QCB 508 – Week 7

John D. Storey

Spring 2020

Contents

Parametric and Nonparametric Inference

Parametric Inference

Parametric inference is based on a family of known probability distributions governed by a defined parameter space.

The goal is to perform inference (or more generally statistics) on the values of the parameters.

Nonparametric Inference

Nonparametric inference or modeling can be described in two ways (not mutually exclusive):

1. An inference procedure or model that does not depend on or utilize the parametrized probability distribution from which the data are generated.
2. An inference procedure or model that may have a specific structure or based on a specific formula, but the complexity is adaptive and can grow to arbitrary levels of complexity as the sample size grows.

In *All of Nonparametric Statistics*, Larry Wasserman says:

> ... it is difficult to give a precise definition of nonparametric inference.... For the purposes of this book, we will use the phrase nonparametric inference to refer to a set of modern statistical methods that aim to keep the number of underlying assumptions as weak as possible.

He then lists five estimation examples (see Section 1.1): distributions, functionals, densities, regression curves, and Normal means.

Nonparametric Descriptive Statistics

Exploratory data analysis methods tend to be nonparametric. Why?

Sometimes the exploratory methods are calibrated by known probability distributions, but they are usually informative regardless of the underlying probability distribution (or lack thereof) of the data.

Semiparametric Inference

Semiparametric inference or modeling methods contain both parametric and nonparametric components.

An example is $X_i|\mu_i \sim \text{Normal}(\mu_i, 1)$ and $\mu_i \overset{\text{iid}}{\sim} F$ for some arbitrary distribution F.

Non-likelihood Inference Topics

A range of parametric and nonparametric topics:

- t-distribution
- Goodness of fit
- Exact tests
- Method of moments
- Permutation methods
- Empirical distribution functions
- Bootstrap

The t Distribution

Normal, Unknown Variance

Suppose a sample of n data points is modeled by $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} \text{Normal}(\mu, \sigma^2)$ where σ^2 is *unknown*. Recall that $S = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \overline{X})^2}{n-1}}$ is the sample standard deviation.

The statistic

$$\frac{\overline{X} - \mu}{S/\sqrt{n}}$$

has a t_{n-1} distribution, a t-distribution with $n-1$ degrees of freedom.

Aside: Chi-Square Distribution

Suppose $Z_1, Z_2, \ldots, Z_v \overset{\text{iid}}{\sim} \text{Normal}(0, 1)$. Then $Z_1^2 + Z_2^2 + \cdots + Z_v^2$ has a χ_v^2 distribution, where v is the degrees of freedom.

This χ_v^2 rv has a pdf, expected value equal to v, and variance equal to $2v$.

Also,

$$\frac{(n-1)S^2}{\sigma^2} \sim \chi_{n-1}^2.$$

Theoretical Basis of the t

Suppose that $Z \sim \text{Normal}(0, 1)$, $X \sim \chi_v^2$, and Z and X are independent. Then $\frac{Z}{\sqrt{X/v}}$ has a t_v distribution.

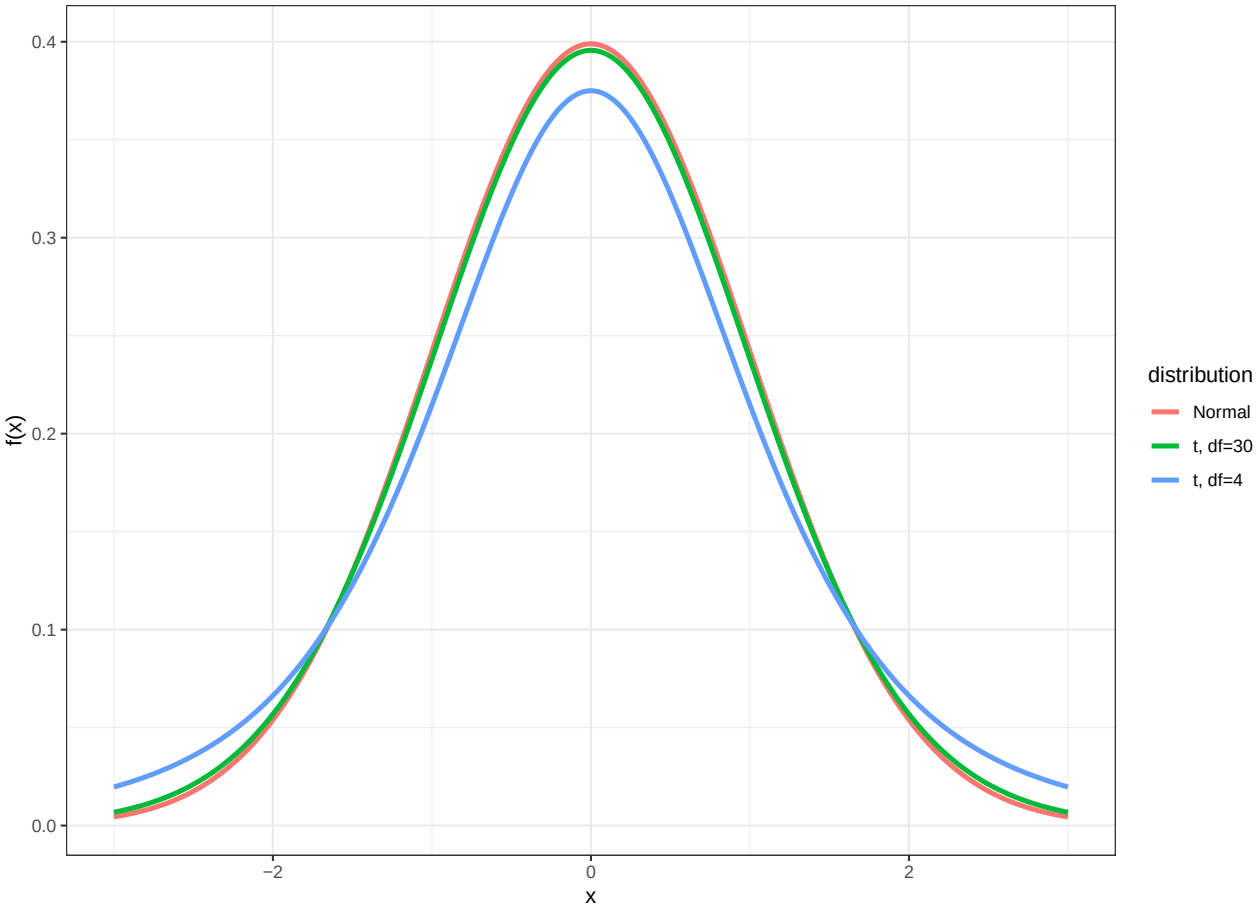
Since $\frac{\overline{X}-\mu}{\sigma/\sqrt{n}} \sim \text{Normal}(0, 1)$ and $\overline{X}$ and S^2 are independent (shown later), it follows that the following has a t_{n-1} distribution:

$$\frac{\overline{X} - \mu}{S/\sqrt{n}}.$$

When Is t Utilized?

- The t distribution and its corresponding CI's and HT's are utilized when the data are Normal (or approximately Normal) and n is small
- Small typically means that $n < 30$
- In this case the inference based on the t distribution will be more accurate
- When $n \geq 30$, there is very little difference between using t-statistics and z-statistics

t vs Normal



t Percentiles

We calculated percentiles of the Normal(0,1) distribution (e.g., z_α). We can do the analogous calculation with the t distribution.

Let t_α be the α percentile of the t distribution. Examples:

```
> qt(0.025, df=4) # alpha = 0.025
[1] -2.776445
> qt(0.05, df=4)
[1] -2.131847
> qt(0.95, df=4)
[1] 2.131847
> qt(0.975, df=4)
[1] 2.776445
```

Confidence Intervals

Here is a $(1-\alpha)$-level CI for μ using this distribution:

$$\left(\hat{\mu} - |t_{\alpha/2}|\frac{s}{\sqrt{n}}, \hat{\mu} + |t_{\alpha/2}|\frac{s}{\sqrt{n}}\right),$$

where as before $\hat{\mu} = \overline{x}$. This produces a wider CI than the z statistic analogue.

Hypothesis Tests

Suppose we want to test $H_0 : \mu = \mu_0$ vs $H_1 : \mu \neq \mu_0$ where μ_0 is a known, given number.

The t-statistic is

$$t = \frac{\hat{\mu} - \mu_0}{\frac{s}{\sqrt{n}}}$$

with p-value

$$\Pr(|T^*| \geq |t|)$$

where $T^* \sim t_{n-1}$.

Two-Sample t-Distribution

Let $X_1, X_2, \ldots, X_{n_1} \overset{\text{iid}}{\sim} \text{Normal}(\mu_1, \sigma_1^2)$ and $Y_1, Y_2, \ldots, Y_{n_2} \overset{\text{iid}}{\sim} \text{Normal}(\mu_2, \sigma_2^2)$ have unequal variances.

We have $\hat{\mu}_1 = \overline{X}$ and $\hat{\mu}_2 = \overline{Y}$. The unequal variance two-sample t-statistic is

$$t = \frac{\hat{\mu}_1 - \hat{\mu}_2 - (\mu_1 - \mu_2)}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}}.$$

Two-Sample t-Distribution

Let $X_1, X_2, \ldots, X_{n_1} \overset{\text{iid}}{\sim} \text{Normal}(\mu_1, \sigma^2)$ and $Y_1, Y_2, \ldots, Y_{n_2} \overset{\text{iid}}{\sim} \text{Normal}(\mu_2, \sigma^2)$ have equal variance.

We have $\hat{\mu}_1 = \overline{X}$ and $\hat{\mu}_2 = \overline{Y}$. The equal variance two-sample t-statistic is

$$t = \frac{\hat{\mu}_1 - \hat{\mu}_2 - (\mu_1 - \mu_2)}{\sqrt{\frac{S^2}{n_1} + \frac{S^2}{n_2}}}.$$

where

$$S^2 = \frac{\sum_{i=1}^{n_1}(X_i - \overline{X})^2 + \sum_{i=1}^{n_2}(Y_i - \overline{Y})^2}{n_1 + n_2 - 2}.$$

Two-Sample t-Distributions

When the two populations have equal variances, the pivotal t-statistic follows a $t_{n_1+n_2-2}$ distribution.

When there are unequal variances, the pivotal t-statistic follows a t distribution where the degrees of freedom comes from an approximation using the Welch–Satterthwaite equation (which R calculates).

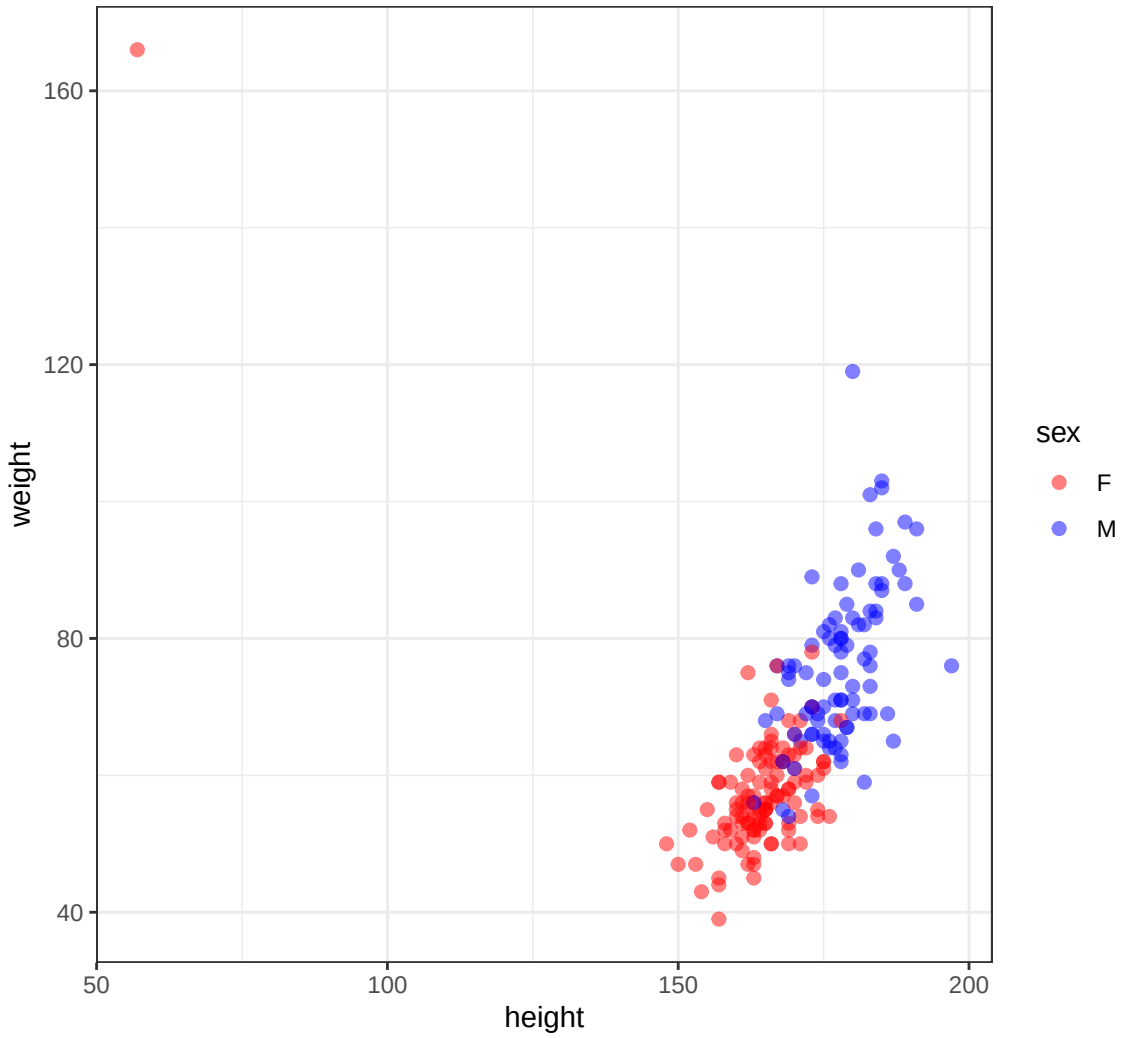
Normal-ish Data: "Davis" Data Set

```
> library("car")
> data("Davis")
```

```
> htwt <- tbl_df(Davis)
> htwt
# A tibble: 200 x 5
   sex   weight height repwt repht
   <fct>  <int>  <int> <int> <int>
 1 M         77    182    77   180
 2 F         58    161    51   159
 3 F         53    161    54   158
 4 M         68    177    70   175
 5 F         59    157    59   155
 6 M         76    170    76   165
 7 M         76    167    77   165
 8 M         69    186    73   180
 9 M         71    178    71   175
10 M         65    171    64   170
# ... with 190 more rows
```

Height vs Weight

```
> ggplot(htwt) +
+ geom_point(aes(x=height, y=weight, color=sex), size=2, alpha=0.5) +
+ scale_colour_manual(values=c("red", "blue"))
```



