The Bootstrap

Paired Warmup (call this `activity-3.Rmd`)

Using the `Default` data set in the `ISLR` package, you'll be estimating the 2, 5, and 10 fold CV MSE for a logistic regression model predicting `default` based on `balance`. Start by loading relevant packages and running following code:

```
set.seed(42)
k <- 2
partition_index <- rep(1:k, each = nrow(Default)/k) %>%
  sample()
MSE_i <- rep(NA, k)
```

Use the following scaffold to computer to compute three MSE estimates.

```
# add partition index column to Default data set
for (i in 1:k) {
  # create training data set
  # create test data set
  # fit model
  # use model to predict into test data set
  # store MSE_i
}
# compute final MSE estimate
```


Start Pulling!

The Bootstrap

A widely applicable and powerful statistical tool used to quantify the uncertainty of a given estimate or model.

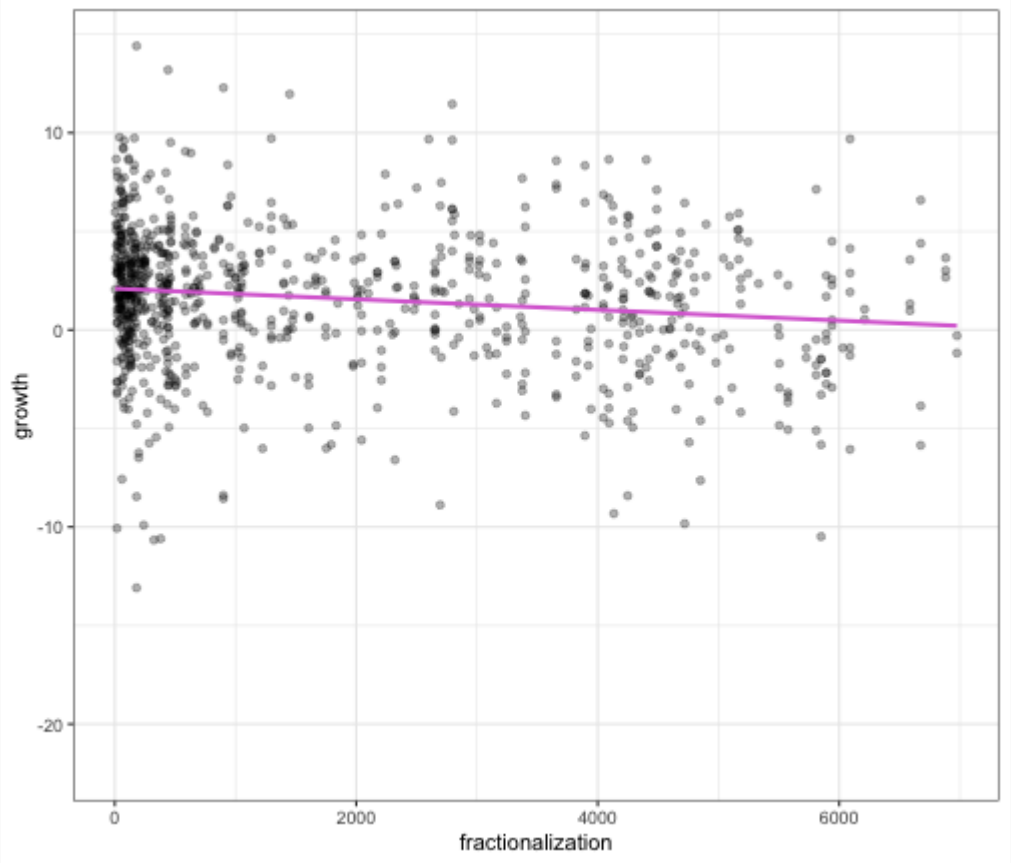
Basic Idea

With a dataset of n obs to which you've fit an estimate $\hat{\theta}$.

