ROBUST STATISTICS FOR PERSONALITY AND INDIVIDUAL DIFFERENCES

BERTINORO, ITALY, JULY 16-21, 2011

Organizers: Jens B. Asendorpf & Marco Perugini

Supported by EAPP and ISSID

The goal is to briefly summarize the many issues and techniques that have been developed. The emphasis is on a conceptual understanding of modern robust methods plus the ability to apply robust methods using the software R. More details can be found in

Wilcox, R. R. (2011) *Modern Statistics for the Social and Behavioral Sciences: A Practical Introduction*. New York: Chapman & Hall/CRC press.

Wilcox, R. R. (in press). *Introduction to Robust Estimation and Hypothesis Testing*. 3rd Edition. San Diego, CA: Academic Press.

Other books that might be of interest:

Wilcox, R. R. (2010). *Fundamentals of Modern Statistical Methods: Substantially Improving Power and Accuracy*, 2nd Edition. New York: Springer.

This book provides a very nontechnical introduction to robust methods assuming little or no training in statistics.

Wilcox, R. R. (2009). *Basics Statistics: Understanding Conventional Methods and Modern Insights*. New York: Oxford University Press.

I use this book in our undergraduate, introductory statistics course. It covers the usual topics, but at the end of each chapter is a brief description of advances and insights from the last half century. In effect, the need for robust methods is described and illustrated plus a very short description of a few modern methods aimed at dealing with nonnormality and heteroscedasticity.

TENTATIVE SCHEDULE:

Day 1: Basic Issues and Methods

I. Practical concerns with methods based on means and least squares regression

II. Strategies that might seem reasonable but are highly unsatisfactory.

III. Robust Measures of location and Variation

IV. Estimating Standard Errors of Robust Estimators

V. Bootstrap Methods

VI. Computing confidence intervals and testing hypotheses

VII. Comparing Two independent Groups.

- a. Shift function
- b. Comparing robust measures of location:
  trimmed means, medians, M-estimators
- c. Modern rank-based methods

VIII. Measuring effect size

Day 2: One-way and higher designs, including repeated measures

I. Two-Sample repeated Measures.

- a. Comparing all quantiles
- b. Three perspectives based on robust measures of location

III. One-way ANOVA

IV. Multiple comparisons

V. Two-way ANOVA

a. Methods based on robust measures of location

b. Rank-based methods

VI. Multiple comparisons

VII. Three-way ANOVA

VIII. Multivariate methods and issues

- a. Measures of location
- b. Measures of scatter
- c. Detecting outliers
- d. One-sample hypothesis testing
- e. Two-sample hypothesis testing
- f. Projection-type analog of Wilcoxon--Mann--Whitney,
  with comments on effect size
- g. Comparisons based on depth of points.

IX. Robust Principal Components

Day 3: Regression and Measures of Association

I. Robust measures of correlation

```
I. Robust regression estimators
II. Eliminating leverage points
III.  Inferential methods
IV.  Dealing with curvature
V.   Measures of association based on a given fit
VI. Moderator analysis
VII. Mediator  analysis
VIII. ANCOVA
```

1 SOME PRELIMINARY REMARKS

HUNDREDS OF PAPERS PUBLISHED DURING THE LAST HALF CENTURY HAVE DEALT WITH FUNDAMENTAL CONCERNS REGARDING CLASSIC, ROUTINELY USED METHODS FOR COMPARING GROUPS AND STUDYING ASSOCIATIONS.

PRACTICAL REASONS FOR TAKING ADVANTAGE OF MORE MODERN METHODS:

- The possibility of substantially higher power relative to methods that assume normality and homoscedasticity.
- More accurate confidence intervals and better control over the probability of a Type I error.
- A deeper and more accurate sense of how groups compare and how variables are related. This includes better measures of effect size and measures of association.

THREE MAJOR INSIGHTS WITH MANY RELATED RESULTS:

- Heavy-tailed distributions (outliers are likely to occur) are commonly encountered and can destroy power when using means or least squares regression, they can result in an inaccurate sense of what is typical, and they can result in seemingly small measures of effect size that are in fact large from a graphical point of view, and strong associations among variables can be missed.
- The sample size needed to assume normality, when using means, can be very large, contrary to what was once believed. In some situations a sample size greater than 300 is required, as will be illustrated.
- Heteroscedasticity can be much more serious than once thought.

BROAD GOALS

- Understand when and why classic routinely used methods perform poorly relative to more modern methods.
- Understand why some seemingly natural methods for dealing with nonnormality, outliers and heteroscedasticity are generally unsatisfactory.
- Learn how to deal with nonnormality, outliers and heteroscedasticity in a theoretically sound manner. Technical details are kept to a minimum, but it is important to stress what works well and what does not.
- Elaborate on what various methods tell us and what they don't tell us.

ROBUST PARAMETERS AND HEAVY-TAILED DISTRIBUTIONS

Roughly, a parameter is said to be robust if arbitrarily small changes in a distribution cannot have an arbitrarily large effect on the value of the parameter. The population mean μ and variance σ^2 are not robust. In practical terms, regardless of how large the sample size might be, methods based on the sample mean can be highly unsatisfacory in terms of both power and measures of effect size. When using Pearson's correlation, our understanding of the strength of the association could be highly misleading and true associations might be missed.

There are mathematical methods for characterizing the degree of robustness enjoyed by a parameter, but here we only illustrate the notion of robustness and why it has practical importance. However, when dealing with estimators, such as the mean and median, one aspect of the notion of robustness is easy to describe. And it is related to one method used to characterize the robustness of the population mean and variance. This is the breakdown point of an estimator, which is discussed later.

A common misconception is that if a distribution is symmetric, methods based on means are robust. But because σ^2 is not robust, this is not necessarily true. Figure 1 illustrates the basic concern.

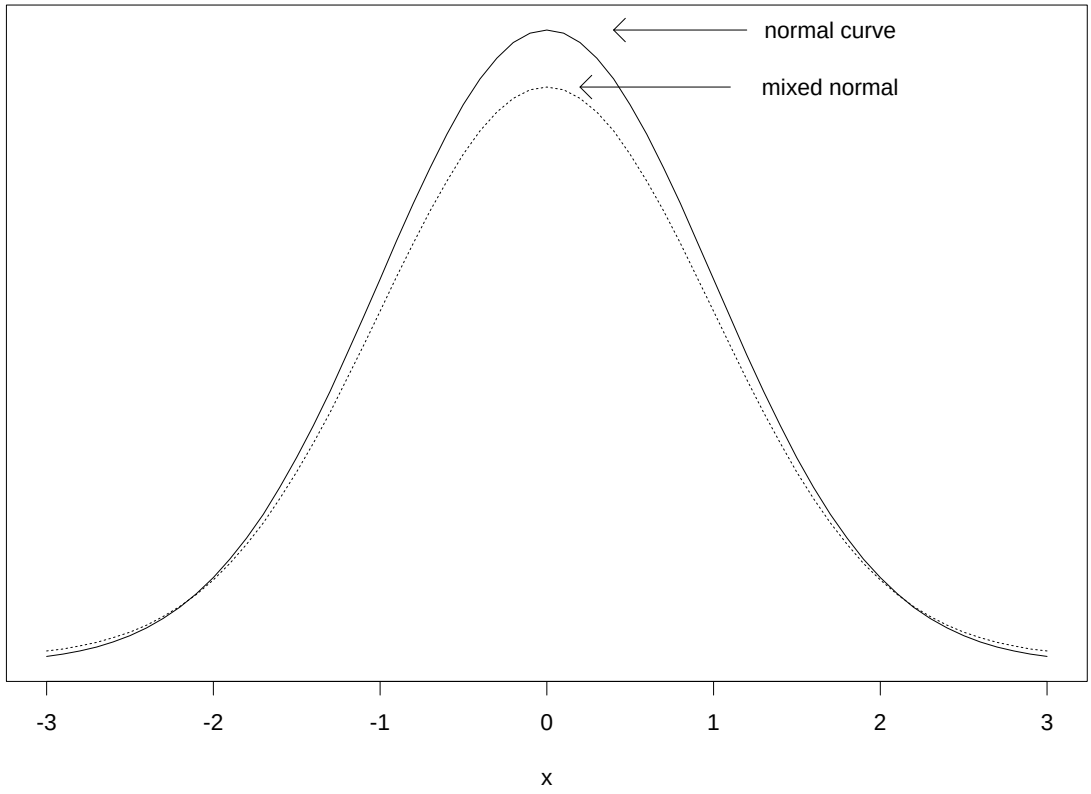

Figure 1: For normal distributions, increasing the standard deviation from 1 to 1.5 results in a substantial change in the distribution, as illustrated in a basic statistics course. But when considering non-normal distributions, a seemingly large difference in the variances does not necessarily mean that there is a large difference in the graphs of the distributions. The two curves shown here have an obvious similarity, yet the variances are 1 and 10.9.

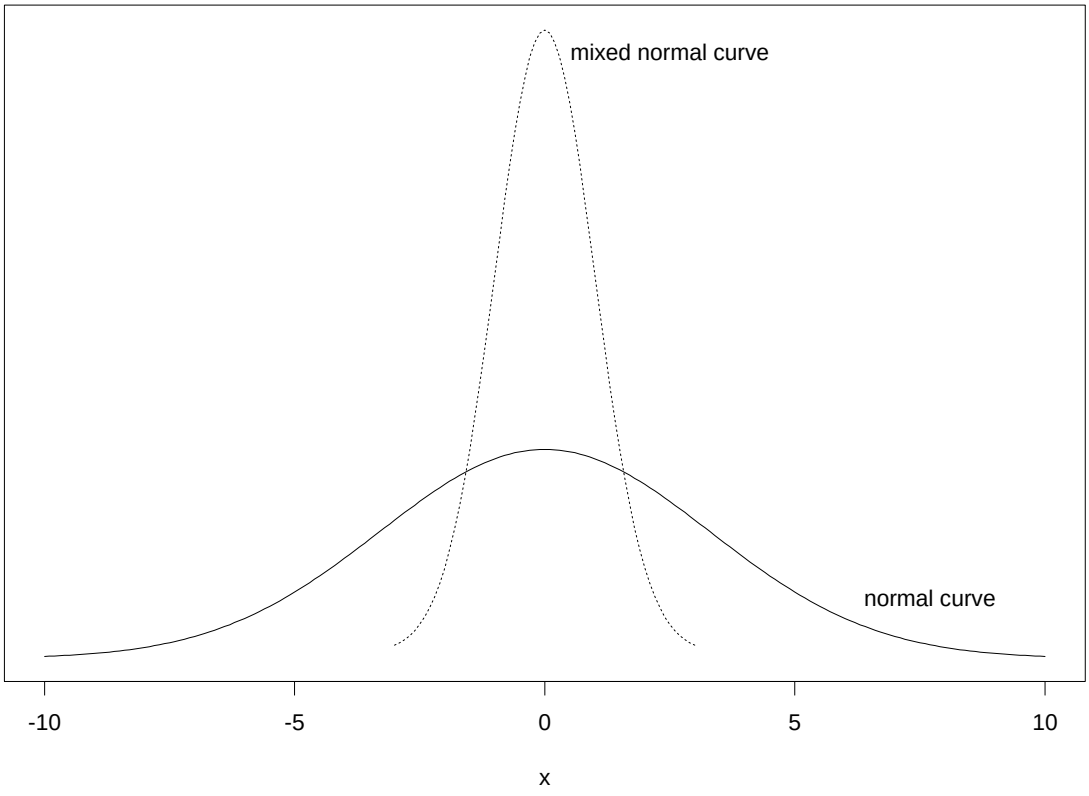

Figure 2: Two probability curves having equal means and variances.

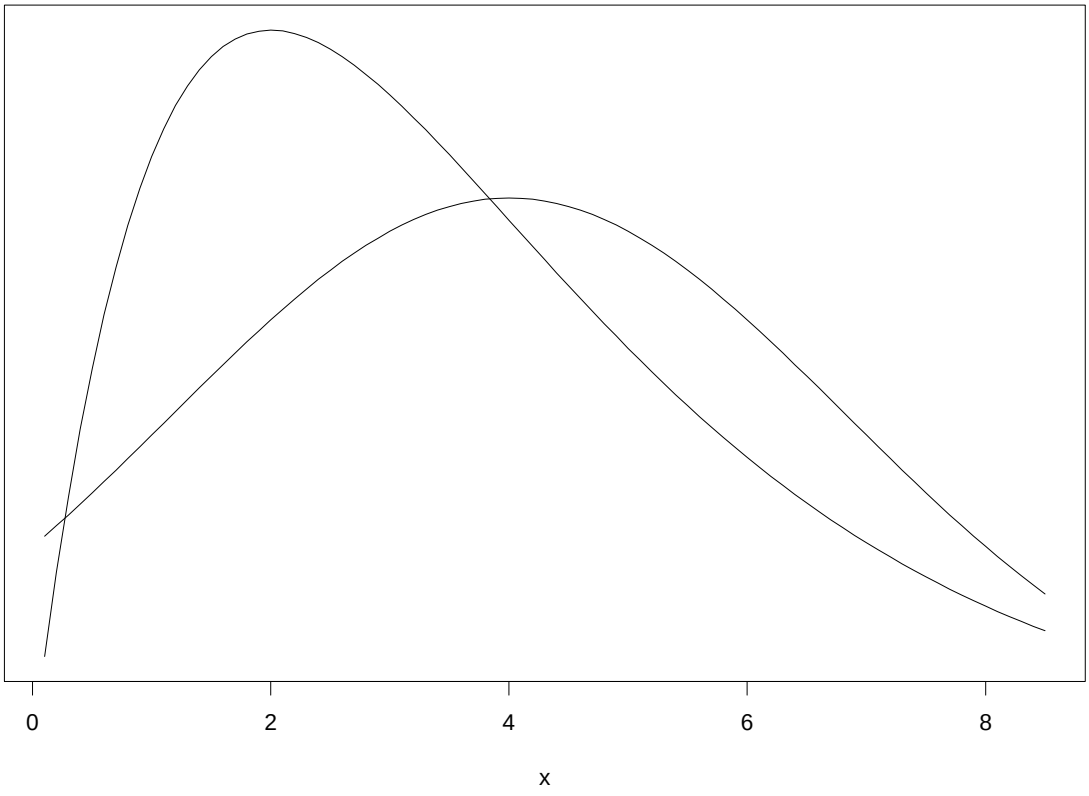

Figure 3: Two probability curves having equal means and variances.

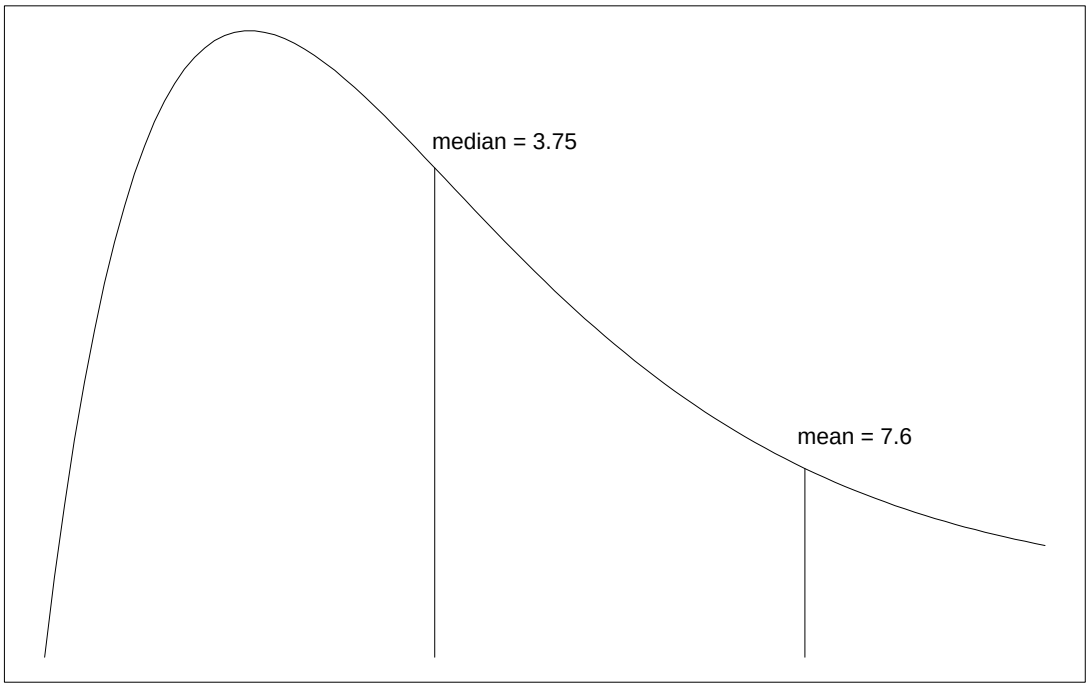

Figure 4: The population mean can be located in the extreme portion of the tail of a distribution. That is, the mean can represent a highly atypical response.

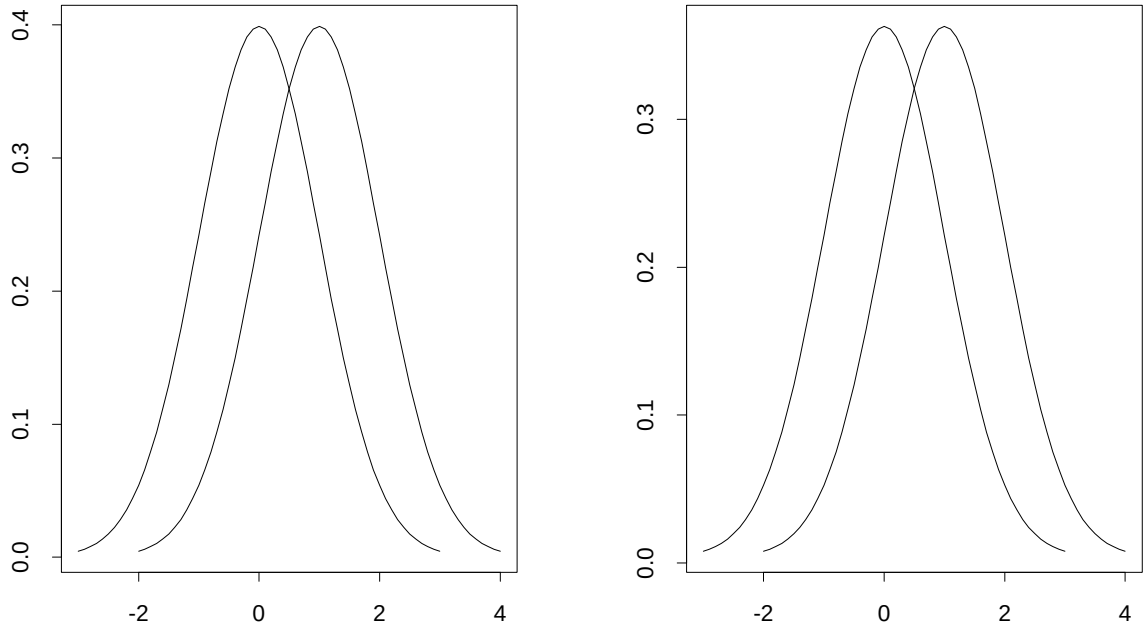

Figure 5: In the left panel, power is .96 based on Student's T, $\alpha = .05$. But in the left panel, power is only .28, illustrating the general principle that slights changes in the distributions being compared can have a large impact on the ability to detect true differences between the population means.

DEALING WITH OUTLIERS: THE FIRST TWO STRATEGIES LISTED HERE ARE REASONABLE. THE NEXT IS RELATIVELY INEFFECTIVE AND THE FOURTH IS A SEEMINGLY NATURAL STRATEGY THAT SHOULD NEVER BE USED.

1. TRIM. BEST-KNOWN EXAMPLE IS THE MEDIAN, BUT TYPICALLY IT TRIMS TOO MUCH

2. REMOVE OUTLIERS AND AVERAGE THE VALUES THAT REMAIN.

3. TRANSFORM THE DATA, FOR EXAMPLE TAKE LOGARITHMS, BUT THIS APPROACH PERFORMS POORLY BY MODERN STANDARDS.

4. HIGHLY UNSATISFACTORY STRATEGY: DISCARD OUTLIERS AND APPLY STANDARD HYPOTHESIS TESTING METHODS FOR MEANS USING THE REMAINING DATA. IF, FOR EXAMPLE THE SAMPLE SIZE IS REDUCED FROM n to m AFTER TRIMMING, USE A METHOD FOR MEANS THAT ASSUMES WE HAVE m OBSERVATIONS. THIS RESULTS IN USING THE WRONG STANDARD ERROR. METHODS THAT REMOVE OUTLIERS CAN BE USED, BUT IT IS IMPERATIVE THAT A CORRECT ESTIMATE OF THE STANDARD BE USED, WHICH DEPENDS ON HOW OUTLIERS ARE TREATED. (DETAILS ARE GIVEN LATER.)

ALL INDICATIONS ARE THAT TYPICALLY, A 20% TRIMMED MEAN IS A GOOD CHOICE FOR GENERAL USE. MORE DETAILS WILL BE COVERED LATER IN THE COURSE. BUT THERE ARE ALWAYS EXCEPTIONS. NO SINGLE METHOD IS ALWAYS BEST. ONLY EFFECTIVE WAY TO DETERMINE WHETHER ANOTHER CHOICE MAKES A PRACTICAL DIFFERENCE IS TO TRY IT.

METHOD GENERALLY AGREED TO BE LEAST SATISFACTORY: X is an outlier if

$$\frac{|X - \bar{X}|}{s} > 2. \tag{1}$$

SUFFERS FROM MASKING. (The very presence of outliers causes them to be missed.)

BETTER: BOXPLOT RULE (USING IDEAL FOURTHS q_1 and q_2). Values greater than $q_2 + 1.5(q_2 - q_1)$ or less than $q_1 - 1.5(q_2 - q_1)$ are declared outliers

The R function

```
outbox(x,mbox=F)
```

applies this rule.

ALSO GOOD IS THE MAD-MEDIAN RULE:

$$\frac{|X - M|}{\text{MADN}} > 2.24. \tag{2}$$

where M is the median, MAD is the median of $|X_1 - M|, \ldots, |X_n - M|$ and MADN is MAD/.6745; it estimates σ under normality. MADN is a robust measure of variation: it's breakdown point is .5.

The R function

```
out(x)
```

MODIFIED ONE-STEP M-ESTIMATOR: REMOVE VALUES DECLARED OUTLIERS VIA THE MAD-MEDIAN RULE, AVERAGE THE REMAINING VALUES.

$$\bar{X}_{\text{mom}} = \frac{1}{n - \ell - u} \sum_{i=\ell+1}^{n-u} X_{(i)}$$

ONE-STEP M-ESTIMATOR:

X is declared an outlier if

$$\frac{|X - M|}{\text{MADN}} > 1.28. \tag{3}$$

If the number of small values declared outliers (ℓ) is not equal to the number of large values desclared outliers (u), the one-step M-estimator is given by

$$\bar{X}_{\text{os}} = \frac{1.26(\text{MADN})(u - \ell)}{n - u - \ell} + \frac{1}{n - u - \ell} \sum_{i=\ell+1}^{n-u} X_{(i)}. \tag{4}$$

The R function

```
onestep(x)
```

computes the one-step M-estimator just illustrated. And the function

```
mom(x)
```

computes the modified one-step M-estimator.

BREAKDOWN POINT OF AN ESTIMATOR: HOW MANY VALUES NEED TO BE CHANGED TO MAKE AN ESTIMATOR ARBITRARILY LARGE OR SMALL?

MEAN: 1 OUT OF N

20% TRIMMED MEAN, 20%

10% TRIMMED MEAN, 10%

ONE-STEP M-ESTIMATOR AND MEDIAN, ABOUT 50%.

(THE NOTION OF A BREAKDOWN POINT CAN BE EXTENDED TO PARAMETERS. THE MEAN AND VARIANCE HAVE A BREAKDOWN POINT OF ZERO, FOR THE 20% TRIMMED IT IS .2, AND FOR THE ONE-STEP AND MEDIAN IT IS .5, THE HIGHEST POSSIBLE VALUE.)

THE BREAKDOWN POINT IS VERY IMPORTANT, BUT OTHER CRITERIA NEED TO BE TAKEN INTO ACCOUNT WHEN DECIDING HOW TO ANALYZE DATA, AS WILL BE ILLUSTRATED.

n=40, can assume normality when using a mean?

NO: two major insights, one of which is particularly devastating.

Early studies considered unusually light-tailed distributions. And the implicit assumption was that if the sample mean has approximately a normal distribution, the usual T test statistic will have approximately Student's t distribution. This is not necessarily true, as will be illustrated. In practical terms, when using T, might need $n > 300$.

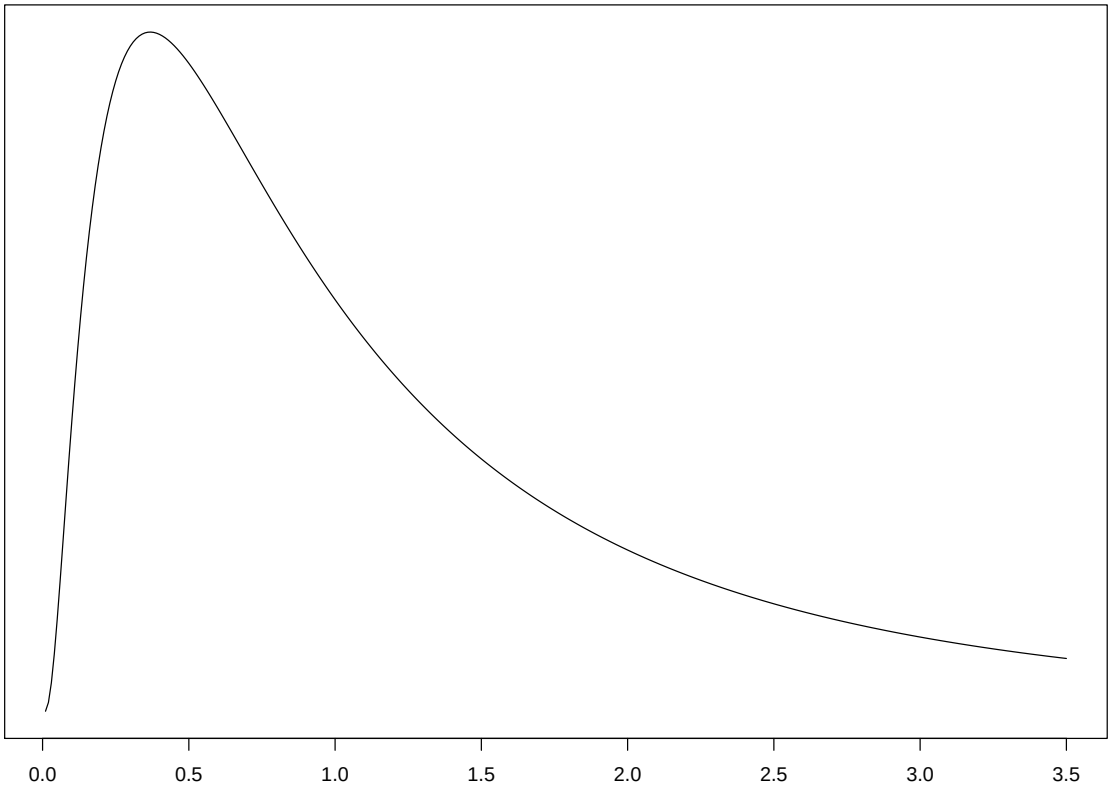

Figure 6: Shown is a lognormal distribution, which is skewed and relatively light-tailed, roughly meaning that the proportion of outliers found under random sampling is relatively small.

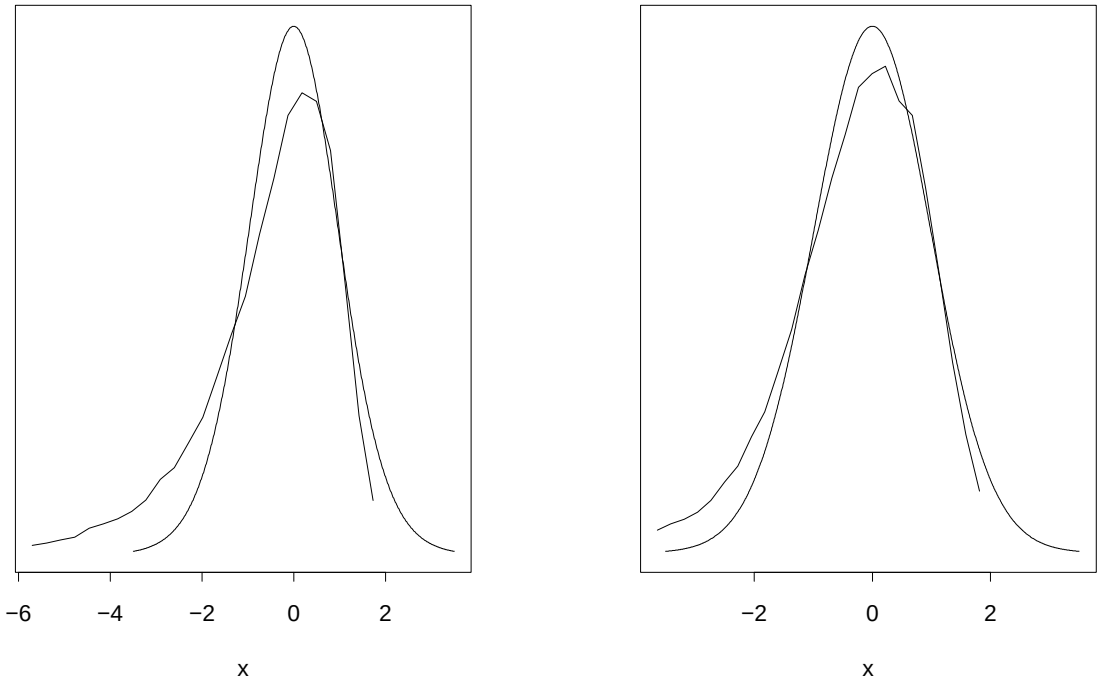

Figure 7: The left panel shows the distribution of 5000 T values, with each T value based on 20 observations generated from the lognormal distribution. The symmetric solid line is the distribution of T under normality. The right panel is the same as the left, only now the sample size is $n = 100$.