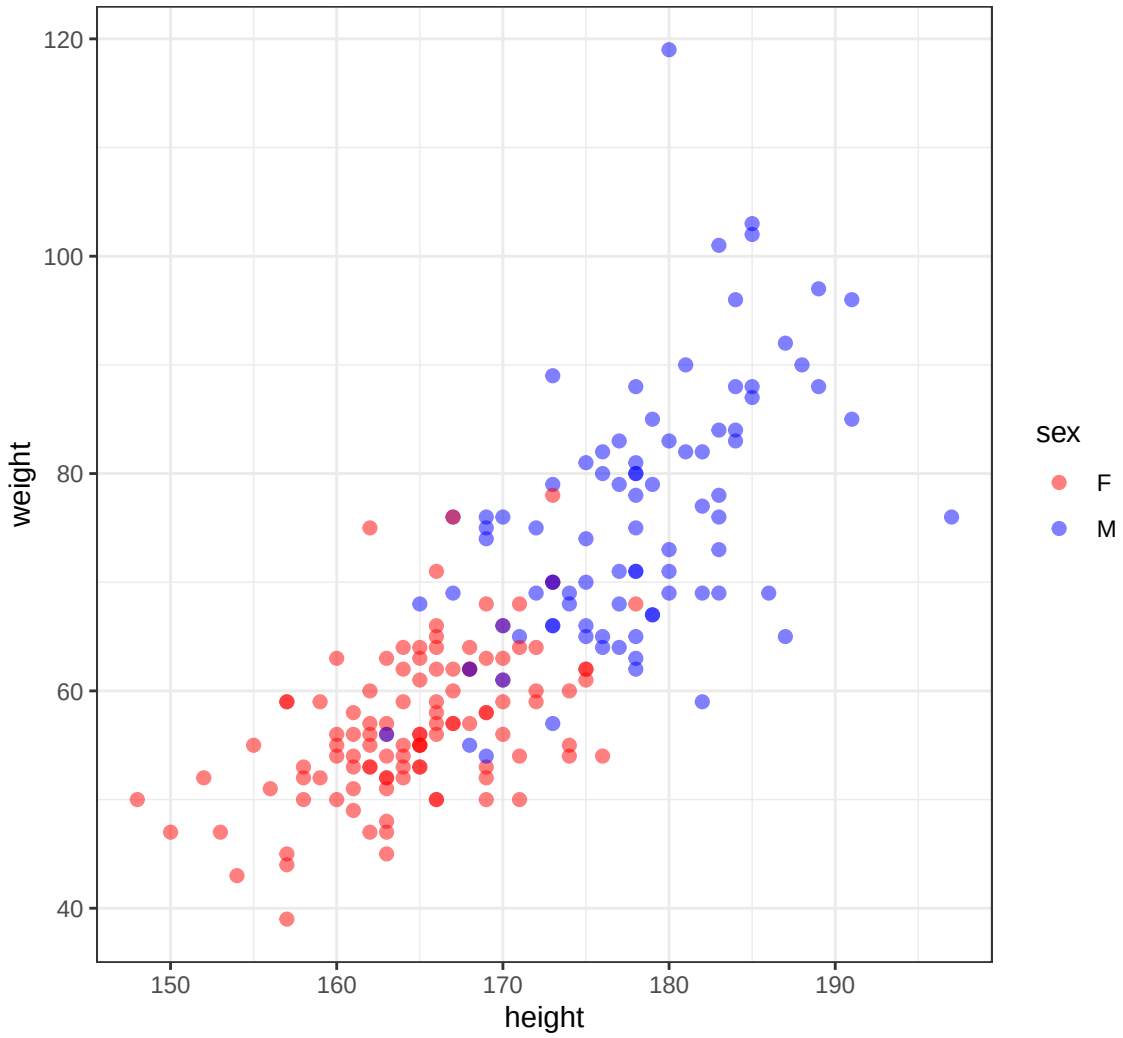
An Error?

```
> which(htwt$height < 100)
[1] 12
> htwt[12,]
# A tibble: 1 x 5
  sex   weight height repwt repht
  <fct>  <int>  <int> <int> <int>
1 F        166     57    56   163
```

```
> htwt[12,c(2,3)] <- htwt[12,c(3,2)]
```

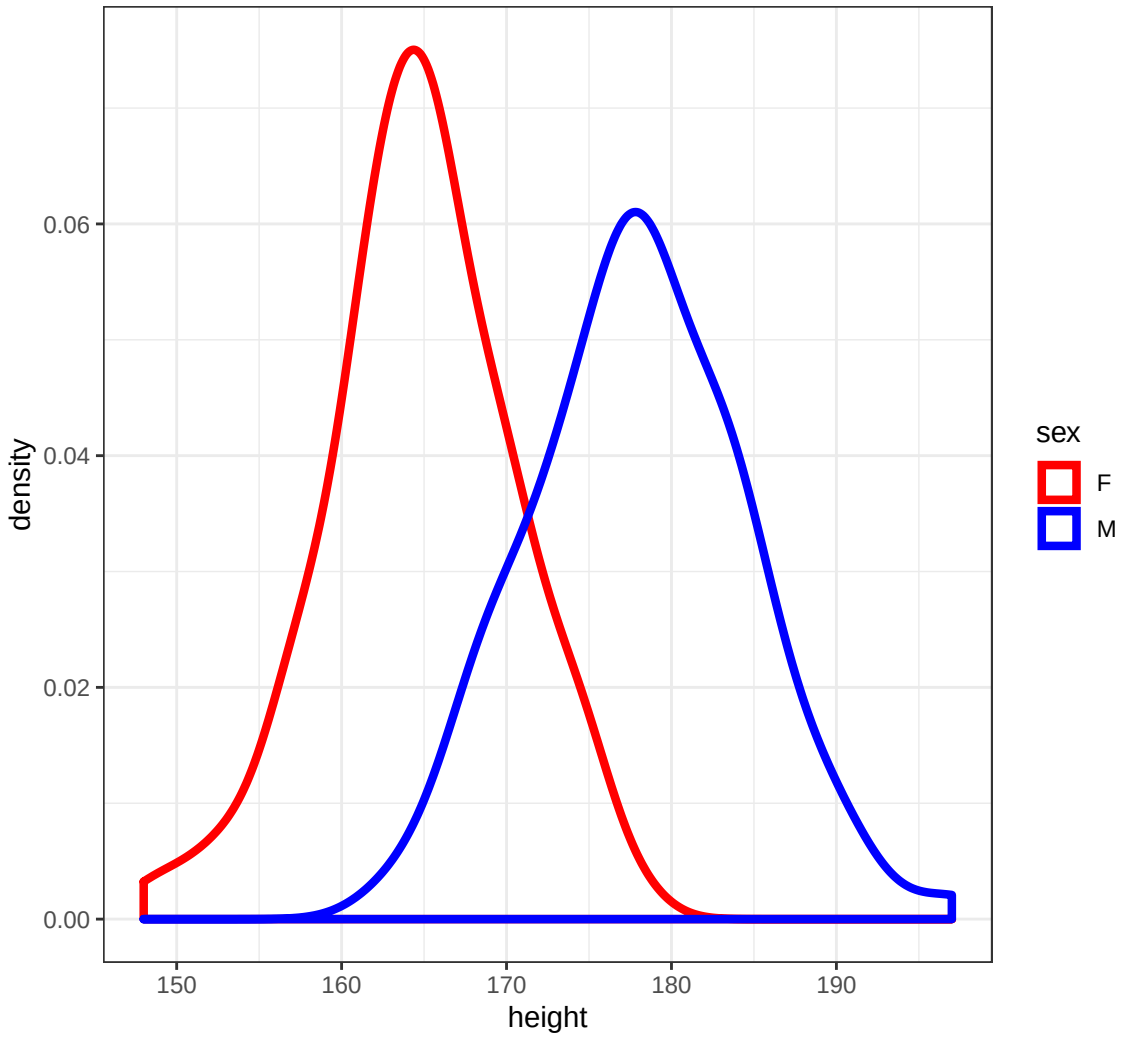
Updated Height vs Weight

```
> ggplot(htwt) +
+   geom_point(aes(x=height, y=weight, color=sex), size=2, alpha=0.5) +
+   scale_color_manual(values=c("red", "blue"))
```



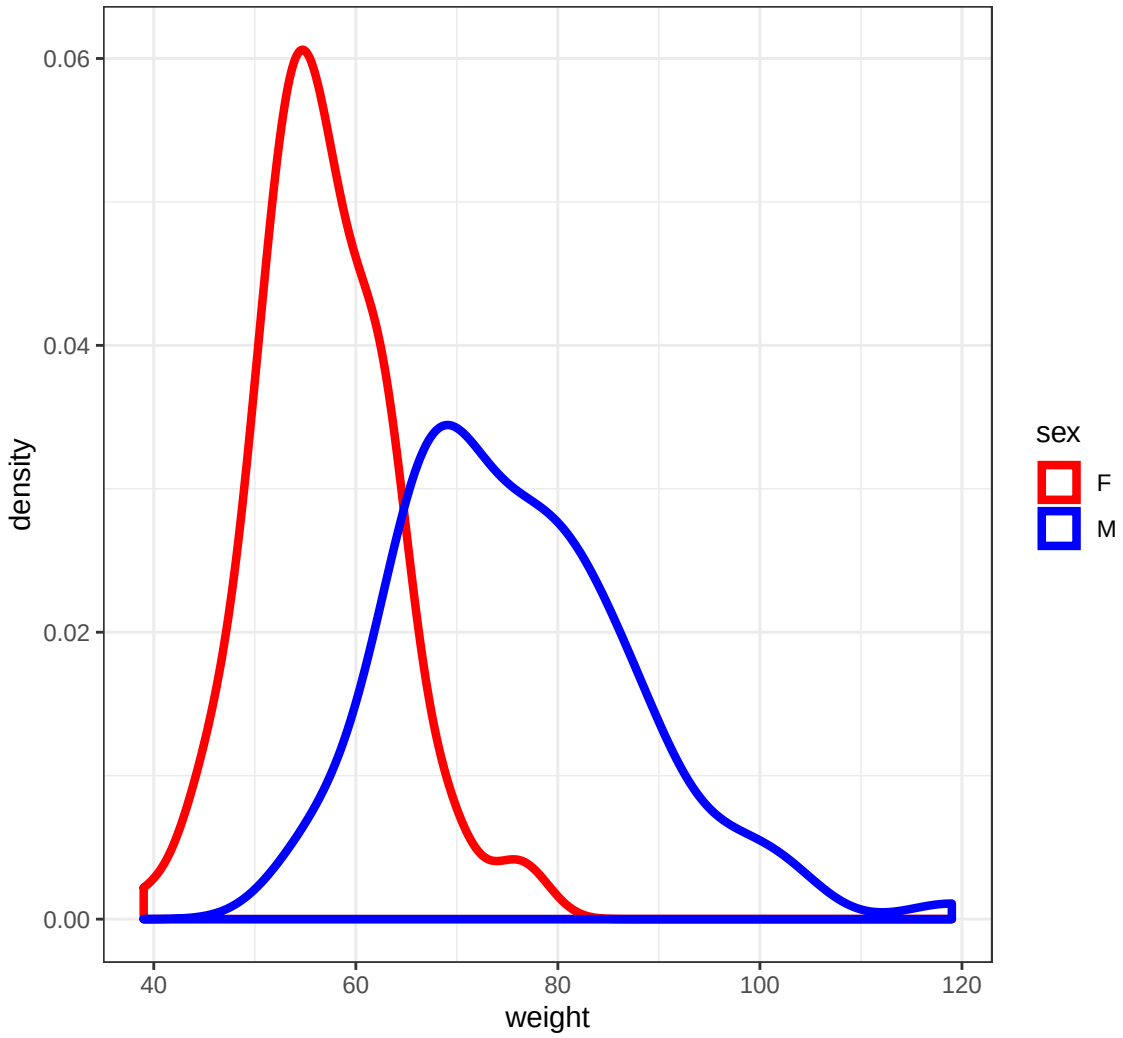
Density Plots of Height

```
> ggplot(htwt) +
+   geom_density(aes(x=height, color=sex), size=1.5) +
+   scale_color_manual(values=c("red", "blue"))
```



Density Plots of Weight

```
> ggplot(htwt) +
+   geom_density(aes(x=weight, color=sex), size=1.5) +
+   scale_color_manual(values=c("red", "blue"))
```



`t.test()` Function

From the help file...

```
Usage

t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula'
t.test(formula, data, subset, na.action, ...)
```

Two-Sided Test of Male Height

```
> m_ht <- htwt %>% filter(sex=="M") %>% select(height)
> testresult <- t.test(x = m_ht$height, mu=177)
```

```
> class(testresult)
[1] "htest"
> is.list(testresult)
[1] TRUE
```

Output of `t.test()`

```
> names(testresult)
 [1] "statistic"   "parameter"   "p.value"     "conf.int"
 [5] "estimate"    "null.value"  "stderr"      "alternative"
 [9] "method"      "data.name"
> testresult

    One Sample t-test

data:  m_ht$height
t = 1.473, df = 87, p-value = 0.1443
alternative hypothesis: true mean is not equal to 177
95 percent confidence interval:
 176.6467 179.3760
sample estimates:
mean of x
 178.0114
```

Tidying the Output

```
> library(broom)
> tidy(testresult)
# A tibble: 1 x 8
  estimate statistic p.value parameter conf.low conf.high
     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
1     178.      1.47   0.144        87     177.      179.
# ... with 2 more variables: method <chr>, alternative <chr>
```

Two-Sided Test of Female Height

```
> f_ht <- htwt %>% filter(sex=="F") %>% select(height)
> t.test(x = f_ht$height, mu = 164)

    One Sample t-test

data:  f_ht$height
t = 1.3358, df = 111, p-value = 0.1844
alternative hypothesis: true mean is not equal to 164
95 percent confidence interval:
 163.6547 165.7739
sample estimates:
mean of x
 164.7143
```

Difference of Two Means

```
> t.test(x = m_ht$height, y = f_ht$height)

    Welch Two Sample t-test

data:  m_ht$height and f_ht$height
t = 15.28, df = 174.29, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 11.57949 15.01467
sample estimates:
mean of x mean of y
 178.0114  164.7143
```

Test with Equal Variances

```
> htwt %>% group_by(sex) %>% summarize(sd(height))
# A tibble: 2 x 2
  sex   `sd(height)`
  <fct>        <dbl>
1 F             5.66
2 M             6.44
> t.test(x = m_ht$height, y = f_ht$height, var.equal = TRUE)

    Two Sample t-test

data:  m_ht$height and f_ht$height
t = 15.519, df = 198, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 11.60735 14.98680
sample estimates:
mean of x mean of y
 178.0114  164.7143
```

Paired Sample Test (v. 1)

First take the difference between the paired observations. Then apply the one-sample t-test.

```
> htwt <- htwt %>% mutate(diffwt = (weight - repwt),
+                         diffht = (height - repht))
> t.test(x = htwt$diffwt) %>% tidy()
# A tibble: 1 x 8
  estimate statistic p.value parameter conf.low conf.high
     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
1  0.00546    0.0319   0.975       182   -0.332     0.343
# ... with 2 more variables: method <chr>, alternative <chr>
```

```
> t.test(x = htwt$diffht) %>% tidy()
# A tibble: 1 x 8
  estimate statistic  p.value parameter conf.low conf.high
     <dbl>     <dbl>    <dbl>     <dbl>    <dbl>     <dbl>
1     2.08      13.5 2.64e-29       182     1.77      2.38
```

```
# ... with 2 more variables: method <chr>, alternative <chr>
```

Paired Sample Test (v. 2)

Enter each sample into the `t.test()` function, but use the `paired=TRUE` argument. This is operationally equivalent to the previous version.

```
> t.test(x=htwt$weight, y=htwt$repwt, paired=TRUE) %>% tidy()
# A tibble: 1 x 8
  estimate statistic p.value parameter conf.low conf.high
     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
1  0.00546    0.0319   0.975       182   -0.332     0.343
# ... with 2 more variables: method <chr>, alternative <chr>
```

```
> t.test(x=htwt$height, y=htwt$repht, paired=TRUE) %>% tidy()
# A tibble: 1 x 8
  estimate statistic  p.value parameter conf.low conf.high
     <dbl>     <dbl>    <dbl>     <dbl>    <dbl>     <dbl>
1     2.08      13.5 2.64e-29       182     1.77      2.38
# ... with 2 more variables: method <chr>, alternative <chr>
>
> htwt %>% select(height, repht) %>% na.omit() %>%
+   summarize(mean(height), mean(repht))
# A tibble: 1 x 2
  `mean(height)` `mean(repht)`
           <dbl>         <dbl>
1           171.          168.
```

Goodness of Fit

Rationale

Sometimes we want to figure out which probability distribution is a reasonable model for the data.

This is related to nonparametric inference in that we wish to go from being in a nonparametric framework to a parametric framework.

Goodness of fit (GoF) tests allow one to perform a hypothesis test of how well a particular parametric probability model explains variation observed in a data set.