1. Draw a bootstrap sample, of size n **with replacement**.
2. Fit your estimate, $\hat{\theta}^*$ to the bootstrap sample.
3. Repeat 1-2 many times and assess the variability in your estimate by looking at the *bootstrap distribution*.

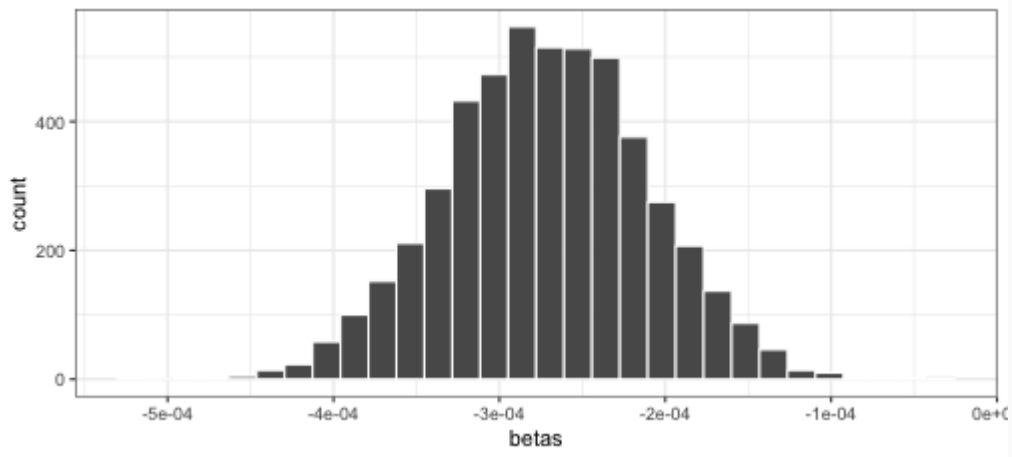
Ex: Simple Regression

Is there a relationship between fractionalization and growth?



Bootstrapping $\hat{\beta}_1$

```
betas <- rep(NA, 5000)
for(i in 1:5000) {
  boot_ind <- sample(1:nrow(war),
                     size = nrow(war),
                     replace = TRUE)
  war_boot <- war[boot_ind, ]
  betas[i] <- coef(lm(growth ~ fractionalization,
                      data = war_boot))[2]
}
```



Bootstrap distribution

```
mean(betas)
```

```
## [1] -0.0002717769
```

```
sd(betas)
```

```
## [1] 6.088999e-05
```

```
summary(m1)$coef
```

```
##                              Estimate   Std. Error   t value
## (Intercept)            2.0964034623 1.592597e-01 13.163424
## fractionalization -0.0002704172 5.956005e-05 -4.540245
```

A common argument

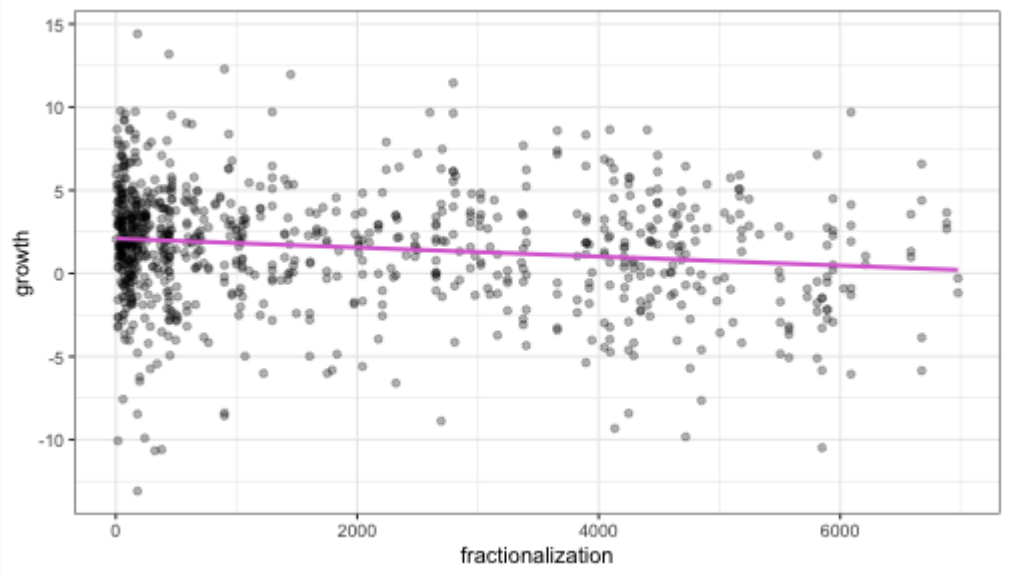
Parametric methods have assumptions that often aren't reasonable, therefore the bootstrap is preferable because it's assumption free.