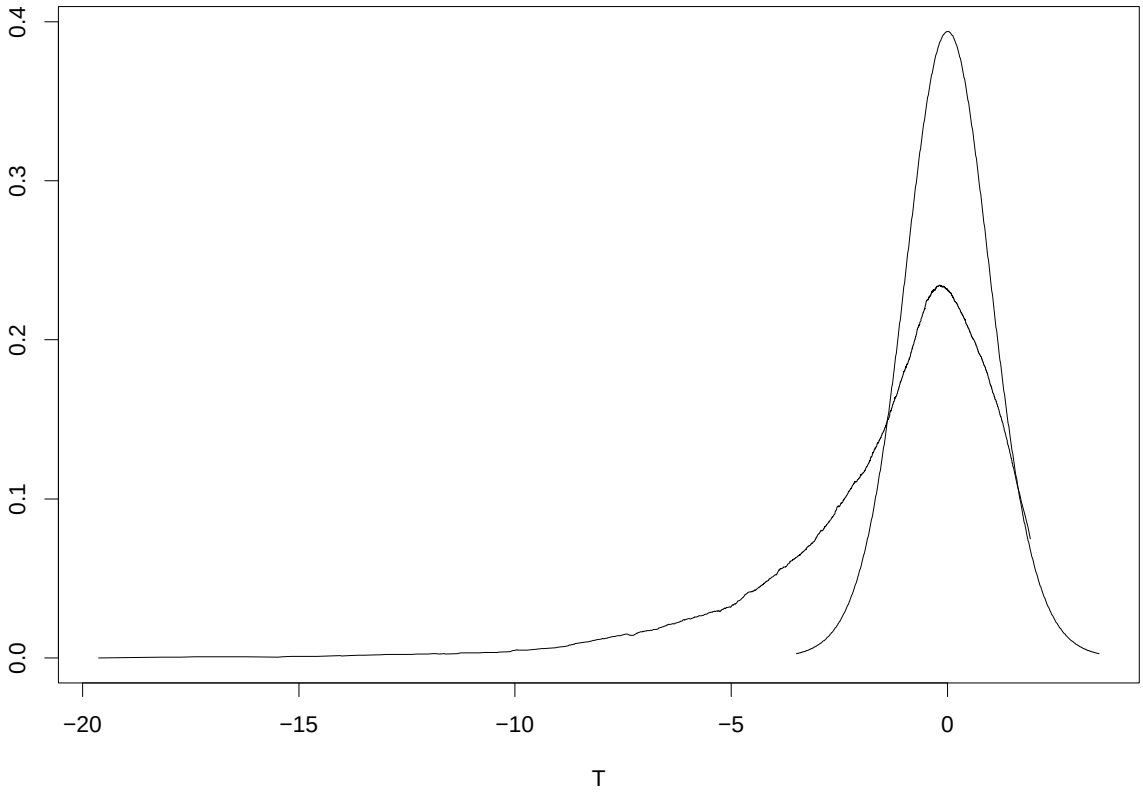

Figure 8: The asymmetric curve is the distribution of T when sampling 20 observations from a contaminated lognormal distribution, which is heavy-tailed. The symmetric curve is the distribution of T under normality.

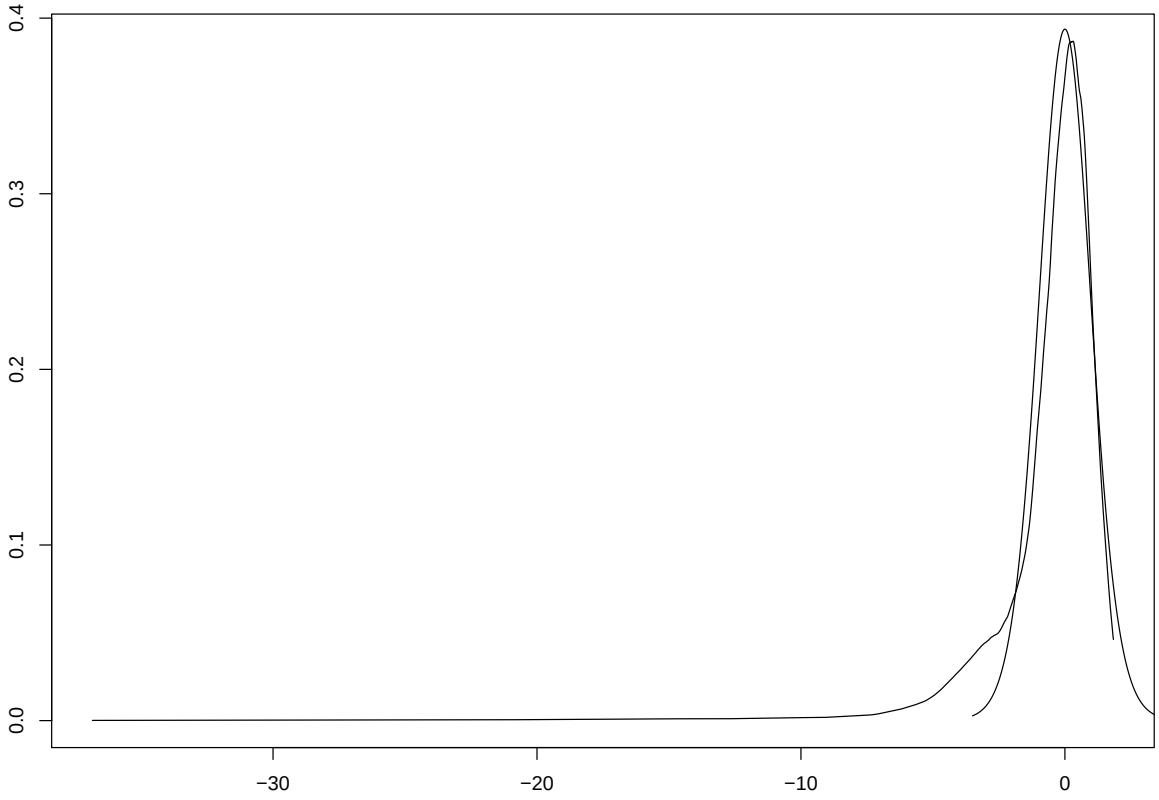

Figure 9: The distribution of T when randomly sampling from the hangover data. The symmetric curve is the distribution of T under normality.

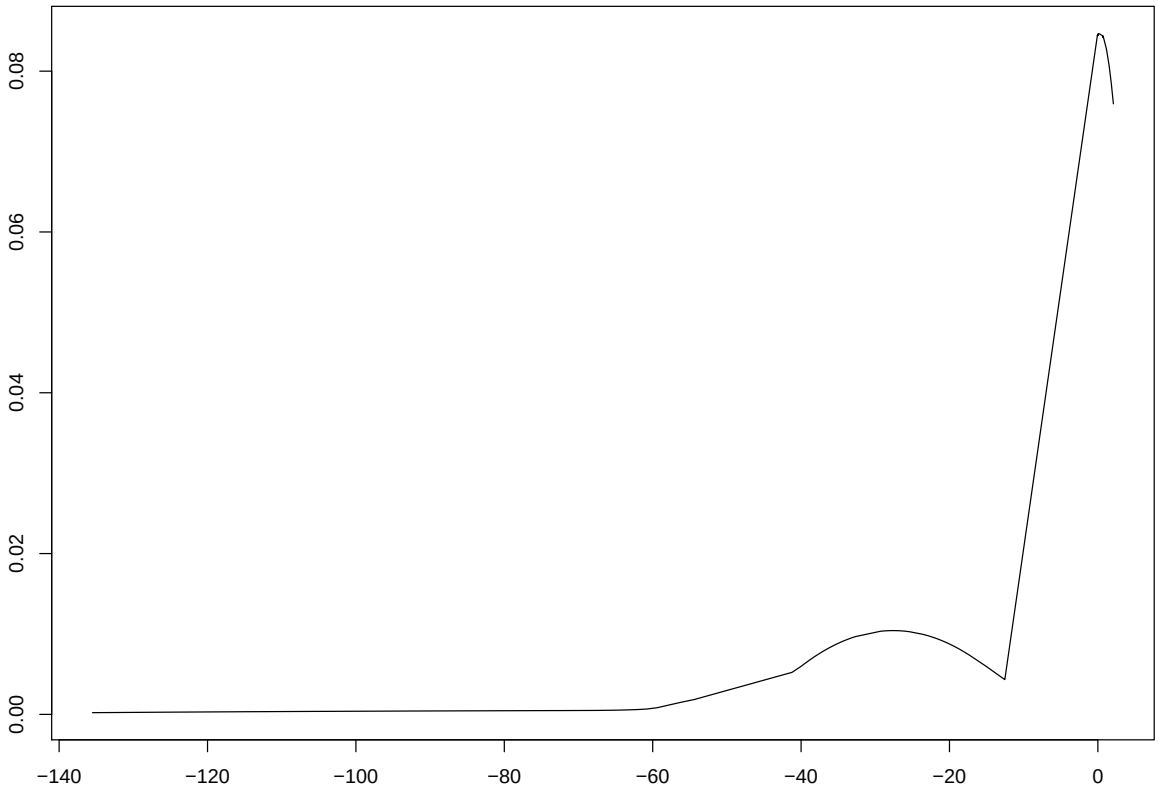

Figure 10: The distribution of T when randomly sampling from the sexual attitude data.

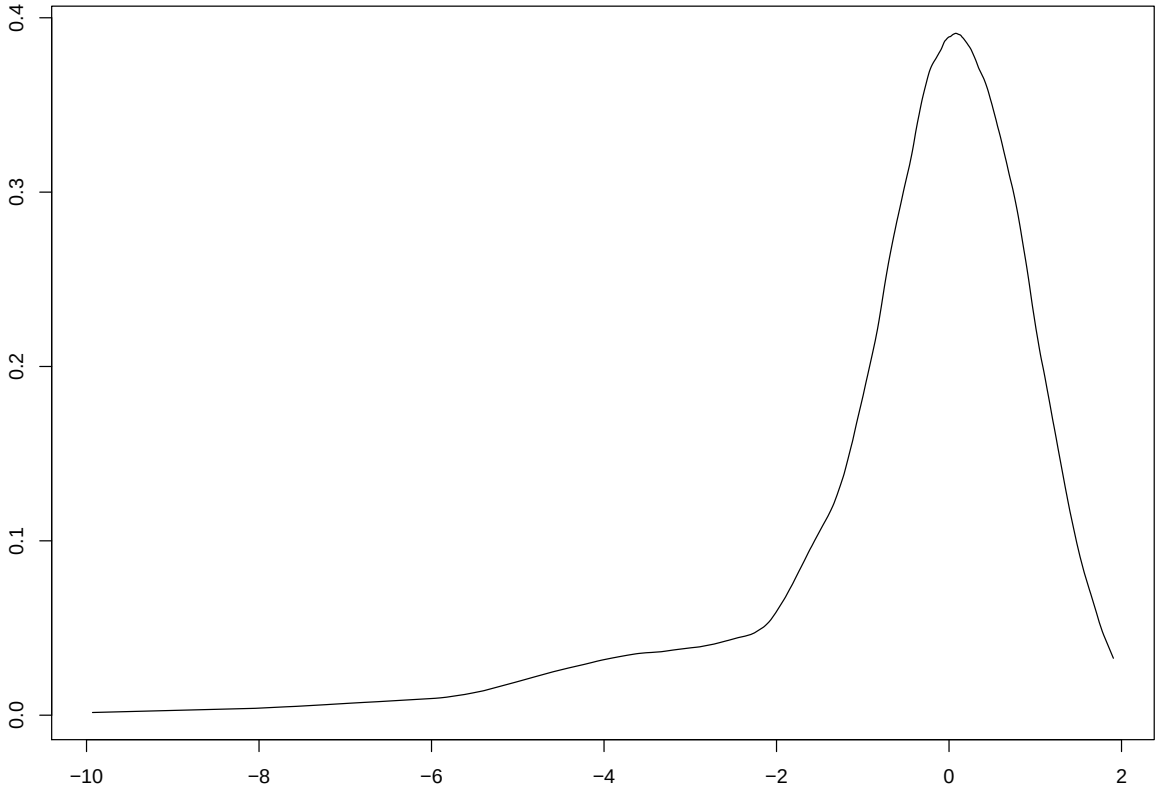

Figure 11: The distribution of T when randomly sampling from the sexual attitude data with the extreme outlier removed.

2 TRIMMED MEANS

TRIMMING REDUCES PROBLEMS ASSOCIATED WITH SKEWED DISTRIBUTIONS AND LOW POWER DUE TO OUTLIERS.

BASIC ISSUE: ESTIMATING THE SQUARED STANDARD ERROR OF THE TRIMMED MEAN

RECALL ESTIMATE OF VAR($\bar{X}$) = s^2/n

FOR A 20% TRIMMED MEAN, ESTIMATE IS

$$\frac{s_w^2}{.6^2 n},$$

where s_w^2 is the 20% Winsorized sample variance.

Simply estimating the standard error using the sample variance applied to the data left after trimming is technically unsound and can differ substantially from the estimator just described, which is motivated by theory and supported by simulations.

MORE BROADLY, SIMPLY DISCARDING OUTLIERS AND APPLYING METHODS FOR MEANS TO THE REMAINING DATA CAN RESULT IN HIGHLY INACCURATE RESULTS. STANDARD ERRORS DEPEND ON THE METHOD USED TO DETECT AND REMOVE THE INFLUENCE OF OUTLIERS.

EXAMPLE

For sexual attitude data, $n = 105$ males, the estimated standard error of the 20% trimmed mean is .53.

Imagine we use the method for the sample mean on the remaining 63 values left after trimming. That is, we compute s using these 63 values only and then compute $s/\sqrt{63}$. This yields 0.28, which is about half of the value based on a theoretically sound technique.

The R function

```
trimse(x,tr=.2)
```

estimates the standard error of a trimmed mean.

A CONFIDENCE INTERVAL FOR THE POPULATION TRIMMED MEAN: TUKEY-MCLAUGHLIN METHOD

$$\left(\bar{X}_t - c\frac{s_w}{(1-2G)\sqrt{n}},\ \bar{X}_t + c\frac{s_w}{(1-2G)\sqrt{n}}\right),$$

where G is the proportion of values trimmed, c is the $1-\alpha/2$ quantile of the Student's t distribution with $h-1$ degrees of freedom, where $h = n - 2G$ is the number of observations left after trimming. With 20% trimming, a $1-\alpha$ confidence interval is given by

$$\left(\bar{X}_t - c\frac{s_w}{.6\sqrt{n}},\ \bar{X}_t + c\frac{s_w}{.6\sqrt{n}}\right). \tag{5}$$

The R function

```
trimci(x,tr=.2,alpha=.05,nv=0)
```

computes a $1-\alpha$ confidence interval for μ_t. The amount of trimming, indicated by the argument `tr`, defaults to 20%. If the argument alpha is unspecified, $\alpha = .05$ is used.

EXAMPLE

Suppose a test of open mindedness is administered to 10 participants yielding the observations

$$5, 60, 43, 56, 32, 43, 47, 79, 39, 41.$$

$\bar{X}_t = 44.8$ and $\bar{X} = 44.5$. Confidence interval for the trimmed mean is

$$44.8 \pm 2.57\frac{7.385}{0.6\sqrt{10}} = (34.8,\ 54.8).$$

In contrast, the 0.95 confidence interval for the mean is (30.7, 58.3). The ratio of the lengths of the confidence intervals is $(54.8 - 34.8)/(58.3 - 30.7) = .72$. That is, the length of the confidence interval based on the trimmed mean is substantially shorter.

NOTE IMPLICATION REGARDING POWER.

HYPOTHESIS TESTING

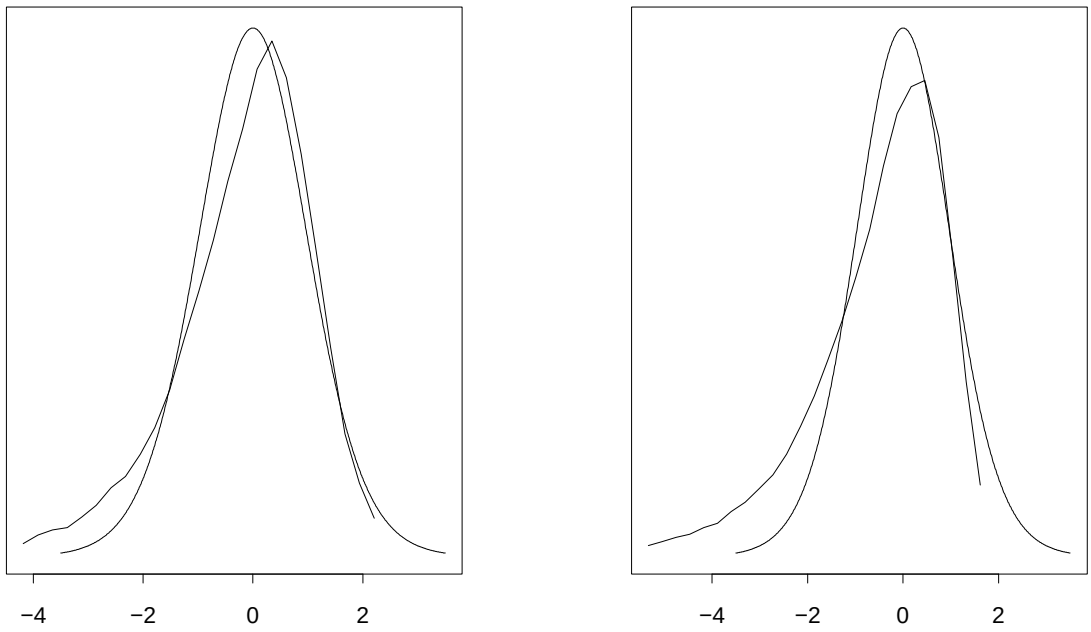

Figure 12: The left panel shows actual distribution of T_t and the approximation based on Student's t distribution when sampling from the lognormal distribution in Figure 6. The right panel shows the actual distribution of T. In practical terms, using Student's t to compute confidence intervals for the 20% trimmed mean tends to be more accurate than using Student's t to compute a confidence interval for the mean.

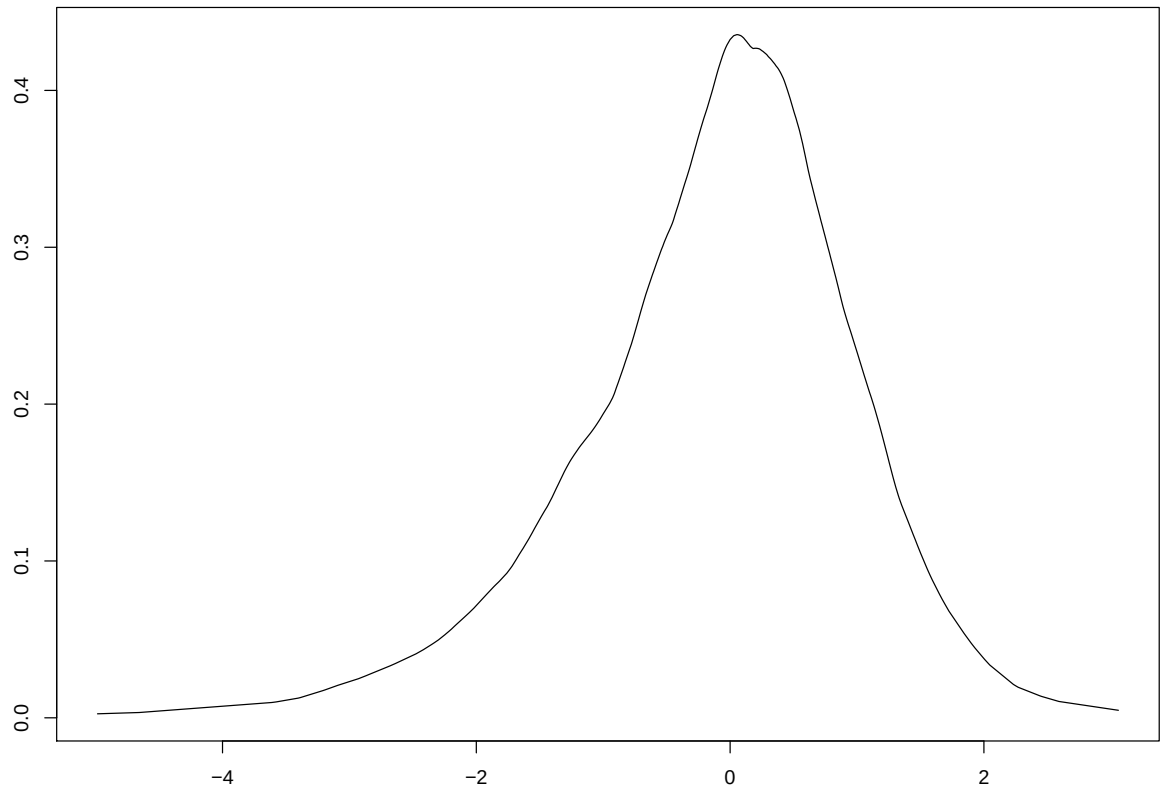

Figure 13: The distribution of T_t for the sexual attitude data, with the extreme outlier included.

Test statistic is

$$T_t = \frac{0.6(\bar{X}_t - \mu_0)}{s_w/\sqrt{n}}, \tag{6}$$

where μ_0 is some specified value of interest

MEDIAN: REQUIRES SPECIAL METHODS

FOR DISTRIBUTION FREE CONFIDENCE INTERVAL, USE THE R FUNCTION

```
sint(x,alpha=.05)
```

The R function

```
msmedse(x)
```

computes an estimate of the standard error of sample median, M

BUT THIS ESTIMATE OF THE STANDARD ERROR OF THE MEDIAN, AS WELL AS ALL OTHERS THAT HAVE BEEN PROPOSED, CAN PERFORM POORLY WHEN TIED (DUPLICATED) VALUES OCCUR.

PRACTICAL IMPLICATION: NEED ALTERNATIVE METHOD FOR COMPARING GROUPS BASED ON MEDIAN.

WE CAN DEAL WITH TIED VALUES USING A PERCENTILE BOOTSTRAP METHOD, WHICH WILL BE DESCRIBED LATER.

NOTE: STUDENT'S T CAN BE BIASED. THAT IS, PROBABILITY OF REJECTING IS NOT MINIMIZED WHEN THE NULL HYPOTHESIS IS TRUE (e.g., Pratt, JASA, 1964).

COMMENTS ON M-ESTIMATOR AND MODIFIED ONE-STEP M-ESTIMATOR

STANDARD ERROR OF M-ESTIMATOR CAN BE ESTIMATED. BUT RESULTING TEST STATISTIC IS UNSATISFACTORY, IN TERMS OF TYPE I ERRORS, WHEN DEALING WITH SKEWED DISTRIBUTIONS. NEED TO USE A PERCENTILE BOOTSTRAP TO BE DESCRIBED. SAME IS TRUE WHEN USING THE MODIFIED ONE-STEP M-ESTIMATOR.

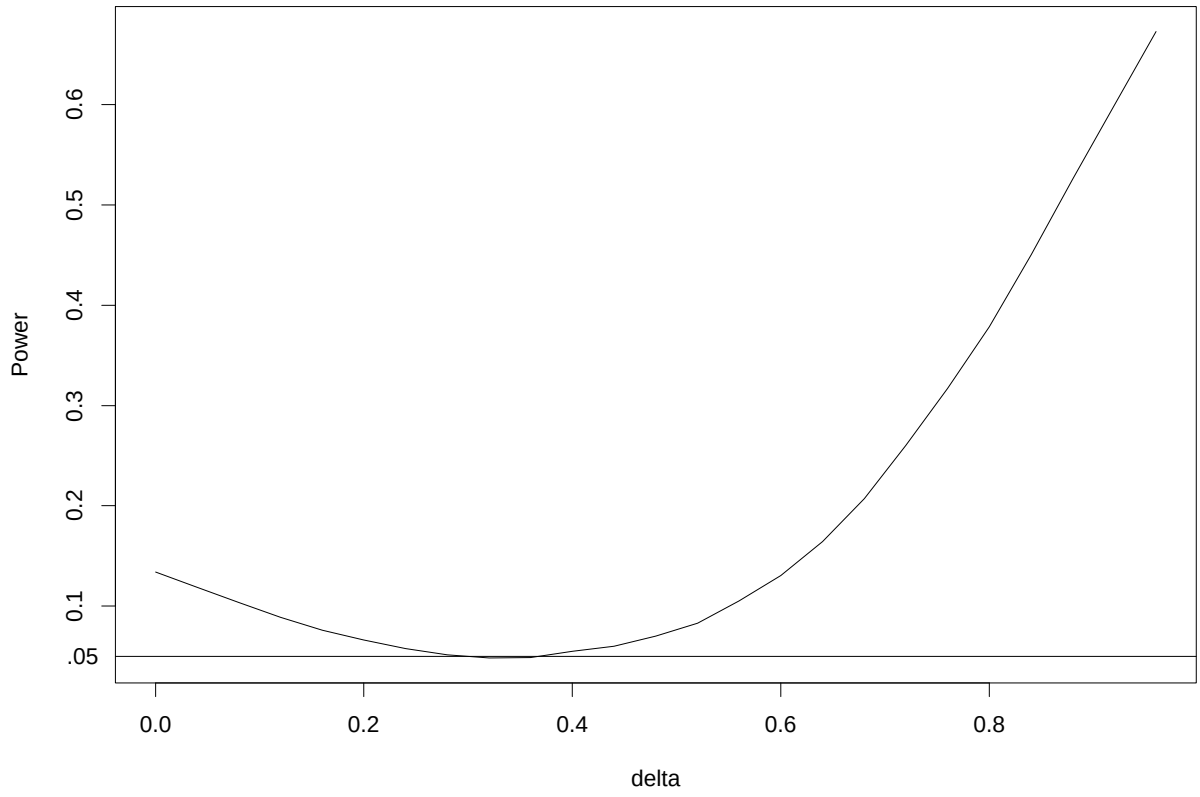

Figure 14: Power curve of Student's T when sampling from a lognormal distribution, $n = 20$, $\alpha = 0.05$. The null hypothesis corresponds to delta=0. Ideally the power curve should be strictly increasing as delta gets large.

3 BOOTSTRAP METHODS

NOT A PANACEA FOR DEALING WITH THE MANY PRACTICAL PROBLEMS ENCOUNTERED WHEN TRYING TO UNDERSTAND DATA.

BUT THEY HAVE CONSIDERABLE PRACTICAL VALUE FOR A WIDE RANGE OF SITUATIONS.

UNDERSTANDING THE BASICS:

First consider *Determining Critical Values via Simulation Studies*

If the goal is to have a Type I error probability of 0.05, this can be accomplished if the distribution of

$$T = \frac{\bar{X} - \mu_0}{s/\sqrt{n}} \tag{7}$$

can be determined.

Momentarily assume the distribution of T is symmetric. If

$$P(T \leq t) = 0.975.$$

a 0.95 confidence interval for μ is given by

$$\left(\bar{X} - t\frac{s}{\sqrt{n}}, \bar{X} + t\frac{s}{\sqrt{n}}\right).$$

Conceptually, when the null hypothesis is true, t can be determined to a high degree of accuracy if a study could be repeated millions of times. If, for example, it were repeated 1000 times, put the the resulting T values in ascending order and label the results $T_{(1)} \leq \cdots \leq T_{(1000)}$. Then one way of estimating t is with $T_{(975)}$, simply because 97.5% of the T values are less than or equal to $T_{(975)}$.

Here is how we would use R to determine t if sampling from a lognormal distribution:

- Generate n observations from the distribution of interest. (For a lognormal distribution the R function `rlnorm` accomplishes this goal.)

- Compute the sample mean $\bar{X}$, the sample standard deviation s, and then T.
- Repeat steps 1 and 2 many times. For illustrative purposes, assume they are repeated 1000 times yielding 1000 T values: $T_1, \ldots, T_{1000}$.
- Put these values in ascending order yielding $T_{(1)} \leq \cdots \leq T_{(1000)}$.
- An estimate of the .975 quantile of the distribution of T is $T_{(975)}$ again because the proportion of observed T values less than or equal to $T_{(975)}$ is 0.975. If, for example, $T_{(975)} = 2.1$, this suggests using $t = 2.1$ when computing a 0.95 confidence interval.

But this assumes T has a symmetric distribution. Alternative solution: also estimate the .025 quantile with $T_{(25)}$.

$$\left(\bar{X} - T_{(u)}\frac{s}{\sqrt{n}}, \bar{X} - T_{(\ell+1)}\frac{s}{\sqrt{n}}\right).$$

CALLED AN EQUAL-TAILED CONFIDENCE INTERVAL

NOTE: It might appear that $T_{(u)}$ should be used to compute the upper end of the confidence interval, not the lower end, but a little algebra shows that this is not the case.

BUT WE DON'T KNOW THE DISTRIBUTION WE ARE SAMPLING FROM.

HOWEVER, WE HAVE AN ESTIMATE BASED ON THE OBSERVED DATA.

SO CONDUCT A SIMULATION ON THE OBSERVED DATA TO DETERMINE DISTRIBUTION OF T.

REPLACE T WITH

$$T^* = \frac{\bar{X}^* - \bar{X}}{s^*/\sqrt{n}}.$$

THIS IS THE STRATEGY USED BY THE BOOTSTRAP-T METHOD.

SYMMETRIC CONFIDENCE INTERVAL

Let

$$T^* = \frac{|\bar{X}^* - \bar{X}|}{s^*/\sqrt{n}} \tag{8}$$

Table 1: Actual Type I error probabilities for three methods based on the mean when testing at the $\alpha = .05$ level

		Method		
	Dist.	BT	SB	T
$n = 20$	N	.054	.051	.050
	LN	.078	.093	.140
	MN	.100	.014	.022
	SH	.198	.171	.202
$n = 100$	N	.048	.038	.050
	LN	.058	.058	.085
	MN	.092	.018	.041
	SH	.168	.173	.190

N=Normal, LN=Lognormal, MN=Mixed normal, SH=Skewed, heavy-tailed,
BT=Equal-tailed, bootstrap-t, SB=Symmetric bootstrap-t, T=Student's T

and reject $H_0 : \mu = \mu_0$ if $|T| \geq T^*_{(c)}$, where $c = (1 - \alpha)B$ rounded to the nearest integer and again $T^*_{(1)} \leq \cdots \leq T^*_{(B)}$ are the B bootstrap T^* values written in ascending order. An approximate $1 - \alpha$ confidence interval for μ is now given by

$$\bar{X} \pm T^*_{(c)} \frac{s}{\sqrt{n}}. \tag{9}$$

4 PERCENTILE BOOTSTRAP METHOD

NOT RECOMMENDED WHEN TESTING HYPOTHESES ABOUT MEANS, BUT EASIEST TO EXPLAIN WHEN WORKING WITH THE MEAN

GOAL: TEST

$$H_0 : \mu = \mu_0.$$

STRATEGY:

GENERATE BOOTSTRAP SAMPLES AND ESTIMATE THE PROBABILITY THAT A BOOTSTRAP SAMPLE MEAN IS GREATER THAN μ_0.