Chi-Square GoF Test

Suppose we have data generating process $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F$ for some probability distribution F. We wish to test $H_0 : F \in \{F_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \boldsymbol{\Theta}_0\}$ vs H_1 : not H_0. Suppose that $\boldsymbol{\Theta}_0$ is d-dimensional.

Divide the support of $\{F_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \boldsymbol{\Theta}_0\}$ into k bins $I_1, I_2, \ldots, I_k$.

For $j = 1, 2, \ldots, k$, calculate

$$q_j(\boldsymbol{\theta}) = \int_{I_j} dF_{\boldsymbol{\theta}}(x).$$

Suppose we observe data $x_1, x_2, \ldots, x_n$. For $j = 1, 2, \ldots, k$, let n_j be the number of values $x_i \in I_j$.

Let $\tilde{\theta}_1, \tilde{\theta}_2, \ldots, \tilde{\theta}_d$ be the values that maximize the multinomial likelihood

$$\prod_{j=1}^{k} q_j(\boldsymbol{\theta})^{n_j}.$$

Form GoF statistic

$$s(\boldsymbol{x}) = \sum_{j=1}^{k} \frac{\left(n_j - nq_j\left(\tilde{\boldsymbol{\theta}}\right)\right)^2}{nq_j\left(\tilde{\boldsymbol{\theta}}\right)}$$

When H_0 is true, $S \sim \chi^2_v$ where $v = k-d-1$. The p-value is calculated by $\Pr(S^* \geq s(\boldsymbol{x}))$ where $S^* \sim \chi^2_{k-d-1}$.

Example: Hardy-Weinberg

Suppose at your favorite SNP, we observe genotypes from 100 randomly sampled individuals as follows:

AA	AT	TT
28	60	12

If we code these genotypes as 0, 1, 2, testing for Hardy-Weinberg equilibrium is equivalent to testing whether $X_1, X_2, \ldots, X_{100} \overset{\text{iid}}{\sim} \text{Binomial}(2, \theta)$ for some unknown allele frequency of T, θ.

The parameter dimension is such that $d = 1$. We will also set $k = 3$, where each bin is a genotype. Therefore, we have $n_1 = 28$, $n_2 = 60$, and $n_3 = 12$. Also,

$$q_1(\theta) = (1-\theta)^2, \quad q_2(\theta) = 2\theta(1-\theta), \quad q_3(\theta) = \theta^2.$$

Forming the multinomial likelihood under these bin probabilities, we find $\tilde{\theta} = (n_2 + 2n_3)/(2n)$. The degrees of freedom of the χ^2_v null distribution is $v = k - d - 1 = 3 - 1 - 1 = 1$.

Let's carry out the test in R.

```
> n <- 100
> nj <- c(28, 60, 12)
>
> # parameter estimates
> theta <- (nj[2] + 2*nj[3])/(2*n)
> qj <- c((1-theta)^2, 2*theta*(1-theta), theta^2)
>
> # gof statistic
> s <- sum((nj - n*qj)^2 / (n*qj))
> s
[1] 5.36048
>
> # p-value
> 1-pchisq(s, df=1)
[1] 0.02059811
```

Let's use the `HardyWeinberg` R package.

```
> library(HardyWeinberg)
> x <- c(28, 60, 12)
```

```
> names(x) <- c("AA", "AT", "TT")
> HWChisq(x, cc=0)
Chi-square test for Hardy-Weinberg equilibrium (autosomal)
Chi2 =  5.36048 DF =  1 p-value =  0.02059811 D =  5.64 f =  -0.2315271
```

Exact Tests

Definition

An **exact test** is a hypothesis test where the distribution of the test statistics is known *exactly* when the null hypothesis is true.

An example is the one-sample t-test when the data are exactly iid Normal-distributed. This is not realistic (when do you know data are exactly Normal?), but there are real examples where exact tests are compelling.

Fisher's Exact Test of Independence

Suppose $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} \text{Bernoulli}(p)$ and $Y_1, Y_2, \ldots, Y_n \overset{\text{iid}}{\sim} \text{Bernoulli}(q)$. We observed pairs $(X_1, Y_1), \ldots, (X_n, Y_n)$, and we want to test if X and Y are independent rv's.

Let N_{xy} be the number of observed (x, y) pairs for $x, y \in \{0, 1\}$.

We can calculate a statistic based on N_{xy} that does not depend on p or q.

Tabulated Data

We observe $n_{00} = a, n_{01} = b, n_{10} = c, n_{11} = d$. Compile this into a table:

	$Y = 0$	$Y = 1$	Totals
$X = 0$	a	b	$a + b$
$X = 1$	c	d	$c + d$
Totals	$a + c$	$b + d$	n

Probability of Observed Data

$$\Pr(N_{00} = a | N_{00} + N_{01} = a + b, N_{00} + N_{10} = a + c) =$$

$$\frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}}$$

This does not depend on p or q!

P-value

Calculating Fisher exact test p-value can be confusing and/or controversial. Here is one way to get a two-sided test P-value.

Consider all a^*, b^*, c^*, d^* such that $a^* + b^* = a + b$ and $c^* + d^* = c + d$. We need to keep the conditional part of the probability intact.

The p-value is the sum of probabilities over all configurations such that

$$\Pr(N_{00} = a^* | N_{00} + N_{01} = a^* + b^*, N_{00} + N_{10} = a^* + c^*) \leq$$

$$\Pr(N_{00} = a | N_{00} + N_{01} = a + b, N_{00} + N_{10} = a + c)$$

Example: Self-Isolation and Infection

X = self-isolation (no or yes)

Y = infection (no or yes)

	$Y = 0$	$Y = 1$	Totals
$X = 0$	1	9	10
$X = 1$	11	3	14
Totals	12	12	24

```
> x <- matrix(c(1, 9, 11, 3), nrow=2, byrow=TRUE)
> fisher.test(x)$p.value
[1] 0.002759456
> fisher.test(x, alternative="less")$p.value
[1] 0.001379728
> fisher.test(x, alternative="greater")$p.value
[1] 0.9999663
```

Other Scenarios

Fisher's exact test has been derived for cases where X and Y are mulinomial.

There is a one-dimensional Fisher's exact test. A great example of this is Fisher's exact test of HWE.

Fisher's Exact Test of HWE

Suppose at your favorite SNP, we observe genotypes from 100 randomly sampled individuals as follows:

AA	AT	TT
28	60	12

Let's do Fisher's exact test of HWE on these data.

Observe $n_{AA} = 28, n_{AT} = 60, n_{TT} = 12$ with $n = 100$ observed genotypes.

We also observe marginal allele counts, $n_A = 116, n_T = 84$.

Let p be the true allele frequency of T. Then, under HWE,

$$\Pr(N_{AA} = n_{AA}, N_{AT} = n_{AT}, N_{TT} = n_{TT} | N_A = n_A, N_T = n_T) \tag{1}$$

$$= \frac{\Pr(N_{AA} = n_{AA}, N_{AT} = n_{AT}, N_{TT} = n_{TT})}{\Pr(N_A = n_A, N_T = n_T)} \tag{2}$$

$$= \frac{\binom{n}{n_{AA}\ n_{AT}\ n_{TT}}(1-p)^{2n_{AA}}(2p(1-p))^{n_{AT}}p^{2n_{TT}}}{\binom{2n}{n_T}(1-p)^{n_A}p^{n_T}} \tag{3}$$

$$= \frac{\binom{n}{n_{AA}\ n_{AT}\ n_{TT}}2^{n_{AT}}}{\binom{2n}{n_T}} \tag{4}$$

Let's use the `HardyWeinberg` R package to perform an exact test of HWE.

```
> library(HardyWeinberg)
> x <- c(28, 60, 12)
> names(x) <- c("AA", "AT", "TT")
> HWExact(x)
Haldane Exact test for Hardy-Weinberg equilibrium (autosomal)
using SELOME p-value
sample counts: nAA =  28 nAT =  60 nTT =  12
H0: HWE (D==0), H1: D <> 0
D =  5.64 p-value =  0.02565977
```

Let's compare the result to the χ^2 goodness of fit test and the generalized likelihood ratio test.

```
> HWChisq(x, cc=0) # chi-square gof
Chi-square test for Hardy-Weinberg equilibrium (autosomal)
Chi2 =  5.36048 DF =  1 p-value =  0.02059811 D =  5.64 f =  -0.2315271
>
> HWLratio(x) # generalized lrt
Likelihood ratio test for Hardy-Weinberg equilibrium
G2 = 5.467661 DF = 1 p-value = 0.01937154
```

Method of Moments

Rationale

Suppose that $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F$. By the strong law of large numbers we have, as $n \to \infty$

$$\frac{\sum_{i=1}^n X_i^k}{n} \xrightarrow{\text{a.s.}} \mathrm{E}_F\left[X^k\right]$$

when $\mathrm{E}_F\left[X^k\right]$ exists.

This means that we can nonparametrically estimate the moments of a distribution. Also, in the parametric setting, these moments can be used to form parameter estimates.

Definition

Suppose that $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F_{\boldsymbol{\theta}}$ where $\boldsymbol{\theta}$ is d-dimensional.

Calculate moments $\mathrm{E}\left[X^k\right]$ for $k = 1, 2, \ldots, d'$ where $d' \geq d$.

For each parameter $j = 1, 2, \ldots, d$, solve for θ_j in terms of $\mathrm{E}\left[X^k\right]$ for $k = 1, 2, \ldots, d'$.

The method of moments estimator of θ_j is formed by replacing the function of moments $\mathrm{E}\left[X^k\right]$ that equals θ_j with the empirical moments $\sum_{i=1}^n X_i^k / n$.

Example: Normal

For a Normal(μ, σ^2) distribution, we have

$$\mathrm{E}[X] = \mu$$

$$\mathrm{E}\left[X^2\right] = \sigma^2 + \mu^2$$

Solving for μ and σ^2, we have $\mu = \mathrm{E}[X]$ and $\sigma^2 = \mathrm{E}[X^2] - \mathrm{E}[X]^2$. This yields method of moments estimators

$$\tilde{\mu} = \frac{\sum_{i=1}^n X_i}{n}, \quad \tilde{\sigma}^2 = \frac{\sum_{i=1}^n X_i^2}{n} - \left[\frac{\sum_{i=1}^n X_i}{n}\right]^2.$$

Example: Balding-Nichols Model

In the BN model, we have

$$Z_i | Q_i \sim \text{Binomial}(2, Q_i)$$

independently for $i = 1, 2, \ldots, n$, and

$$Q_1, Q_2, \ldots, Q_n \overset{\text{iid}}{\sim} \text{BN}(p, f).$$

Recall that, marginally, $Z_1, Z_2, \ldots, Z_n$ are iid.

We showed that

$$\text{E}[Z] = 2p \text{ and } \text{Var}[Z] = 2p(1-p)(1+f).$$

Therefore,

$$p = \frac{\text{E}[Z]}{2} \text{ and } f = \frac{\text{Var}[Z]}{\text{E}[Z](1 - \text{E}[Z]/2)} - 1.$$

To estimate p and f, replace $\text{E}[Z]$ and $\text{Var}[Z]$ with estimates:

$$\widetilde{\text{E}}[Z] = \frac{\sum_{i=1}^n Z_i}{n}$$

$$\widetilde{\text{Var}}[Z] = \frac{\sum_{i=1}^n Z_i^2}{n} - \left(\frac{\sum_{i=1}^n Z_i}{n}\right)^2$$

Exercise: RNA-Seq

Recall the Gamma-Poisson distribution of RNA-Seq data covered in Week 3.

Derive the method-of-moments estimates of α and β from that model.

Exploring Goodness of Fit

As mentioned above, moments can be nonparametrically estimated. At the same time, for a given parametric distribution, these moments can also be written in terms of the parameters.

For example, consider a single parameter exponential family distribution. The variance is going to be defined in terms of the parameter. At the same time, we can estimate variance through the empirical moments

$$\frac{\sum_{i=1}^n X_i^2}{n} - \left[\frac{\sum_{i=1}^n X_i}{n}\right]^2.$$

In the scenario where several sets of variables are measured, the MLEs of the variance in terms of the single parameter can be compared to the moment estimates of variance to assess goodness of fit of that distribution.

Permutation Methods

Rationale

Permutation methods are useful for testing hypotheses about equality of distributions.

Observations can be permuted among populations to simulate the case where the distributions are equivalent.

Many permutation methods only depend on the ranks of the data, so they are a class of robust methods for performing hypothesis tests. However, the types of hypotheses that can be tested are limited.

Permutation Test

Suppose $X_1, X_2, \ldots, X_m \overset{\text{iid}}{\sim} F_X$ and $Y_1, Y_2, \ldots, Y_n \overset{\text{iid}}{\sim} F_Y$.

We wish to test $H_0 : F_X = F_Y$ vs $H_1 : F_X \neq F_Y$.

Consider a general test statistic $S = S(X_1, X_2, \ldots, X_m, Y_1, Y_2, \ldots, Y_n)$ so that the larger S is the more evidence there is against the null hypothesis.

Under the null hypothesis, any reordering of these values, where m are randomly assigned to the "X" population and n are assigned to the "Y" population, should be equivalently distributed.

For B permutations (possibly all unique permutations), we calculate

$$S^{*(b)} = S\left(Z_1^{*(b)}, Z_2^{*(b)}, \ldots, Z_m^{*(b)}, Z_{m+1}^{*(b)}, \ldots, Z_{m+n}^{*(b)}\right)$$

where $Z_1^{*(b)}, Z_2^{*(b)}, \ldots, Z_m^{*(b)}, Z_{m+1}^{*(b)}, \ldots, Z_{m+n}^{*(b)}$ is a random permutation of the values $X_1, X_2, \ldots, X_m, Y_1, Y_2, \ldots, Y_n$.

Example permutation in R:

```
> z <- c(x, y)
> zstar <- sample(z, replace=FALSE)
```

The p-value is calculated as proportion of permutations where the resulting permutation statistic exceeds the observed statistics:

$$\text{p-value}(s) = \frac{1}{B}\sum_{b=1}^{B} 1\left(S^{*(b)} \geq S\right).$$

This can be (1) an exact calculation where all permutations are considered, (2) a Monte Carlo approximation where B random permutations are considered, or (3) a large $\min(m, n)$ calculation where an asymptotic probabilistic approximation is used.

Wilcoxon Rank Sum Test

Also called the Mann-Whitney-Wilcoxon test.

Consider the ranks of the data as a whole, $X_1, X_2, \ldots, X_m, Y_1, Y_2, \ldots, Y_n$, where $r(X_i)$ is the rank of X_i and $r(Y_j)$ is the rank of Y_j. Note that $r(\cdot) \in \{1, 2, \ldots, m+n\}$. The smallest value is such that $r(X_i) = 1$ or $r(Y_j) = 1$, the next smallest value maps to 2, etc.

Note that

$$\sum_{i=1}^{m} r(X_i) + \sum_{j=1}^{n} r(Y_j) = \frac{(m+n)(m+n+1)}{2}.$$

The statistic W is calculated by:

$$R_X = \sum_{i=1}^{m} r(X_i) \qquad R_Y = \sum_{j=1}^{n} r(Y_j)$$
$$W_X = R_X - \frac{m(m+1)}{2} \quad W_Y = R_Y - \frac{n(n+1)}{2}$$
$$W = \min(W_X, W_Y)$$

In this case, the *smaller* W is, the more significant it is. Note that $mn - W = \max(W_X, W_Y)$, so we just as well could utilize large $\max(W_X, W_Y)$ as a test statistic.

Wilcoxon Signed Rank-Sum Test

The Wilcoxon signed rank test is similar to the Wilcoxon two-sample test, except here we have paired observations $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$.

An example is an individual's clinical measurement before (X) and after (Y) treatment.

In order to test the hypothesis, we calculate $r(X_i, Y_i) = |Y_i - X_i|$ and also $s(X_i, Y_i) = \text{sign}(Y_i - X_i)$.

The test statistic is $|W|$ where

$$W = \sum_{i=1}^{n} r(X_i, Y_i) s(X_i, Y_i).$$

Both of these tests can be carried out using the `wilcox.test()` function in R.

