0 BOOTSTRAP

Lets simulate a random sample n = 3 from a normal population (mean = 100, sd = 20).

```
set.seed(1)  # makes random number generation reproducible
Sample <- rnorm(n = 3, mean = 100, sd = 20)
Sample
```

```
## [1]  87.47092 103.67287  83.28743
```

```
mean(Sample)
```

```
## [1] 91.47707
```

```
sd(Sample)
```

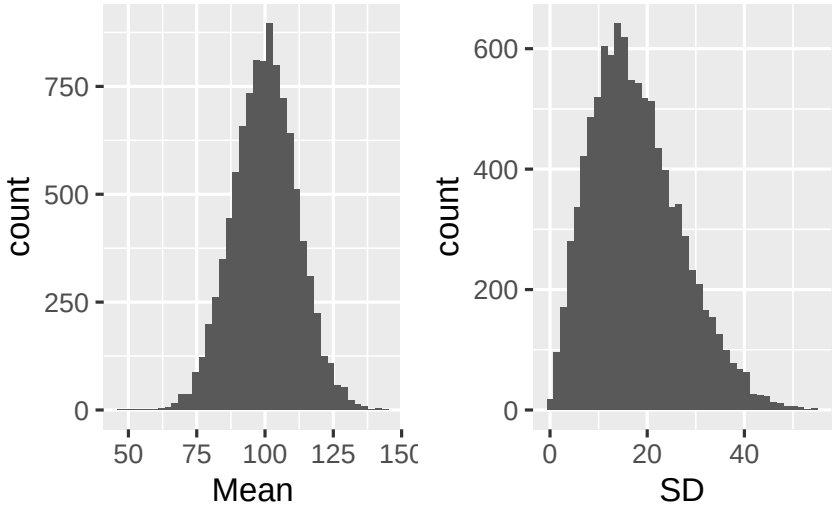
```
## [1] 10.76701
```

Sample mean is ca 10% lower than population mean and sample sd is twice less than pop sd! Of course, we only know this because this is a simulation.

Next we draw 10 000 random samples and calculate 10 000 means and 10 000 sd-s.

```
Summary <- replicate(10000, rnorm(3, 100, 20)) %>% as_tibble() %>% melt() %>% group_by(variable) %>%
    summarise(Mean = mean(value), SD = sd(value))

a <- Summary %>% ggplot(aes(Mean)) + geom_histogram(bins = 40)
b <- Summary %>% ggplot(aes(SD)) + geom_histogram(bins = 40)
grid.arrange(a, b, nrow = 1)
```




```
mean(Summary$Mean)
```

```
## [1] 99.98043
```

```
mean(Summary$SD)
```

```
## [1] 17.76452
```

```
mode <- function(x, adjust = 1) {
    x <- na.omit(x)
    dx <- density(x, adjust = adjust)
    dx$x[which.max(dx$y)]
}
mode(Summary$SD)
```

```
## [1] 14.07554
```

the average of means 10 000 samples is pretty close to population average, but the average of sd-s is not (it tends to be lower because the algorithm for SD calcultion is biased at small samples; if you don't believe me, repeat the simulation using N = 30)

Also note that the distribution of SD-s is asymmetric, and that it is the the mode of the distribution that is arguably the most typical (the most probable) sd value. This value is only 14 (remember, the population value is 20).

Moreover, in addition to underestimated mode we have here a fat tail, which ensures that it is nevertheless quite probable that the sample SD grossly overestimates population SD!

What is the probability that we get a sample, whose SD > 25?

```
mean(Summary$SD > 25)
```

```
## [1] 0.2114
```

Yuk! There is >20% probability for that.

```
mean(Summary$SD < 15)
```

```
## [1] 0.4344
```

And there is >40% Pr of grossly underestimating the population SD, which leaves only a <40% chance of a reasonably accurate estimate. Small samples are simply tragic, and all because the algorithm used to calculate SD sucks!

Note: Here we used a little trick, which is based on R-s internal encoding of logical values. The code `Summary$SD > 25` *produces a vector of 10 000 TRUE/FALSE logical values, and as TRUE is encoded as 1 and FALSE is encoded as 0, taking the mean of this vector gives us the proportion of TRUE-s, meaning the proportion of SD values that are larger than 25. If we instead calculated the* `sum(Summary$SD > 25)`*, we would get the number of TRUE-s/1-s in the vector.*

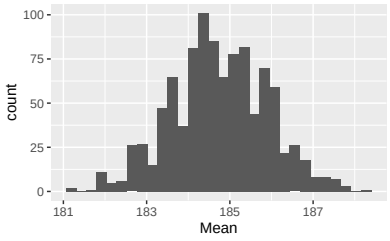
Bootstrap

What if we have a single sample and want to estimate the true population value? Bootstapping is a method that allows to draw extra samples (bootstrap samples) from the existing sample (empirical sample), rather than from the population, like we did in the simulations. As long as the empirical sample reflects the population, this method works surprisingly well and allows to draw realistic error bars for our estimates of the population mean, median, or quantiles (but not min or max). As an added bonus, bootstrapping does not use distributional assumptions, making its implementation both easy and flexible. But this means that when we can in good faith add such assumptions, then bootstrapping results are suboptimal in relation to full Bayesian (or frequentist) modelling.

1. From the empirical sample of size N draw with replacement B bootstrap samples, each with size N.
2. For each of the B bootstrap samples calculate your desired statistic (you will end up with B numbers)
3. From these B numbers draw a histogram or density plot. Using this plot we can ask some pertinent questions.