IN SYMBOLS, GENERATE A BOOTSTRAP SAMPLE MEAN. THE GOAL IS TO DETERMINE

$$p = P(\bar{X}^* > \mu_0),$$

the probability that the bootstrap sample mean is greater than the hypothesized value.

Let μ_0 be any hypothesized value. Here are the steps used by percentile bootstrap method when dealing with the mean:

1. Generate a bootstrap sample $X_1^*, \ldots, X_n^*$. (Randomly sample with replacement n observations from $X_1, \ldots, X_n$.)
2. Compute the mean of this bootstrap sample, $\bar{X}^*$.
3. Repeat steps 1 and 2 B times yielding $\bar{X}_1^*, \ldots, \bar{X}_B^*$.
4. Estimate $p = P(\bar{X}^* > \mu_0)$ with $\hat{p}$, the proportion of bootstrap sample means greater than μ_0.

The percentile bootstrap p-value when testing for exact equality is

$$P = 2\min(\hat{p}, 1 - \hat{p}).$$

That is, the p-value is either $2\hat{p}$ or $2(1 - \hat{p})$, whichever is smallest.

The R function

```
onesampb(x, est = onestep, alpha = 0.05, nboot = 2000)
```

computes a confidence interval based on the one-step M-estimator, where x is an R variable containing data, alpha is α, which defaults to 0.05, and nboot is B, the number of bootstrap samples to be used, which defaults to 2000. (This function contains two additional arguments, the details of which can be found in Wilcox, 1997a, p. 85.) This function can be used with any measure of location via the argument est. For example,

```
onesampb(x, est = tmean)
```

would return a confidence interval based on the 20% trimmed mean.

For convenience, the R function

```
momci(x,alpha=.05,nboot=2000)
```

computes a confidence interval based on the modified one-step M-estimator and the function

```
trimpb(x,tr=.2,alpha=.05,nboot=2000)
```

uses a trimmed mean. The argument `tr` indicates the amount of trimming and defaults to 0.2 if not specified. Again, alpha is α and defaults to 0.05. It appears that $B = 500$ suffices, in terms of achieving accurate probability coverage with 20% trimming. But to be safe, B (`nboot`) defaults to 2,000. (An argument for using $B = 2000$ can be made along the lines used by Booth and Sarker, 1998. If B is relatively small,this might result in relatively low power)

EXAMPLE

For 15 law schools, the undergraduate GPA of entering students, in 1973, was

3.39 3.30 2.81 3.03 3.44 3.07 3.00 3.43 3.36 3.13 3.12 2.74 2.76 2.88 2.96.

The 0.95 confidence interval returned by the R function `onesampb` is (2.95, 3.29). So among all law schools, it is estimated that the typical GPA of entering students is between 2.95 and 3.29. Using the MOM-estimator instead (the R function `momci`), the 0.95 confidence interval is (2.92, 3.35). Using the function `trimpb`, the .95 confidence interval for the 20% trimmed mean is (2.94, 3.25). Setting the argument tr=0 in `trimpb` results in a 0.95 confidence interval for the mean: (2.98, 3.21), illustrating that in some cases, switching from the mean to a 20% trimmed mean makes little difference.

5 BOOTSTRAP-T METHOD BASED ON TRIMMED MEANS

If the amount of trimming is 20% or more, it seems that using a percentile bootstrap method is best for general use, but with the amount of trimming close to zero, it currently seems that using a bootstrap-t method is preferable. (With 10% trimming, it is unclear whether a bootstrap-t is preferable to a percentile bootstrap method.) The computational steps of the most basic version of the bootstap-t method are summarized here when using a trimmed mean.

Generate a bootstrap sample of size n and compute the trimmed mean and Winsorized standard deviation, which we label $\bar{X}_t^*$ and s_w^*, respectively. Let γ be the amount of trimming. With 20% trimming, $\gamma = 0.2$ and with 10% trimming, $\gamma = 0.1$. Next, compute

$$T_t^* = \frac{(1-2\gamma)(\bar{X}_t^* - \bar{X}_t)}{s_w^*/\sqrt{n}}. \tag{10}$$

Repeating this process B times yields B T_t^* values. Writing these B values in ascending order we get $T_{t(1)}^* \leq T_{t(2)}^* \leq \cdots \leq T_{t(B)}^*$. Letting $\ell = .025B$, rounded to the nearest integer, and $u = B - \ell$, an estimate of the 0.025 and 0.975 quantiles of the distribution of T_t is $T_{(\ell+1)}^*$ and $T_{(u)}^*$. The resulting 0.95 confidence interval for μ_t (the population trimmed mean) is

$$\left(\bar{X}_t - T_{t(u)}^* \frac{s_w}{(1-2\gamma)\sqrt{n}}, \bar{X}_t - T_{t(\ell+1)}^* \frac{s_w}{(1-2\gamma)\sqrt{n}}\right). \tag{11}$$

Hypothesis Testing.

As for testing $H_0 : \mu_t = \mu_0$, compute

$$T_t = \frac{(1-2\gamma)(\bar{X}_t - \mu_0)}{s_w/\sqrt{n}}$$

and reject if

$$T_t \leq T_{t(\ell+1)}^*,$$

or if

$$T_t \geq T_{t(u)}^*.$$

Table 2: Actual Type I error probabilities using 20% trimmed means, $\alpha = .05$

		Method			
	Dist.	BT	SB	P	TM
$n = 20$	N	.067	.052	.063	.042
	LN	.049	.050	.066	.068
	MN	.022	.019	.053	.015
	SH	.014	.018	.066	.020

N=Normal, LN=Lognormal, MN=Mixed normal, SH=Skewed, heavy-tailed, BT=Equal-tailed, bootstrap-t, SB=Symmetric bootstrap-t, P=Percentile bootstrap, TM=Tukey-McLaughlin

The symmetric bootstrap-t method can be used as well when testing a two-sided hypothesis. Now we use

$$T_t^* = \frac{|(1-2\gamma)(\bar{X}_t^* - \bar{X}_t)|}{s_w^*/\sqrt{n}}. \tag{12}$$

and reject H_0 if $|T_t| > T_{t(c)}^*$, where $c = (1-\alpha)B$ rounded to the nearest integer. An approximate $1-\alpha$ confidence interval for μ_t is

$$\bar{X}_t \pm T_{t(c)}^* \frac{s_w}{(1-2\gamma)\sqrt{n}}. \tag{13}$$

The R function

```
trimcibt(x, tr = 0.2, alpha = 0.05, nboot = 599, side = T)
```

computes a bootstrap-t confidence interval for a trimmed mean. The argument side indicates whether an equal-tailed or a symmetric confidence interval is to be computed. The default is side=T resulting in a symmetric confidence interval. Using side=F means that an equal-tailed confidence interval will be computed. The argument `tr` indicates the amount of trimming, which defaults to 20%. So to compute a confidence interval for the mean, set

tr=0.

EXAMPLE

Data on the desired number of sexual partners over next 30 years among 105 college males. The Tukey-McLaughlin .95 confidence interval for the 20% is (1.62, 3.75). Using the R function `trimcibt` with side=F yields an equal-tailed .95 confidence interval of (1.28, 3.61). With side=T it is (1.51, 3.61). Using the percentile bootstrap method, the R function `trimpb` returns (1.86, 3.95). So in this particular case, the lengths of the confidence intervals do not vary that much among the methods used, but the intervals are centered around different values, which might affect any conclusions made. If trimcibt is used to compute a 0.95 confidence for the mean (by setting the argument tr=0), the result is (−2.46, 4704.59), which differs drastically for the confidence interval for a 20% trimmed mean.

In summary, all indications are that the percentile bootstrap is more stable (with at least 20% trimming) than the bootstrap-t method. That is, the actual Type I error probability tends to be closer to the nominal level. And it has the added advantage of more power, at least in some situations, compared to any other method we might choose.

However, there are situations where the bootstrap-t method outperforms the percentile method. And there are additional situations where the percentile bootstrap is best. So both methods are important to know.

FOUR GENERAL APPROACHES WHEN COMPARING TWO GROUPS:

1. Compare measures of location, such as the mean or median.
2. Compare measures of variation.
3. Focus on the probability that a randomly sampled observation from the first group is smaller than a randomly sampled observation from second group.
4. Simultaneously compare all of the quantiles to get a global sense of where the distributions differ and by how much. For example, low scoring participants in group 1 might be very similar to low scoring participants in group 2, but for high scoring participants, the reverse might be true.

6 Concerns About Student's t

EXAMPLE

Consider again the two normal distributions shown in the left panel of Figure 5. Both have variances one and the means differ by 1. As previously noted, using Student's T with $\alpha = .05$, power is 0.96 with $n_1 = n_2 = 25$. But now look at the two contaminated normal distributions shown in the right panel of Figure 5. The difference in the means is the same as in the left panel and the plot of the distributions has an obvious similarity to the distributions in the left panel. But for the right panel, power is only 0.28. One reason power is low is that when sampling from a heavy-tailed distribution, the actual probability of a Type I error can be substantially lower than the specified α value. For example, if you use Student's T with $\alpha = 0.05$, the actual probability of a Type I error can drop below .01. If an adjustment could be made so that the actual probability of a Type I error is indeed .05, power would be better, but it would still be low relative to alternative methods that are less sensitive to outliers. The reason is that in the right panel, the variances are 10.9, compared to 1 for the distributions in left panel. Said another way, outliers are more likely to occur when sampling from the distributions in the left panel which can inflate the standard deviations. And even when outliers are not a concern, having unequal variances or even different degrees of skewness can result in relatively poor power as well.

Finally, we note that when using Student's T, even a single outlier in only one group can result in a rather undesirable property. The following example illustrates the problem.

EXAMPLE

Consider the following values.

Group 1: 4 5 6 7 8 9 10 11 12 13
Group 2: 1 2 3 4 5 6 7 8 9 10

The corresponding sample means are $\bar{X}_1 = 8.5$ and $\bar{X}_2 = 5.5$ and $T = 2.22$. With $\alpha = 0.05$, the critical value is $t = 2.1$, so Student's T would reject the hypothesis of equal means and conclude that the first group has a larger population mean than the second (because the first group has the larger sample mean). Now, if we increase the largest observation in the first group from 13 to 23, the sample mean increases to $\bar{X}_1 = 9.5$. So the difference between $\bar{X}_1$ and $\bar{X}_2$ has increased from 3 to 4 and this would seem to suggest that we have stronger evidence that the population means differ and in fact the first group has the larger population mean. However, increasing the largest observation in the first group also inflates the corresponding sample variance, s_1^2. In particular, s_1^2 increases from 9.17 to 29.17. The result is that T *decreases* to $T = 2.04$ and we no longer reject. That is, increasing the largest observation has more of an effect on the sample variance than the sample mean in the sense that now we are no longer able to conclude that the population means differ. Increasing the largest observation in the first group to 33, the sample mean increases to 10.5, the difference between the two sample means increases to 5 and now $T = 1.79$. So again we do not reject and in fact our test statistic is getting smaller! This illustration provides another perspective on how outliers can mask differences between population means.

Yuen's Method for Trimmed Means

The R function

```
yuen(x,y,alpha=.05,tr=.2)
```

EXAMPLE.

In a study of sexual attitudes, 1327 males and 2282 females were asked how many sexual

partners they desired over the next 30 years.[1] We can compare the means with the R function `yuen` by setting `tr=0`. (This results in using Welch's test.) The 0.95 confidence interval for the difference between the means is (-1491.087, 4823.244) and the p-value is 0.30. Given the large sample sizes, a tempting conclusion might be that power is adequate and that the groups do not differ. However, the 0.95 confidence interval for the difference between the 20% trimmed means is (0.408 2.109) with a p-value less than 0.001. In terms of Tukey's three-decision rule, we can be reasonably certain that males have the larger population 20% trimmed mean.

The function

```
fac2list(x,g)
```

can be used to separate data into groups, where `x` is the R variable (often the column of a matrix or the column of a data frame) containing the data to be analyzed and `g` indicates the column where that indicates the level of the corresponding value stored in `x`.

EXAMPLE.

Plasma retinol data are available from

`http://lib.stat.cmu.edu/datasets/Plasma_Retinol.`

Retinol is vitamin A, and plasma retinol appears to be related to the utilization of vitamin A in rats. For future reference, the variable names in the plasma retinol data set are:

```
1       AGE: Age (years)
2       SEX: Sex (1=Male, 2=Female).
3       SMOKSTAT: Smoking status (1=Never,2=Former,3=Current Smoker)
4       QUETELET: Quetelet (weight/(height^2))
5       VITUSE: Vitamin Use (1=Yes,fairly often,2=Yes,not often,3=No)
6       CALORIES: Number of calories consumed per day.
7       FAT: Grams of fat consumed per day.
8       FIBER: Grams of fiber consumed per day.
```

[1]The data in Table 2.2 are based on the same question but are from a different study. The data used in this example, supplied by Lynn Miller, are stored in the file miller.dat and can be downloaded from the author's web page given in Chapter 1 of Wilcox, 2011.

```
9       ALCOHOL: Number of alcoholic drinks consumed per week.
10      CHOLESTEROL: Cholesterol consumed (mg per day).
11      BETADIET: Dietary beta-carotene consumed (mcg per day).
12      RETDIET: Dietary retinol consumed (mcg per day)
13      BETAPLASMA: Plasma beta-carotene (ng/ml)
14      RETPLASMA: Plasma Retinol (ng/ml)
```

The first few lines of the data set look like this:

```
64  2  2 21.48380  1 1298.8 57.0  6.3  0.0 170.3 1945 890 200 915
76  2  1 23.87631  1 1032.5 50.1 15.8  0.0  75.8 2653 451 124 727
38  2  2 20.01080  2 2372.3 83.6 19.1 14.1 257.9 6321 660 328 721
40  2  2 25.14062  3 2449.5 97.5 26.5  0.5 332.6 1061 864 153 615
```

We compare males and females based on the daily consumption of cholesterol using Yuen's test. Assuming the data are stored in the R variable plasma, column 2 indicates sex (male=1, female=2) and column 10 indicates the daily consumption of cholesterol. So the R command

```
z=fac2list(plasma[,10],plasma[,2])
```

separates the data into groups corresponding to males and females and stores the results in z in list mode. Because male is indicated by the value 1, which is less than the value used to indicate female, z[[1]] will contain the data for the males and z[[2]] contains the data for the females. The R command

```
yuen(z[[1]],z[[2]])
```

will compare males to females using 20% trimmed means.

COMPARING MEDIANS

Best overall method, especially when dealing tied values: percentile bootstrap, which is described in the next section. With no tied values, method based on McKean-Schrader estimate of standard error works well.

The R function

```
msmed(x,y,alpha=.05)
```

compares medians using the McKean-Schrader estimate of the standard error.

7 Percentile Bootstrap Methods for Comparing Measures of Location

Let $\bar{X}_1^*$ and $\bar{X}_2^*$ be the bootstrap means corresponding to groups 1 and 2, respectively. Reject if

$$P(\bar{X}_1^* > \bar{X}_2^*)$$

is relatively close to 0 or 1. (A p-value can be computed as indicated momentarily.)

A $1-\alpha$ confidence interval for $\mu_1 - \mu_2$ is computed as follows. Let

$$D^* = \bar{X}_1^* - \bar{X}_2^*$$

be the difference between the bootstrap means. Now suppose we repeat this process B times yielding $D_1^*, \ldots, D_B^*$. (The software written for my books uses $B = 2000$.) The middle 95% of these values, after putting them in ascending order, yields a .95 confidence interval for the difference between the population means. In symbols, put the values $D_1^*, \ldots, D_B^*$ in ascending order yielding $D_{(1)}^* \leq \cdots \leq D_{(B)}^*$. Then an approximate $1-\alpha$ confidence interval for the difference between the population means, $\mu_1 - \mu_2$, is

$$(D_{(\ell+1)}^*, D_{(u)}^*), \tag{14}$$

where $\ell = \alpha B/2$, rounded to the nearest integer, and $u = B - \ell$. So for a 0.95 confidence interval, $\ell = 0.025B$.

Computing a p-value

A p-value can be computed and is based on the probability that a bootstrap mean from

the first group is greater than a bootstrap mean from the second. In symbols, a p-value can be computed if we can determine

$$p^* = P(\bar{X}_1^* > \bar{X}_2^*). \tag{15}$$

The value of p^* reflects the degree of separation between the two (bootstrap) sampling distributions. If the means based on the observed data are identical, meaning that $\bar{X}_1 = \bar{X}_2$, then p^* will have a value approximately equal to 0.5. (Under normality, p^* is exactly equal to 0.5.) Moreover, the larger the difference between the sample means $\bar{X}_1$ and $\bar{X}_2$, the closer p^* will be to 0 or 1. If $\bar{X}_1$ is substantially larger than $\bar{X}_2$, p^* will be close to 1, and and if $\bar{X}_1$ is substantially smaller than $\bar{X}_2$, p^* will be close to 0. (Hall, 1988a, provides relevant theoretical details and results in Hall 1988b are readily extended to trimmed means.) Theoretical results not covered here suggest the following decision rule when the goal is to have a Type I error probability α: Reject the hypothesis of equal means if p^* is less than or equal to $\alpha/2$, or greater than or equal to equal to $1 - \alpha/2$. Said another way, if we let

$$p_m^* = \min(p^*, 1 - p^*),$$

meaning that p_m^* is equal to p^* or $1 - p^*$, whichever is smaller, then reject if

$$p_m^* \leq \frac{\alpha}{2}. \tag{16}$$

We do not know p^*, but it can be estimated by generating many bootstrap samples and computing the proportion of times a bootstrap mean from the first group is greater than a bootstrap mean from the second. That is, if A represents the number of values among $D_1^*, \ldots, D_B^*$ that are greater than zero, then we estimate p^* with

$$\hat{p}^* = \frac{A}{B}. \tag{17}$$

Finally, reject the hypothesis of equal population means if $\hat{p}^*$ is less than or equal to $\alpha/2$ or greater than or equal to $1 - \alpha/2$. Or setting

$$\hat{p}_m^* = \min(\hat{p}^*, 1 - \hat{p}^*),$$

reject if

$$\hat{p}_m^* \leq \frac{\alpha}{2}. \tag{18}$$

The p-value is

$$2\hat{p}_m^*$$

(Liu & Singh, 1997).

When dealing with the median, and tied values occur, now

$$p^* = P(M_1^* > M_2^*) + .5P(M_1^* = M_2^*).$$

This method can be used with any measure of location, but it should not be used when comparing means.

EXAMPLE

As a simple illustration, imagine that we generate bootstrap samples from each group and compute the difference between the bootstrap sample means. Further imagine that this process is repeated 10 times (so $B = 10$ is being used) resulting in the following D^* values:

-2, -.5, -.1, -1.2, 1, -1.3, -2.3, -.01, -1.7, -.8

There is one positive difference, so $\hat{p}^* = 1/10$. The smaller of the two numbers $\hat{p}^*$ and $1 - \hat{p}^*$ is 0.1. Consequently, the p-value is $2(0.1) = 0.2$.

THE SAME STRATEGY IS USED WITH OTHER MEASURES OF LOCATION.

WITH SKEWED DISTRIBUTIONS, WHEN COMPARING M-ESTIMATORS, THIS IS THE BEST APPROACH TO HYPOTHESIS TESTING.

WORKS WELL WHEN USING 20% TRIMMED MEANS

COMPARING MEDIANS, WORKS WELL WHEN THERE ARE TIED VALUES. ONLY METHOD FOUND TO PERFORM WELL IN SIMULATIONS WHEN THERE ARE TIED VALUES.

The R function

```
medpb2(x,y,alpha=.05,nboot=2000,SEED=T)
```

tests the hypothesis of equal medians using the percentile bootstrap method just described. The function also returns a $1-\alpha$ confidence interval for the difference between the population medians.

The R function

```
trimpb2(x, y, tr = 0.2, alpha = 0.05, nboot = 2000)
```

compares trimmed means, including medians as a special case, using the percentile bootstrap method just described. Here `x` is any R variable containing the data for group 1 and `y` contains the data for group 2. The amount of trimming, `tr`, defaults to 20%, α defaults to 0.05, and `nboot` (B) defaults to 2000. This function returns a p-value plus a $1-\alpha$ confidence interval for the difference between the trimmed means.

The R function

```
pb2gen(x, y, alpha = 0.05, nboot = 2000, est = onestep, ...)
```

can be used to compare groups using any measure of location in conjunction with the percentile bootstrap method. Again, `x` and `y` are any R variables containing data and `nboot` is B, the number of bootstrap samples to be used. By default, $B = 2000$. The argument est indicates which measure of location is to be employed. It can be any R function that computes a measure of location and defaults to the function `onestep`, which is the one-step M estimator

8 Bootstrap-t Methods for Comparing Measures of Location

When testing the hypothesis of equal population means, Welch's test statistic is

$$W = \frac{(\bar{X}_1 - \bar{X}_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}.$$

The probability of a Type I error can be controlled exactly if the distribution of W (over many studies) can be determined when the null hypothesis of equal means is true. Welch's strategy was to approximate the distribution of W with a Student's t distribution and the degrees of freedom estimated based on the sample variances and sample sizes. The bootstrap-t strategy is to use bootstrap samples instead to estimate the distribution of W when the null hypothesis is true.

An outline of the method is as follows. Generate a bootstrap sample of size n_1 from the first group and label the resulting sample mean and standard deviation $\bar{X}_1^*$ and s_1^*, respectively. Do the same for the second group and label the bootstrap sample mean and

standard deviation $\bar{X}_2^*$ and s_2^*. Let

$$W^* = \frac{(\bar{X}_1^* - \bar{X}_2^*) - (\bar{X}_1 - \bar{X}_2)}{\sqrt{\frac{(s_1^*)^2}{n_1} + \frac{(s_2^*)^2}{n_2}}}. \tag{19}$$

Repeat this process B times yielding B W^* values: $W_1^*, \ldots, W_B^*$. Next, put these B values in ascending order, which we label $W_{(1)}^* \leq \cdots \leq W_{(B)}^*$. Let $\ell = \alpha B/2$, rounded to the nearest integer, and $u = B - \ell$. Then an approximate $1 - \alpha$ confidence interval for the difference between the means $(\mu_1 - \mu_2)$ is

$$\left((\bar{X}_1 - \bar{X}_2) - W_{(u)}^* \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}, (\bar{X}_1 - \bar{X}_2) - W_{(\ell+1)}^* \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}, \right). \tag{20}$$

METHOD READILY EXTENDED TO TRIMMED MEANS

The R function

```
yuenbt(x, y, tr = 0.2, alpha = 0.05, nboot = 599, side=F)
```

uses a bootstrap-t method to compare trimmed means. The arguments are the same as those used by trimpb2 plus an additional argument labeled side, which indicates whether a symmetric or equal-tailed confidence interval will be used. Side defaults to F for false meaning that the equal-tailed confidence interval will be computed. Setting side equal to T yields the symmetric confidence interval.

9 Rank-Based and Nonparametric Methods

Let p be the probability that a randomly sampled observation from the first group is less than a randomly sampled observation for the second. (Momentarily, it is assumed that tied values never occur.) If the groups do not differ, then in particular it should be the case that

$$H_0 : p = 0.5 \tag{21}$$

is true. The quantity p has been called a *probabilistic measure of effect size*, the *probability of concordance*, the measure of *stochastic superiority* and the *common language measure of effect size*.

Currently, a method derived by Cliff seems best for general use. (Another method that performs well was derived by Brunner and Munzel.) It deals effectively with both heteroscedasticity and tied vales, unlike the classic Wilcoxon-Mann-Whitney test. The R function

```
cid(x,y,alpha=.05,plotit=F),
```

performs the calculations.

Let

$$p_1 = P(X_{i1} > X_{i2}),$$

$$p_2 = P(X_{i1} = X_{i2}),$$

and

$$p_3 = P(X_{i1} < X_{i2}).$$

The function also reports a confidence interval for $P = p_3 + .5p_2$, which is labeled `ci.p`. The estimate of P is labeled `phat`. To get a p-value, use the function

```
cidv2(x,y,plotit=F).
```

When the argument `plotit=T`, these functions plot the distribution of D, where D is the difference between a randomly sampled observation from the first group, minus a randomly sampled observation from the second group. D will have a symmetric distribution around zero when the distributions are identical. The plot provides perspective on the extent to which this is the case.

COMPARING ALL QUANTILES SIMULTANEOUSLY

Roughly, when comparing medians, the goal is to compare the central values of the two distributions. But an additional issue is how low scoring individuals in the first group compare to low scoring individuals in the second. And in a similar manner, how do relatively high scores within each group compare? A way of addressing this issue is to compare the 0.25 quantiles of both groups as well as the 0.75 quantiles. Or to get a more detailed sense of how the distributions differ, all of the quantiles might be compared. There is a method for comparing all quantiles in a manner that controls the probability of a Type I error exactly assuming random sampling only. The method was derived by Doksum and Sievers (1976) and is based on an extension of the Kolmogorov-Smirnov method. Complete computational details are not provided, but a function that applies the method is supplied and illustrated next.

The R function

```
sband(x,y, flag = F, plotit = T, xlab = "x (First Group)", ylab = "Delta")
```

computes confidence intervals for the difference between the quantiles using the data stored in the R variables x and y. Moreover, it plots the estimated differences as a function of the estimated quantiles associated with the first group, the first group being the data stored in the first argument, x. This difference between the quantiles, viewed as a function of the quantiles of the first group, is called a *shift function*. To avoid the plot, set the argument plotit=F.

EXAMPLE

In a study by J. Victoroff et al. (2008), 52 14-year-old refugee boys in Gaza were classified into one of two groups according to whether a family member had been wounded or killed by an Israeli. One issue was how these two groups compare based on a measures of depression. In particular, among boys with relatively high depression, does having a family member killed or wounded have more of an impact than among boys with relatively low measures of depression? Here is a portion of the output from sband:

```
            qhat lower upper
[1,] 0.03448276    NA    18
[2,] 0.06896552    NA    15
[3,] 0.10344828    NA    15
```

```
 [4,] 0.13793103    NA    15
 [5,] 0.17241379    NA    16
 [6,] 0.20689655    NA    16
 [7,] 0.24137931    NA    16
 [8,] 0.27586207    NA    15
 [9,] 0.31034483    NA    16
[10,] 0.34482759    NA    16
[11,] 0.37931034    NA    16
[12,] 0.41379310   -10    19
[13,] 0.44827586    -7    20
[14,] 0.48275862    -5    26
[15,] 0.51724138    -5    26
[16,] 0.55172414    -4    26
[17,] 0.58620690    -2    34
[18,] 0.62068966    -1    NA
[19,] 0.65517241    -2    NA
[20,] 0.68965517     2    NA
[21,] 0.72413793     1    NA
[22,] 0.75862069     2    NA
[23,] 0.79310345     2    NA
[24,] 0.82758621     1    NA
[25,] 0.86206897     2    NA
[26,] 0.89655172     2    NA
[27,] 0.93103448     0    NA
[28,] 0.96551724    -3    NA
[29,] 1.00000000    -2    NA
```

The column headed by `qhat` indicates the quantile being compared. The first value listed is 0.03448276, meaning that a confidence interval for the difference between the 0.03448276 quantiles is given in the next two columns. In the column headed by lower, entries NA indicate $-\infty$. In the column headed by upper, NA indicates ∞. So the first row of the output says that the confidence interval for the difference between the 0.03448276 quantiles is $(-\infty, 18)$. This interval contains 0, so you would fail to conclude that the quantiles differ. The function also returns a value labeled `numsig`, which indicates how many differences among all of the estimated quantiles are significant. That is, the confidence interval does not contain 0.

Now look at row 20 of the output. This says that when comparing the 0.68965517

quantiles, the confidence interval is $(2, \infty)$. This interval does not contain 0, so reject. Looking at rows 21-26, we again reject. So no difference between the groups is found when looking at the lower quantiles, but a difference is found from the 0.69 to 0.90 quantiles. Roughly, the results indicate than among boys who have had a family member wounded or killed, the effect is more pronounced among boys with high depression scores. Moreover, the probability that all of these confidence intervals simultaneously contain the true differences is approximately 0.95. If it is desired to compute the exact probability, this can be done by setting the argument flag=T. If flag=T is used in the example, the output labeled pc (probability coverage) has the value 0.96762 meaning that all 29 confidence intervals contain the true differences with probability 0.96762. Said another way, the probability of making at least one Type I error among the 29 quantiles being compared is $1 - 0.96762 = 0.03238$.

10 Measuring Effect Size

Generally, how might we measure or characterize the difference between two groups? Some possibilities are:

- Compute a confidence interval for the difference between some measure of location.
- Use a standardized difference or some related method that measures the difference between the means relative to the standard deviations.
- Compare and plot the differences between all of the quantiles. (This is done by the shift function, to be described).
- Plot the distributions. (For example, use the R function g2plot or use boxplots.)
- Estimate the probability that a randomly sampled observation from the first group is larger than a randomly sampled from the second.