The bootstrap requires a sample that captures the important structure in the data. Difficult with small samples of skewed data.

But it sure is flexible . . .

Bootstrapping r

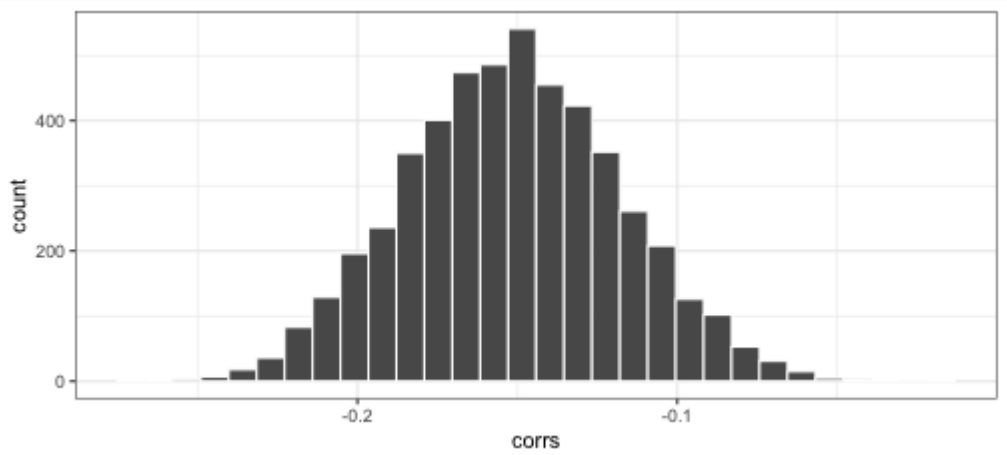
Is there a relationship between fractionalization and growth in terms of the **correlation coefficient**?



r = -0.151

Bootstrapping r

```
corrs <- rep(NA, 5000)
for(i in 1:5000) {
  boot_ind <- sample(1:nrow(war),
                     size = nrow(war),
                     replace = TRUE)
  war_boot <- war[boot_ind, ]
  corrs[i] <- cor(war_boot$fractionalization,
                  war_boot$growth)
}
```



Bootstrapping v. CV

Both are computationally intensive methods that involve sampling from your data set to learn more about your estimate/model.

Cross-validation

Often used for *model assessment* and *model selection*.

- Partition data into test and train
- Fit model to train, predict on test
- Iterate though all possible *folds* (not for VS)
- Compute aggregate measure of predictive ability

Bootstrapping

Often used for quantifying uncertainty.

- Draw a bootstrap sample of size n from your data *with replacement*.
- Compute estimate of interest
- Consider distribution of bootstrap estimates over many samples

Activity 3, continued

Take a look at the `law82` dataset inside the `bootstrap` package.

Compute a statistic of interest and construct the bootstrap distribution to find its standard error.

Activity 3, continued continued

Take a look at the `law82` dataset inside the `bootstrap` package.

1. Fit two models, linear and quadratic, to predict `GPA` based on `LSAT`.
2. Compute the cross-validated MSE for both models using one of the three methods we've discussed.
3. Bootstrap this statistic to estimate its standard error.