DoTA2

Daavid Stein

Radiant wins more often?

In our Data Munging Project, our group found that Radiant wins more often than dire. We found this out by doing a simple barchart, but we didn't actually get the exact proportions. So let's find out exactly how often Radiant wins.

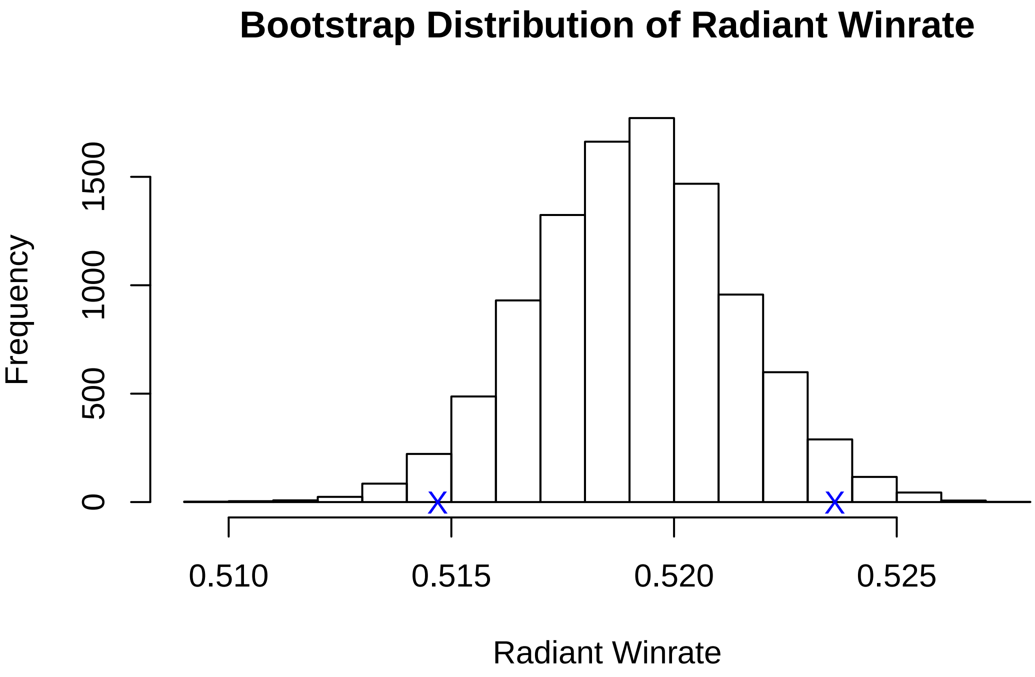
```
sum(match_trim$radiant_win)/dim(match_trim)[1]
```

```
## [1] 0.5191494
```

So Radiant wins about 52% of the time. This means that Dire wins 100 - 0.52 = 48% of the time. Let's see if this is significant by constructing the bootstrap distribution for Radiant's winrate.

```
## Joining, by = "match_id"
```

```
##      2.5%     97.5%
## 0.5146908 0.5236085
```



```
##         V1
##  Min.   :0.5097
##  1st Qu.:0.5176
##  Median :0.5191
##  Mean   :0.5191
##  3rd Qu.:0.5206
##  Max.   :0.5277
```

Our 95% bootstrap confidence inerval is (0.515 0.524). The bootstrap distribution is centered 51.9%. The lower bound of the distribution is 50.9%. So we should feel confident that Radiant wins more often than Dire.

Since the data is from a randomized experiment and we have more than 15 successes and failures, we can find a confidence interval for the population proportion p using the following formula:

$$\hat{p} \pm 1.96 \cdot \frac{\sqrt{\hat{p}(1-\hat{p})}}{\sqrt{n}}$$

```
## [1] 0.5155614 0.5244386
```

We are 95% confident that the true population winrate for Radiant lies between 0.516 and 0.524. This is very close to the 95% confidence interval we got from the bootstrap distribution (0.515, 0.524)

We can also construct the contingency table and ratio of proportions:

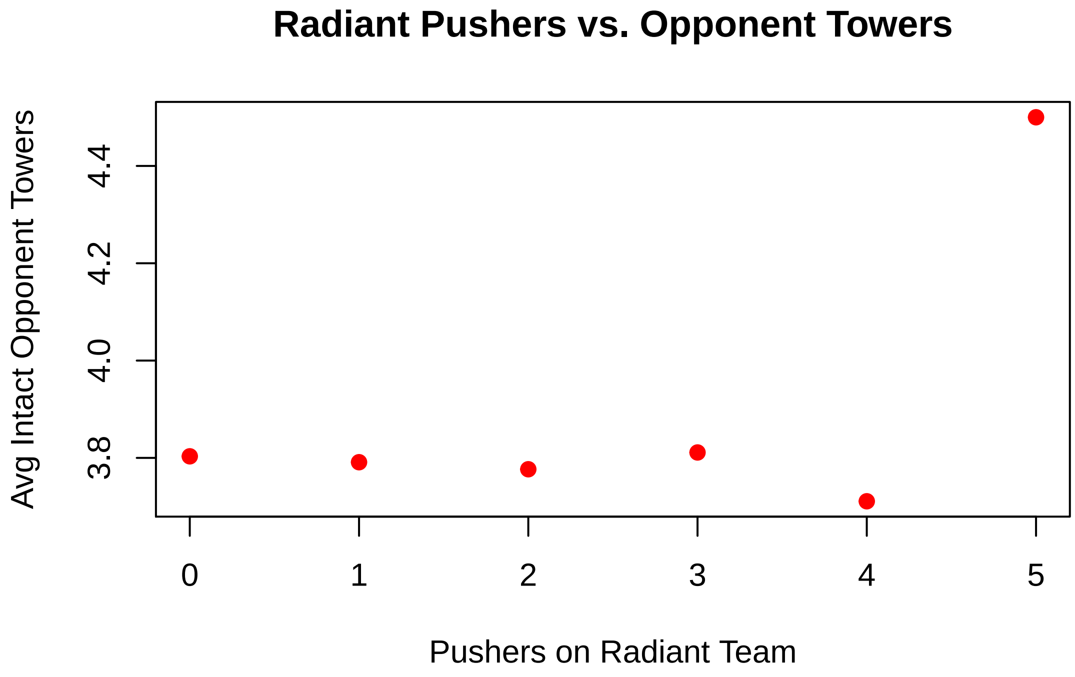
```
##           winner
## Team        Dire Win Radiant Win
##   Dire     0.4808506   0.5191494
##   Radiant 0.4808506   0.5191494
```

```
## [1] 1.079648
```

In our sample, Radiant is 8 percent more likely to win than Dire. These results are extremely surprising, since the only difference between the teams is where they start on the map! This should indicate to developers that to achieve balance, either starting locations should be randomized or players should not be able to pick their team.

Players can pick different heroes with different roles. Different hero roles are better at different things. One role is called a "pusher." A pusher's job is to "push" lanes and destroy enemy structures like towers and barracks. We could guess then, that the number of pushers on a team might be negatively associated with the number of enemy structures still standing at the end of the game. Let's find out if this is indeed the case.

In order to get a sensible scatterplot and correlation coefficient that makes sense we need to get the AVERAGE tower_status per num_pushers.



There definitely does appear to be a negative association between the number of pushers on a Radiant team [this was much clearer from the plot until something happened last night and my aggregate data changed somehow] and the number of towers still standing on the opponent's (Dire) team - at least until the number of pushers is 5. This does make a certain amount of sense, because a role-homogeneous team is not healthy - you probably want at least one "hard" carry. (a different role). Furthermore, there actually isn't much data on teams with 5 pushers:

```
avg_tower_dire[6, ]
```

```
## # A tibble: 1 x 3
##   num_pushers mean_tower_status_dire     n
##         <int>                  <dbl> <int>
## 1           5                    4.5     4
```

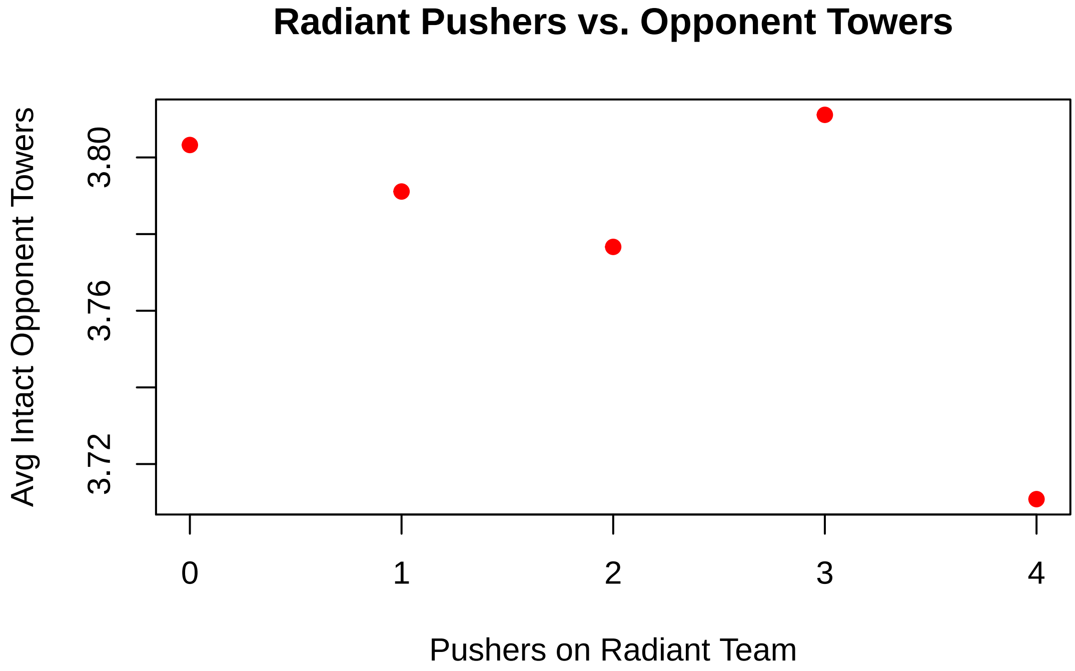
There are actually only 4 such teams in the dataset. So let's remove teams with 5 pushers before we check the correlation coefficient.

```
avg_tower_dire_trim <- avg_tower_dire[1:5, ]
```

```
stats::cor(avg_tower_dire_trim$num_pushers, avg_tower_dire_trim$mean_tower_status_dire)
```

```
## [1] -0.6506111
```

```
plot(avg_tower_dire_trim$num_pushers, avg_tower_dire_trim$mean_tower_status_dire,
    col = "red", pch = 19, xlab = "Pushers on Radiant Team", ylab = "Avg Intact Opponent Towers",
    main = "Radiant Pushers vs. Opponent Towers")
```