A commonly used approach assumes the two groups have a common variance, which we label σ^2. That is, $\sigma_1^2 = \sigma_2^2 = \sigma^2$ is assumed. Then the so-called standardized difference between the groups is

$$\delta = \frac{\mu_1 - \mu_2}{\sigma}. \tag{22}$$

Cohen (1977) suggests that as a general guide, when dealing normal distributions, $\delta = 0.2$, 0.5 and 0.8 correspond to small, medium and large effect sizes, respectively. An estimate of δ is

$$d = \frac{\bar{X}_1 - \bar{X}_2}{s_p}, \tag{23}$$

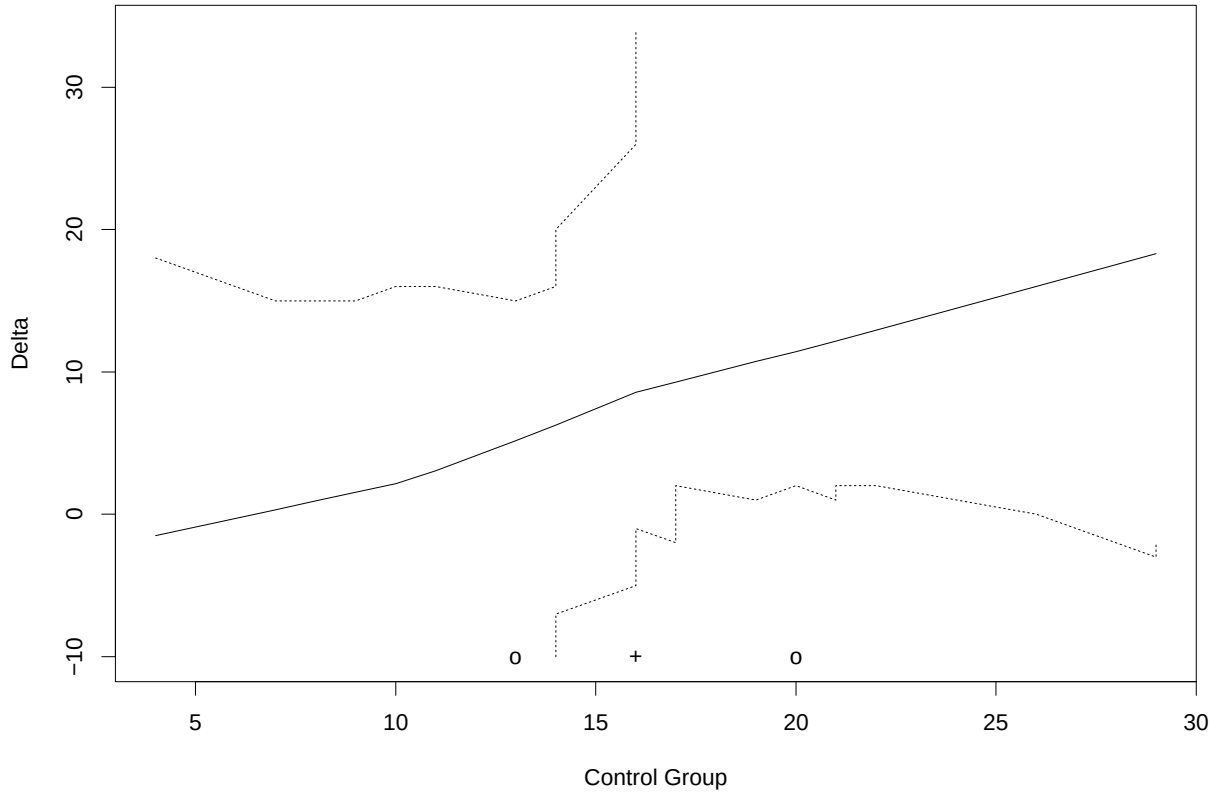

Figure 15: A plot of the shift function based on the Gaza data. The plot indicates that among boys with low measures of depression, there is little difference between the two groups. But as we move toward subpopulations of boys with high depression, the difference between the two groups increases.

which is often called *Cohen's d*, where s_p is the pooled standard deviation.

TWO SERIOUS CONCERNS

1. The measure of effect size δ can be seriously affected by nonnormality. The left panel of Figure 16, which is the same as in Figure 5, shows two normal distributions where the difference between the means is 1 ($\mu_1 - \mu_2 = 1$) and both standard deviations are one. So

$$\delta = 1,$$

which is often viewed as being relatively large. Now look at the right panel of Figure 16 As is evident, the difference between the two distributions appears to be very similar to the difference shown in the left panel, so according to Cohen we again have a large effect size. However, in the right panel, $\delta = 0.3$ because these two distributions are mixed normals with variances 10.9. This illustrates the general principle that arbitrarily small departures from normality can render the magnitude of δ meaningless. In practical terms, if we rely exclusively on δ to judge whether there is a substantial difference between two groups, situations will arise where we will grossly underestimate the degree to which groups differ, particularly when outliers occur.

2. Assumes homoscedasticity.

Here is another concern about δ when trying to characterize how groups differ. Look again at Figures 2 and 3. The distributions have equal means and equal variances but they differ in an obvious way that might have practical importance. Although the difference between measures of location provide a useful measure of effect size, we might need additional ways of gaining perspective on the extent to which groups differ such as the shift function, already described.

DEALING WITH NONNORMALITY

One way of dealing with nonnormality is to use a generalization of δ based on 20% trimmed means and Winsorized variances, where the Winsorized variances are rescaled so that under normality they estimate the variance (Algina, Keselman, and Penfield, 2005). With 20% trimming, this means that the Winsorized variance is divided by 0.4121. That is, under normality, $s_w^2/0.4142$ estimates σ^2. If we assume the groups have equal Winsorized variances, δ becomes

$$\delta_t = .642\frac{\bar{X}_{t1} - \bar{X}_{t2}}{S_w},$$

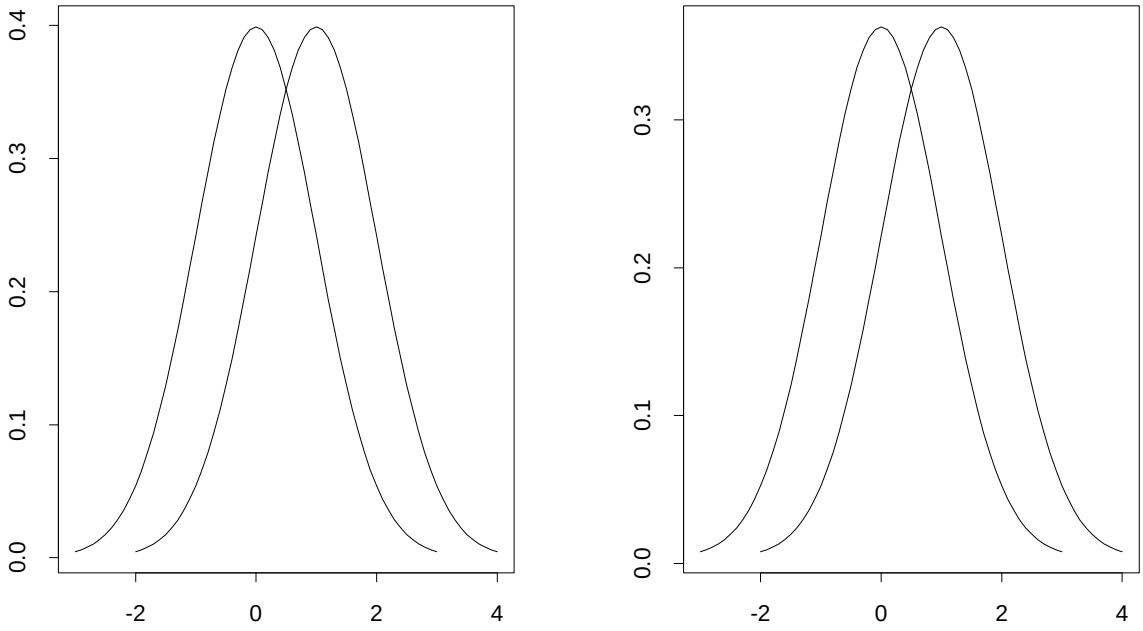

Figure 16: In the left panel, $\delta = 1$. In the right panel, $\delta = .3$, illustrating that a slight departure from normality can lower δ substantially.

where

$$S_W^2 = \frac{(n_1 - 1)s_{w1}^2 + (n_2 - 1)s_{w2}^2}{n_1 + n_2 - 2}$$

is the pooled Winsorized variance. Under normality, and when the variances are equal, $\delta = \delta_t$. If the Winsorized variances are not equal, Algina et al. suggest using both

$$\delta_{t1} = 0.642\frac{\bar{X}_{t1} - \bar{X}_{t2}}{s_{w1}},$$

and

$$\delta_{t2} = 0.642\frac{\bar{X}_{t1} - \bar{X}_{t2}}{s_{w2}}.$$

10.1 A Heteroscedastic and Robust Measure of Effect Size

Momentarily imagine that an equal number of observations is sampled from each group and let σ^2_{pool} be the (population) variance corresponding to the pooled observations. That is, we are pooling together two variables that might have different variances resulting in a variable that has variance σ^2_{pool}. Then a heteroscedastic, explanatory measure of effect size, xi , is

$$\xi = \sqrt{\frac{\sigma_\mu^2}{\sigma^2_{\text{pool}}}}.$$

To add perspective, it is noted that in the context of least squares regression, the approach leading to ξ^2 results in Pearson's squared correlation coefficient, the coefficient of determination. (Kulinskaya and Staudte, 2006, studied another approach that is somewhat related to ξ^2.) Also, as previously noted, Cohen suggested that under normality and homoscedasticity, $\delta = 0.2$, 0.5 and 0.8 correspond to small, medium and large effect sizes, respectively. The corresponding values of ξ are approximately, 0.15, 0.35 and 0.50.

Estimation of ξ^2 is straightforward when there are equal sample sizes. First estimate σ_μ^2 by replacing μ_1 and μ_2 with the corresponding sample means. Next, pool all $2n$ values and compute the sample variance, say s^2_{pool}, which estimates σ^2_{pool}. (Note that s^2_{pool} and s_p^2 are not the same. The latter is based on a weighted average of the individual sample variances.) But when there are unequal sample sizes, this estimation method can be shown to be unsatisfactory. To deal with this, suppose the sample sizes are $n_1 < n_2$ for groups 1 and 2, respectively. If we randomly sample (without replacement) n_1 observations from the second group, we get a satisfactory estimate of ξ^2. To use all of the data in the second group, we repeat this process many times yielding a series of estimates for ξ^2, which are then

averaged to get a final estimate, which we label $\hat{\xi}^2$. And the estimate of ξ is just

$$\hat{\xi} = \sqrt{\hat{\xi}^2}.$$

THIS MEASURE OF EFFECT SIZE IS NOT ROBUST

BUT A ROBUST VERSION IS READILY OBTAINED BY REPLACING THE MEANS AND VARIANCES WITH SOME ROBUST MEASURE OF LOCATION AND SCATTER.

Robust Variations

The measure of effect size xi can be made more robust. Again replace the means with trimmed means, and replace the pooled sample variance with the Winsorized variance that has been rescaled to estimate the variance under normality. Henceforth, when using the measure of effect size ξ, the version based on 20% trimmed means and the 20% Winsorized variances will be assumed unless stated otherwise.

EXAMPLE

In the right panel of Figure 16, $\xi = 0.56$ (based on 20% trimmed means) indicating a large effect size. This is in contrast to δ, which indicates a relatively small effect size.

EXAMPLE

For data comparing hangover symptoms of sons of alcoholics versus a control group, we reject the hypothesis of equal 20% trimmed means. If we use a standardized difference between the two groups based on the means and the standard deviation of the first group we get $\hat{\delta}_1 = 0.4$. Using the standard deviation of the second group yields $\hat{\delta}_2 = 0.6$. So taken together, and assuming normality, these results suggest a medium effect size. The estimate of ξ is 0.44 suggesting that this robust, heteroscedastic measure of effect size is fairly large.

EXAMPLE

Sexual attitude data. The sample sizes are $n_1 = 1327$ and $n_2 = 2282$, Welch's test returns a p-value of 0.30, but Yuen's test has a p-value less than 0.001. Cohen's effect size, d, is less

than 0.0001. In contrast, $\hat{\delta}_t = 0.48$, suggesting a medium effect size and $\hat{\xi} = 0.47$, suggesting a large effect size.

The R function

```
yuenv2(x,y)
```

is exactly like the R function `yuen`, only it also reports the effect size ξ. The R function

```
akp.effect(x,y,EQVAR=T,tr=.2)
```

computes the measure of effect size δ_t, which defaults to using a 20% trimmed mean. If the argument `EQVAR=F`, the function returns both δ_1 and δ_2. Setting `tr=0`, `akp.effect` returns Cohen's d when `EQVAR=T`.

11 Comparing Correlations and Least Squares Regression Slopes

The goal is to test

$$H_0 : \rho_1 = \rho_2,$$

the hypothesis that the two groups have equal population correlation coefficients.

DO NOT USE FISHER'S R-TO-Z TRANSFORMATION. CAN PERFORM POORLY UNDER NONNORMALITY.

Currently, one of the more effective approaches is to use a (modified) percentile bootstrap method.

When comparing slopes, based on least squares, a wild bootstrap method can be used as well as a non-bootstrap method based in part of the HC4 estimate of the standard, which deals well with heteroscedasticity.

IGNORING HETEROSCEDASTICITY CAN RESULT IN USING THE WRONG STANDARD ERROR, INVALIDATING THE RESULTS.

The R function

```
twopcor(x1,y1,x2,y2)
```

computes a confidence interval for the difference between two Pearson correlations corresponding to two independent groups using the modified bootstrap method just described. The data for group 1 are stored in the R variables x1 and y1, and the data for group 2 are stored in x2 and y2.

The R function

```
twolsreg(x1,y1,x2,y2)
```

computes a confidence interval for the difference between the slopes based on the least squares estimator.

The R function

```
tworegwb(x1,y1,x2,y2)
```

tests the hypothesis of equal slopes, via the least squares regression estimator, using the heteroscedastic wild bootstrap method in conjunction with the moderated multiple regression model. That is, assume

$$Y = \beta_0 + \beta_1 X_i + \beta_2 G_i + \beta_3 X_i G_i + e, i = 1, \ldots n,$$

where X_i is the predictor variable and G_i is a *dummy variable*, $G_i = 0$ if a pair of observations (X_i, Y_i) comes from the first group, otherwise $G_i = 1$. When $G_i = 0$, the model becomes

$$Y_i = \beta_0 + \beta_1 X_i + e;$$

and when $G_i = 1$ the model is

$$Y_i = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) X_i + e.$$

So if $\beta_3 = 0$, both groups have slope β_1 with different intercepts if $\beta_2 \neq 0$. That is, the hypothesis that both groups have equal slopes is

$$H_0 : \beta_3 = 0.$$

EXAMPLE

In an unpublished study by L. Doi, there was interest in whether a measure of orthographic ability (Y) is associated with a measure of sound blending (X). Here we consider

whether an auditory analysis variable (Z) *modifies* the association between X and Y. One way of approaching this issue is to partition the pairs of points (X, Y) according to whether Z is $\leq$ or > 14, and then enter the resulting pairs of points into the R function `twopcor`. The 0.95 confidence interval for $\rho_1 - \rho_2$, the difference between the correlations, is $(-0.64, 0.14)$. This interval contains zero so we would not reject the hypothesis of equal correlations. If we compare regression slopes instead, the 0.95 confidence interval is $(-0.55, 0.18)$ and again we fail to reject. It is stressed, however, that this analysis does not establish that the association does not differ for the two groups under study. A concern is that power might be low when attention is restricted to Pearson's correlation or least squares regression. Methods covered later indicate that the measure of auditory analysis does modify the association between orthographic ability and sound blending.

12 COMPARING TWO DEPENDENT GROUPS

There are several general ways two dependent groups might be compared:

1. Compare measures of location, such as the mean or median.
2. Compare measures of variation.
3. Focus on the probability that a randomly sampled observation from the first group is less than a randomly sampled observation from second group.
4. Use a rank-based method to test the hypothesis that the marginal distributions are identical.
5. Simultaneously compare all of the quantiles to get a global sense of where the distributions differ and by how much.

All of the methods for comparing independent groups have analogs for the case of two dependent groups.

12.1 When Does the Paired T Test Perform Well?

The good news about the paired T test is that if the observations in the first group (the X_{i1} values) have the same population distribution as the observations in the second group, so in particular they have equal means, variances and the same amount of skewness, generally Type I error probabilities substantially higher than the nominal level can be avoided. The reason is that for this special case, the difference scores (the D_i values) have a symmetric

(population) distribution in which case methods based on means perform reasonably well in terms of avoiding Type I error probabilities substantially higher than the nominal level. However, practical problems can arise when the two groups (or the two dependent variables) differ in some manner. Again, arbitrarily small departures from normality can destroy power, even when comparing groups having symmetric distributions. And if groups differ in terms of skewness, the difference scores (the D_i values) have a skewed distribution, in which case the paired T test can be severely biased, meaning that power can actually decrease as the difference between the means gets large. Yet another problem is poor probability coverage when computing a confidence interval for the difference between the means. If the goal is to test the hypothesis that the two variables under study have identical distributions, the paired T test is satisfactory in terms of Type I errors. But if we reject, there is doubt as to whether this is primarily due to the difference between the means or some other way the distributions differ. With a large enough sample size, these concerns become negligible, but just how large the sample size must be is difficult to determine.

13 Comparing Robust Measures of Location

Again, one way of dealing with the risk of relatively low power when comparing means is to use 20% trimmed means, medians or M-estimators instead. A possible appeal of methods for comparing 20% trimmed means is that by design, they perform about as well as the paired T test for means when the normality assumption is true. But in fairness, if a distribution is sufficiently light-tailed, comparing means might have a power advantage. And if distributions are sufficiently heavy-tailed, medians or M-estimators might provide more power. As usual, if distributions are skewed, comparing means is not the same as comparing a 20% trimmed mean or other robust measures of location. And despite any concerns about the mean, there are situations where it is more appropriate than a 20% trimmed mean. (See Chapter 2 of Wilcox, 2011.) But to get good power under a broad range of conditions, as well as improved control over the Type I error probability, using a 20% trimmed mean has appeal.

SOME PRELIMINARIES REGARDING DIFFERENCE SCORES VERSUS MARGINAL DISTRIBUTIONS. Imagine have the data

```
x
       HUSBAND        WIFE

             [,1]         [,2]
[1,] -1.512550659 -2.518859882
[2,]  0.851913806  0.315954303
```

```
 [3,]  1.411331832 -0.009223448
 [4,] -0.263640976 -0.691615289
 [5,] -0.065369901 -0.919673851
 [6,]  0.005091679 -0.185889866
 [7,] -0.350801421  0.330545513
 [8,]  0.797314918  0.041423880
 [9,]  0.872460313 -0.412274755
[10,] -0.456663115  0.891793351
[11,]  0.959838975 -1.442279774
[12,] -0.448607861 -1.731769107
[13,]  0.655643063  0.306367011
[14,]  0.427942746  0.640430419
[15,] -0.381066248 -1.161691294
[16,] -0.881927109  1.845648473
[17,] -0.364925416  0.620919115
[18,]  1.165781298  0.987349169
[19,] -1.708818501 -0.659770555
[20,] -2.237646390  0.606067220

> median(x[,1]-x[,2]) ---  median difference between husband and wife
[1] 0.3886252
> median(x[,1])-median(x[,2]) --- median of all males minus median of all females
[1] -0.1806057
> loc2dif(x[,1],x[,2]) ---   median of the difference between any male and any female
[1] 0.1210479
```

Same is true for trimmed means (tr>0) and M-estimators

For convenience, the focus is on trimmed means, but the methods about to be described extend to medians and M-estimators as will be made clear. If two dependent groups have identical distributions, then the population trimmed mean of the difference scores is equal to the difference between the individual trimmed means, which is zero. In symbols, $\mu_{tD} = \mu_{t1} - \mu_{t2} = 0$. However, if the distributions differ, in general it will be the case that $\mu_{tD} \neq \mu_{t1} - \mu_{t2}$. In practical terms, computing a confidence interval for μ_{tD} is not necessarily the same as computing a confidence interval for $\mu_{t1} - \mu_{t2}$. And the same is true when using medians or an M-estimator. So an issue is whether one should test

$$H_0 : \mu_{tD} = 0, \tag{24}$$

or

$$H_0 : \mu_{t1} = \mu_{t2}. \tag{25}$$

The latter approach is called comparing the *marginal trimmed means*. Put another way, the distributions associated with each of the dependent groups, ignoring the other group, are called *marginal distributions*. The goal is to compare the trimmed means of the marginal distributions.

If we test Equation (24), the goal in effect is to assess how the typical husband compares to his wife. Testing Equation (25) instead, the goal is to compare how the typical female compares to the typical male.

In terms of controlling the Type I error probability, currently it seems that there are only slight differences between the two approaches. A rough rule is that using difference scores, Type I errors. For example, under normality, if the pairs of observations have correlation $\rho = .4$, testing (24) at the 0.05 level (using the R function `trimci`), the actual Type I error probability is approximately 0.059 compared to 0.044 when testing (25) using the R function

yuend(x,y,tr=.2,alpha=.05).

With $\rho = 0$, the actual Type I error probabilities are approximately 0.059 and 0.047, respectively. So in this particular case, comparing marginal trimmed means results in a Type I error probability that is closer to the nominal 0.05 level. But if the distributions are lognormal, the Type I error probabilities are approximately 0.043 and 0.033, suggesting that using difference scores might have more power. In terms of maximizing power, the optimal choice depends on how the groups differ, which of course is unknown. If forced to choose one approach over the other, with goal of maximizing power, using difference scores seems to have an advantage. But perhaps situations arise where, from a substantive point of view, the choice between testing Equation (24) and (25) is relevant.

14 Missing Values

All indications are that from a robust point of view, something other than imputation should be used. If values are missing at random, can use a percentile bootstrap method to compare marginal trimmed means, which uses all of the available data.

The R function

`rmmismcp(x,y,est=tmean)`

test the hypothesis H_0: $\mu_{t1} = \mu_{t2}$ using this approach. With 20% trimming or more, it appears to be one of the better methods for general use when there are missing values. By default, a 20% trimmed mean is used, but other measures of location can be used via the argument est. For example, `rmmismcp(x,y,est=onestep)` would compare the groups with a one-step M-estimator. The function also computes a confidence interval.

EXAMPLE:

```
x
       HUSBAND        WIFE

              [,1]          [,2]
 [1,] -1.512550659       NA
 [2,]  0.851913806  0.315954303
 [3,]  1.411331832 -0.009223448
 [4,] -0.263640976 -0.691615289
 [5,] -0.065369901 -0.919673851
 [6,]  0.005091679 -0.185889866
 [7,] -0.350801421  0.330545513
 [8,]  0.797314918  0.041423880
 [9,]  0.872460313 -0.412274755
[10,] -0.456663115  0.891793351
[11,]  NA          -1.442279774
[12,] -0.448607861 -1.731769107
[13,]  0.655643063  0.306367011
[14,]  0.427942746  0.640430419
[15,] -0.381066248 -1.161691294
[16,] -0.881927109  1.845648473
[17,] -0.364925416       NA
[18,]  1.165781298  0.987349169
[19,] -1.708818501 -0.659770555
[20,] -2.237646390  0.606067220
```

Case wise deletion: remove any row with missing value.

But can use all of the data via the R function

```
rm2miss(x,y,tr=0)
```

to test H_0: $\mu_{t1} = \mu_{t2}$ assuming any missing values occur at random. It uses a bootstrap-t method in conjunction with an extension of the method derived by Lin and Stivers (1974), and by default it tests the hypothesis of equal means. It appears to be one of the better methods when the amount of trimming is small.

The function

l2drmci(x,y=NA,est=median,alpha=0.05,na.rm=T)

compares the groups using all pairwise differences among all males and females. Setting na.rm=F means that all of the data are used assuming missing values occur at random.

15 Comparing All Quantiles

Again, it might be of interest to compare low scoring individuals in one group to low scoring participants in another. This might be done by comparing the lower quartiles. Simultaneously, one might want to compare high scoring participants using the upper quartiles. This can done using an analog of the shift function for dependent groups.

The R function

```
lband(x,y=NA,alpha=0.05,plotit=T)
```

compares all quantiles of two dependent groups and plots the shift function if the argument plotit=T. If the argument y is not specified, it is assumed that the argument x is a matrix with two columns.

EXAMPLE

A study was performed where EEG measures of convicted murderers were compared to a control group. Measures for both groups were taken at four sites in the brain. For illustrative purposes, the first two sites are compared.

The R function lband finds differences at the 0.14, 0.5 and 0.57 quantiles. That is, the 0.14 quantile for the first site differs significantly different from the 0.14 quantile of the second site, and the same is true for the 0.5 and 0.57 quantiles.

16 One-Way ANOVA

When using means, problems due to nonnormality and heteroscedasticity are exacerbated as the number of groups increases.

All of the methods for comparing two groups in a robust manner can be extended to more than two groups.

TEST THE ASSUMPTIONS OF NORMALITY AND HOMOSCEDASTICITY? NOT SUPPORTED BASED ON PUBLISHED STUDIES. SUCH TESTS OFTEN DO NOT HAVE ENOUGH POWER TO DETECT VIOLATIONS OF ASSUMPTIONS THAT HAVE PRACTICAL IMPORTANCE.

TRIMMED MEANS:

Use the R function

16.1 R Functions t1way, t1wayv2 and t1wayF

The R function

```
t1way(x,tr=.2,grp=NA)
```

tests the hypothesis of equal trimmed. The argument x can have list mode or it can be a matrix. In the former case, the data for group 1 are stored in the variable x[[1]], group 2 is stored in x[[2]] and so on. In the latter case, x is an n by J matrix where column 1 contains the data for group 1, column 2 contains the data for group 2 and so forth. The argument tr indicates the amount of trimming and when tr=0, this function performs Welch's method for means. The argument grp allows you to compare a selected subset of the groups. By default all groups are used. If you set grp=c(1,3,4), then the trimmed means for groups 1, 3 and 4 will be compared with the remaining data ignored. The function returns the value of the test statistic and the corresponding significance level (so specifying a value for α is not necessary). The R function

```
t1wayv2(x,tr=.2,grp=NA)
```

is the same as `t1way`, only the measure of effect size ξ is reported as well.

EXAMPLE

Here is an example of a portion of the output from t1way:

```
$TEST:
[1] 5.059361

$nu1:
[1] 3

$nu2:
[1] 10.82531

$siglevel:
[1] 0.01963949
```

This says that the test statistic F_t is equal to 5.06 and the p-value, labled siglevel, is 0.0194. So in particular you would reject the hypothesis of equal trimmed means with $\alpha = 0.05$ or even 0.02. In contrast, as previously indicated, if we compare means with Welch's method or the ANOVA F test, we fail to reject with $\alpha = 0.05$, illustrating that the choice of method can alter the conclusions reached. Setting the argument tr to 0, t1way reports the results of Welch's test to be

```
$TEST:
[1] 2.038348

$nu1:
[1] 3

$nu2:
[1] 19.47356

$siglevel:
[1] 0.1417441
```

So switching from means to 20% trimmed means, the p-value drops from 0.14 to about 0.02.

EXAMPLE

It is common to find situations where we get a smaller p-value using trimmed means rather than means. However, the reverse situation can and does occur. This point is illustrated with data taken from Le (1994) where the goal is to compare the testosterone levels of four groups of male smokers: heavy smokers (group 1), light smokers (group 2), former smokers (group 3) and non-smokers (group 4). The data are:

G1	G2	G3	G4
.29	.82	.36	.32
.53	.37	.93	.43
.33	.77	.40	.99
.34	.42	.86	.95
.52	.74	.85	.92
.50	.44	.51	.56
.49	.48	.76	.87
.47	.51	.58	.64
.40	.61	.73	.78
.45	.60	.65	.72

The p-value using Welch's method is 0.0017. Using 20% trimmed means instead, the p-value is 0.029. One reason this is not surprising is that a boxplot for each group reveals no outliers and it can be seen that the estimated standard errors for the means are smaller than the corresponding standard errors of the 20% trimmed means. However, a boxplot for the first group suggests that the data are skewed which might be having an effect on the p-value of Welch's test beyond any differences among the means.

The R function

```
t1wayF(x,fac,tr=.2,nboot=100,SEED=T)
```

is like the R function `t1wayv2`, only `x` is a column of data and the argument `fac` is a factor variable. In essence, this function eliminates the need to use the R function `fac2list`.

EXAMPLE

Assuming the plasma retinol data (described in Chapter 1 of Wilcox, 2011) are stored in the R variable `plasma`,

t1wayF(plasma[,13],plasma[,3]).

would compare the three groups indicated by column 3 (smoking status) using the data in column 13, which are plasma beta-carotene measures.

16.2 BOOTSTRAP METHODS FOR ONE-WAY ANOVA

The bootstrap-t method (in Chapter 8 of Wilcox, 2011) can be extended to the problem of testing

$$H_0 : \mu_{t1} = \mu_{t2} = \cdots = \mu_{tJ},$$

the hypothesis of equal trimmed means. When comparing means or trimmed means with a small amount of trimming, a bootstrap-t method appears to be a relatively good method for general use. The strategy is to use the available data to estimate an appropriate critical value for the test statistic

The R function

```
t1waybt(x, tr = 0.2, alpha = 0.05, grp = NA, nboot = 599)
```

performs the bootstrap-t method for trimmed means that was just described. The argument x is any R variable containing data that are stored in list mode or in a matrix. In the first case x[[1]] contains the data for group 1, x[[2]] contains the data for group 2 and so on. If x is a matrix, column 1 contains the data for group 1, column 2 the data for group 2, and so forth. The argument `grp` can be used to analyze a subset of the groups. For example, `grp=c(2,4,5)` would compare groups 2, 4 and 5 only. As usual, alpha is α and nboot is B,

the number of bootstrap samples to be used.

EXAMPLE

Data from a study dealing with schizophrenia are used to illustrate the bootstrap-t method. If the data are stored in the R variable skin, the command

```
t1waybt(skin,tr=0)
```

tests the hypothesis of equal means and returns

```
$test:
[1] 2.04

$p.value
[1] 0.2307692
```

Although not indicated, if the goal were to test the hypothesis of equal means at the 0.05 level, the critical value would have been 4.65. The critical value, assuming normality, is 3.1.

There are situations where a percentile bootstrap method is preferable a bootstrap-t method or any non-bootstrap method that uses standard errors. If the goal is to compare M-estimators, for example, all indications are that a percentile bootstrap method is best for general use. If the amount of trimming is reasonably large, say at least 20%, again a percentile bootstrap method performs relatively well. But when comparing medians, again tied values can create serious practical problems and the methods described here are not recommended. (Use the percentile bootstrap for medians, in Chapter 13 of Wilcox, 2011, in conjunction with the R function `medpb`.)

Method SHPB

Let θ be any population measure of location, such as the 20% trimmed mean (μ_t) or the median. There are many variations of the percentile bootstrap method that can be used to test

$$H_0 : \theta_1 = \cdots = \theta_J,$$

the hypothesis that J groups have a common measure of location, but only two are described here. The first is related to a test statistic mentioned by Schrader and Hettmansperger (1980) and studied by He, Simpson and Portnoy (1990), and will be called *method SHPB*.