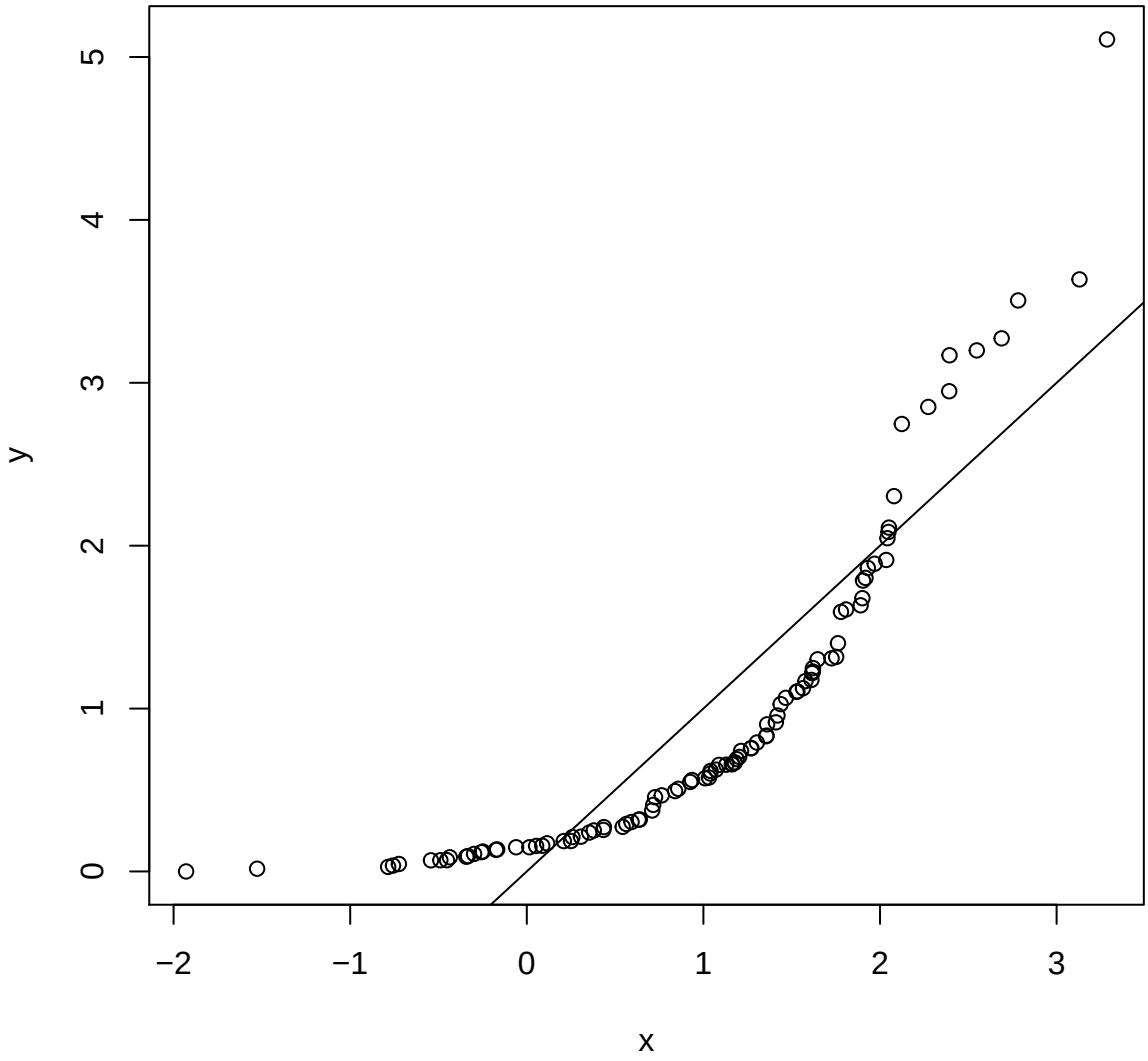
```
wilcox.test(x, y = NULL,
            alternative = c("two.sided", "less", "greater"),
            mu = 0, paired = FALSE, exact = NULL, correct = TRUE,
            conf.int = FALSE, conf.level = 0.95, ...)
```

Examples

Same population mean and variance.

```
> x <- rnorm(100, mean=1)
> y <- rexp(100, rate=1)
> wilcox.test(x, y)

    Wilcoxon rank sum test with continuity correction

data:  x and y
W = 5596, p-value = 0.1457
alternative hypothesis: true location shift is not equal to 0
```

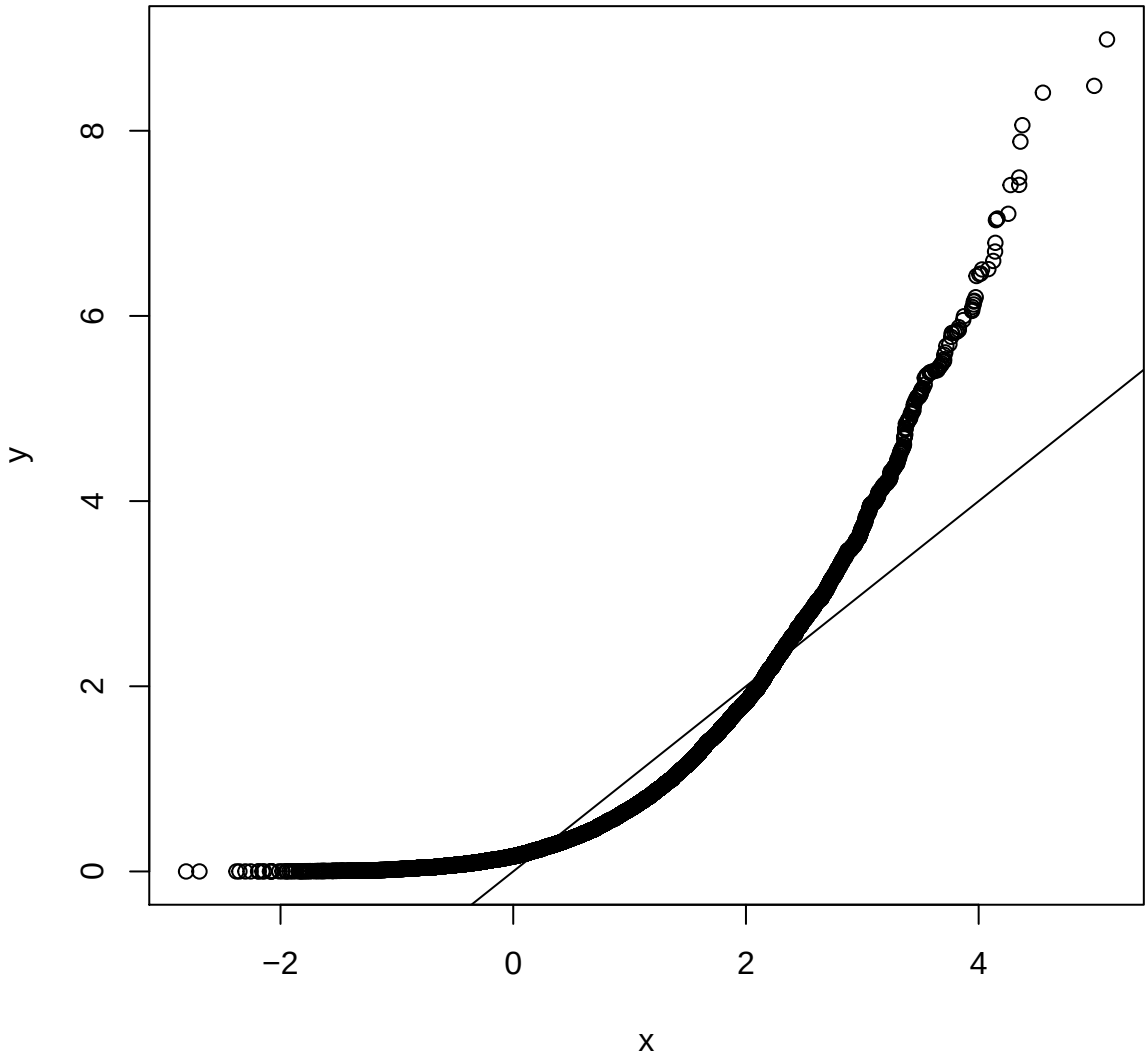
```
> qqplot(x, y); abline(0,1)
```



Same population mean and variance. Large sample size.

```
> x <- rnorm(10000, mean=1)
> y <- rexp(10000, rate=1)
> wilcox.test(x, y)

    Wilcoxon rank sum test with continuity correction

data:  x and y
W = 54175539, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

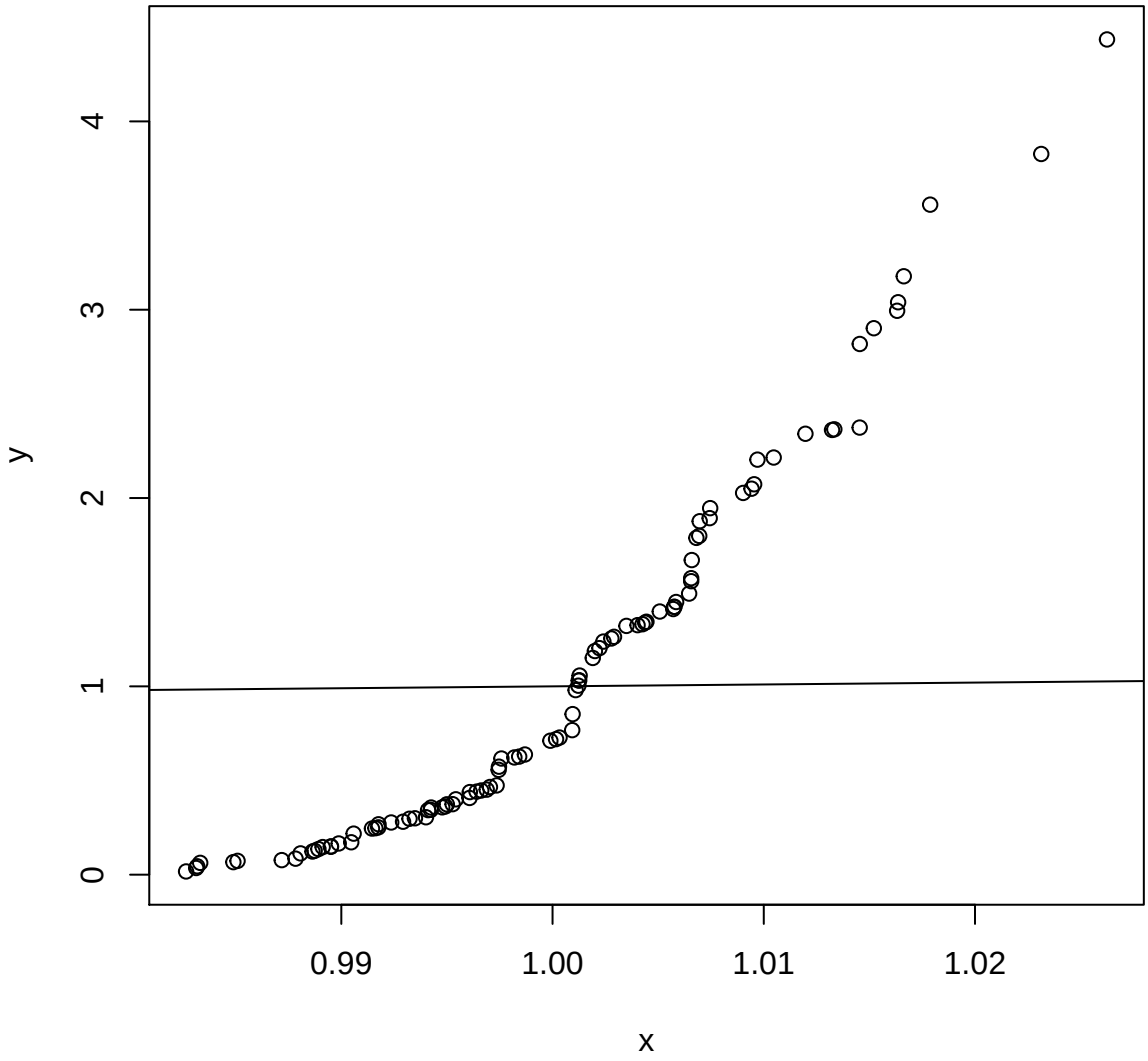
```
> qqplot(x, y); abline(0,1)
```



Same mean, very different variances.

```
> x <- rnorm(100, mean=1, sd=0.01)
> y <- rexp(100, rate=1)
> wilcox.test(x, y)

    Wilcoxon rank sum test with continuity correction

data:  x and y
W = 5435, p-value = 0.2884
alternative hypothesis: true location shift is not equal to 0
```

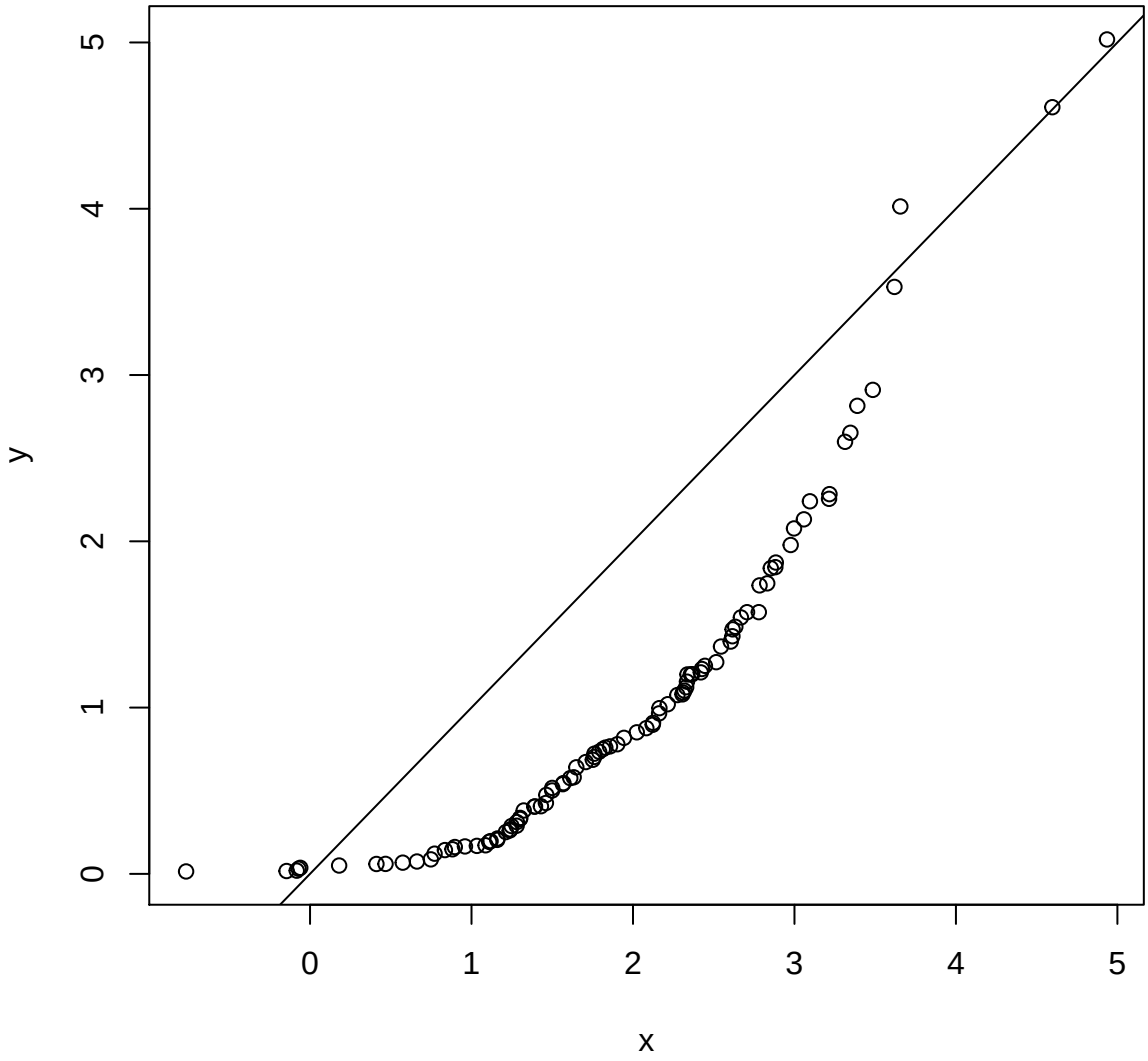
```
> qqplot(x, y); abline(0,1)
```



Same variances, different means.

