total as a possible explanatory variable and also seeing if we can add our own variable that says if the Pokemon is a dual type, but are having trouble with creating the variable at the moment.