We see that there is a strong negative correlation between pushers and opponents' towers.

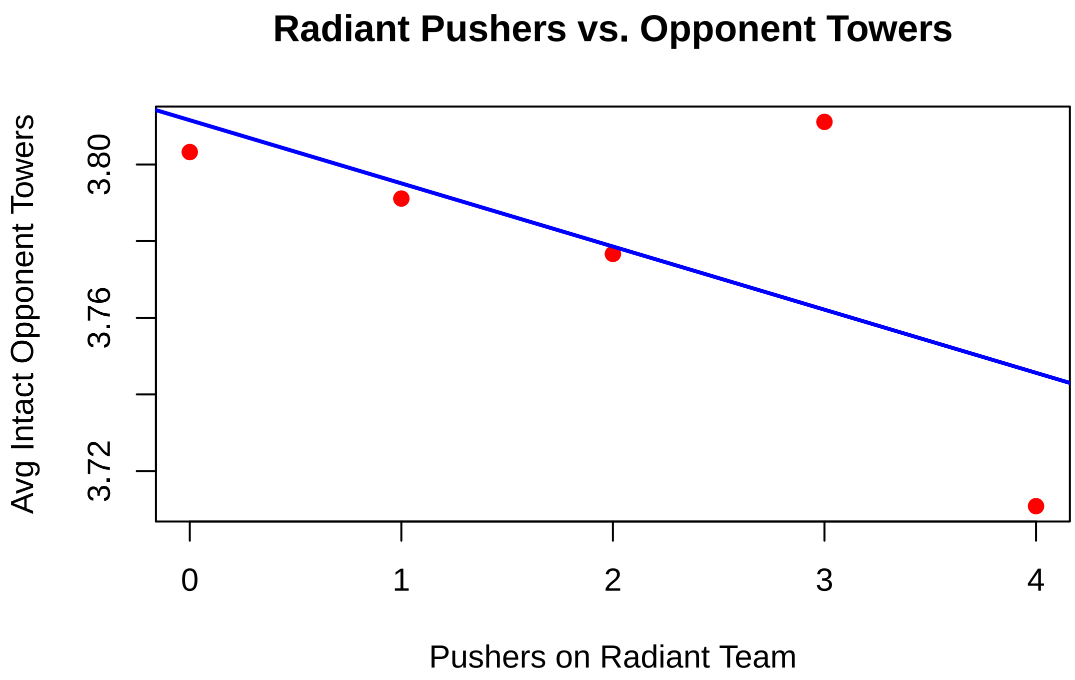
Now let's fit a model.

```
fit <- lm(mean_tower_status_dire ~ num_pushers, data = avg_tower_dire_trim)
coefficients(fit)
```

```
## (Intercept) num_pushers
##  3.81154961 -0.01647891
```

Interpretation of Slope: For every pusher on the Radiant team, we predict the average # of Dire towers remaining will be decreased by 0.02.

```
plot(avg_tower_dire_trim$num_pushers, avg_tower_dire_trim$mean_tower_status_dire,
    col = "red", pch = 19, xlab = "Pushers on Radiant Team", ylab = "Avg Intact Opponent Towers",
    main = "Radiant Pushers vs. Opponent Towers")
abline(fit, lwd = 2, col = "blue")
```



R^2

```
stats::cor(avg_tower_dire_trim$num_pushers, avg_tower_dire_trim$mean_tower_status_dire)^2
```

```
## [1] 0.4232948
```

Looking at r^2, however, we see that only 42% of the variability in average Dire tower status can be explained by pushers. From this I would conclude that doing a multiple linear regression is indicated, possibly with # of carries as a second variable with an interaction term.