```
> x <- rnorm(100, mean=2)
> y <- rexp(100, rate=1)
> wilcox.test(x, y)

    Wilcoxon rank sum test with continuity correction

data:  x and y
W = 7672, p-value = 6.687e-11
alternative hypothesis: true location shift is not equal to 0
```

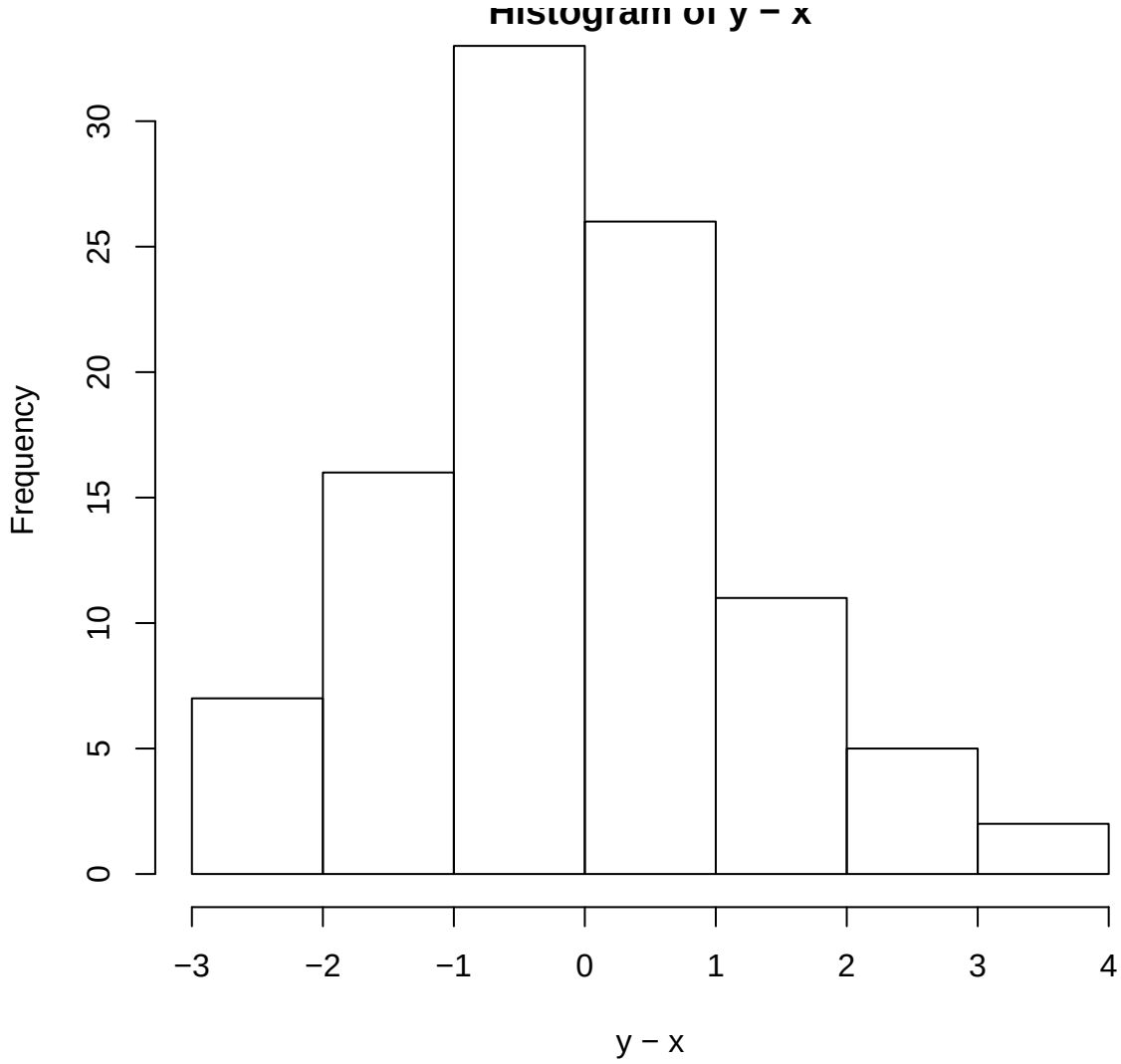
```
> qqplot(x, y); abline(0,1)
```



Same population mean and variance.

```
> x <- rnorm(100, mean=1)
> y <- rexp(100, rate=1)
> wilcox.test(x, y, paired=TRUE)

    Wilcoxon signed rank test with continuity correction

data:  x and y
V = 2838, p-value = 0.2826
alternative hypothesis: true location shift is not equal to 0
```

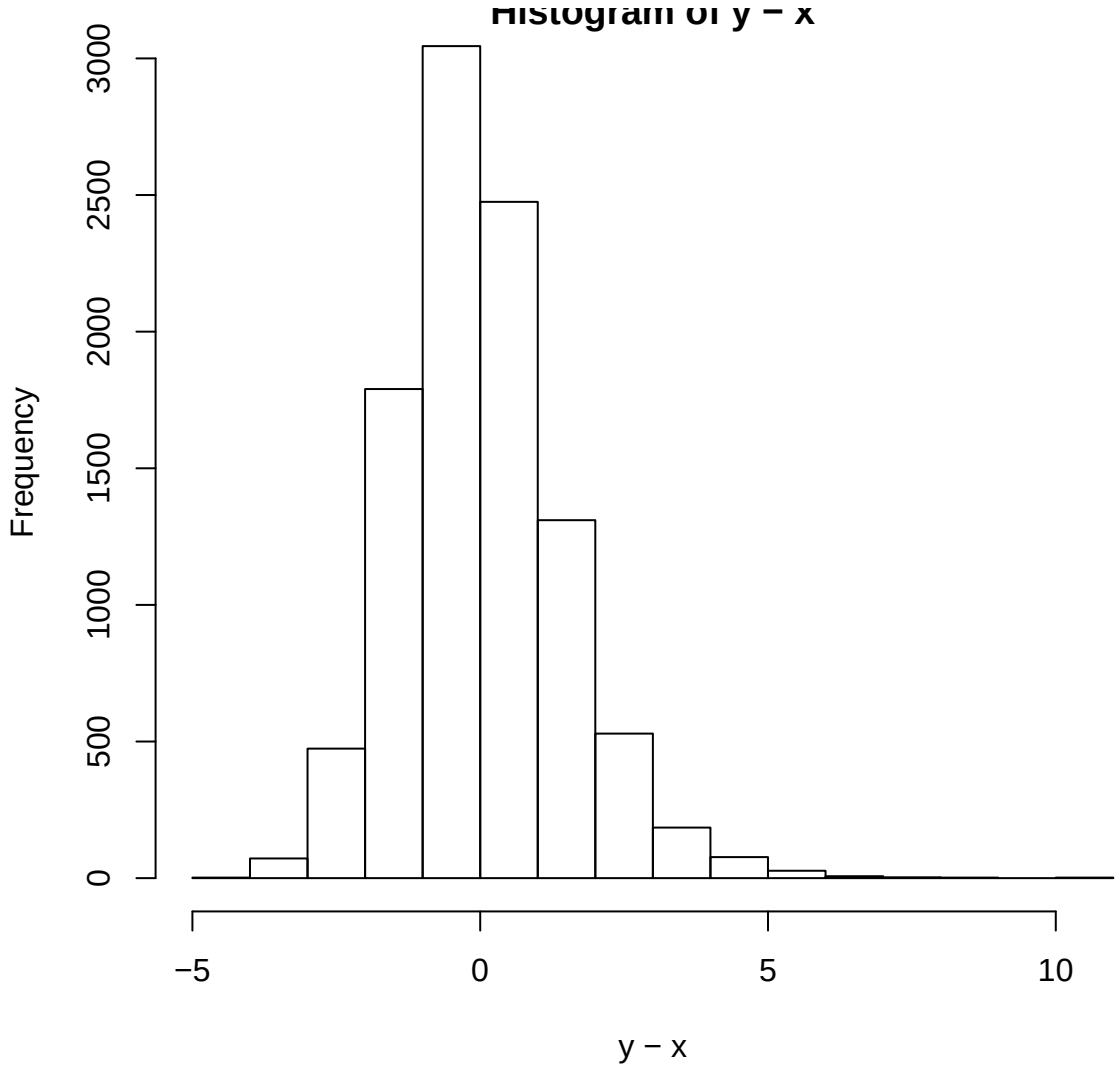
```
> hist(y-x)
```



Same population mean and variance. Large sample size.

```
> x <- rnorm(10000, mean=1)
> y <- rexp(10000, rate=1)
> wilcox.test(x, y, paired=TRUE)

    Wilcoxon signed rank test with continuity correction

data:  x and y
V = 26199685, p-value = 3.371e-05
alternative hypothesis: true location shift is not equal to 0
```