Let $\hat{\theta}_j$ be an estimate of θ based on data from the jth group ($j = 1, \ldots, J$). The test statistic is

$$H = \frac{1}{N} \sum n_j (\hat{\theta}_j - \bar{\theta})^2,$$

where $N = \sum n_j$, and

$$\bar{\theta} = \frac{1}{J} \sum \hat{\theta}_j.$$

To determine a critical value, shift the empirical distributions of each group so that the measure of location being used is equal to zero. That is, set $Y_{ij} = X_{ij} - \hat{\theta}_j$. Then generate bootstrap samples from each group in the usual way from the Y_{ij} values and compute the test statistic based on the bootstrap samples yielding H^*. Repeat this B times resulting in $H_1^*, \ldots, H_B^*$, and put these B values in order yielding $H_{(1)}^* \leq \cdots \leq H_{(B)}^*$. An estimate of an appropriate critical value is $H_{(u)}^*$, where $u = (1 - \alpha)B$, rounded to the nearest integer, and H_0 is rejected if $H \geq H_{(u)}^*$. (For simulation results on how this method performs when comparing M-estimators, see Wilcox, 1993b.)

Method LSPB

The second method stems from general results derived by Liu and Singh (1997) and will be called *method LSPB*. To convey the basic strategy, momentarily consider three groups only, let

$$\delta_{12} = \theta_1 - \theta_2,$$

be the difference between the measures of location for groups 1 and 2, let

$$\delta_{13} = \theta_1 - \theta_3$$

be the difference between the measures of location for groups 1 and 3 and let

$$\delta_{23} = \theta_2 - \theta_3$$

be the difference between the measures of location for groups 2 and 3. If the null hypothesis is true, then $\delta_{12} = \delta_{13} = \delta_{23} = 0$. To simplify matters, momentarily focus on δ_{12} and δ_{13}. Further imagine that we generate bootstrap samples from each group yielding δ_{12}^* and δ_{13}^* and that we repeat this process many times for a situation where the null hypothesis is true. Then we might get the plot shown in the left panel of Figure 17. Note that the point (0, 0)

is the near the center of cloud of points. Said another way, the point (0, 0) is deeply nested within the cloud of bootstrap values. This is what we would expect when null hypothesis is true, which corresponds to $\delta_{12} = \delta_{13} = 0$. But if the null hypothesis is false, a plot of the bootstrap values might yield something like the right panel of Figure 17. Now the point (0, 0) is near the outside edge of the cloud. That is, (0, 0) is not deeply nested within the cloud of bootstrap values suggesting that the point $(\delta_{12}, \delta_{13}) = (0, 0)$ is relatively unusual and that the null hypothesis should be rejected.

Generalizing, let

$$\delta_{jk} = \theta_j - \theta_k,$$

where for convenience it is assumed that $j < k$. That is, the δ_{jk} values represent all pairwise differences among the J groups. When working with means, for example, δ_{12} is the difference between the means of groups 1 and 2, and δ_{35} is the difference for groups 3 and 5. If all J groups have a common measure of location (i.e., $\theta_1 = \cdots = \theta_J$), then in particular

$$H_0 : \delta_{12} = \delta_{13} = \cdots = \delta_{J-1,J} = 0 \tag{26}$$

is true. It can be shown that the total number of δ's is $L = (J^2 - J)/2$. For example, if $J = 3$, there are $L = 3$ values: δ_{12}, δ_{13} and δ_{23}.

For each group, generate bootstrap samples from the *original* values and compute the measure of location for each group. That is, the observations are *not* centered. Said another way, bootstrap samples are *not* generated from the Y_{ij} values but rather from the X_{ij} values. Repeat this B times. The resulting estimates of location are represented by $\hat{\theta}^*_{jb}$ $(j = 1, \ldots, J; b = 1, \ldots, B)$ and the corresponding estimates of δ are denoted by $\hat{\delta}^*_{jkb}$. (That is, $\hat{\delta}^*_{jkb} = \hat{\theta}^*_{jb} - \hat{\theta}^*_{kb}$.) The general strategy is to determine how deeply $\mathbf{0} = (0, \ldots, 0)$ is nested within the bootstrap values $\hat{\delta}^*_{jkb}$ (where $\mathbf{0}$ is a vector having length L). For the special case where only two groups are being compared, this is tantamount to determining the proportion of times $\hat{\theta}^*_{1b} > \hat{\theta}^*_{2b}$, among all B bootstrap samples. But here we need special techniques for comparing more than two groups.

Method SHPB can be applied with the R function

```
b1way(x,est=onestep,alpha=.05,nboot=599).
```

By default it uses an M-estimator (with Huber's Ψ). The function

```
pbadepth(x, est=onestep, con=0, alpha=0.05, nboot=2000, grp=NA, op=1, MM =
                    F, MC=F, SEED=T, na.rm = F, ...)
```

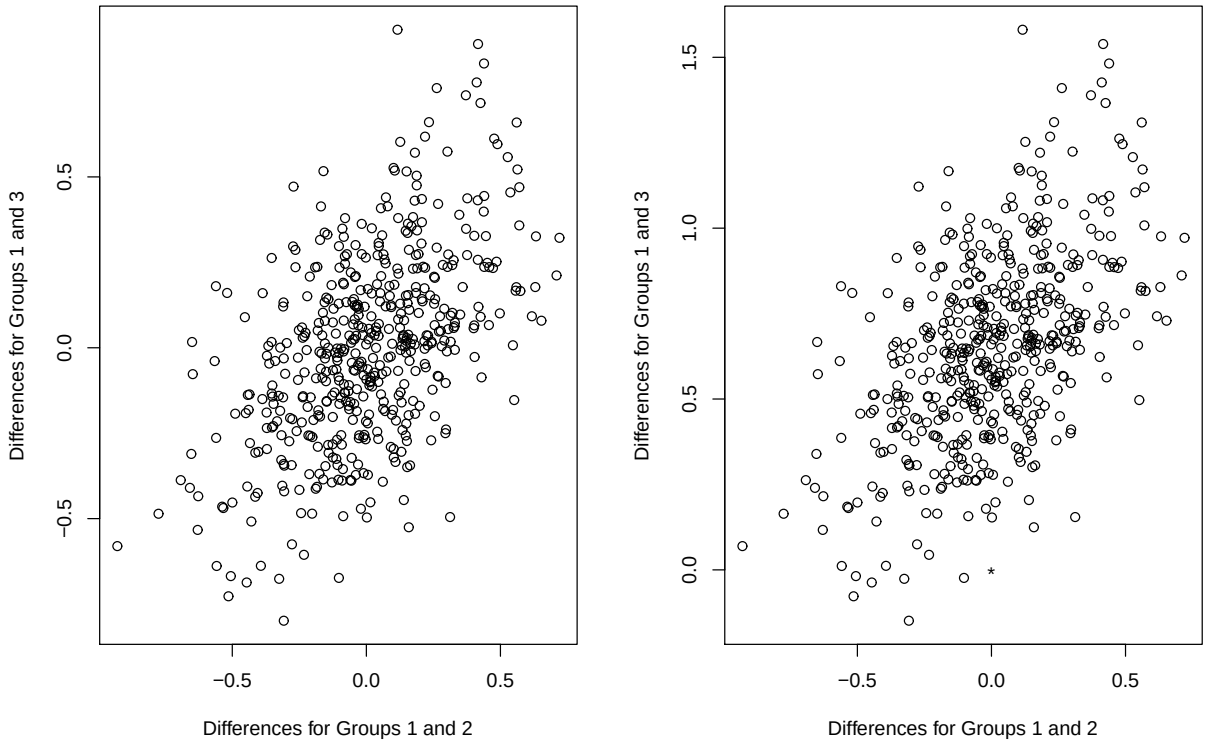

Figure 17: Plots of the differences between bootstrap measures of location. The left panel corresponds to a situation where the null hypothesis of equal population measures of location is true, in which case the differences should be clustered around the point (0, 0). That is, the point (0, 0) should be nested deeply within the cloud of points. The right panel corresponds to a situation where the null hypothesis is false. Note that now the point (0, 0), marked by *, is near the edge of the cloud suggesting that the point (0, 0) is relatively unusual.

performs method LSPB. As usual, the argument ... can be used to reset default settings associated with the estimator indicated by the argument `est`. The argument `op` determines how the distance from the center of the bootstrap cloud is measured. By default, Mahalanobis distance is used. (The argument `con` is explained in Chapters 11 and 13.) To use projection distances, set `op=3`. If `op=3` and `MC=T`, the function uses a multi-core processor, assuming one is available and the R package multicore has been installed; this will reduce execution time.

16.3 Rank-Based Methods

Brunner, Dette and Munk (1997) method currently stands out. (It improves on the Kruskal-Wallis test.) Roughly, testing the hypothesis of identical distributions in a manner that is sensitive to the differences among the means of the ranks. (Ranks are determined based on the pooled data.) The R function

```
bdm(x)
```

performs the BDM rank-based ANOVA

16.4 Two-Way ANOVA

R has a built-in function for plotting interactions, which has the form

```
interaction.plot(x.factor, trace.factor, response, fun = mean,trace.label =
deparse(substitute(trace.factor)), xlab = deparse(substitute(x.factor)),ylab
                                    = ylabel)
```

(This function has some additional arguments not described here.) The first argument, `x.factor`, indicates the levels (the groups) that will be indicated along the x-axis. (It is typically a factor variable.) The argument `trace.factor` indicates the factor that will be plotted within the plot. The argument `response` contains the data and `fun` indicates the measure of location that will be be used.

If data are stored in a matrix or list mode, with no factor variables, it is more convenient to plot interactions with the R function

```
interplot(J, K, x, locfun = mean, locvec = NULL, g1lev = NULL, g2lev = NULL,
 type = c("l", "p", "b"), xlab = "Fac 1", ylab = "means", trace.label = "Fac
                                        2").
```

The measure of location can be chosen via the argument `locfun`. But if measures of location are stored in `locvec`, these values are used to create the plot instead. If the data are stored in list mode, the first K groups are assumed to be the data for the first level of Factor A, the next K groups are assumed to be data for the second level of Factor A, and so on. In R notation, x[[1]] is assumed to contain the data for level 1 of Factors A and B, x[[2]] is assumed to contain the data for level 1 of Factor A and level 2 of Factor B, and so forth. If, for example, a 2-by-4 design is being used, the data are stored as follows:

		Factor B		
Factor	x[[1]]	x[[2]]	x[[3]]	x[[4]]
A	x[[5]]	x[[6]]	x[[7]]	x[[8]]

For instance, x[[5]] contains the data for the second level of Factor A and the first level of Factor B.

If the data are stored in a matrix, the first K columns are assumed to be the data for the first level of Factor A, the next K columns are assumed to be data for the second level of Factor A, and so on.

EXAMPLE

A 2-by-2 ANOVA. Weight gain based two sources of protein and two amounts of protein. Imagine that the data stored in the R variable `mm` with column one containing the data, column 2 indicating the source of protein (beef and cereal) and column 3 indicating the amount of protein (high and low). Then the R command

```
interaction.plot(mm[,2],mm[,3],mm[,1],trace.label="Amount",
       xlab="Source",ylab="Trimmed Means",fun=mean)
```

produces a plot of the interactions based on means with the factor source on the x-axis. If the data are stored in a matrix called cereal, the command

```
interplot(2,2,cereal,g1lev=c("B","C"),g2lev=c("H","L"),
xlab="Source",trace.label="Amount",ylab="Trimmed Means",locfun=tmean)
```

would a produce a similar plot but with means replaced by 20% trimmed means. The groups for the factor source would be labeled B and C and indicated along the x-axis.

16.5 Two-Way ANOVA: Hypothesis Testing

A positive feature of the ANOVA F test is that if groups have identical distributions, the probability of a Type I error seems to be controlled reasonably well under nonnormality. But violating the assumption of equal variances can result can result in poor power properties (power can go down as the means become unequal), unsatisfactory control over the probability of a Type I error, and relatively low power compared to other methods that might be used.

Numerous methods have been proposed for dealing with heteroscedasticity when comparing measures of location. Here the focus is on comparing trimmed means, which of course contains comparing means as a special case. The method used here is based on an extension of a method derived by Johansen (1980), which represents a heteroscedastic approach to what is known as the general linear model. The computational details are described in chapter 11 of Wilcox (2011).

16.6 R Function t2way

The R function

```
t2way(J, K, x, tr = 0.2, grp=c(1:p), p=J*K)
```

tests the hypotheses of no main effects and no interaction. Here, the arguments `J` and `K` denote the number of levels associated with Factors A and B. Like t1way, the data are assumed to be stored in `x`, which can be any R variable that is a matrix or has list mode.

The data are assumed to be stored as previously described. If they are not stored in the assumed order, the argument `grp` can be used to correct this problem. As an illustration, suppose the data are stored as follows:

		Factor B		
Factor	x[[2]]	x[[3]]	x[[5]]	x[[8]]
A	x[[4]]	x[[1]]	x[[6]]	x[[7]]

That is, the data for level 1 of Factors A and B are stored in the R variable x[[2]], the data for level 1 of A and level 2 of B is stored in x[[3]], and so forth. To use t2way, first enter the R command

```
grp=c(2,3,5,8,4,1,6,7).
```

Then the command `t2way(2,4,x,grp=grp)` tells the function how the data are ordered. In the example, the first value stored in `grp` is 2, indicating that x[[2]] contains the data for level 1 of both Factors A and B, the next value is 3, indicating that x[[3]] contains the data for level 1 of A and level 2 of B, and the fifth value is 4, meaning that x[[4]] contains the data for level 2 of Factor A and level 1 of B. As usual, `tr` indicates the amount of trimming, which defaults to 0.2, and alpha is α, which defaults to 0.05. The function returns the test statistic for Factor A, V_a, in the R variable t2way$test.A, and the significance level is returned in t2way$sig.A. Similarly, the test statistics for Factor B, V_b, and interaction, V_{ab}, are stored in t2way$test.B and t2way$test.AB, with the corresponding significance levels stored in t2way$sig.B and t2way$sig.AB.

As a more general example, the command

```
t2way(2,3,z,tr=.1,grp=c(1,3,4,2,5,6))
```

would perform the tests for no main effects and no interactions for a 2-by-3 design for the data stored in the R variable z, assuming the data for level 1 of Factors A and B are stored in z[[1]], the data for level 1 of A and level 2 of B are stored in z[[3]], and so on. The analysis would be based on 10% trimmed means.

Note that t2way contains an argument `p`. Generally this argument can be ignored; it is used by t2way to check whether the total number of groups being passed to the function is equal to JK. If JK is not equal to the number of groups in `x`, the function prints a warning message. If, however, if the goal is to perform an analysis using some subset of the groups stored in `x`, this can be done simply by ignoring the warning message. For example, suppose `x` contains data for 10 groups and it is desired to use groups 3, 5, 1, and 9 in a 2-by-2 design. That is, groups 3 and 5 correspond to level 1 of the first factor and levels 1 and 2 of the second. The command

```
t2way(2,2,x,grp=c(3,5,1,9))
```

accomplishes this goal.

EXAMPLE

A total of $N = 50$ male Sprague-Dawley rats were assigned to one of six conditions corresponding to a 2-by-3 ANOVA. (The data in this example were supplied by U. Hayes.) The two levels of the first factor have to do with whether an animal was placed on a fluid restriction schedule one week prior to the initiation of the experiment. The other factor had to do with the injection of one of three drugs. One of the outcome measures was sucrose consumption shortly after acquisition of a LiCl-induced conditioned taste avoidance. The output from t2way appears as follows:

```
$Qa
[1] 11.0931

$A.p.value
[1] 0.001969578

$Qb
[1] 3.764621

$B.p.value
[1] 0.03687472

$Qab
[1] 2.082398

$AB.p.value
[1] 0.738576
```

So based on 20% trimmed means, there is a main effect for both Factors A and B, but no interaction is detected.

EXAMPLE

Using the plasma data, this next example illustrates how to use `t2way` if columns of the data are used to indicate the levels of the factors. First, use the R command

```
z=fac2list(plasma[,10],plasma[,2:3])
```

to convert the data into list mode. In effect, the data in column 10 are divided into groups according to the data stored in columns 2 and 3. Note that there is no need to indicate that the second argument is a factor variable. (The function assumes that this should be the case and converts it to a factor variable if necessary.) Then

```
t2way(2,3,z)
```

performs the analysis and returns

```
$Qa
[1] 18.90464

$A.p.value
[1] 0.001

$Qb
[1] 4.702398

$B.p.value
[1] 0.151

$Qab
[1] 2.961345

$AB.p.value
[1] 0.286
```

The R function

```
pbad2way(J,K,x,est=onestep,conall=T,alpha = 0.05,nboot=2000,grp =
NA,pro.dis=F,...)
```

performs tests the hypothesis of main effects and no interactions using a percentile bootstrap method.

To explain the argument conall, let θ be any measure of location and let

$$\begin{aligned}
\Upsilon_1 &= \frac{1}{K}(\theta_{11} + \theta_{12} + \cdots + \theta_{1K}),\\
\Upsilon_2 &= \frac{1}{K}(\theta_{21} + \theta_{22} + \cdots + \theta_{2K}),\\
&\vdots\\
\Upsilon_J &= \frac{1}{K}(\theta_{J1} + \theta_{J2} + \cdots + \theta_{JK}).
\end{aligned}$$

So Υ_j is the average of the K measures of location associated with the jth level of Factor A. The hypothesis of no main effects for Factor A is

$$H_0 : \Upsilon_1 = \Upsilon_2 = \cdots = \Upsilon_J.$$

One variation of the percentile bootstrap method is to test

$$H_0 : \Delta_1 = \cdots = \Delta_{J-1} = 0, \tag{27}$$

where

$$\Delta_j = \Upsilon_j - \Upsilon_{j+1},$$

$j = 1, \ldots, J-1$. Briefly, generate bootstrap samples in the usual manner yielding $\hat{\Delta}_j^*$, a bootstrap estimate of Δ_j. Then proceed as described in Section 7.6. That is, determine how deeply $\mathbf{0} = (0, \ldots, 0)$ is nested within the bootstrap samples. If $\mathbf{0}$ is relatively far from the center of the bootstrap samples, reject.

The method just described is satisfactory when dealing with the probability of a type I error, but when the groups differ, this approach might be unsatisfactory in terms of power depending on the pattern of differences among the Υ_j values. One way of dealing with this issue is to compare all pairs of the Υ_j instead. That is, for every $j < j'$, let

$$\Delta_{jj'} = \Upsilon_j - \Upsilon_{j'},$$

and then test

$$H_0 : \Delta_{12} = \Delta_{13} = \cdots = \Delta_{J-1,J} = 0. \tag{28}$$

This is accomplished with the argument conall=T. Of course, a similar method can be used when dealing with Factor B.

With `conall=F`, the hypothesis given by Equation (27) is tested. Using `conall=T` can result in a numerical error as illustrated momentarily. (It uses Mahalanobis distance, which requires the inverse of a covariance matrix. But the covariance matrix can be singular.) If this numerical error occurs, there are two options. The first is to use `conall=F`, which is acceptable if all of the hypotheses are rejected. But if one or more are not rejected, the suggestion is to use `pro.dis=T`, which avoids the numerical error by replacing Mahalanobis distance with what is called projection distance. (Projection distance does not require the inverse of a covariance matrix.) A possible appeal of using `conall=F` is that it can result in faster execution time compared to using `pro.dis=T`.

The R function

```
t2waybt(J, K, x, tr = 0.2, grp = c(1:p), p = J * K, nboot = 599, SEED = T)
```

compares groups using a bootstrap-t method.

EXAMPLE

For the plasma retinol data, we compare plasma retinol levels (stored in column 14 of the R variable `plasma`) based on one-step M-estimators and two factors: sex and smoking status. The two R commands

```
z=fac2list(plasma[,14],plasma[,2:3])
pbad2way(2,3,z)
```

attempt to accomplish this goal using Mahalanobis distance, but this returns

```
Error in solve.default(cov, ...):
system is computationally singular:
reciprocal condition number = 0.
```

To avoid this error, use projection distance via the R command

```
pbad2way(2,3,z,pro.dis=T),
```

which reports the following p-values:

```
$sig.levelA
[1] 0.204

$sig.levelB
[1] 0.102

$sig.levelAB
[1] 0.338
```

So in particular, no differences are found when testing at the 0.05 level.

However, comparing 20% trimmed means via a bootstrap-t method, using the R command

```
t2waybt(2,3,z)
```

returns

```
$A.p.value
[1] 0.1218698
$B.p.value
[1] 0.03505843
$AB.p.value
[1] 0.2504174
```

So now we conclude that there are differences among the three groups of smokers, demonstrating again that the choice of method can make a practical difference. If `t2way` is used instead (the non-bootstrap method for comparing trimmed means), the p-value for Factor B is now 0.041.

RANK-BASED METHODS

The R function

```
bdm2way(J,K,x,alpha=0.05)
```

performs rank-based method. The function returns p-values for each of the hypotheses tested as well as an estimate of what is called the relative effects. Following Akritas et al., relative effects refer to the average ranks associated with each of the JK groups minus 1/2. That is, for levels j and k of Factors A and B, respectively, the relative effect is $\bar{R}_{jk} - 0.5$, where $\bar{R}_{jk}$ is the average of the ranks. (The relative effects are related to a weighted sum of the cumulative probabilities, where the weights depend on the sample sizes.)

Patel and Hoel (1973) proposed an alternative approach to interactions in a 2-by-2 design. For level 1 of Factor A, let $P_{11,12}$ be the probability that a randomly sampled observation from level 1 of Factor B is less than a randomly sampled observation from a level 2 of Factor B. Similarly, for level 2 of Factor A, let $P_{21,22}$ be the probability that that a randomly sampled observation from level 1 of Factor B is less than a randomly sampled observation from a level 2 of Factor B. The Patel-Hoel definition of no interaction is that

$$p_{11,12} = p_{21,22}.$$

The R function `rimul` performs this test.

17 Three-Way ANOVA

To deal with heteroscedasticity, and for the more general goal of comparing trimmed means, use the R function

```
t3way(J,K,L,x,tr=.2, grp = c(1:p)).
```

The data are assumed to be arranged such that the first L groups correspond to level 1 of Factors A and B ($J = 1$ and $K = 1$) and the L levels of Factor C. The next L groups correspond to the first level of Factor A, the second level of Factor B, and the L levels of Factor C. If, for example, a 3-by-2-by-4 design is being used and the data are stored in list mode, it is assumed that for $J = 1$ (the first level of the first factor), the data are stored in the R variables x[[1]],...,x[[8]] as follows:

		Factor C		
Factor	x[[1]]	x[[2]]	x[[3]]	x[[4]]
B	x[[5]]	x[[6]]	x[[7]]	x[[8]]

For the second level of the first factor, $J = 2$, it is assumed that the data are stored as

	Factor C			
Factor	x[[9]]	x[[10]]	x[[11]]	x[[12]]
B	x[[13]]	x[[14]]	x[[15]]	x[[16]]

If the data are not stored as assumed by t3way, `grp` can be used to indicate the proper ordering. As an illustration, consider a 2-by-2-by-4 design and suppose that for $J = 1$, the data are stored as follows:

	Factor C			
Factor	x[[15]]	x[[8]]	x[[3]]	x[[4]]
B	x[[6]]	x[[5]]	x[[7]]	x[[8]]

and for $J = 2$

	Factor C			
Factor	x[[10]]	x[[9]]	x[[11]]	x[[12]]
B	x[[1]]	x[[2]]	x[[13]]	x[[16]]

Then the R command

```
grp=c(15,8,3,4,6,5,7,8,10,9,11,12,1,2,13,16)
```

and the command `t3way(2,2,3,x,grp=grp)` will test all of the relevant hypotheses at the 0.05 level using 20% trimmed means.

If the data are stored with certain columns containing information about the levels of the factors, again the R function `fac2list` can be used to convert the data into list mode so that the R function `t3way` can be used.

18 COMPARING MORE THAN TWO DEPENDENT GROUPS

We can fail to reject when comparing the marginal trimmed means, but we reject when using difference scores, the point being that the choice of method can make a practical difference. This is not to suggest, however, that difference scores always give the highest power. The only certainty is that the reverse can happen, so both approaches are covered in this chapter.

18.1 Inferences Based on Difference Scores

Another approach is to test some appropriate hypothesis based on difference scores. One possibility is to form difference scores between the first group and the remaining $J-1$ groups and then test the hypothesis that the corresponding (population) measures of location are all equal to zero. But this approach is unsatisfactory in the sense that if we rearrange the order of the groups, power can be affected substantially.

We can avoid this problem by instead forming difference scores among all pairs of groups. There are a total of

$$L = \frac{J^2 - J}{2}$$

such differences. In symbols, we compute

$D_{i1} = X_{i1} - X_{i2}$,

$D_{i2} = X_{i1} - X_{i3}$,

$\vdots$

$D_{iL} = X_{i,J-1} - X_{iJ}$.

The R function

```
rmdzero(x,est=onestep, grp = NA, nboot = NA, ...)
```

performs the test on difference scores.

EXAMPLE

Rao's cork data consists of 4 measures taken from the north, south east and west sides of a tree. Setting the argument `est=tmean`, `rmdzero` returns a p-value of 0.014, so in particular reject with $\alpha = 0.05$. When these data were analyzed using bd1way, the p-value is 0.20. This result is in sharp contrast to the p-value obtained here, which illustrates that the choice of method can make a substantial difference in the conclusions reached.

18.2 Rank-Based Methods

Numerous rank-based methods have been proposed with the goal of improving on Friedman's test in terms of both Type I errors and power. A fundamental criticism of the Friedman test is that it is based on a highly restrictive assumption regarding the covariance matrix, called *compound symmetry*, which means that all J groups have a common variance (homoscedasticity) and all pairs of groups have the same covariance. An improvement on Friedman's test that currently stands out is described by Brunner, Domhof and Langer (2002, Section 7.2.2), and will be called *method BPRM*. This method also improves on a technique derived by Agresti and Pendergast (1986), which in turn improves on a method derived by Quade (1979). (R also has a built-in function for applying Quade's test.)

The R function

```
bprm(x)
```

performs method BPRM just described, where the argument `x` is assumed to be a matrix with J columns corresponding to groups, or it can have list mode. This function returns a p-value.

18.3 Between-by-Within Designs

Consider a two-way ANOVA design involving with J levels associated with the first factor and K levels with the second. A between-by-within or a split-plot design refers to a two-way design where the levels of the first factor are independent, but the measures associated with the K levels of the second factor are dependent instead. The R function

```
bwtrim(J, K, x, tr = 0.2, grp = c(1:p))
```

tests hypotheses about trimmed means. Here, J is the number of independent groups, K is the number of dependent groups, `x` is any R variable that is a matrix or has list mode, and as usual, the argument `tr` indicates the amount of trimming, which defaults to 0.2 if unspecified. If the data are stored in list mode, it is assumed that x[[1]] contains the data for level 1 of both factors, x[[2]] contains the data for level 1 of the first factor and level 2 of the second, and so on. If the data are stored in a matrix (or a data frame), it is assumed that the first K columns correspond to level 1 of Factor A, the next K columns correspond to level 2 of Factor A, and so on. If the data are not stored in the proper order, the argument

`grp` can be used to indicate how they are stored. For example, if a 2-by-2 design is being used, the R command

```
bwtrim(2,2,x,grp=c(3,1,2,4))
```

indicates that the data for the first level of both factors are stored in x[[3]], the data for level 1 of Factor A and level 2 of Factor B are in x[[1]], and so forth.

EXAMPLE

R has a built-in variable ChickWeight, which is a matrix. There are four independent groups of chicks that received a different diet. The diet they received is indicated in column 4 (by the values 1, 2, 3 and 4). Each chick was measured at 12 different times, with times indicated in column 2. So we have a 4-by-12 between-by-within design. Column 1 contains the weight of the chicks. To analyze the data with the R function `bwtrim`, the data need to be stored as described in this section, which can be done with the R command

```
z=fac2list(ChickWeight[,1],ChickWeight[,c(4,2)]).
```

Then the 10% trimmed means are compared with the command

```
bwtrim(4,12,z,tr=.1).
```

The p-value when comparing the four diets is 0.061. The p-value when comparing weight over time is extremely small, but of course the result that chicks gain weight over time is not very interesting. The goal here was merely to illustrate a feature of R.

Important: In this last example, look at the last argument when using the R function `fac2list`, `ChickWeight[,c(4,2)]`, and notice the use of `c(4,2)`. This indicates that the levels of the between factor are stored in column 4 and the levels of the within factor are stored in column 2. If we had used `c(2,4)`, this would indicate that the levels in column 2 correspond to the between factor, which is incorrect.

Situations are encountered where data are stored in a matrix or a data frame with one column indicating the levels of the between factor and one or more columns containing

data corresponding to different times. Imagine, for example, three medications are being investigated regarding their effectiveness to lower cholesterol and that column 3 indicates which medication a participant received. Moreover, columns 5 and 8 contain the participants' cholesterol level at times 1 and 2, respectively. Then the R function

```
bw2list(x, grp.col, lev.col).
```

will convert the data to list mode so that the R function `bwtrim` can be used. The argument grp.col indicates the column indicating the levels of the independent groups. And the argument lev.col indicates the K columns where the within group data are stored. For the situation just described, if the data are stored in the R variable `chol`, the R command

```
z=bw2list(chol,3,c(5,8))
```

will store the data in `z`, after which the R command

```
bwtrim(3,2,z)
```

will perform the analysis.

The R function

```
tsplitbt(J,K,x,tr=.2,alpha=.05,grp=c(1:JK),nboot=599)
```

performs a bootstrap-t method for a split-plot (between-by-within) design.

18.4 Percentile Bootstrap Method

The R function

```
sppba(J,K,x,est=onestep,grp = c(1:JK),avg=F,nboot=500,...)
```

tests the hypothesis of no main effects. Setting the argument `avg=T` (for true) indicates that the averages of the measures of location (the $\bar{\theta}_{j.}$ values) will be used. That is, test the hypothesis of no main effects for Factor A. The remaining arguments have their usual meaning. The R function

```
sppbb(J,K,x,est=onestep,grp = c(1:JK),nboot=500,...)
```

tests the hypothesis of no main effects for Factor B and

```
sppbi(J,K,x,est=onestep,grp = c(1:JK),nboot=500,...)
```

tests the hypothesis of no interactions.

EXAMPLE

A study was conducted where the goal was to compare the EEG measures for murderers to a control group. For each participant, eeg measures were taken at four sites in the brain. Imagine that the data are stored in the R variable `eeg`, a matrix with 8 columns, with the control group data stored in the first 4 columns. Then the hypothesis of no main effects (the control groups does not differ from the murderers) can be tested with the command

```
sppba(2,4,x).
```

The hypotheses of no main effects among sites as well as no interactions are tested in a similar manner using `sppbb` and `sppbi`.