Example: what is the mean height of US presidents? We have the heights of the last 11 presidents.

```
heights <- tibble(value = c(183, 192, 182, 183, 177, 185, 188, 188, 182, 185, 188))
N <- nrow(heights)  #empirical sample size
B <- 1000  #nr of bootstrap samples
boot1 <- replicate(B, sample_n(heights, size = N, replace = TRUE)) %>%
as.data.frame() %>%
melt() %>%
group_by(variable) %>%
summarise(Mean = mean(value))
ggplot(boot1, aes(Mean)) + geom_histogram()
```



Now we have a distribution of 1000 estimates of the mean height of the US presidents. We assume that the most typical values are closer to the true "population" value than the rarer values in the tails. As this distribution seems to be normal (and there are theoretical reasons to think it should be normal) we can take the mean to characterize its expected value (other possibilities include median and mode). Furthermore, we are justified in working with this distribution as it was a posterior distribution, which conveys our beliefs about the mean height of the presidents (but only insofar as we have

no prior beliefs on this mean value, nor beliefs about the distribution of presidents heights – here a full Bayesian approach is clearly appropriate, but lets pretend for now that we don't care). Anyway, this posterior holds any and all information about the mean presidents height. And to extract it is really easy.

Lets start be asking what is the most likely value of mean heights of US presidents

```
mean(boot1$Mean)
```

```
## [1] 184.7475
```

This is basically the same as taking the empirical sample mean

```
mean(heights$value)
```

```
## [1] 184.8182
```

NB! Bootstrapping does not change the most likely estimate, all it does is quantifying our uncertainty concerning this estimate.

Lets next ask, what is the probability that the true mean height of presidents >186 cm.

```
mean(boot1$Mean > 186)
```

```
## [1] 0.129
```

millise tõenäosusega jääb presidentide keskmine pikkus vahemikku 185-186 cm?

```
(sum(boot1$Mean >= 185) - sum(boot1$Mean > 186))/length(boot1$Mean)
```

```
## [1] 0.297
```

What could be a 92% credible interval for the mean presidents height? Or, eqivalently, into which interval will the true mean fall with 92% probability?

```
quantile(boot1$Mean, probs = c(0.06, 0.96))
```

```
##       6%      96% 
## 182.9091 186.7273
```

These quantiles remove an equal fraction (4%) of estimates from both tails of the posterior distribution, and whats left in the middle is the CI. An arguably better method is to isolate 92% of the highest density region of the posterior (92% of the area under the curve so that at no point outside of this interval is the curve higher than at any point inside the interval). This interval is called HDPI or HDI (Highest Density Probability Interval).

```
posterior_summary(boot1$Mean, prob = c(0.05, 0.9))
```

```
##      Estimate Est.Error       Q5      Q90
## [1,] 184.7475  1.158073 182.7273 186.1818
```

Usually the quantile method and HDI give very similar results. When that does not happen, the full posterior should be examined.

What is the probability that the mean presidents height is larger than the mean height of adult US males (178.3 cm)?

```
mean(boot1$Mean > 178.3)
```

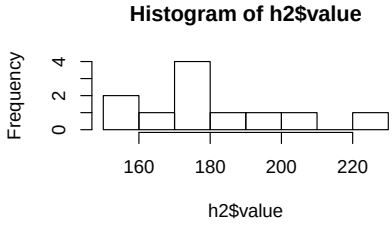
```
## [1] 1
```

This number shows weakness of the bootstrapping approach – based on my prior beliefs there is no way I would say with absolute certainty that the mean height of US presidents is higher than the mean height of US males. Sure, my subjective probability for this is high, but it is definitely not 100%. And I am sure as hell willing to change it based on additional evidence that might come along.

To get from bootstrap to full Bayesianism we must add to the mix 2 models: a data model, which is the likelihood, and a model of our prior belief. If our likelihood is normal distribution, which has 2 parameters (mean and sd), then we need a separate prior for both of them. These prior functions need not be normal (although they can be). Thus the Bayesian solution for normal likelihood gives 2 posterior distributions, one for the mean height and another for the standard deviation at the original data (individual heights) level. We will show, how to do it, later.

lets simulate 11 values from random us male heights

```
h2 <- data.frame(value = rnorm(11, 178.3, 20))
hist(h2$value)
```

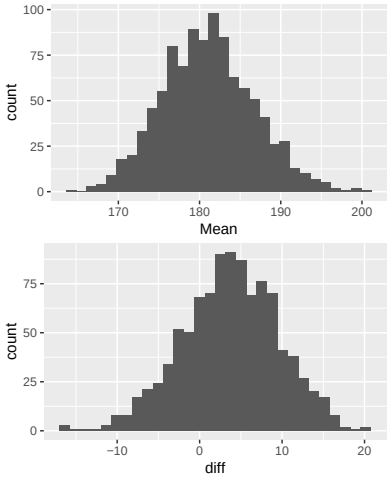



```
mean(h2$value)
```

```
## [1] 181.0954
```

```
N <- nrow(h2)  #empirical sample size
B <- 1000  #nr of bootstrap samples
boot2 <- replicate(B, sample_n(h2, size = N, replace = TRUE)) %>%
as.data.frame() %>%
melt() %>%
group_by(variable) %>%
summarise(Mean = mean(value))
ggplot(boot2, aes(Mean)) + geom_histogram()
```

Now, if these are our best estimates of US presidents mean height and us citizen mean height, what is our best estimate of the difference of presidents mean height from population mean height?




```
diff <- boot1$Mean - boot2$Mean
ggplot(data = NULL, aes(diff)) + geom_histogram()
```

```
posterior_summary(diff)
```

```
##      Estimate Est.Error      Q2.5    Q97.5
## [1,] 3.684254  5.890063 -8.170399 14.97985
```