```
> hist(y-x)
```



Permutation *t*-test

As above, suppose $X_1, X_2, \ldots, X_m \overset{\text{iid}}{\sim} F_X$ and $Y_1, Y_2, \ldots, Y_n \overset{\text{iid}}{\sim} F_Y$, and we wish to test $H_0 : F_X = F_Y$ vs $H_1 : F_X \neq F_Y$. However, suppose we additionally know that $\text{Var}(X) = \text{Var}(Y)$. We can use a t-statistic to test this hypothesis:

$$t = \frac{\overline{x} - \overline{y}}{\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right) s^2}}$$

where s^2 is the pooled sample variance.

To obtain the null distribution, we randomly permute the observations to assign m data points to the X sample and n to the Y sample. This yields permutation data set $x^* = (x_1^*, x_2^*, \ldots, x_m^*)^T$ and $y^* = (y_1^*, y_2^*, \ldots, y_n^*)^T$. We form null t-statistic

$$t^* = \frac{\overline{x}^* - \overline{y}^*}{\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right) s^{2*}}}$$

where again s^{2*} is the pooled sample variance.

In order to obtain a p-value, we calculate $t^{*(b)}$ for $b = 1, 2, \ldots, B$ permutation data sets.

The p-value of t is then the proportion of permutation statistics as or more extreme than the observed statistic:

$$\text{p-value}(t) = \frac{1}{B} \sum_{b=1}^{B} 1\left(|t^{*(b)}| \geq |t|\right).$$

Empirical Distribution Functions

Definition

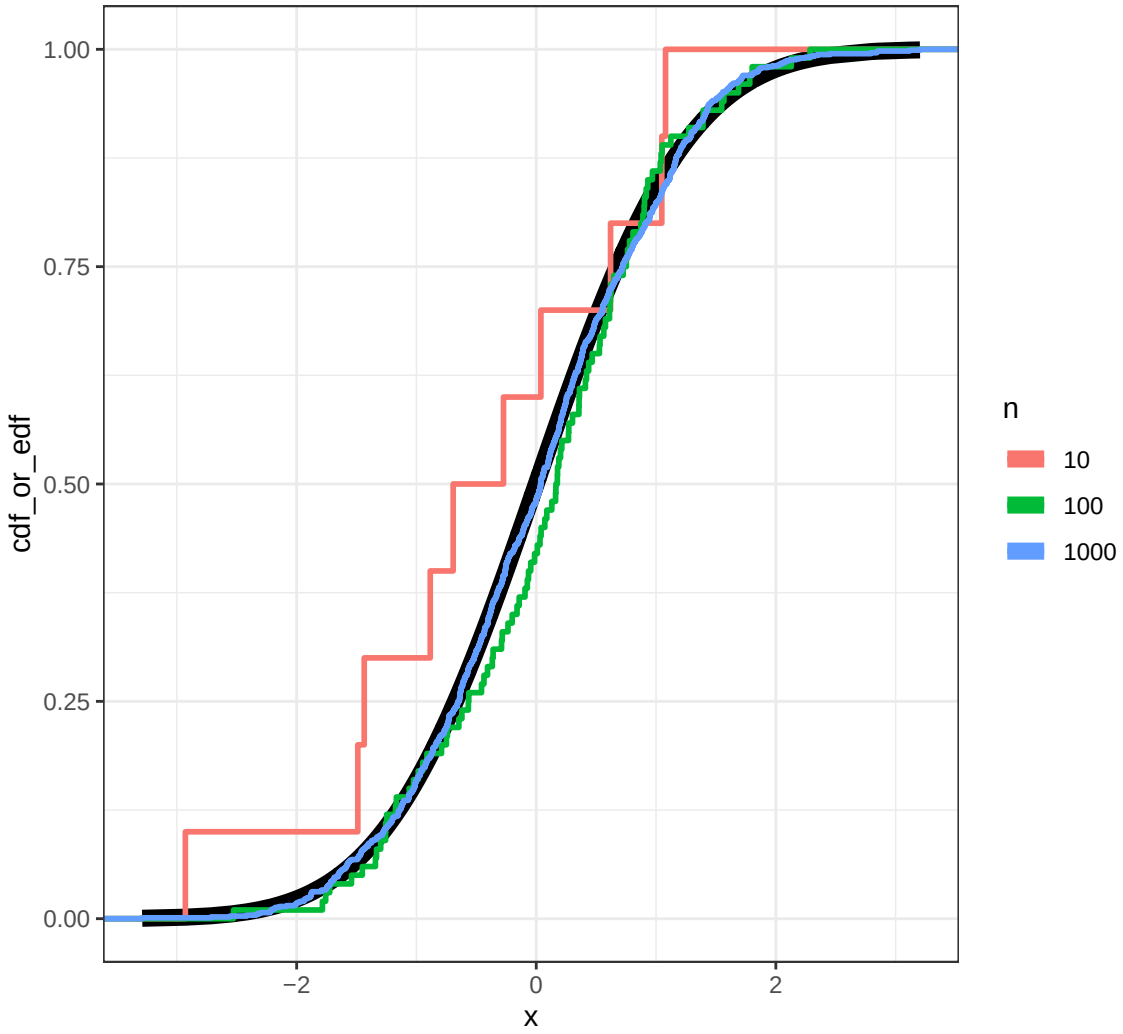
Suppose $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F$. The **empirical distribution function** (edf) – or **empirical cumulative distribution function** (ecdf) – is the distribution that puts probability $1/n$ on each observed value X_i.

Let $1(X_i \leq y) = 1$ if $X_i \leq y$ and $1(X_i \leq y) = 0$ if $X_i > y$.

$$\text{Random variable: } \hat{F}_{\boldsymbol{X}}(y) = \frac{1}{n} \sum_{i=1}^{n} 1(X_i \leq y)$$

$$\text{Observed variable: } \hat{F}_{\boldsymbol{x}}(y) = \frac{1}{n} \sum_{i=1}^{n} 1(x_i \leq y)$$

Example: Normal



Pointwise Convergence

Under our assumptions, by the strong law of large numbers for each $y \in \mathbb{R}$,

$$\hat{F}_{\boldsymbol{X}}(y) \xrightarrow{\text{a.s.}} F(y)$$

as $n \to \infty$.

Glivenko-Cantelli Theorem

Under our assumptions, we can get a much stronger convergence result:

$$\sup_{y \in \mathbb{R}} \left| \hat{F}_{\boldsymbol{X}}(y) - F(y) \right| \xrightarrow{\text{a.s.}} 0$$

as $n \to \infty$. Here, "sup" is short for *supremum*, which is a mathematical generalization of *maximum*.

This result says that even the worst difference between the edf and the true cdf converges with probability 1 to zero.

Dvoretzky-Kiefer-Wolfowitz (DKW) Inequality

This result gives us an upper bound on how far off the edf is from the true cdf, which allows us to construct confidence bands about the edf.

$$\Pr\left(\sup_{y\in\mathbb{R}}\left|\hat{F}_{\boldsymbol{X}}(y)-F(y)\right|>\epsilon\right)\leq 2\exp -2n\epsilon^2$$

As outlined in *All of Nonparametric Statistics*, setting

$$\epsilon_n=\sqrt{\frac{1}{2n}\log\left(\frac{2}{\alpha}\right)}$$

$$L(y)=\max\{\hat{F}_{\boldsymbol{X}}(y)-\epsilon_n,0\}$$

$$U(y)=\min\{\hat{F}_{\boldsymbol{X}}(y)+\epsilon_n,1\}$$

guarantees that $\Pr(L(y)\leq F(y)\leq U(y) \text{ for all } y)\geq 1-\alpha$.

Statistical Functionals

A **statistical functional** $T(F)$ is any function of F. Examples:

- $\mu(F)=\int xdF(x)$
- $\sigma^2(F)=\int(x-\mu(F))^2dF(x)$
- $\text{median}(F)=F^{-1}(1/2)$

A **linear statistical functional** is such that $T(F)=\int a(x)dF(x)$.

Plug-In Estimator

A plug-in estimator of $T(F)$ based on the edf is $T(\hat{F}_{\boldsymbol{X}})$. Examples:

- $\hat{\mu}=\mu(\hat{F}_{\boldsymbol{X}})=\int x\hat{F}_{\boldsymbol{X}}(x)=\frac{1}{n}\sum_{i=1}^n X_i$
- $\hat{\sigma}^2=\sigma^2(\hat{F}_{\boldsymbol{X}})=\int(x-\hat{\mu})^2\hat{F}_{\boldsymbol{X}}(x)=\frac{1}{n}\sum_{i=1}^n(X_i-\hat{\mu})^2$
- $\text{median}(\hat{F}_{\boldsymbol{X}})=\hat{F}_{\boldsymbol{X}}^{-1}(1/2)$

EDF Standard Error

Suppose that $T(F)=\int a(x)dF(x)$ is a linear functional. Then:

$$\text{Var}(T(\hat{F}_{\boldsymbol{X}}))=\frac{1}{n^2}\sum_{i=1}^n\text{Var}(a(X_i))=\frac{\text{Var}_F(a(X))}{n}$$

$$\text{se}(T(\hat{F}_{\boldsymbol{X}}))=\sqrt{\frac{\text{Var}_F(a(X))}{n}}$$

$$\hat{\text{se}}(T(\hat{F}_{\boldsymbol{X}}))=\sqrt{\frac{\text{Var}_{\hat{F}_{\boldsymbol{X}}}(a(X))}{n}}$$

Note that

$$\mathrm{Var}_F(a(X)) = \int (a(x) - T(F))^2 dF(x)$$

because $T(F) = \int a(x)dF(x) = \mathrm{E}_F[a(X)]$. Likewise,

$$\mathrm{Var}_{\hat{F}_{\boldsymbol{X}}}(a(X)) = \frac{1}{n}\sum_{i=1}^{n}(a(X_i) - T(\hat{F}_{\boldsymbol{X}}))^2$$

where $T(\hat{F}_{\boldsymbol{X}}) = \frac{1}{n}\sum_{i=1}^{n} a(X_i)$.

EDF CLT

Suppose that $\mathrm{Var}_F(a(X)) < \infty$. Then we have the following convergences as $n \to \infty$:

$$\frac{\mathrm{Var}_{\hat{F}_{\boldsymbol{X}}}(a(X))}{\mathrm{Var}_F(a(X))} \xrightarrow{P} 1 \ , \ \frac{\hat{\mathrm{se}}(T(\hat{F}_{\boldsymbol{X}}))}{\mathrm{se}(T(\hat{F}_{\boldsymbol{X}}))} \xrightarrow{P} 1$$

$$\frac{T(F) - T(\hat{F}_{\boldsymbol{X}})}{\hat{\mathrm{se}}(T(\hat{F}_{\boldsymbol{X}}))} \xrightarrow{D} \mathrm{Normal}(0,1)$$

The estimators are very easy to calculate on real data, so this a powerful set of results.

Kolmogorov–Smirnov Test

The KS test is a goodness of fit test that can be used to compare a sample of data to a particular distribution, or to compare two samples of data.

The former is a parametric GoF test, and the latter is a nonparametric test of equal distributions.

One Sample KS Test

Suppose we have data generating process $X_1, X_2, \ldots, X_n \sim F$ for some probability distribution F. We wish to test $H_0 : F = F_{\boldsymbol{\theta}}$ vs $H_1 : F \neq F_{\boldsymbol{\theta}}$ for some parametric distribution $F_{\boldsymbol{\theta}}$.

For observed data $x_1, x_2, \ldots, x_n$ we form the edf $\hat{F}_{\boldsymbol{x}}$ and test-statistic

$$D(\boldsymbol{x}) = \sup_z \left| \hat{F}_{\boldsymbol{x}}(z) - F_{\boldsymbol{\theta}}(z) \right|.$$

The null distribution of this test can be approximated based on a stochastic process called the Brownian bridge.

Two Sample KS Test

Suppose $X_1, X_2, \ldots, X_m \overset{\text{iid}}{\sim} F_X$ and $Y_1, Y_2, \ldots, Y_n \overset{\text{iid}}{\sim} F_Y$. We wish to test $H_0 : F_X = F_Y$ vs $H_1 : F_X \neq F_Y$.

For observed data $x_1, x_2, \ldots, x_m$ and $y_1, y_2, \ldots, y_n$ we form the edf's $\hat{F}_{\boldsymbol{x}}$ and $\hat{F}_{\boldsymbol{y}}$. We then form test-statistic

$$D(\boldsymbol{x}, \boldsymbol{y}) = \sup_z \left| \hat{F}_{\boldsymbol{x}}(z) - \hat{F}_{\boldsymbol{y}}(z) \right|.$$

The null distribution of this statistic can be approximated using results on edf's.

Both of these tests can be carried out using the `ks.test()` function in R.

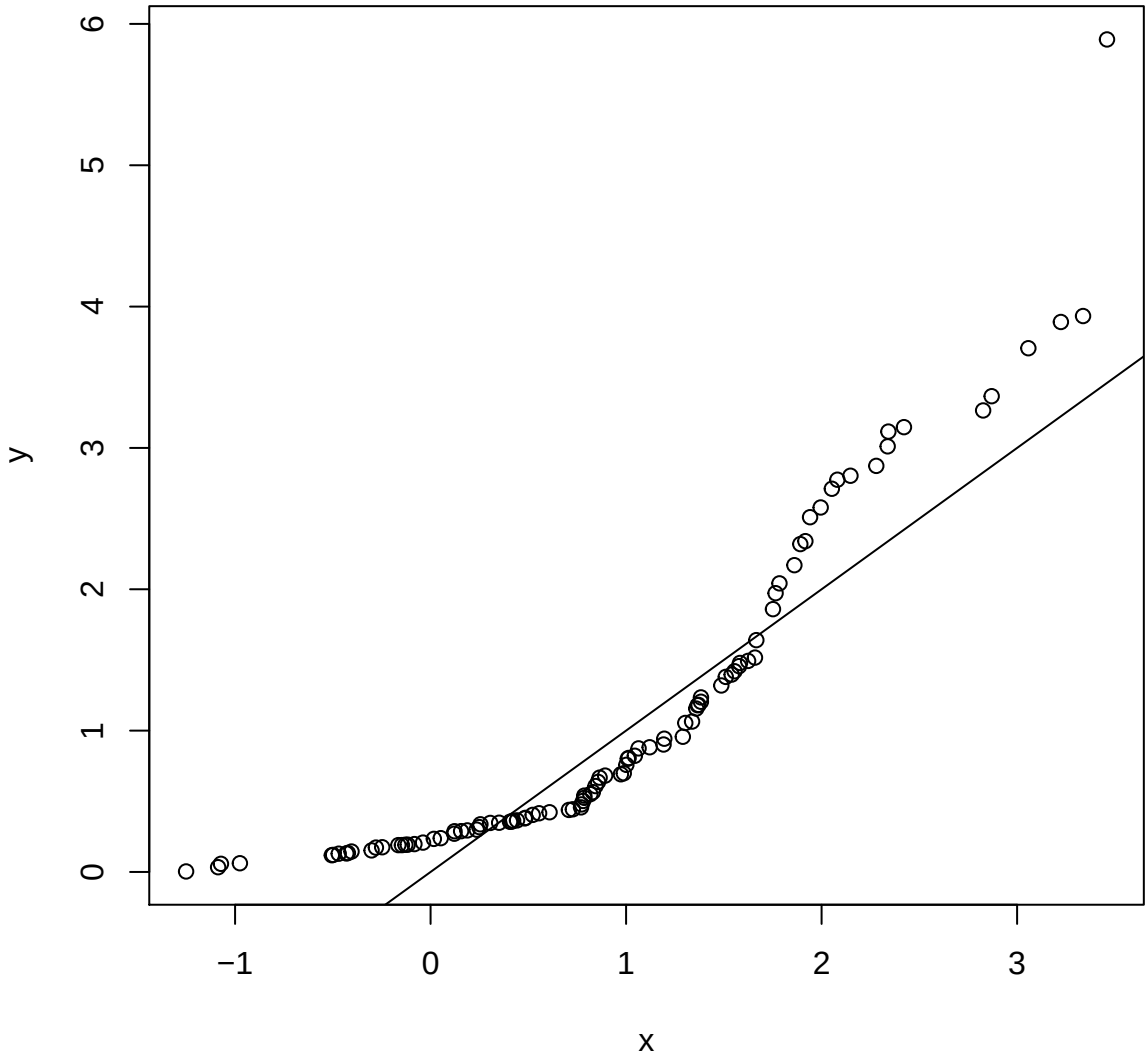
```
ks.test(x, y, ...,
        alternative = c("two.sided", "less", "greater"),
        exact = NULL)
```

Example: Exponential vs Normal

Two sample KS test.

```
> x <- rnorm(100, mean=1)
> y <- rexp(100, rate=1)
> wilcox.test(x, y)

    Wilcoxon rank sum test with continuity correction

data:  x and y
W = 4957, p-value = 0.9173
alternative hypothesis: true location shift is not equal to 0
> ks.test(x, y)

    Two-sample Kolmogorov-Smirnov test

data:  x and y
D = 0.19, p-value = 0.0541
alternative hypothesis: two-sided
```

```
> qqplot(x, y); abline(0,1)
```



One sample KS tests.

```
> ks.test(x=x, y="pnorm")

    One-sample Kolmogorov-Smirnov test

data:  x
D = 0.37038, p-value = 2.429e-12
alternative hypothesis: two-sided
>
> ks.test(x=x, y="pnorm", mean=1)

    One-sample Kolmogorov-Smirnov test

data:  x
D = 0.064807, p-value = 0.795
alternative hypothesis: two-sided
```

Standardize (mean center, sd scale) the observations before comparing to a Normal(0,1) distribution.

```
> ks.test(x=((x-mean(x))/sd(x)), y="pnorm")

    One-sample Kolmogorov-Smirnov test

data:  ((x - mean(x))/sd(x))
D = 0.037869, p-value = 0.9988
alternative hypothesis: two-sided
>
> ks.test(x=((y-mean(y))/sd(y)), y="pnorm")