18.5 Rank-Based Method

The R function

```
bwrank(J,K,x)
```

performs a between by within ANOVA based on ranks. Null hypotheses are stated in terms of the various distributions. For example, for Factor A (independent groups), let

$$\bar{F}_{j.}(x) = \frac{1}{K}\sum_{k=1}^{K} F_{jk}(x),$$

the average of the distributions among the K levels of Factor B corresponding to the jth level of Factor A. The hypothesis of no main effects for Factor A is

$$H_0 : \bar{F}_{1.}(x) = \bar{F}_{2.}(x) = \cdots = \bar{F}_{J.}(x).$$

for any x.

EXAMPLE

The notation is illustrated with data taken from Lumley (1996) dealing with shoulder pain after surgery; the data are from a study by Jorgensen et al. (1995). Table 3 shows a portion of the results where two treatment methods are used and measures of pain are taken at three different times. So $F_{11}(3)$ refers to the probability of observing the value 3 or less for a randomly sampled participant from the first group (active treatment) at time 1. For the first group, $\bar{F}_{1.}(3)$ is the average of the probabilities over the three times of observing the value 3 or less. The hypothesis of no main effects for Factor A is that for any x we might pick, $\bar{F}_{1.}(x) = \bar{F}_{2.}(x)$.

For the shoulder pain data in Table 3 the output from `bwrank` is

```
$test.A:
[1] 12.87017
$sig.A:
[1] 0.001043705
$test.B:
[1] 0.4604075
$sig.B:
[1] 0.5759393
$test.AB:
[1] 8.621151
$sig.AB:
[1] 0.0007548441
$avg.ranks:
         [,1]     [,2]     [,3]
[1,] 58.29545 48.40909 39.45455
[2,] 66.70455 82.36364 83.04545

$rel.effects:
          [,1]      [,2]      [,3]
[1,] 0.4698817 0.3895048 0.3167036
[2,] 0.5382483 0.6655580 0.6711013
```

Table 3: Shoulder Pain Data(1=low, 5=high)

Active Treatment			No Active Treatment		
Time 1	Time 2	Time 3	Time 1	Time 2	Time 3
1	1	1	5	2	3
3	2	1	1	5	3
3	2	2	4	4	4
1	1	1	4	4	4
1	1	1	2	3	4
1	2	1	3	4	3
3	2	1	3	3	4
2	2	1	1	1	1
1	1	1	1	1	1
3	1	1	1	5	5
1	1	1	1	3	2
2	1	1	2	2	3
1	2	2	2	2	1
3	1	1	1	1	1
2	1	1	1	1	1
1	1	1	5	5	5
1	1	1	3	3	3
2	1	1	5	4	4
4	4	2	1	3	3
4	4	4			
1	1	1			
1	1	1			

So treatment methods are significantly different and there is a significant interaction, but no significant difference is found over time.

18.6 Within-by-Within

The R function

```
wwtrim(J,K,x,grp=c(1:p),p=J*K,tr=.2,bop=F)
```

compares trimmed means for a within-by-within design.

18.7 Three-Way Designs

Three-way designs can be analyzed based on an extension of the method in Johansen (1980). The following R functions can be used to perform the analysis.

For a three-way, J-by-K-by-L design where the third factor involves dependent groups, and the other two factors deal with independent groups, use the R function

```
bbwtrim(J,K,L,x,grp=c(1:p),tr=.2)
```

For a between-by-within-by-within design use

```
bwwtrim(J,K,L,x,grp=c(1:p),tr=.2).
```

And for a within-by-within-by-within design use

```
wwwtrim(J,K,L,x,grp=c(1:p),tr=.2).
```

18.8 Data Management: R Functions bw2list and bbw2list

The R function

```
bw2list(x, grp.col, lev.col).
```

deals with data that are stored in a matrix or a data frame with one column indicating the levels of the between factor and one or more columns containing data corresponding to different within group levels.

EXAMPLE

Imagine that column 3 of the data frame `m` indicates which of two medications were used to treat some malady. Further imagine that columns 6, 7 and 8 contain measures taken at times 1, 2 and 3 for husbands, and columns 10, 11 and 12 contain the results for wives. Then the command

```
z=bw2list(m, 3, c(6,7,8,10,11,12)).
```

will store the data in `z` in list mode with time the third factor. The command

```
bwwtrim(2,2,3,z)
```

will compare the groups based on a 20% trimmed mean.

The R function

```
z=bbw2list(x, grp.col, lev.col)
```

is like the function `bw2list`, only designed to handle a between-by-between-by-within design. Now the argument `grp.col` should have two values, which indicate the columns of `x` containing data on the levels of the two between factors, and `lev.col` indicates the columns containing the dependent outcomes.

EXAMPLE

The last example is repeated, only now it is assumed that independent samples of males and females are used. Moreover, column 4 is assumed to indicate whether a participant is male or female and measures, taken at three different times, are contained in columns 6, 7 and 8. Then the command

```
z=bw2list(m, c(3,4), c(6,7,8))
```

stores the data in z, in list mode, and the command

```
bbwtrim(2,2,3,z)
```

will compare the groups based on a 20% trimmed mean.

19 Multiple Comparisons

J independent groups.

Dunnett's T3 method is readily extended to trimmed means. Briefly, for each pair of groups, apply Yuen's method and control FWE using the critical value employed by T3, with the degrees of freedom adjusted based on the amount of trimming.

The R function

```
lincon(x,con=0,tr=0.2,alpha=0.05) ,
```

compares all pairs of groups using the extension of the T3 method to trimmed means. The argument `x` is assumed to be a matrix with J columns or to have list mode. The argument `tr` controls the amount of trimming and defaults to 20%.

19.1 Percentile Bootstrap Methods for Comparing Trimmed Means, Medians and M-estimators

The only known method for comparing medians that remains reasonably accurate when tied values occur is based on a percentile bootstrap method. Also, when there are no tied values, there seems to be no compelling reason to use a non-bootstrap method.

When comparing M-estimators or MOM, the same method used to compare medians appears to be best for general use when the sample sizes are large. That is, for each group, an M-estimator is computed rather than a median, otherwise the computations are the same. But with small sample sizes, roughly meaning that all sample sizes are less than or equal to 80, the method is too conservative in terms of Type I errors: the actual level can drop well below the nominal level. An adjustment is available that appears to be reasonably effective.

The R function

```
medpb(x,alpha=.05,nboot=NA,grp=NA,est=median,con=0,bhop=F) ,
```

performs all pairwise comparisons of J independent groups using the method for medians described in the previous section. The argument `nboot` determines how many bootstrap samples will be used. By default, `nboot=NA`, meaning that the function will choose how many bootstrap samples will be used depending how many groups are being compared.

The R function

```
tmcppb(x,alpha=.05,nboot=NA,grp=NA,est=tmean,con=0,bhop=F,...) ,
```

is exactly the same as `mdepb`, only it defaults to using a 20% trimmed mean.

The R function

```
mcppb20(x)
```

compares all pairs of groups via 20% trimmed means using the method in Wilcox (2001d). It is limited to $\alpha = 0.05$ and can be used with 20% trimmed means only.

The R function

```
pbmcp(x,alpha=.05,nboot=NA,grp=NA,est=onestep,con=0,bhop=F)
```

can be used to compare measures of location indicated by the argument `est`, which defaults to a one-step M-estimator. If the largest sample size is less than 80 and `bhop=F`, method SR is used to control FWE. But if the largest sample size exceeds 80, Hochberg's method is used, still assuming that `bhop=F`. If `bhop=T`, the Benjamini - Hochberg method is used. With small sample sizes, this function appears to be best when comparing groups based on an M-estimator or MOM. If the goal is to compare groups using MOM, set the argument `est=mom`.

The R function

```
linconb(x,con=0,tr=0.2,alpha = 0.05) ,
```

compares all pairs of groups using the bootstrap-t method just described.

19.2 Two-Way Designs

The R functions `lincon` and `linconb` introduced in Sections 13.1.9 and 13.1.14, respectively, as well as `kbcon`, contain an argument `con` that can be used to specify the linear contrast coefficients relevant to the hypotheses of interest. To help analyze a J-by-K design, the R function

```
con2way(J,K)
```

is supplied, which generates the linear contrast coefficients needed to compare all main effects and interactions. The contrast coefficients for main effects associated with Factor A and Factor B are returned in `conA` and `conB`, respectively, which are matrices. The columns of the matrices correspond to the various hypotheses to be tested. The rows contain the contrast coefficients corresponding to the groups. Contrast coefficients for all interactions are returned in `conAB`. When using `con2way` in conjunction with the R function `lincon`, it is assumed that the groups are arranged as in Section 11.1.2. The following example provides an illustration.

EXAMPLE

Consider again a 3-by-2 design where the means are arranged as follows:

		Factor B	
		1	2
	1	μ_1	μ_2
Factor A	2	μ_3	μ_4
	3	μ_5	μ_6

The R command

```
con2way(3,2)
```

returns

```
$conA
     [,1] [,2] [,3]
[1,]    1    1    0
```

```
[2,]    1    1    0
[3,]   -1    0    1
[4,]   -1    0    1
[5,]    0   -1   -1
[6,]    0   -1   -1

$conB
     [,1]
[1,]    1
[2,]   -1
[3,]    1
[4,]   -1
[5,]    1
[6,]   -1

$conAB
     [,1] [,2] [,3]
[1,]    1    1    0
[2,]   -1   -1    0
[3,]   -1    0    1
[4,]    1    0   -1
[5,]    0   -1   -1
[6,]    0    1    1
```

Each matrix has 6 rows because there are a total of 6 groups. The first column of the matrix `conA` contains 1, 1, -1, -1, 0, 0, which are the contrast coefficients for comparing the first two levels of Factor A, as explained in the second example at the beginning of Section 13.2. The second column contains the contrast coefficients for comparing levels 1 and 3. And the final column contains the coefficients for comparing levels 2 and 3. In a similar manner, `conB` contains the contrast coefficients for the two levels of Factor B and `conAB` contains the contrast coefficients for all interactions.

EXAMPLE

For the plasma retinol data, stored in R, consider two factors: sex and smoking status. The goal is to perform all multiple comparisons based on 20% trimmed means. First use the R command

```
z=fac2list(plasma[,10],plasma[,2:3])
```

to convert the data into list mode and store it in the R variable `z`. We have a 2-by-3 design, and so the command

```
w=con2way(2,3)
```

creates the contrast coefficients and stores them in `w`. Then the R command

`lincon`(z,con=w$conA)

will compare the two levels of Factor A (male and female). The command

`lincon`(z,con=w$conB)

will perform all pairwise comparisons based on the three levels of Factor B (smoking status) and

`lincon`(z,con=w$conAB)

will perform all interactions associated with any two rows and columns.

The R function

linconb(x, con = 0, tr = 0.2, alpha = 0.05, nboot = 599)

performs the bootstrap-t method for trimmed means just described. As usual, the argument con can be used to specify the contrast coefficients. If con is not passed to the function, all pairwise comparisons are performed.

When dealing with a two-way ANOVA design, the R function

mcp2a(J,K,x,est=onestep,con=0,alpha=0.05,nboot=NA,grp=NA,...)

performs all pairwise multiple comparisons among the rows and columns, and it tests all linear contrasts relevant to interactions. Method SR is used to control the probability of at least one Type I error if the largest sample size is less than 80. Otherwise, Rom's method is used. By default, `mcp2a` uses the one-step M-estimator of location.

The R function

bbmcppb(J, K, x, tr = 0.2, JK = J * K, alpha = 0.05, grp = c(1:JK), nboot = 500, bhop = F, SEED = T)

performs multiple comparisons for a two-way ANOVA design based on trimmed means in conjunction with a percentile bootstrap method. It uses the linear contrasts for main effects and interactions as previously described. Rom's method is used to control the probability of one or more Type I errors. For $C > 10$ hypotheses, or when the goal is to test at some level other than 0.05 and 0.01, Hochberg's method is used. Setting the argument bhop=T, the Benjamini - Hochberg method is used instead.

19.3 Methods for Dependent Groups

The R function

```
rmmcp(x,con = 0, tr = 0.2, alpha = 0.05, dif=T)
```

performs multiple comparisons among dependent groups using trimmed means and Rom's method for controlling FWE. By default, difference scores are used and all pairwise comparisons are performed. Setting dif=F results in comparing marginal trimmed means. And linear contrasts can be tested via the argument `con`.

The R function

```
rmmcppb(x,y=NA,alpha=.05,con=0,est=onestep,plotit=T,dif=T,
grp=NA,nboot=NA,BA=F,hoch=F,xlab="Group 1",ylab="Group 2",pr=T, SEED=T,...)
```

performs all pairwise comparisons using the percentile bootstrap method just described. The argument `y` defaults to NA meaning that the argument `x` is assumed to be a matrix with J columns or to have list mode. If data are stored in the argument `y`, then it is assumed that two dependent groups are to be compared with the data in first group stored in `x`. To test hypotheses based on the marginal measures of location, set the argument `dif=F`; otherwise difference scores are used. The R function

```
dmedpb(x,y=NA,alpha=.05,con=0,est=median,plotit=T,dif=F,grp=NA,
hoch=F,nboot=NA,xlab="Group 1",ylab="Group 2",pr=T,SEED=T,...)
```

is the same as rmmcppb, only it defaults to comparing medians and it is designed to handle tied values.

The R function

```
bptd(x, tr = 0.2, alpha = 0.05, con = 0, nboot = 599)
```

tests hypotheses about linear contrasts associated with dependent groups using a bootstrap-t method.

19.4 Between-by-Within

The R function

```
bwmcp(J, K, x, tr = 0.2, alpha = 0.05,con=0, nboot=599)
```

performs method BWMCP described in the previous section. By default, it creates all relevant linear contrasts for main effects and interactions and then applies a method that takes into account which pairs of groups are independent. (Other methods are and appropriate T functions are covered in Wilcox, 2010.)

19.5 Three-Way Designs

This section describes three R functions aimed at facilitating the analysis of a three-way design. The first function

```
con3way(J,K,L)
```

generates all of the linear contrast coefficients needed to test hypotheses about main effects and interactions.

EXAMPLE

We illustrate how to use `con3way` when dealing with a between-by-between-by-between design and the goal is to test the hypothesis of no three-way interaction. (Main effects

and two-way interactions are handled in a similar manner.) Consider again the weight-gain illustration in Section 11.1 where there are two factors: amount of protein (high and low) and source of protein (beef and cereal). Here we imagine that there is a third factor, type of medication, having two levels, namely, placebo and experimental. It is convenient to represent the means as follows:

Factor A: placebo

		Source	
		Beef	Cereal
Amount	High	μ_1	μ_2
	Low	μ_3	μ_4

Factor A: experimental

		Source	
		Beef	Cereal
Amount	High	μ_5	μ_6
	Low	μ_7	μ_8

For the first level of Factor A (placebo), interaction refers to

$$(\mu_1 - \mu_2) - (\mu_3 - \mu_4).$$

For the second level of Factor A, interaction refers to

$$(\mu_5 - \mu_6) - (\mu_7 - \mu_8).$$

The hypothesis of no three-way interaction corresponds to

$$(\mu_1 - \mu_2) - (\mu_3 - \mu_4) = (\mu_5 - \mu_6) - (\mu_7 - \mu_8).$$

So in terms of a linear contrast, the hypothesis of no three-way interaction is

$$H_0 : \mu_1 - \mu_2 - \mu_3 + \mu_4 - \mu_5 + \mu_6 + \mu_7 - \mu_8 = 0. \tag{29}$$

What is needed is a convenient way of generating the linear contrast coefficients, and this is accomplished with the R function `con3way`. For example, the R command

```
m=con3way(2,2,2)
```

stores in `m$conABC` the linear contrast coefficients for testing the hypothesis of no three-way interaction. When the factors have more than two levels, con3way generates all sets of linear contrasts relevant to all three-way interactions and again returns them in `con3way$conABC`. In a similar manner, all linear contrasts relevant to main effects for factor A are stored in `m$conA`

The next step is to test the hypothesis of no three-way interaction with an R function function described in Section 13.1 or 13.2 that is designed for independent groups. For example, if the data are stored in the R variable `dat`, the command

```
lincon(dat,con=m$conABC)
```

would test the hypothesis of no three-way interaction based on a 20% trimmed mean.

For convenience, the R function

```
mcp3atm(J, K, L, x, tr = 0.2, con = 0, alpha = 0.05, grp = NA, pr = T)
```

is provided that performs all of the multiple comparisons based on the linear contrast coefficients generated by `con3way` when dealing with a between-by-between-by-between design.

A within-by-within-by-within design is handled in a manner similar to a between-by-between-by-between design. The only difference is that now we use a method for linear contrasts that is designed for dependent groups, which were described in Section 13.4. Assuming that all linear contrasts generated by `con3way` are of interest, the R function

```
rm3mcp(J, K, L, x, tr = 0.2, alpha = 0.05, dif = T, op = F, grp = NA)
```

can be used.

As for situations where there are both between and within factors, use a bootstrap method via the R functions described in Section 13.6.4.

EXAMPLE

We repeat the last example assuming all groups are dependent. Again use the R function `con3way` to generate the linear contrast coefficients. If the contrast coefficients are stored in `m$conABC`, the R command

```
rmmcp(dat,con=m$conABC)
```

tests the hypothesis of no three-way interaction.

Bootstrap Methods are available.

20 SOME MULTIVARIATE METHODS

20.1 Detecting Outliers

Usual Mahalanobis distance: Suffers from masking. One of the worst possible methods.

Need a method that uses a robust measure of location and scatter. Numerous methods have been proposed. Two that perform relatively well are a projection method and what is called the minimum generalized variance method.

The R function

```
outpro(m,gval = NA, center = NA, plotit = T, op = T, MM = F, cop = 3, xlab =
                        ‘‘VAR 1’’, ylab = ‘‘VAR 2’’)
```

checks for outliers using the projection method described in the previous section. By default, the method uses the boxplot rule on each projection. To use the MAD-median rule, set the argument `MM=T`. The argument cop determines how the center of the data cloud is computed. By default, the marginal medians are used. The options cop=2, 3, 4, 5, 6, 7 correspond to MCD, marginal medians, MVE, TBS, RMBA and a measure of location called the *spatial*

(L1) *median*, respectively. These choices can make a difference in the results, but from a practical point of view it is unclear which method is best for routine use. An alternative choice for the center of the data can be specified with the argument center. (The argument op has to do with how the data are plotted; see Wilcox, 2005, for more information.) The argument `gval` corresponds to c in the equation

$$\frac{|X - M|}{\text{MADN}} > c.$$

The function chooses a value for `gval` if one is not specified.

Filzmoser, Maronna and Werner (2008) noted that under normality, if the number of variables is large, the proportion of points declared outliers by the better-known outlier detection methods can be relatively high. This concern applies to all the methods covered here with the projection method seemingly best at avoiding this problem. But with more than nine variables ($p > 9$), it breaks down as well. Currently, the best way of dealing with this problem is to use the projection method, but rather than use the boxplot rule or the MAD-median rule (as described in Chapter 2 of Wilcox, 2011), determine an adjustment so that the proportion of points declared outliers is small, say 5%, when sampling from a normal distribution.

The R function

```
outproad(m,gval = NA, center = NA, plotit = T, op = T, MM = F, cop = 3, xlab
                = ‘‘VAR 1’’, ylab = ‘‘VAR 2’’,rate = 0.05)
```

is the same as `outpro`, only the decision rule is adjusted so that the expected proportion of points declared outliers, under normality, is approximately equal to the value indicated by the argument `rate`, which defaults to 0.05. Use this function if the number of variables is greater than 9.

The R functions

```
outproMC(m, gval,center = NA, plotit = T, op = T, MM = F, cop = 3, xlab =
                        ‘‘VAR 1’’, ylab = ‘‘VAR 2’’)
```

and

```
outproadMC(m, center = NA, plotit = T, op = T, MM = F, cop = 3, xlab = ‘‘VAR
                    1’’, ylab = ‘‘VAR 2’’,rate = 0.05)
```

are the same as outpro and outproad, respectively, but they take advantage of a multi-core processor if one is available. (They require the R package multicore.)

When working with three dimensional data, the R function

```
out3d(x,outfun=outpro,xlab="Var 1",ylab="Var 2",zlab="Var 3",
                    reg.plane=F,regfun=tsreg)
```

will plot the data and indicate outliers with an *. If the argument `reg.plane=T`, this function also plots the regression plane based on the regression estimator indicated by the argument `regfun`, assuming that `regfun` returns the intercept and slope estimates in `regfun$coef`. When plotting the regression surface, the function assumes the outcome variable is stored in the third column of the argument `x`. And the two predictors are stored in the first two columns. To plot the least squares regression surface, use `regfun=lsift`. By default, the function uses the Theil - Sen regression estimator described in Section 15.1.1. (The function out3d requires the R package scatterplot3d, which can be installed as described in Chapter 1 of Wilcox, 2011.)

EXAMPLE

Predictors of reading ability study. Figure 18 shows the points flagged as outliers plus the regression plane based on the Theil-Sen estimator. Figure 19 is the same, only the least squared regression line is shown. As is evident, the two regression planes differ substantially.

20.2 Robust MANOVA Based on Trimmed Means

Johansen (1980) derived an alternative to the classic MANOVA method that allows the covariance matrices among the J groups to differ. Johansen's method can be generalized to trimmed means and has been found to compare well to other methods that have been derived (Wilcox, 1995). Hypotheses based on linear contrasts can be tested using a percentile bootstrap method.

The R function

```
MULtr.anova(x, J = NULL, p = NULL, tr = 0.2,alpha=.05)
```

performs the robust MANOVA method based on trimmed means. The argument J defaults

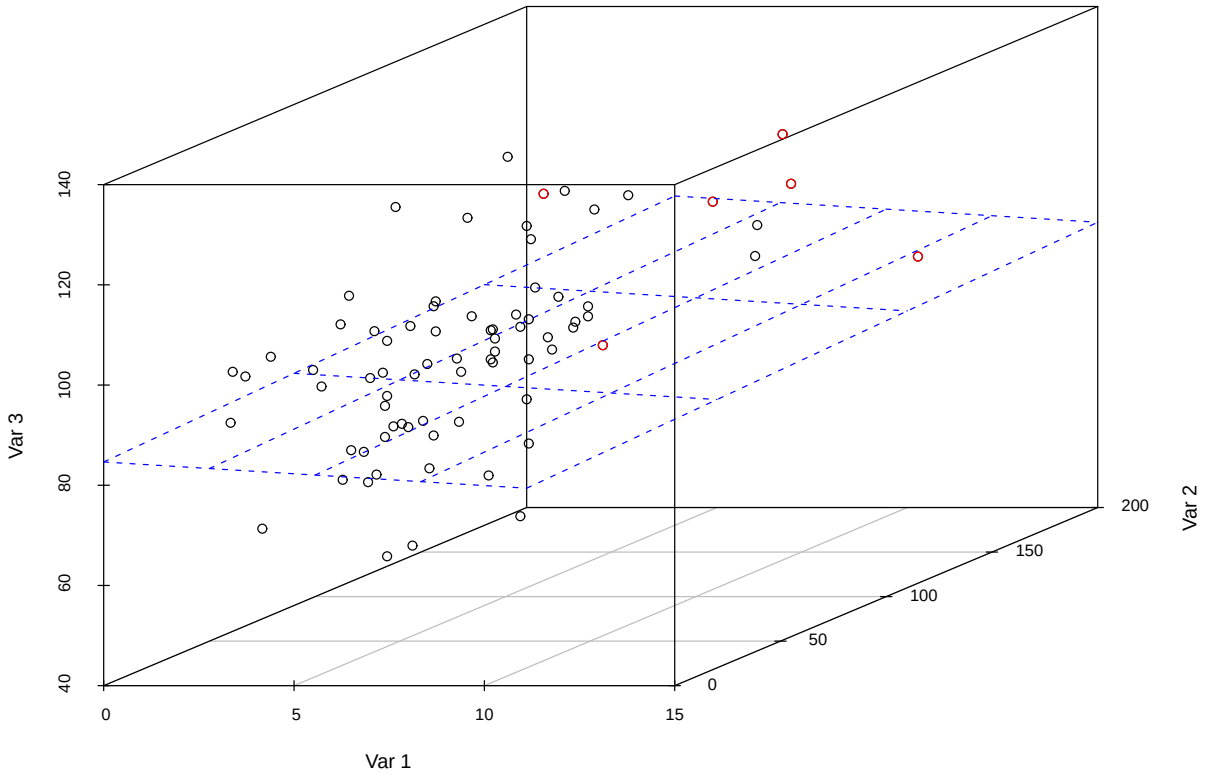

Figure 18: A scatterplot of reading data based on the R function out3d.

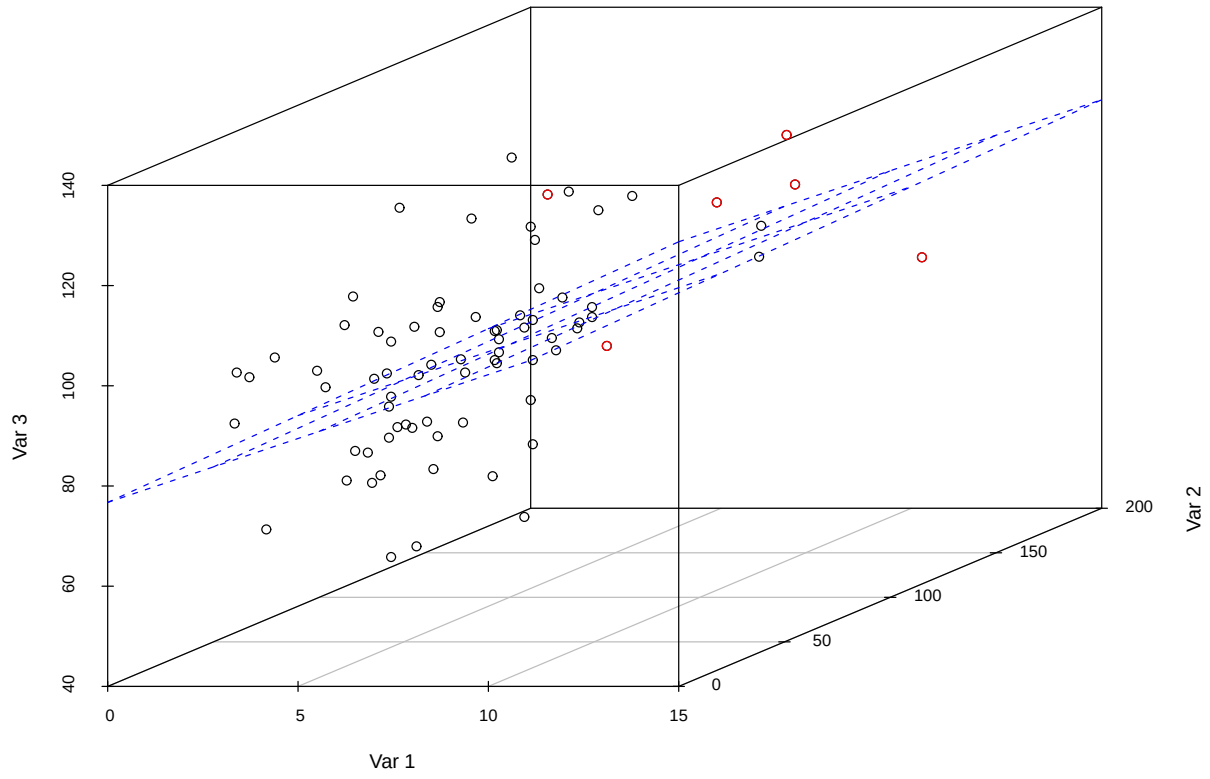

Figure 19: A scatterplot of reading data based on the R function out3d, only the least squares regression line is shown.

to NULL, meaning that x is assumed to have list mode with length J, where x[[j]] contains a matrix with n_j rows and p columns, $j = 1, \ldots, J$.

The R function

MULAOVp(x, J = NULL, p = NULL, tr = 0.2)

also performs the robust MANOVA method based on trimmed means, only it returns a p-value.

20.3 R Functions linconMpb, linconSpb, fac2Mlist and fac2BBMlist

The R function

linconMpb(x, alpha = 0.05, nboot = 1000, grp = NA, est = tmean, con = 0, bhop = F, SEED = T, PDIS = F, J = NULL, p = NULL, ...)

tests hypotheses, based on linear contrasts involving multivariate data, assuming that for each group some marginal measure of location is used. By default all pairwise comparisons are performed based on the marginal 20% trimmed means, but M-estimators, for example, could be used by setting the argument est=onestep. The probability of at least one type I error is set via the argument alpha and is controlled using Rom's method. As usual, contrast coefficients can be specified via the argument con. The argument x is assumed to have list mode, x[[1]] contains a matrix of p variables for first group, x[[2]] contains a matrix of data for the second group, and so on. If x does not have list mode, but rather is a matrix or data frame with the first p columns corresponding group 1, the next p columns corresponding to group 2, and so forth, then specify how many groups there are via the argument J, or how specify how many variables there via the argument p.

For each group, it might be desired to use a multivariate measure of location that takes into account the overall structure of the data. That is, use something like the MCD estimator or the OP estimator. This can be done via the R function

linconSpb(x, alpha = 0.05, nboot = 1000, grp = NA, est = smean, con = 0, bhop = F, SEED = T, PDIS = F, ...)

By default, the OP estimator of location is used, but this might result in relatively high execution time.

Data Management

The following two R functions might help with data management. The R function

fac2Mlist(x,grp.col,lev.col,pr=T)

sorts p-variate data stored in the matrix (or data frame) x into groups based on the values stored in the column of x indicated by the argument grp.col. The results are stored in list mode in a manner that can be used by linconMpb and linconSpb. For example, the command

z=fac2Mlist(plasma,2,c(7:8))

will create groups based on the data in column 2. The result is that z[[1]] will contain the data for the first group stored as a matrix. The first column of this matrix corresponds to data stored in column 7 of the R variable plasma and the second column corresponds to data stored in column 8. Similarly, z[[2]] will contain the data for group 2, and so on. So the command

linconSpb(z)

would perform all pairwise comparisons.

The R function

fac2BBMlist(x,grp.col,lev.col,pr=T)

is like the function fac2Mlist, only it is designed to handle a between-by-between design. Now the argument grp.col is assumed to contain two values indicating the columns of x that contain the levels of the two factors. The multivariate data are stored in the columns indicated by the argument lev.col. For a J-by-K design, the result is an R variable having list mode with length JK.