    One-sample Kolmogorov-Smirnov test

data:  ((y - mean(y))/sd(y))
D = 0.17723, p-value = 0.00374
alternative hypothesis: two-sided
```

Bootstrap

Rationale

Suppose $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F$. If the edf $\hat{F}_{\boldsymbol{X}}$ is an accurate approximation for the true cdf F, then we can utilize $\hat{F}_{\boldsymbol{X}}$ in place of F to nonparametrically characterize the sampling distribution of a statistic $T(\boldsymbol{X})$.

This allows for the sampling distribution of more general statistics to be considered, such as the median or a percentile, as well as more traditional statistics, such as the mean, when the underlying distribution is unknown.

When we encounter modeling fitting, the bootstrap may be very useful for characterizing the sampling distribution of complex statistics we calculate from fitted models.

Big Picture

We calculate $T(\boldsymbol{x})$ on the observed data, and we also form the edf, $\hat{F}_{\boldsymbol{x}}$.

To approximate the sampling distribution of $T(\boldsymbol{X})$ we generate B random samples of n iid data points from $\hat{F}_{\boldsymbol{x}}$ and calculate $T(\boldsymbol{x}^{*(b)})$ for each bootstrap sample $b = 1, 2, \ldots, B$ where $\boldsymbol{x}^{*(b)} = (x_1^{*(b)}, x_2^{*(b)}, \ldots, x_n^{*(b)})^T$.

Sampling $X_1^*, \ldots, X_n^* \overset{\text{iid}}{\sim} \hat{F}_{\boldsymbol{x}}$ is accomplished by sampling n times *with replacement* from the observed data $x_1, x_2, \ldots, x_n$.

This means $\Pr(X^* = x_j) = \frac{1}{n}$ for all j.

Bootstrap Variance

For each bootstrap sample $\boldsymbol{x}^{*(b)} = (x_1^{*(b)}, x_2^{*(b)}, \ldots, x_n^{*(b)})^T$, calculate bootstrap statistic $T(\boldsymbol{x}^{*(b)})$.

Repeat this for $b = 1, 2, \ldots, B$.

Estimate the sampling variance of $T(\boldsymbol{x})$ by

$$\hat{\text{Var}}(T(\boldsymbol{x})) = \frac{1}{B}\sum_{b=1}^{B}\left(T\left(\boldsymbol{x}^{*(b)}\right) - \frac{1}{B}\sum_{k=1}^{B}T\left(\boldsymbol{x}^{*(k)}\right)\right)^2$$

Caveat

Why haven't we just been doing this the entire time?!

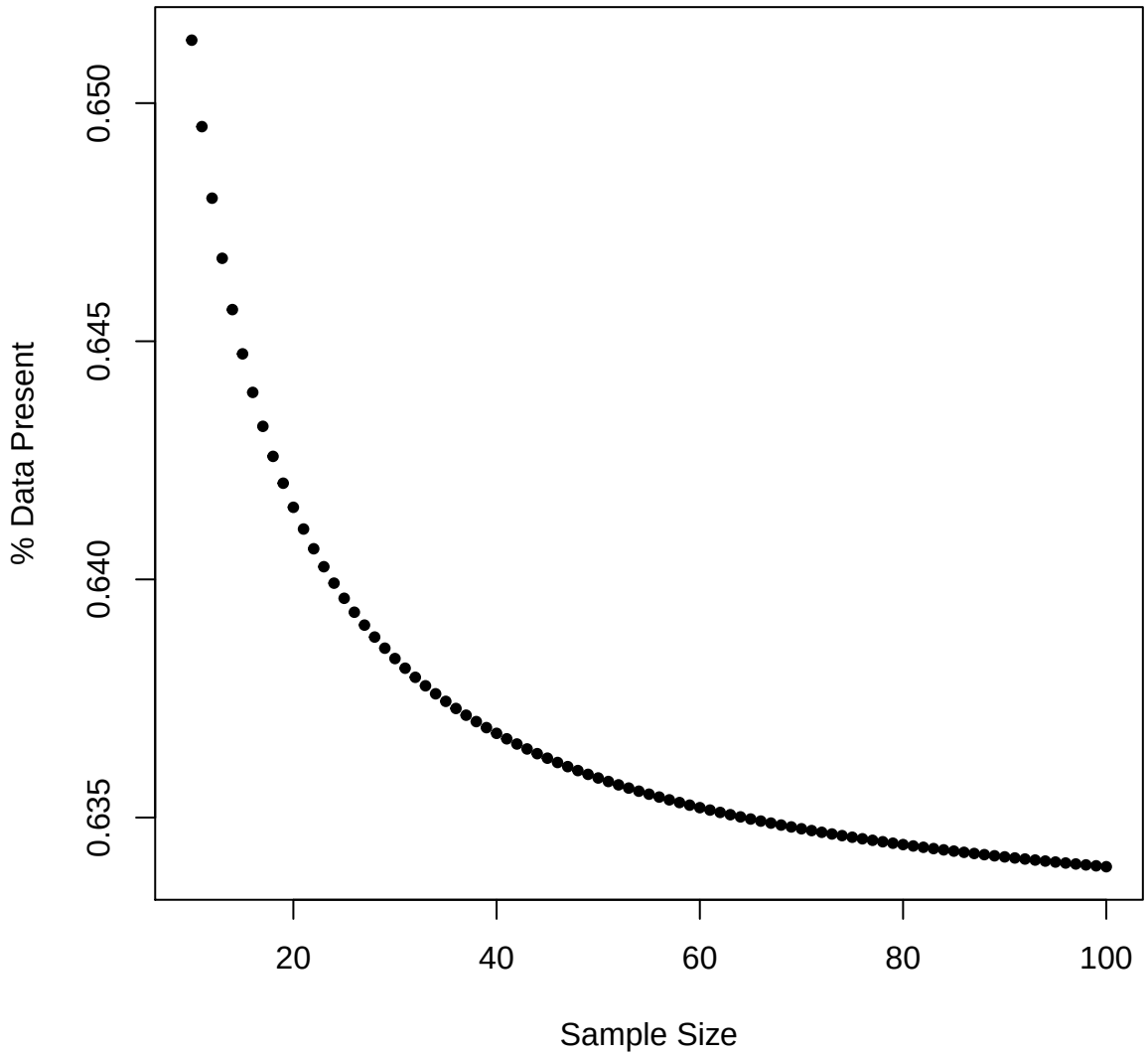
In *All of Nonparametric Statistics*, Larry Wasserman states:

> There is a tendency to treat the bootstrap as a panacea for all problems. But the bootstrap requires regularity conditions to yield valid answers. It should not be applied blindly.

The bootstrap is easy to motivate, but it is quite tricky to implement outside of the very standard problems. It sometimes requires deeper knowledge of statistical theory than likelihood-based inference.

Bootstrap Sample

For a sample of size n, what percentage of the data is present in any given bootstrap sample?



Bootstrap CIs

Suppose that $\theta = T(F)$ and $\hat{\theta} = T(\hat{F}_{\boldsymbol{x}})$.

We can use the bootstrap to generate data from $\hat{F}_{\boldsymbol{x}}$.

For $b = 1, 2, \ldots, B$, we draw $x_1^{*(b)}, x_2^{*(b)}, \ldots, x_n^{*(b)}$ as iid realiztions from $\hat{F}_{\boldsymbol{x}}$, and calculate $\hat{\theta}^{*(b)} = T(\hat{F}_{\boldsymbol{x}^{*(b)}})$.

Let p^*_α be the α percentile of $\left\{\hat{\theta}^{*(1)}, \hat{\theta}^{*(2)}, \ldots, \hat{\theta}^{*(B)}\right\}$.

Let's discuss several ways of calculating confidence intervals for $\theta = T(F)$.

Invoking the CLT

If we have evidence that the central limit theorem can be applied, we can form the $(1-\alpha)$ CI as:

$$(\hat{\theta} - |z_{\alpha/2}| \, \text{se}^*, \hat{\theta} + |z_{\alpha/2}| \, \text{se}^*)$$

where se* is the bootstrap standard error calculated as

$$\text{se}^* = \sqrt{\frac{1}{B}\sum_{b=1}^{B}\left(\hat{\theta}^{*(b)} - \frac{1}{B}\sum_{k=1}^{B}\hat{\theta}^{*(k)}\right)^2}.$$

Note that se* serves as estimate of se$(\hat{\theta})$.

Note that to get this confidence interval we need to justify that the following pivotal statistics are approximately Normal(0,1):

$$\frac{\hat{\theta} - \theta}{\text{se}(\hat{\theta})} \approx \frac{\hat{\theta} - \theta}{\text{se}^*}$$

Percentile Interval

If a *monotone* function $m(\cdot)$ exists so that $m\left(\hat{\theta}\right) \sim \text{Normal}(m(\theta), b^2)$, then we can form the $(1-\alpha)$ CI as:

$$\left(p^*_{\alpha/2}, p^*_{1-\alpha/2}\right)$$

where recall that in general p^*_α is the α percentile of bootstrap estimates $\left\{\hat{\theta}^{*(1)}, \hat{\theta}^{*(2)}, \ldots, \hat{\theta}^{*(B)}\right\}$

Pivotal Interval

Suppose we can calculate percentiles of $\hat{\theta} - \theta$, say q_α. Note that the α percentile of $\hat{\theta}$ is $q_\alpha + \theta$. The $1-\alpha$ CI is

$$(\hat{\theta} - q_{1-\alpha/2}, \hat{\theta} - q_{\alpha/2})$$

which comes from:

$$\begin{aligned}
1 - \alpha &= \Pr(q_{\alpha/2} \leq \hat{\theta} - \theta \leq q_{1-\alpha/2}) \\
&= \Pr(-q_{1-\alpha/2} \leq \theta - \hat{\theta} \leq -q_{\alpha/2}) \\
&= \Pr(\hat{\theta} - q_{1-\alpha/2} \leq \theta \leq \hat{\theta} - q_{\alpha/2})
\end{aligned}$$

Suppose the sampling distribution of $\hat{\theta}^* - \hat{\theta}$ is an approximation for that of $\hat{\theta} - \theta$.

If p^*_α is the α percentile of $\hat{\theta}^*$ then, $p^*_\alpha - \hat{\theta}$ is the α percentile of $\hat{\theta}^* - \hat{\theta}$.

Therefore, $p^*_\alpha - \hat{\theta}$ is the bootstrap estimate of q_α. Plugging this into $(\hat{\theta} - q_{1-\alpha/2}, \hat{\theta} - q_{\alpha/2})$, we get the following $(1-\alpha)$ bootstrap CI:

$$\left(2\hat{\theta} - p^*_{1-\alpha/2}, 2\hat{\theta} - p^*_{\alpha/2}\right).$$

Studentized Pivotal Interval

In the previous scenario, we needed to assume that the sampling distribution of $\hat{\theta}^* - \hat{\theta}$ is an approximation for that of $\hat{\theta} - \theta$. Sometimes this will not be the case and instead we can studentize this pivotal quantity. That is, the distribution of

$$\frac{\hat{\theta} - \theta}{\hat{\text{se}}\left(\hat{\theta}\right)}$$

is well-approximated by that of

$$\frac{\hat{\theta}^* - \hat{\theta}}{\hat{\text{se}}\left(\hat{\theta}^*\right)}.$$

Let z^*_α be the α percentile of

$$\left\{\frac{\hat{\theta}^{*(1)} - \hat{\theta}}{\hat{\text{se}}\left(\hat{\theta}^{*(1)}\right)}, \ldots, \frac{\hat{\theta}^{*(B)} - \hat{\theta}}{\hat{\text{se}}\left(\hat{\theta}^{*(B)}\right)}\right\}.$$

Then a $(1-\alpha)$ bootstrap CI is

$$\left(\hat{\theta} - z^*_{1-\alpha/2}\hat{\text{se}}\left(\hat{\theta}\right), \hat{\theta} - z^*_{\alpha/2}\hat{\text{se}}\left(\hat{\theta}\right)\right).$$

Exercise: Why?

How do we obtain $\hat{\text{se}}\left(\hat{\theta}\right)$ and $\hat{\text{se}}\left(\hat{\theta}^{*(b)}\right)$?

If we have an analytical formula for these, then $\hat{\text{se}}(\hat{\theta})$ is calculated from the original data and $\hat{\text{se}}(\hat{\theta}^{*(b)})$ from the bootstrap data sets. But we probably don't since we're using the bootstrap.

Instead, we can calculate:

$$\hat{\text{se}}\left(\hat{\theta}\right) = \sqrt{\frac{1}{B}\sum_{b=1}^{B}\left(\hat{\theta}^{*(b)} - \frac{1}{B}\sum_{k=1}^{B}\hat{\theta}^{*(k)}\right)^2}.$$

This is what we called se* above. But what about $\hat{\text{se}}\left(\hat{\theta}^{*(b)}\right)$?

To estimate $\hat{\text{se}}\left(\hat{\theta}^{*(b)}\right)$ we need to do a double bootstrap. For each bootstrap sample b we need to bootstrap that daat set another B times to calculate:

$$\hat{\text{se}}\left(\hat{\theta}^{*(b)}\right) = \sqrt{\frac{1}{B}\sum_{v=1}^{B}\left(\hat{\theta}^{*(b)*(v)} - \frac{1}{B}\sum_{k=1}^{B}\hat{\theta}^{*(b)*(k)}\right)^2}$$

where $\hat{\theta}^{*(b)*(v)}$ is the statistic calculated from bootstrap sample v within bootstrap sample b. This can be very computationally intensive, and it requires a large sample size n.

Bootstrap Hypothesis Testing

As we have seen, hypothesis testing and confidence intervals are very related. For a simple null hypothesis, a bootstrap hypothesis test p-value can be calculated by finding the minimum α for which the $(1-\alpha)$ CI does not contain the null hypothesis value. You showed this on your homework.

The general approach is to calculate a test statistic based on the observed data. Then the null distribution of this statistic is approximated by forming bootstrap test statistics under the scenario that the null hypothesis is true. This can often be accomplished because the $\hat{\theta}$ estimated from the observed data is the *population* parameter from the bootstrap distribution.

Example: *t*-test

Suppose $X_1, X_2, \ldots, X_m \overset{\text{iid}}{\sim} F_X$ and $Y_1, Y_2, \ldots, Y_n \overset{\text{iid}}{\sim} F_Y$. We wish to test $H_0 : \mu(F_X) = \mu(F_Y)$ vs $H_1 : \mu(F_X) \neq \mu(F_Y)$. Suppose that we know $\sigma^2(F_X) = \sigma^2(F_Y)$ (if not, it is straightforward to adjust the proecure below).

Our test statistic is

$$t = \frac{\overline{x} - \overline{y}}{\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right) s^2}}$$

where s^2 is the pooled sample variance.

Note that the bootstrap distributions are such that $\mu(\hat{F}_{X^*}) = \overline{x}$ and $\mu(\hat{F}_{Y^*}) = \overline{y}$. Thus we want to center the bootstrap t-statistics about these known means.

Specifically, for a bootstrap data set $x^* = (x_1^*, x_2^*, \ldots, x_m^*)^T$ and $y^* = (y_1^*, y_2^*, \ldots, y_n^*)^T$, we form null t-statistic

$$t^* = \frac{\overline{x}^* - \overline{y}^* - (\overline{x} - \overline{y})}{\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right) s^{2*}}}$$

where again s^{2*} is the pooled sample variance.

In order to obtain a p-value, we calculate $t^{*(b)}$ for $b = 1, 2, \ldots, B$ bootstrap data sets.

The p-value of t is then the proportion of bootstrap statistics as or more extreme than the observed statistic:

$$\text{p-value}(t) = \frac{1}{B} \sum_{b=1}^{B} 1\left(|t^{*(b)}| \geq |t|\right).$$

Parametric Bootstrap

Suppose $X_1, X_2, \ldots, X_n \overset{\text{iid}}{\sim} F_\theta$ for some parametric F_θ. We form estimate $\hat{\theta}$, but we don't have a known sampling distribution we can use to do inference with $\hat{\theta}$.

The parametric bootstrap generates bootstrap data sets from $F_{\hat{\theta}}$ rather than from the edf. It proceeds as we outlined above for these bootstrap data sets.

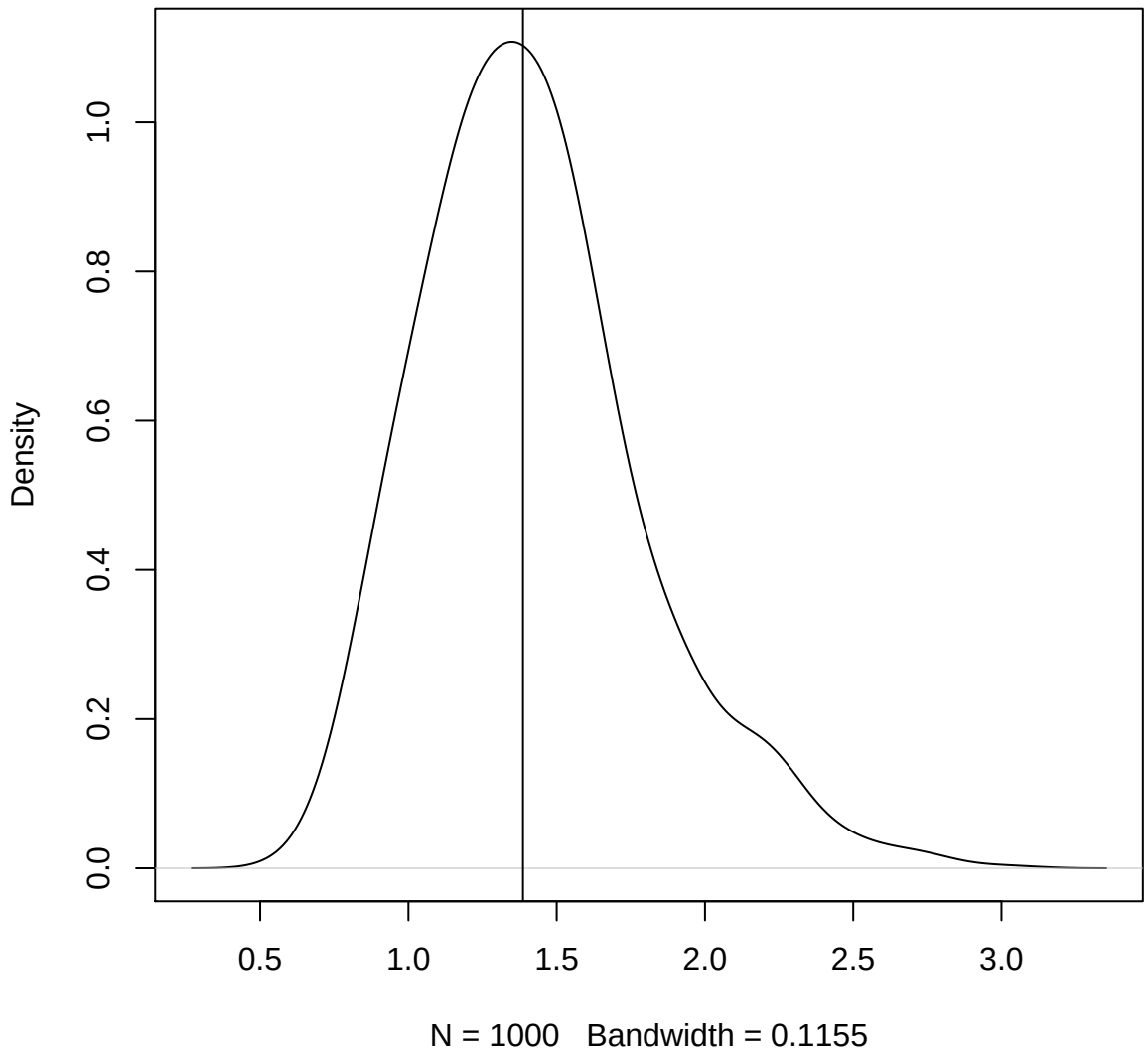
Example: Exponential Data

In the homework, you will be performing a bootstrap t-test of the mean and a bootstrap percentile CI of the median for the following Exponential(λ) data:

```
> set.seed(1111)
> pop.mean <- 2
> X <- matrix(rexp(1000*30, rate=1/pop.mean), nrow=1000, ncol=30)
```

Let's construct a pivotal bootstrap CI of the median here instead.

```
> # population median 2*log(2)
> pop_med <- qexp(0.5, rate=1/pop.mean); pop_med
[1] 1.386294
>
> obs_meds <- apply(X, 1, median)
> plot(density(obs_meds, adj=1.5), main=" "); abline(v=pop_med)
```



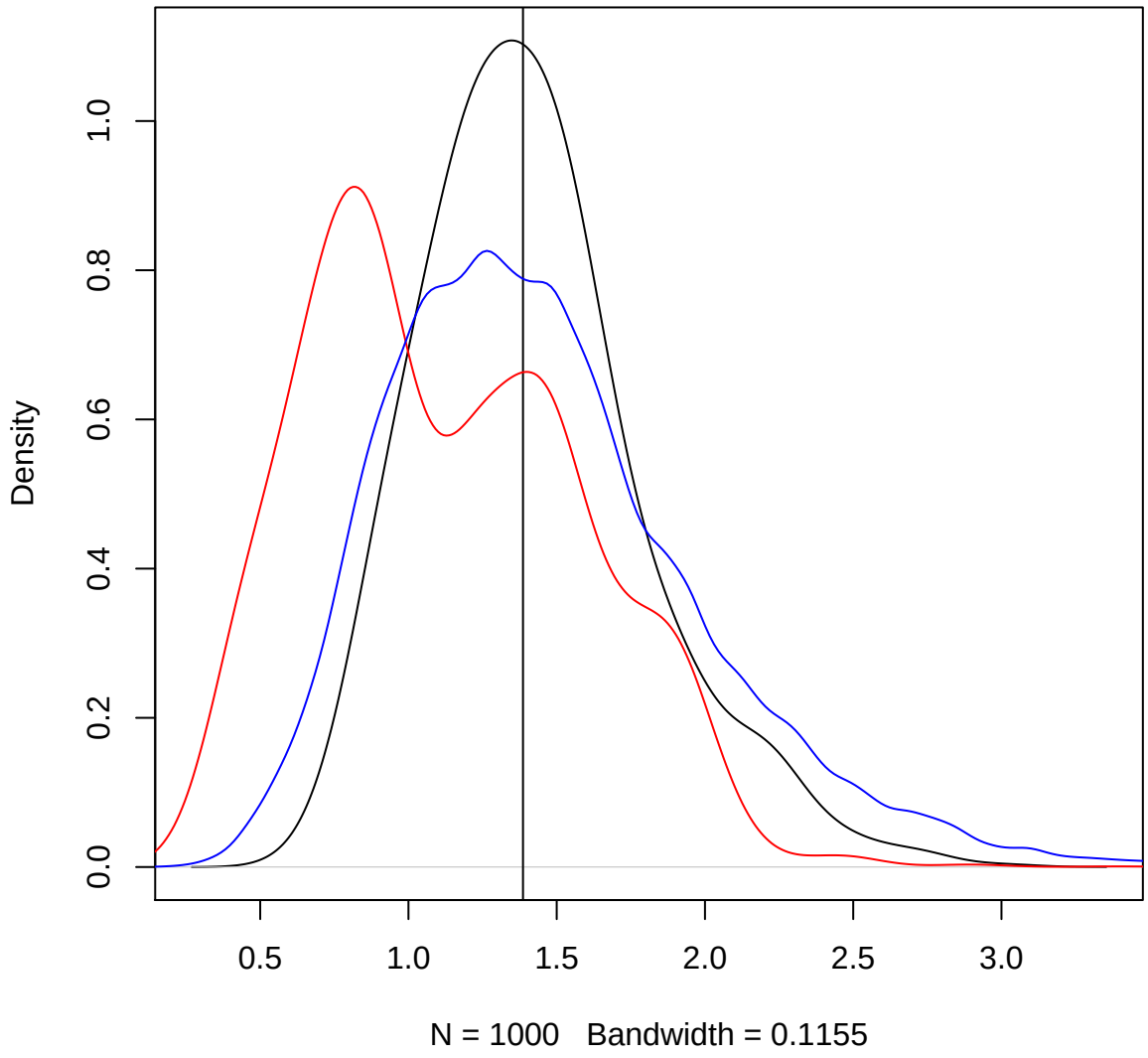
Some embarrassingly inefficient code to calculate bootstrap medians.

```
> B <- 1000
> boot_meds <- matrix(0, nrow=1000, ncol=B)
>
```

```
> for(b in 1:B) {
+   idx <- sample(1:30, replace=TRUE)
+   boot_meds[,b] <- apply(X[,idx], 1, median)
+ }
```

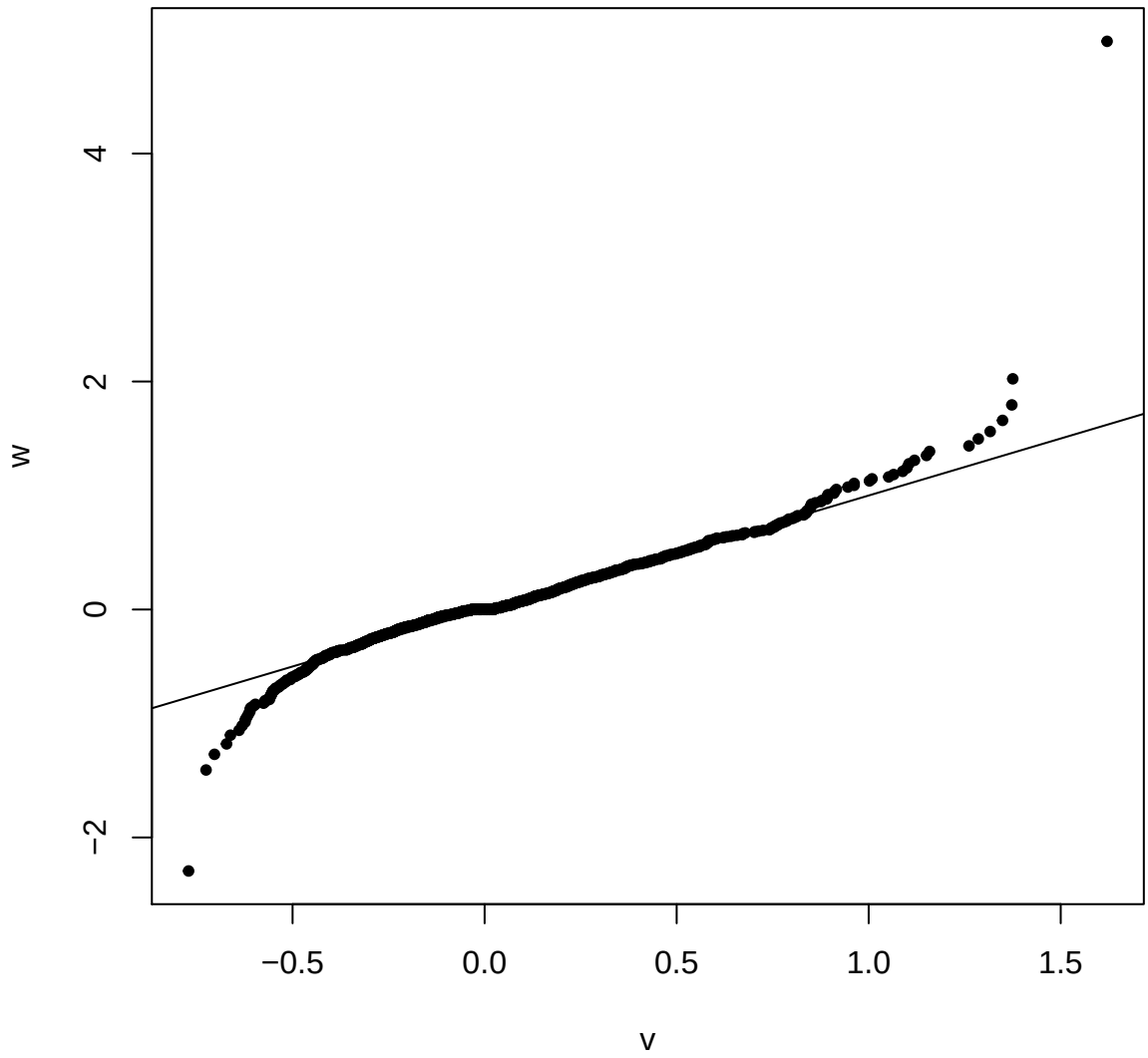
Plot the bootstrap medians.

```
> plot(density(obs_meds, adj=1.5), main=" "); abline(v=pop_med)
> lines(density(as.vector(boot_meds[1:4,]), adj=1.5), col="red")
> lines(density(as.vector(boot_meds), adj=1.5), col="blue")
```



Compare sampling distribution of $\hat{\theta} - \theta$ to $\hat{\theta}^* - \hat{\theta}$.

```
> v <- obs_meds - pop_med
> w <- as.vector(boot_meds - obs_meds)
> qqplot(v, w, pch=20); abline(0,1)
```



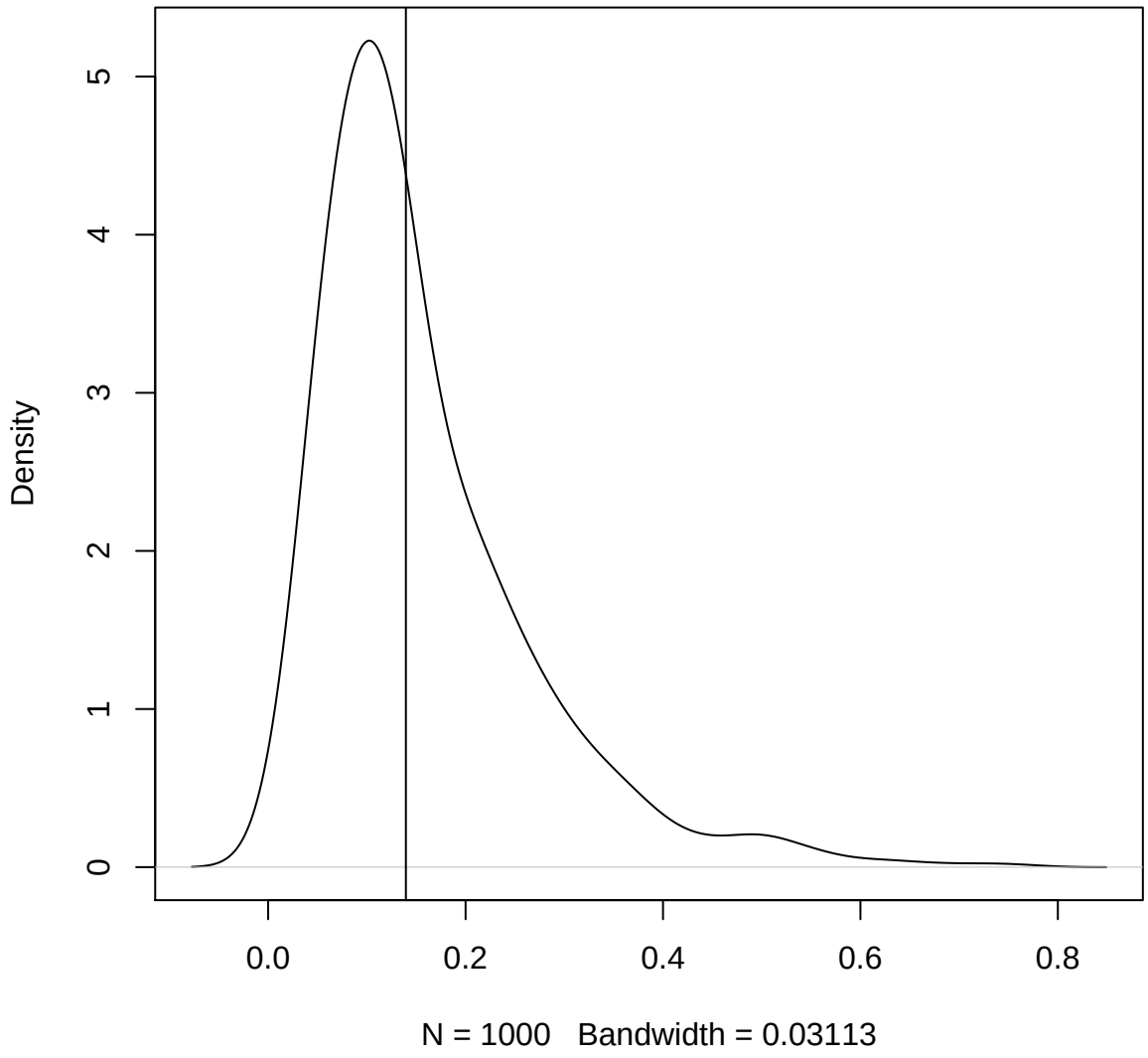
Does a 95% bootstrap pivotal interval provide coverage?

```
> ci_lower <- apply(boot_meds, 1, quantile, probs=0.975)
> ci_upper <- apply(boot_meds, 1, quantile, probs=0.025)
>
> ci_lower <- 2*obs_meds - ci_lower
> ci_upper <- 2*obs_meds - ci_upper
>
> ci_lower[1]; ci_upper[1]
[1] 0.8958224
[1] 2.163612
>
> cover <- (pop_med >= ci_lower) & (pop_med <= ci_upper)
> mean(cover)
[1] 0.808
>
> # :-(
```

Let's check the bootstrap variances.

```
> sampling_var <- var(obs_meds)
> boot_var <- apply(boot_meds, 1, var)
```

```
> plot(density(boot_var, adj=1.5), main=" ")
> abline(v=sampling_var)
```



We repeated this simulation over a range of n and B.

n	B	*coverage*	*avg CI width*
1e+02	1000	0.868	0.7805404
1e+02	2000	0.872	0.7882278
1e+02	4000	0.865	0.7852837
1e+02	8000	0.883	0.7817222
1e+03	1000	0.923	0.2465840
1e+03	2000	0.909	0.2477463
1e+03	4000	0.915	0.2475550
1e+03	8000	0.923	0.2458167
1e+04	1000	0.935	0.0781421
1e+04	2000	0.937	0.0784541
1e+04	4000	0.942	0.0784559
1e+04	8000	0.948	0.0785591
1e+05	1000	0.949	0.0246918

n	*B*	*coverage*	*avg CI width*
1e+05	2000	0.942	0.0246938

Extras

Source

License

Source Code

Session Information

```
> sessionInfo()
R version 3.6.0 (2019-04-26)
Platform: x86_64-apple-darwin15.6.0 (64-bit)
Running under: macOS  10.15.3

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods
[7] base

other attached packages:
 [1] broom_0.5.2         car_3.0-6
 [3] carData_3.0-3       HardyWeinberg_1.6.3
 [5] Rsolnp_1.16         mice_3.8.0
 [7] forcats_0.5.0       stringr_1.4.0
 [9] dplyr_0.8.4         purrr_0.3.3
[11] readr_1.3.1         tidyr_1.0.2
[13] tibble_2.1.3        ggplot2_3.2.1
[15] tidyverse_1.3.0     knitr_1.28

loaded via a namespace (and not attached):
 [1] Rcpp_1.0.3         lubridate_1.7.4    lattice_0.20-40
 [4] utf8_1.1.4         assertthat_0.2.1   digest_0.6.25
 [7] truncnorm_1.0-8    R6_2.4.1           cellranger_1.1.0
[10] backports_1.1.5    reprex_0.3.0       evaluate_0.14
[13] highr_0.8          httr_1.4.1         pillar_1.4.3
[16] rlang_0.4.5        lazyeval_0.2.2     curl_4.3
[19] readxl_1.3.1       data.table_1.12.8  rstudioapi_0.11
[22] rmarkdown_2.1      labeling_0.3       foreign_0.8-75
[25] munsell_0.5.0      compiler_3.6.0     modelr_0.1.6
[28] xfun_0.12          pkgconfig_2.0.3    htmltools_0.4.0
[31] tidyselect_1.0.0   codetools_0.2-16   rio_0.5.16
[34] fansi_0.4.1        crayon_1.3.4       dbplyr_1.4.2
[37] withr_2.1.2        grid_3.6.0         nlme_3.1-144
```

```
[40] jsonlite_1.6.1    gtable_0.3.0      lifecycle_0.1.0
[43] DBI_1.1.0         magrittr_1.5      scales_1.1.0
[46] zip_2.0.4         cli_2.0.2         stringi_1.4.6
[49] farver_2.0.3      fs_1.3.1          xml2_1.2.2
[52] generics_0.0.2    vctrs_0.2.3       openxlsx_4.1.4
[55] tools_3.6.0       glue_1.3.1        hms_0.5.3
[58] abind_1.4-5       parallel_3.6.0    yaml_2.2.1
[61] colorspace_1.4-1  rvest_0.3.5       haven_2.2.0
```