EXAMPLE. The command

z=fac2BBMlist(plasma,c(2,3),c(7,8))

will create groups based on the values in columns 2 and 3 of the R variable plasma. In this particular case, there are two levels for the first factor (meaning that column 2 of plasma has two unique values only) and three for the second. The result will be that z[[1]], ..., z[[6]] will each contain a matrix having two columns stemming from the bivariate data in columns 7 and 8 of plasma. Then the commands

con=con2way(2,3)

linconMpb(z,con=con$conAB)

would test all hypotheses relevant to interactions.

20.4 Multiple Comparisons

Currently, a good method appears to be one based on a generalization of the Yanagihara and Yuan (2005) MANOVA method. Use the R function

YYmcp(x, alpha = 0.05, grp = NA, tr = 0.2, bhop = F, J = NULL, p = NULL, ...)

20.5 Principal Components

Numerous papers have suggested robust approaches to principal components. Here, R functions for three of these methods are provided. The first two are similar to the classic approach in the sense that the goal is to maximize some measure of variation associated with the principal component scores. That is, it is the variation of the marginal distributions that is of concern. This is in contrast to the third approach that maximizes a robust generalized variance, meaning a measure of variation that takes into account the overall structure of the data.

20.6 R Functions outpca, robpca, robpcaS, Ppca, Ppca.summary

The R function

outpca(x,cor=F,SCORES=F,ADJ=F,scree=T, xlab="Principal Component", ylab="Proportion of Variance")

eliminates outliers via the projection method and applies the classic principal component analysis to the remaining data. Following the convention used by R, the covariance matrix is used by default. To use the correlation matrix, set the argument `cor=T`. Setting `SCORES=T`, the principal component scores are returned. If the argument `ADJ=T`, the R function `outproad` is used to check for outliers rather than the R function `outpro`, which is recommended if the number of variables is greater than 9. By default, the argument scree=T, meaning that a scree plot will be created. Another rule that is sometimes used is to retain those components for which the proportion of variance is greater than 0.1. When the proportion is less than 0.1, it has been suggested that the corresponding principal component rarely has much interpretive value.

The function

```
robpcaS(x, SCORES=F)
```

provides a summary of the results based on the method derived by Hubert et al. (2005), including a scree plot based on a robust measure of variation. A more detailed analysis is performed by the function

```
robpca(x, scree=T, xlab = "Principal Component", ylab = "Proportion of
                                Variance"),
```

which returns the eigenvalues and other results discussed by Hubert et al. (2005), but these details are not discussed here.

The R function

```
Ppca(x, p = ncol(x) - 1, locfun = L1medcen, loc.val = NULL, SCORES = F,
  gvar.fun = cov.mba, pr = T, SEED = T, gcov = rmba, SCALE = T, ...)
```

applies the method aimed at maximizing a robust generalized variance. This particular function requires the number of principal components to be specified via the argument `p`, which defaults to $p - 1$. The argument `SCALE=T` means that the marginal distributions will be standardized based on the measure of location and scale corresponding to the argument `gvoc`, which defaults to the median ball algorithm.

The R function

```
Ppca.summary(x, MC=F, SCALE=T)
```

is designed to deal with the issue of how many components should be used. It calls `Ppca` using all possible choices for the number of components, computes the resulting generalized standard deviations, and reports their relative size. If access to a multi-core processor is available, setting the argument `MC=T` will reduce execution time. Illustrations in the next section deal with the issue of how many components to use based on the output from the R function Ppca.summary.

The output from the function `Ppca.summary` differs in crucial ways from the other functions described here. To illustrate it, multivariate normal data were generated with all correlations equal to 0.0. The output from `Ppca.summary` is

```
                   [,1]     [,2]      [,3]      [,4]
Num. of Comp. 1.0000000 2.000000 3.0000000 4.0000000
Gen.Stand.Dev 1.1735029 1.210405 1.0293564 1.0110513
Relative Size 0.9695129 1.000000 0.8504234 0.8353002
```

The second line indicates the (robust) generalized standard deviation given the number of components indicated by the first line. So when using two components, the generalized standard deviation is 1.210405. Note that the generalized standard deviations are not in descending order. Using two components results in the largest generalized standard deviation. But observe that all four generalized standard deviations are approximately equal, which is what we would expect for the situation at hand. The third line of the output is obtained by dividing each value in the second line by the maximum generalized standard deviation. Here, reducing the number of components from 4 to 2 does not increase the generalized standard deviation by very much, suggesting that 4 or maybe 3 components should be used. Also observe that there is no proportion of variance used here, in contrast to classic PCA. In classic PCA, an issue is how many components must be included to capture a reasonably large proportion of the variance. When using the robust generalized variance, it seems more appropriate to first look at the relative size of the generalized standard deviations using all of the components. If the relative size is small, reduce the number of components. In the example, the relative size using all four components is 0.835 suggesting that perhaps all four components should be used.

Now consider data that were generated from a multivariate normal distribution where all of the correlations are 0.9. Now the output from regpca is

```
Importance of components:
```

```
                          PC1    PC2    PC3    PC4
Standard deviation     1.869 0.3444 0.3044 0.2915
Proportion of Variance 0.922 0.0313 0.0244 0.0224
Cumulative Proportion  0.922 0.9531 0.9776 1.0000
```

Note that the first principal component has a much larger standard deviation than the other three principal components. The proportion of variance accounted for by PC1 is 0.922, suggesting that it is sufficient to use the first principal component only to capture the variability in the data.

The output from `Ppca.summary` is

```
                  [,1]      [,2]      [,3]       [,4]
Num. of Comp. 1.000000 2.0000000 3.0000000 4.00000000
Gen.Stand.Dev 2.017774 0.6632588 0.2167982 0.05615346
Relative Size 1.000000 0.3287082 0.1074442 0.02782942
```

As indicated, a single component results in a relatively large generalized standard deviation suggesting that a single component suffices. The relative sizes corresponding to 3 and 4 components are fairly small suggesting that using 3 or 4 components be ruled out. Even with 2 components the relative size is fairly small.

21 REGRESSION AND CORRELATION

Review the notion of homoscedasticity and heteroscedasticity.

WHEN USING LEAST SQUARES REGRESSION, SUGGEST DEALING WITH HETEROSCEDASTICITY USING THE HC4 ESTIMATOR.

IN PRACTICAL TERMS, USE ONE OF THE FOLLOWING R FUNCTIONS:

The R function

```
olshc4(x,y,alpha=.05,xout=F,outfun=out)
```

computes $1 - \alpha$ confidence intervals and p-values for each of the individual parameters. By default, 0.95 confidence intervals are returned. Setting the argument alpha equal to 0.1, for example, will result in 0.9 confidence intervals.

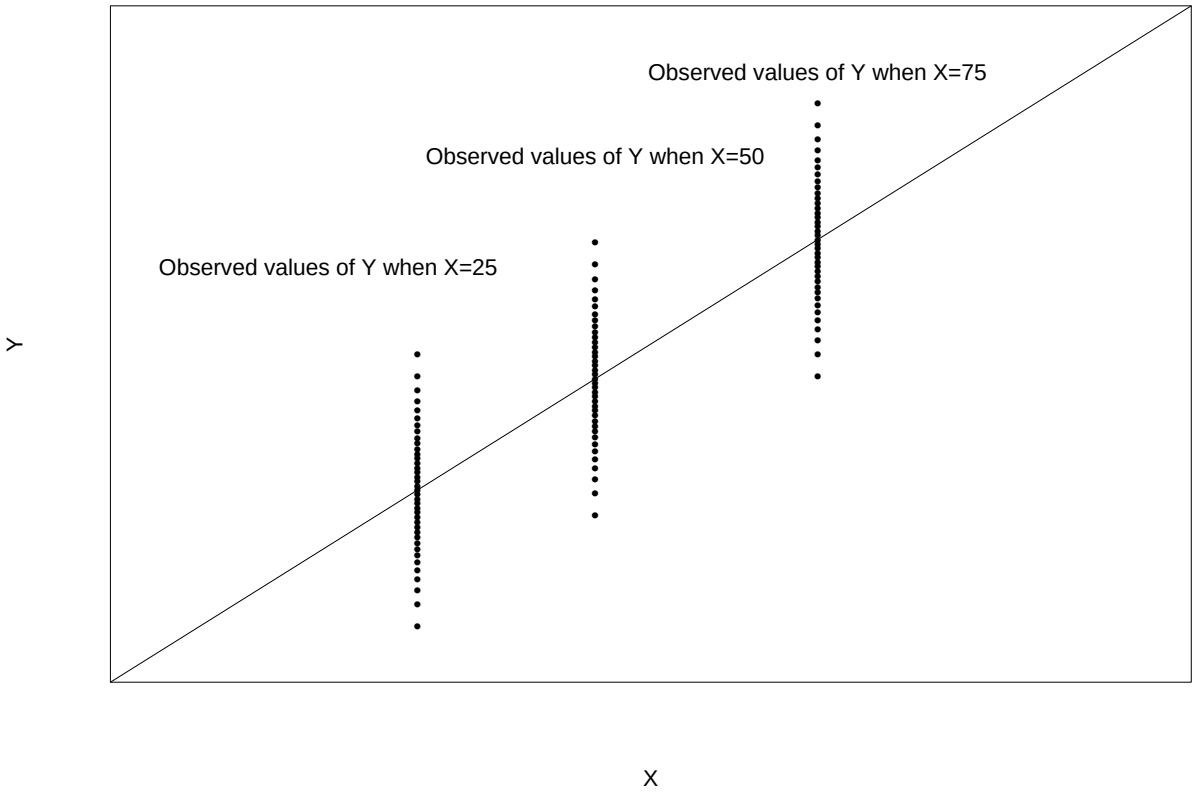

Figure 20: An example of homoscedasticity. The conditional variance of Y, given X, does not change with X.

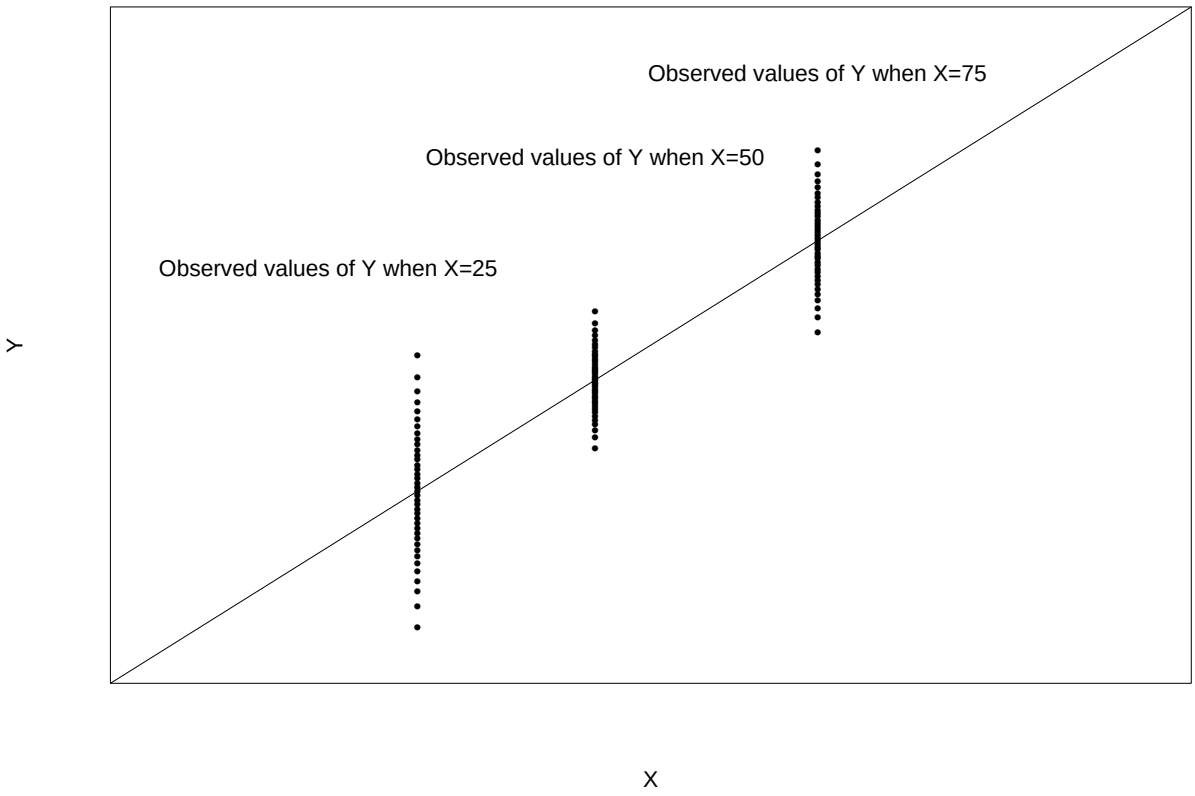

Figure 21: An example of heteroscedasticity. The conditional variance of Y, given X, changes with X.

The function

```
hc4test(x,y,xout=F,outfun=out)
```

tests the hypothesis that all of the slopes are equal to zero. Note that both functions include the option of removing leverage points via the arguments `xout` and `outfun`.

22 Problems with Least Squares

In regression, any outlier among the X values is called a *leverage point*. There are two kinds: good and bad.

Bad Leverage Points

Roughly, bad leverage points are outliers that can result in a misleading summary of how the bulk of the points are associated. That is, bad leverage points are not consistent with the association among most of the data.

EXAMPLE

Consider the lake data.

There are two predictors. When both predictors are included in the model, the p-value, when testing H_0: $\beta_1 = 0$, is 0.56. Ignoring the second predictor, the p-value is 0.72 illustrating that p-values are a function in part of which predictors are entered into the model. If we perform the analysis again with both predictors, but with outliers among the predictors removed, now the p-value when testing H_0: $\beta_1 = 0$ is 0.0009, illustrating that a few bad leverage points can have a large impact when testing hypotheses.

EXAMPLE

Consider again the reading study (conducted by L. Doi) dealing with predictors of reading ability. One portion of her study used a measure of digit naming speed (RAN1T) to predict

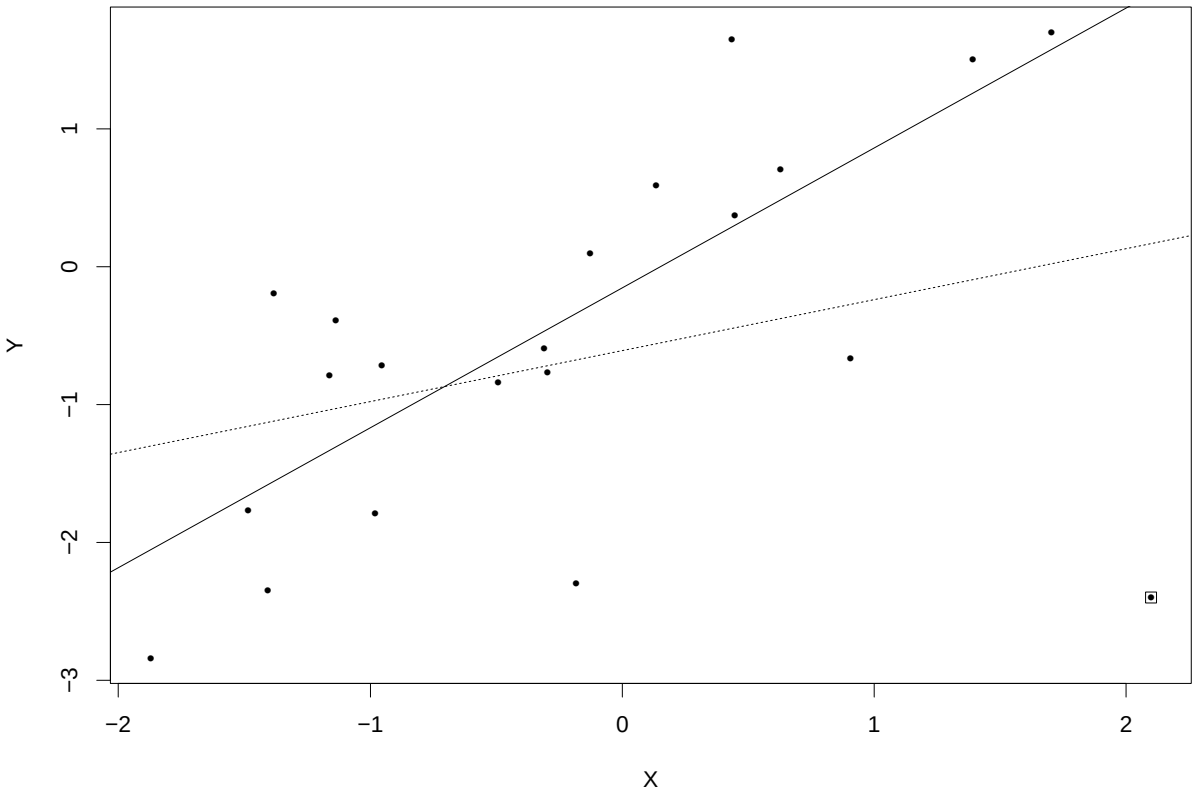

Figure 22: The two points marked by the square in the lower right corner have a substantial impact on the least squares regression line. Ignoring these two points, the least squares regression is given by the solid line. Including them, the least squares regression line is given by the dotted line. Moreover, none of the X or Y values are declared outliers using the boxplot rule of MAD-median rule.

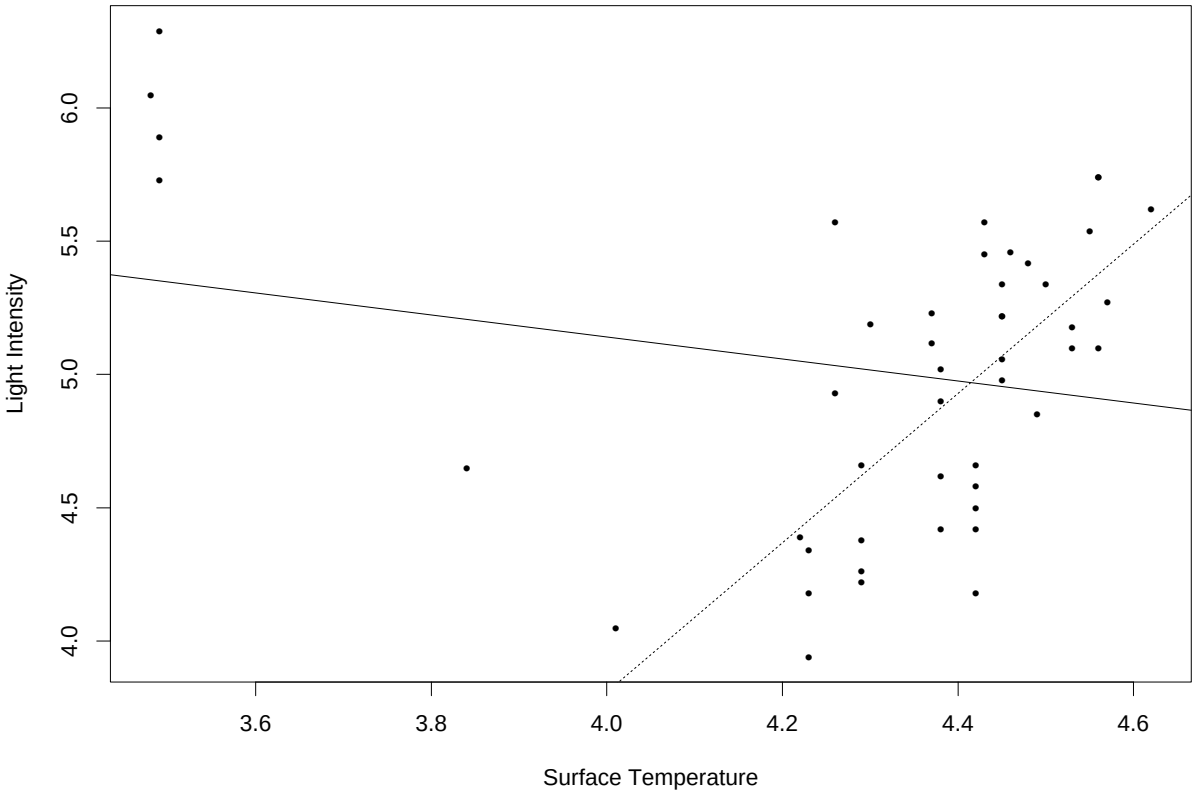

Figure 23: The solid line is the least squares regression line using all of the star data. Ignoring outliers among the X values, the least squares regression line is now given by the dotted line.

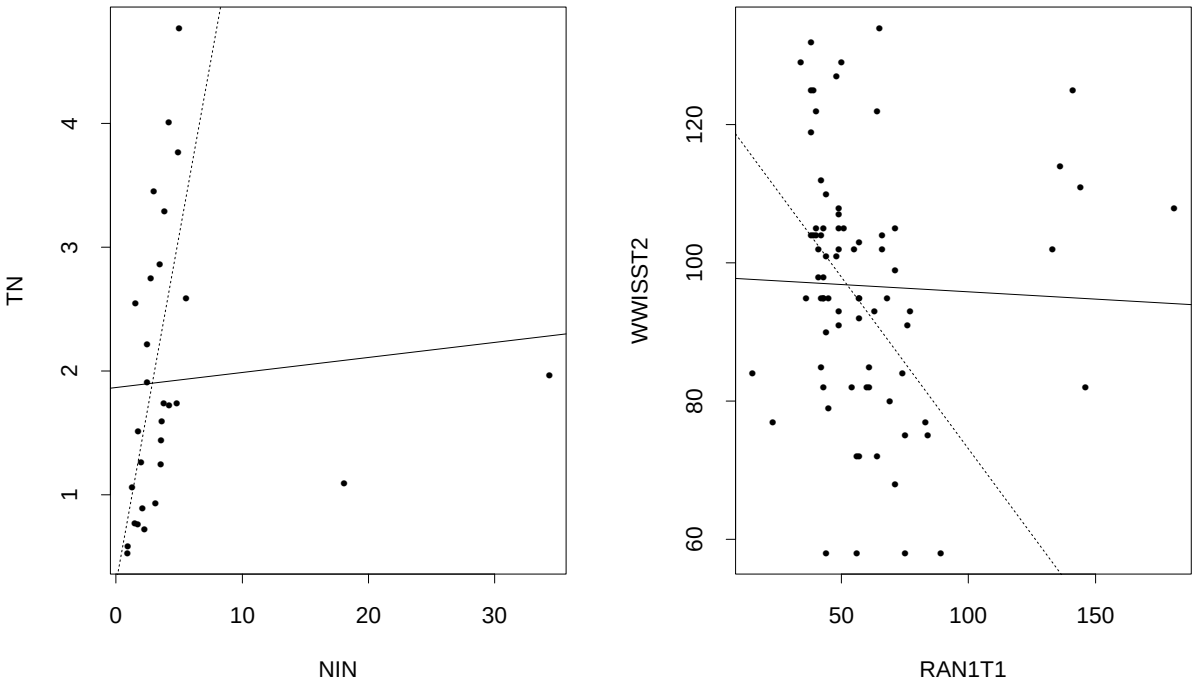

Figure 24: The left panel shows a scatterplot of the lake data. The bad leverage points, located in the lower right portion of the scatterplot, have a tremendous influence on the least squares estimate of the slope resulting in missing any indication of an association when using least squares regression. The right panel shows a scatterplot of the reading data. Now the bad leverage points mask a negative association among the bulk of the points.

the ability to identify words (WWISST2). Using all of the data, the slope of the least squares regression line is nearly equal to zero, which is the nearly horizontal line in the right panel of Figure 24. When testing H_0: $\beta_1 = 0$, the p-value is .76. But the scatterplot clearly reveals that the six largest X values are outliers. When these outliers are removed, the dotted line shows the resulting least squares regression line and now the p-value is 0.002.

Good Leverage Points

Leverage points can distort the association among most of the data, but not all leverage points are bad. A crude description of a *good leverage point* is a point that is reasonably consistent with the regression line associated with most of the data. Figure 25 illustrates the basic idea.

BEWARE OF DISCARDING OUTLIERS AMONG THE Y VALUES. TESTING HYPOTHESES AS IF THE OUTLIERS NEVER EXISTED CAN INVALIDATE THE RESULTS. FOR EXAMPLE, APPLYING STANDARD METHODS ASSOCIATED WITH THE USUAL LEAST SQUARES ESTIMATOR VIOLATES BASIC PRINCIPLES. THIS CAN RESULT IN USING AN INVALID ESTIMATE OF THE STANDARD ERROR. THERE ARE, HOWEVER, METHODS THAT ALLOWS THE REMOVAL OF OUTLIERS AMONG THE Y VALUES.

DO NOT ASSUME THE REGRESSION LINE IS STRAIGHT

EXAMPLE

In a study by C. Chen and F. Manis, one general goal was to investigate the extent Asian immigrants to the United States learn how to pronounce English words and sounds. Figure 26 shows a scatterplot of age versus an individual's score on a standardized test of pronunciation. As indicated, there appears to be curvature as well as heteroscedasticity. (The middle regression line in Figure 26 shows the estimated median score on the pronunciation test, given an individual's age, and was created with the function qsmcobs. (See chapter 15 of Wilcox, 2011, for more details.) The lower and upper regression lines show the predicted

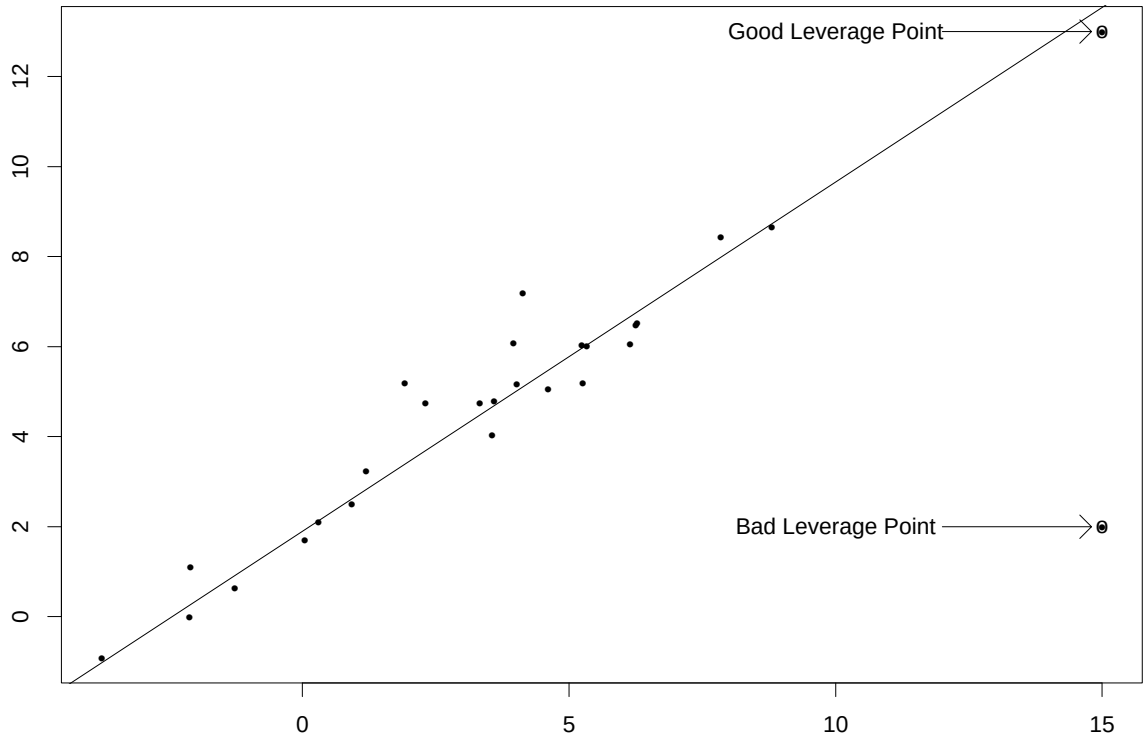

Figure 25: Shown are both good and bad leverage points. Good leverage points do not mask the true association among the bulk of the points and they have the practical advantage of resulting in shorter confidence intervals.

0.25 and 0.75 quantiles, respectively.)

PEARSON'S CORRELATION

The following features of data influence the magnitude of Pearson's correlation:

- The slope of the line around which points are clustered.
- The magnitude of the residuals.
- Outliers
- Restricting the range of the X values, which can cause r to go up or down.
- Curvature.

The R function

```
pcorhc4(x,y)
```

has been provided to compute a confidence interval for ρ using the HC4 method, meaning that it deals with heteroscedasticity, and a p-value is returned as well.

23 Robust Regression and Measures of Association

23.1 The Theil-Sen Estimator

Momentarily consider a single predictor and imagine that we have n pairs of values:

$$(X_1, Y_1), \ldots, (X_n, Y_n).$$

Consider any two pairs of points for which $X_i > X_j$. The slope corresponding to the two points (X_i, Y_i) and (X_j, Y_j) is

$$b_{1ij} = \frac{Y_i - Y_j}{X_i - X_j}. \tag{30}$$

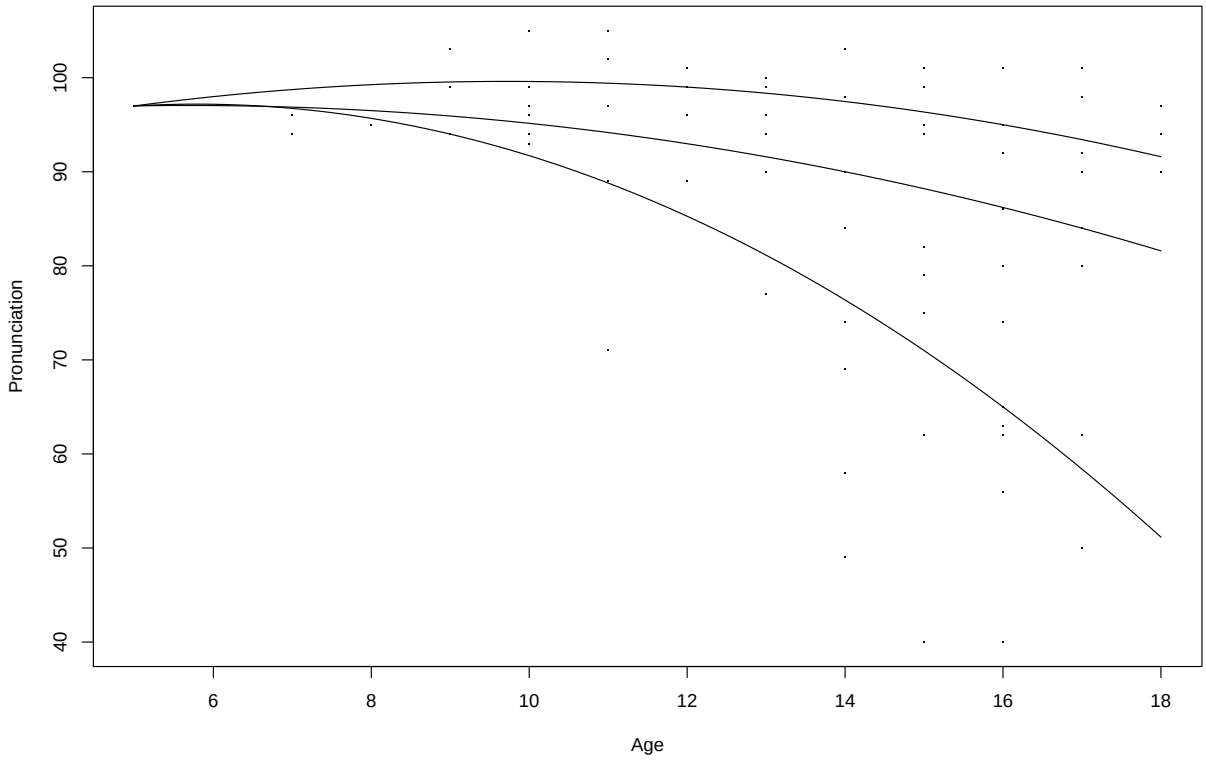

Figure 26: Shown are age and an individual's score on a pronunciation test. The lower, middle and upper regression lines indicate the predicted 0.25 quantile, the median, and the 0.75 quantile, respectively. Note that the regression lines appear to be fairly horizontal among young individuals, but as age increases, scores on the test tend to decrease, and the variation among the scores appears to increase.

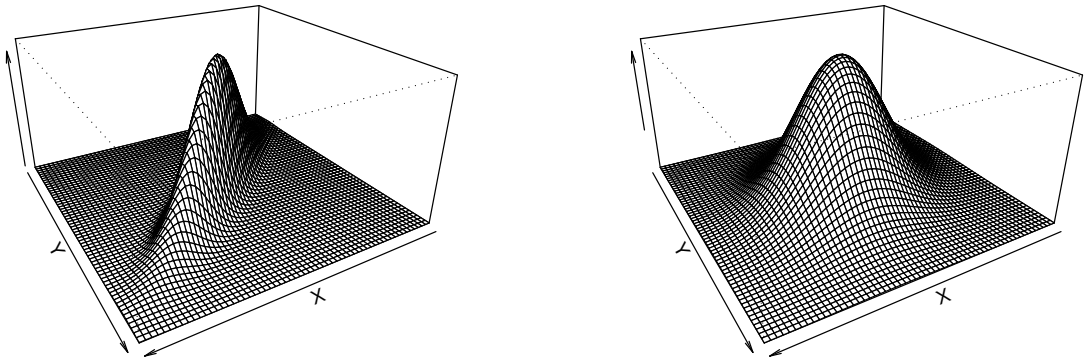

Figure 27: When both X and Y are normal, increasing ρ from .2 to .8 has a noticeable effect on the bivariate distribution of X and Y.

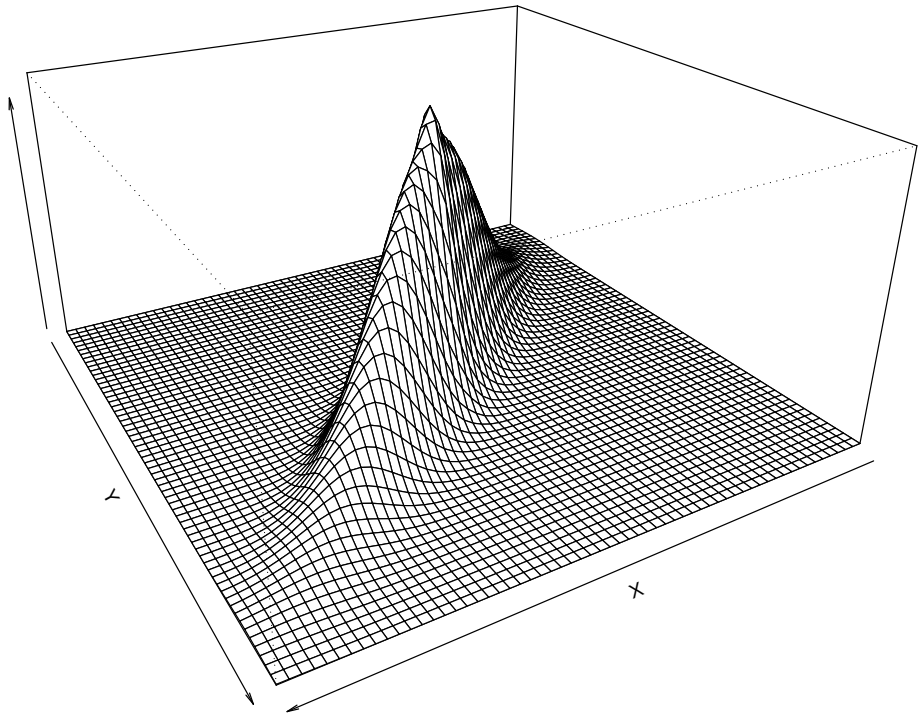

Figure 28: Two bivariate distributions can appear to be very similar yet have substantially different correlations. Shown is a bivariate distribution with $\rho = .2$, but the graph is very similar to the left panel in the previous figure where $\rho = .8$.

The Theil (1950) and Sen (1964) estimate of β_1 is the median of all the slopes represented by b_{1ij}, which is labeled $b_{1\text{ts}}$. The intercept is estimated with

$$b_{0\text{ts}} = M_y - b_{1\text{ts}}M_x,$$

where M_y and M_x are the sample medians corresponding to the Y and X values, respectively.

The R function

```
tsreg(x, y, xout=F, outfun=out, iter=10, varfun=pbvar, corfun=pbcor, ...)
```

computes the Theil - Sen estimator. If the argument `xout=T`, leverage points are removed based on the outlier detection method specified by the argument `outfun`.

23.2 M-Estimators

M-estimators of location can be extended to regression. Many variations have been proposed. Currently, a so-called MM-estimator is popular among some statisticians. The Coakley-Hettmansperger estimator also has excellent theoretical properties. A concern is contamination bias. Also, non-bootstrap methods for making inferences about the MM-estimator are available. But generally, hypothesis testing methods based on M-estimators, that use a non-bootstrap method in conjunction with an estimate of the standard errors, perform poorly when dealing with skewed distributions. Whether this is a practical problem for the MM-estimator has not been investigated.

The R function

chreg(x,y,bend=1.345).

computes the Coakley-Hettmansperger regression estimator. As will all regression functions, the argument x can be a vector or an n by p matrix of predictor values. The argument bend is the value of K used in Huber's Ψ.

The R function

MMreg(x,y)

computes Yohai's MM-estimator, assuming that the R package robustbase has been installed.

EXAMPLE. If the star data in Figure 23 are stored in the R variables starx and stary, the command

chreg(starx,stary)

returns an estimate of the slope equal to 4.0. The R function MMreg estimates the slope to be 2.25, the only point being that the choice of estimator can make a practical difference.

23.3 Other Regression Estimators

Quantile regression is an extension of least absolute value regression.

The R

`rqfit(x,y,qval=.5,xout=F,outfun=out,res=F)`,

performs the calculations; it calls the function `rq`. (Both `rq` and `rqfit` assume that you have installed the R package `quantreg`. To install this package, start R and use the command `install.packages("quantreg")`.) One advantage of `rqfit` over the built-in function `rq` is that, by setting the argument xout=T, it will remove any leverage points found by the function indicated by the argument `outfun`, which defaults to the function `out`.

The R function

`mdepreg(x,y)`

computes the deepest regression line. It is based on the goal of a regression line giving the median of Y given X in a manner that protects against bad leverage points.

Other methods are least trimmed squares, S-estimators, skipped estimators, plus several others listed in Wilcox (2005). Skipped estimators remove outliers, using an appropriate multivariate outlier detection method (such as a projection method or the MGV method), and apply some regression estimator to the data that remain. Least squares is not a good choice. Better is the Theil - Sen estimator. The R functions

opreg

and

mgvreg

apply skipped estimators. They perform well when there is heteroscedasticity.

23.4 Dealing with Curvature: Smoothers

Roughly, given the goal of predicting Y, given $X = x$, look at X_i values close to x and use the corresponding Y_i values to determine the predicted value of Y. Do this for all X_i $(i = 1, \ldots, n)$ yielding $\hat{Y}_i$. Many variations have been proposed.

The built-in R function

```
lowess(x,y,p=2/3)
```

computes Cleveland's smoother. The value for p, the span, defaults to 2/3. You can create a scatterplot of points that contains this smooth with the R commands

```
plot(x,y)
```
lines(lowess(x,y)).

If the line appears to be rather ragged, try increasing p to see what happens. If the line appears to be approximately horizontal, indicating no association, check to see what happens when p is lowered.

The R function

```
lplot(x,y,span=.75,pyhat=F,eout=F,xout=F,outfun=out,plotit=T,
            expand=.5,low.span=2/3,varfun=pbvar,
      scale=F,xlab="X",ylab="Y",zlab="",theta=50,phi=25)
```

(in my library of R functions) plots Cleveland's smoother automatically and provides a variety of options that can be useful. For example, it will remove all outliers if `eout=T`. To remove leverage points only, use `xout=T`. If `pyhat=T`, the function returns $\hat{Y}_i$, the estimate of Y given that $X = X_i$. To suppress the plot, use `plotit=F`. The argument `low.span` is the value of p, the span, when using a single predictor. The arguments `xlab` and `ylab` can be used to label the x-axis and y-axis, respectively. The arguments `theta` and `phi` can be used to rotate three dimensional plots. The argument `scale` is important when plotting a

three-dimensional plot. If there is no association, the default `scale=F` typically yields a good plot. But when there is an association, `scale=T` is usually more satisfactory. (The argument `varfun` has to do with computing a robust analog the Pearson's correlation.)

EXAMPLE

For a diabetes study, the goal was to understand the association between the age of children at diagnosis and their C-peptide levels. The hypothesis of a zero slope is rejected with the R function `hc4test`; the p-value is 0.034. Student's T test of a zero slope has a p-value of 0.008. So a temptation might be to conclude that as age increases, C-peptide levels increase as well. But look at Figure 29, which shows Cleveland's smooth. Note that for children up to about the age of 7, there seems to be a positive association. But after the age of 7, it seems that there is little or no association at all.

EXAMPLE

The plasma retinol data contains data on amount of fat consumed per day and the amount of fiber consumed per day, which are used here to predict plasma retinol levels. Assuming the data are stored in the R variables `x` and `y`, the R command

```
lplot(x,y,scale=T,xlab="FAT",ylab="FIBER",zlab="PLASMA RETINOL")
```

creates the plot shown in Figure 30. The plot suggests that for low fat consumption, as the amount of fiber consumed increases, the typical plasma retinol level increases as well. But for high fat consumption, it appears that as fiber consumption increases, there is relatively little or no increase in plasma retinol levels.

The R function

```
rplot(x,y, est = tmean, scat = T, fr = NA, plotit = T, pyhat = F, efr = 0.5,
  theta = 50, phi = 25, scale = F, expand = 0.5, SEED = T, varfun = pbvar,
 nmin = 0, xout = F, outfun = out, eout = F, xlab = "X", ylab = "Y", zlab =
                                        "")
```

computes a running-interval smooth. The argument est determines the measure of location that is used and defaults to a 20% trimmed mean. The argument fr corresponds to the span

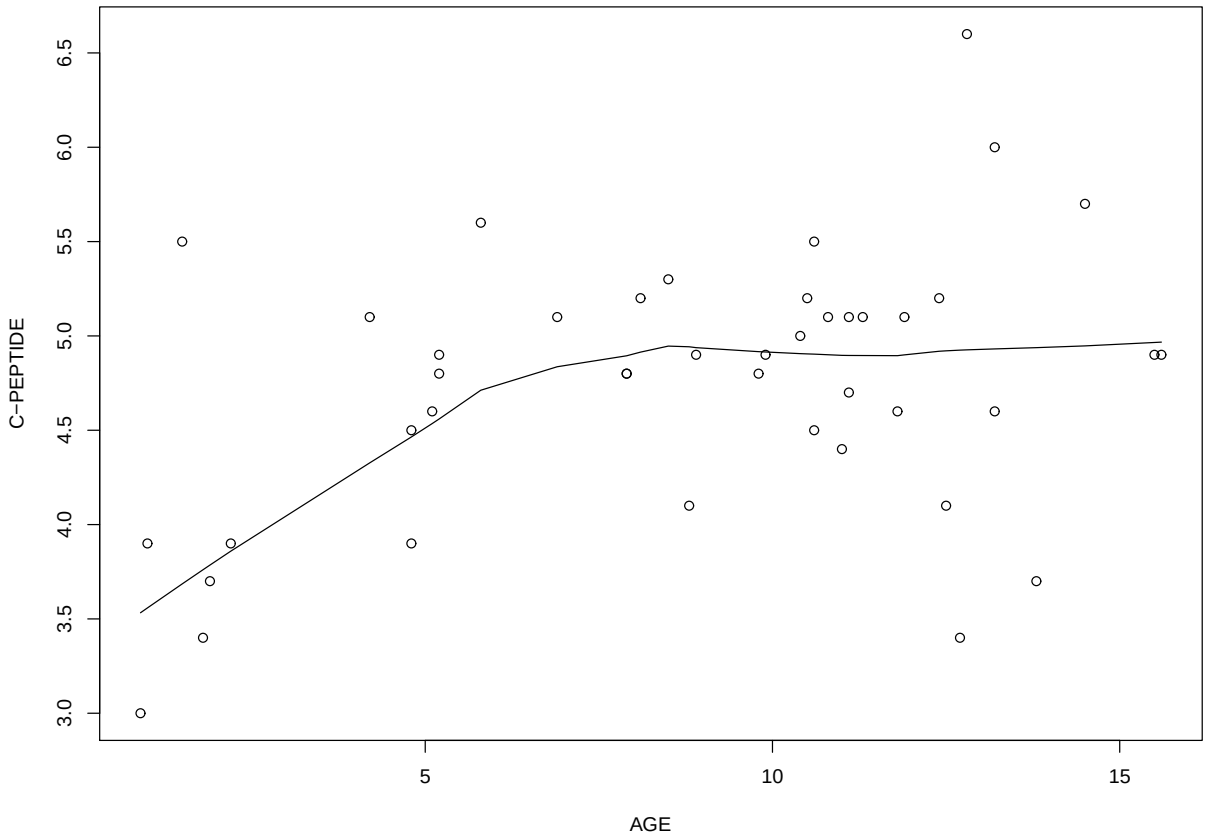

Figure 29: A smooth created by the R function `lplot` using the diabetes data. Note that there seems to be a positive association up to about the age of 7, after which there is little or no association.

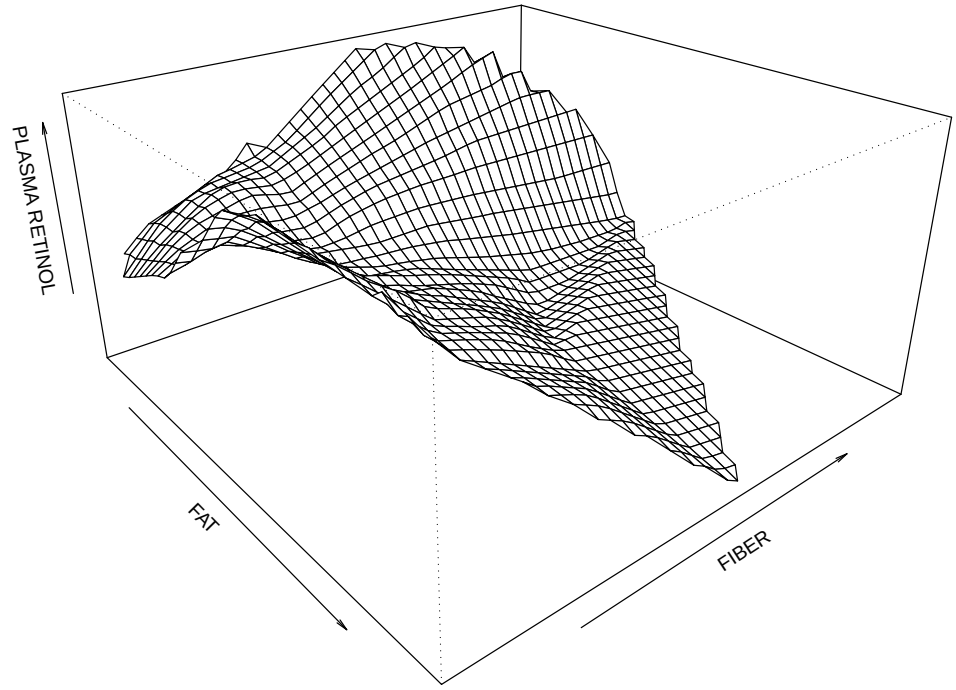

Figure 30: The smooth created by `lplot` when predicting plasma retinol given fat and fiber consumption

(f) and defaults to 0.8. The function returns the $\hat{Y}_i$ values if the argument pyhat is set to T. By default, a scatterplot of the points is created with a plot of the smooth. To avoid the scatterplot, set scat=F. The function

```
rplotsm(x,y,est=tmean,fr=1,plotit=T,pyhat=F,nboot=40,
atr=0,nmin=0,outfun=out,eout=F,xlab="X",ylab="Y",scat=T,
   SEED=T,expand=.5,scale=F, varfun=pbvar,pr=T,...)
```

estimates the regression line with the running-interval smoother in conjunction with bootstrap bagging.

The R function

```
kerreg(x, y, pyhat = F, pts = NA, plotit = T, theta = 50, phi = 25, expand =
0.5, scale = F, zscale = F, eout = F, xout = F, outfun = out, np = 100, xlab
          = "X", ylab = "Y",zlab="", varfun = pbvar, e.pow = T)
```

computes a kernel smoother. If `pts` is a matrix having p columns, and `pyhat=T`, the function estimates Y for each row in `pts`.

The R function

```
runpd(x,y, est=tmean, fr=0.8, plotit=T, pyhat=F, nmin = 0, theta=50, phi=25,
            expand=0.5, scale=F, xlab="X", ylab="Y", MC=F, ...)
```

computes the running-interval smoother based on projection distances. By default it estimate the 20% trimmed mean of Y given values for $X_1, \ldots, X_p$. With MC=T, a multi-core processor will be used if one is available.

The function

```
qsmcobs(qval = 0.5, xlab = "X", ylab = "Y", FIT = T, pc = ".", plotit = T)
```

performs COBS. The quantile to be estimated corresponds to the argument `qval`. Unlike the other smoothers in this section, this particular smoother is limited to a single predictor.

23.5 Comparing the Slopes of Two Independent Groups Based on Robust Estimator

Consider two independent groups and imagine that the goal is to test

$$H_0 : \beta_{11} = \beta_{12}, \tag{31}$$

the hypothesis that the two groups have equal slopes. Use percentile bootstrap method.

The R function

reg2ci(x1, y1, x2, y2, regfun=tsreg, nboot=599, alpha=0.05, plotit=T)

compares the slopes of two groups using a percentile bootstrap method. The data for group 1 are stored in x1 and y1, and for group 2 they are stored in x2 and y2. As usual, nboot is B, the number of bootstrap samples, regfun indicates which regression estimator is to be used and defaults to the Theil - Sen estimator, and plotit=T creates a plot of the bootstrap estimates.

To provide some visual sense of how the regression lines differ, and to provide an informal check on whether both regression lines are reasonably straight, the R function

runmean2g(x1, y1, x2, y2, fr=0.8, est=tmean, xlab="X", ylab="Y", ...)

has been supplied. It creates a scatterplot for both groups (with a + used to indicate points that correspond to the second group) and it plots an estimate of the regression lines for both groups using the running-interval smoother in Section 15.4.3. By default, it estimates the 20% trimmed mean of Y, given X. But some other measure of location can be used via the argument `est`. The smooth for the first group is indicated by a solid line, and a dashed line is used for the other. The R function

```
l2plot(x1, y1, x2, y2,span=2/3,xlab = "X", ylab = "Y")
```

also plots smoothers for each group, only it uses LOESS to estimate the regression lines.

EXAMPLE

A controversial issue is whether teachers' expectancies influence intellectual functioning. A generic title for studies that address this issue is Pygmalion in the classroom. Rosenthal

and Jacobson (1968) argue that teachers' expectancies influence intellectual functioning, and others argue that it does not. A brief summary of some of the counter arguments can be found in Snow (1995). Snow illustrates his concerns with data collected by Rosenthal where children in grades 1 and 2 were used. Here, other issues are examined using robust regression methods. One of the analyses performed by Rosenthal involved comparing a group of experimental children, for whom positive expectancies had been suggested to teachers, to a control children for whom no expectancies had been suggested. (The data used here are taken from Elashoff and Snow, 1970.) One measure was a reasoning IQ pretest score, and a second was a reasoning IQ posttest score. Here we consider whether the slopes of the regression lines differ for the two groups when predicting posttest scores based on pretest scores. Figure 31 shows the output from the R function runmean2g. The 0.95 confidence interval for the difference between the slopes, returned by reg2ci and based on the Theil - Sen estimator, is $(-0.72, 0.18)$ with a p-value of 0.22.

24 Robust Measures of Association

The R Function

```
wincor(x,y,tr=.2)
```

computes the Winsorized correlation and tests the hypothesis H_0: $\rho_w = 0$ using the Student's T test described in the previous section. The amount of Winsorization is controlled by the argument `tr` and defaults to 0.2.

Winsorized correlation does not take into account the overall structure of the data when dealing with outliers. The OP correlation does. It uses the projection method for remove outliers and then computes a correlation using remaining data.

The R Function

```
scor(x,y, plotit = T, xlab = "VAR 1", ylab = "VAR 2", MC = F)
```

computes the OP correlation.

The R Function

```
corb(x,y, corfun = pbcor, nboot = 599,...)
```

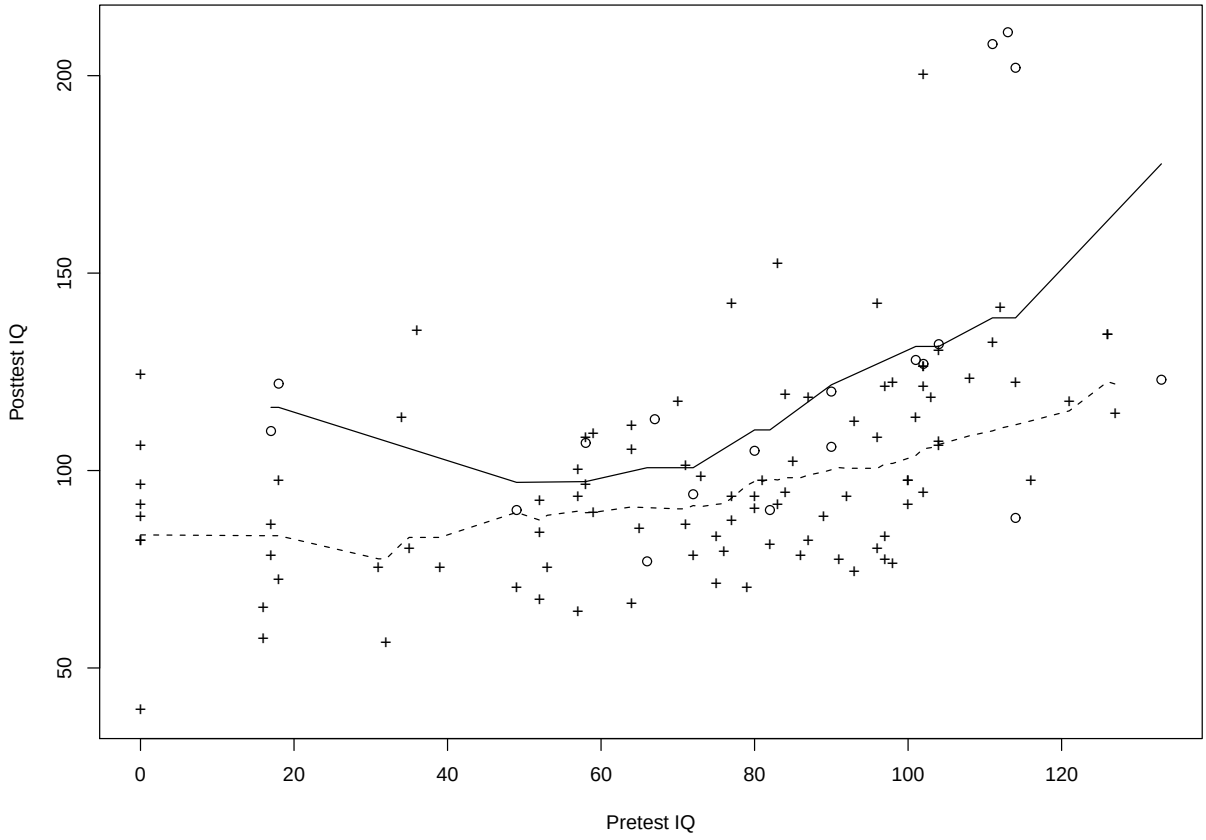

Figure 31: Plots of the regression lines when predicting posttest IQ scores given pretest IQ scores. The solid regression line is based on the experimental group.

can be used to test hypotheses and compute confidence intervals for any of the robust correlations covered in this chapter. By default, the function uses the percentage bend correlation, which is not covered here. (See Wilcox, 2005, Section 9.3.1 for details.) It is an M-type correlation. When using the OP correlation, it is suggested that the command

```
corb(x,y, corfun = scor, plotit=F)
```

be used. Otherwise the function will create a scatterplot for each bootstrap sample, which can increase execution time considerably.

24.1 Measuring the Strength of an Association Based on a Robust Fit

Goal: Given a fit to the data based on a robust regression estimator or smoother, measure the strength of the association. Use a simple generalizations of the notion of explanatory power, which was studied in a general context by Doksum and Samarov (1995). Taking $\tilde{Y}$ to be the predicted value of Y based on any regression estimator or smoother, let $\tau^2(Y)$ be any measure of variation. Then a robust analog of *explanatory power* is

$$\eta^2 = \frac{\tau^2(\tilde{Y})}{\tau^2(Y)}. \tag{32}$$

The *explanatory strength of association* is the (positive) square root of explanatory power, η.

To put η^2 in perspective, if $\tilde{Y}$ is the predicted value of Y based on the least squares regression line, and τ^2 is the usual variance, then η^2 reduces to the squared multiple correlation coefficient. In the case of a single predictor, still using least squares regression, η is just Pearson's correlation, ρ, assuming that the sign of η is taken to be the sign of the slope of the least squares regression line.

Choice of smoother matters in terms of getting an accurate estimate of explanatory power. Using the R function lplot performs relatively well.

24.2 Tests for Linearity

The R function

```
lintest(x,y,regfun=tsreg,nboot=500,alpha=.05)
```

tests the hypothesis that a regression surface is a plane. It uses a wild bootstrap method.

The R function

```
lintestMC(x,y,regfun=tsreg,nboot=500,alpha=.05)
```

is the same as `lintest`, only it takes advantage of a multi-core processor, if one is available, with the goal of reducing execution time.

25 Moderator Analysis

A standard approach uses least squares assuming that

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + e. \tag{33}$$

Rearranging terms

$$Y = (\beta_0 + \beta_2 X_2) + (\beta_1 + \beta_3 X_2) X_1 + e,$$

so the slope for X_1 changes as a linear function of X_2. (An R function, called ols.plot.inter, plots the regression surface when using the least squares estimate of the parameters.) Currently, a commonly used method for testing the hypothesis of no interaction is to test H_0: $\beta_3 = 0$, meaning that the slope for X_1 does not depend on X_2.

Example: predictors of reading ability.

```
model=lm(y~x[,1]*x[,2])
summary.aov(model)
```

```
              Df  Sum Sq Mean Sq F value    Pr(>F)
x[, 1]         1  6997.9  6997.9 30.0057 5.224e-07 ***
x[, 2]         1  3247.4  3247.4 13.9240 0.0003623 ***
x[, 1]:x[, 2]  1    86.9    86.9  0.3724 0.5434918
Residuals     77 17957.9   233.2
```

To deal with heteroscedasticity, still assuming normality, the R function `olshc4` (or the R function `olswbtest`) can be used. Now the appropriate R commands are

```
xx=cbind(x,x[,1]*x[,2])
olshc4(xx,y)
```

To deal with both heteroscedasticity and nonnormality, and simultaneously remove leverage points, one possibility is

```
xx=cbind(x,x[,1]*x[,2])
regci(xx,y,xout=T,outfun=outpro)
```

Now the p-value for H_0: $\beta_3 = 0$, using the reading data, is 0.023.

A more flexible approach for establishing that there is an interaction is to test

$$H_0 : Y = \beta_0 + f_1(X_1) + f_2(X_2) + e, \tag{34}$$

the hypothesis that for some unknown functions f_1 and f_2, a generalized additive model fits the data, versus the alternative hypothesis

$$H_1 : Y = \beta_0 + f_1(X_1) + f_2(X_2) + f_3(X_1, X_2) + e.$$

The R function

```
adtest(x, y, nboot=100, alpha=0.05, xout=F, outfun=out, SEED=T, ...)
```

tests this hypothesis.

Could have substantially more power, as well as substantially less power, versus the robust method just illustrated. (If the regression model used by the robust method is correct, the expectation is that it might have a substantial power advantage compared to using adtest.)

EXAMPLE

A portion of a study conducted by Shelley Tom and David Schwartz dealt with the association between a Totagg score and two predictors: grade point average (GPA) and a measure of academic engagement. The Totagg score was a sum of peer nomination items that were based on an inventory that included descriptors focusing on adolescents' behaviors and social standing. (The peer nomination items were obtained by giving children a roster

sheet and asking them to nominate a certain amount of peers who fit particular behavioral descriptors.) The sample size is $n = 336$. Assuming that the model given by Equation (33) is true, the hypothesis of no interaction (H_0: $\beta_3 = 0$) is not rejected using the least squares estimator. The p-value returned by the R function olswbtest is .6. (And the p-value returned by `olshc4` is 0.64.) But look at the left panel of Figure 32, which shows the plot of the regression surface assuming Equation (33) is true. (This plot was created with the R function `ols.plot.inter` described in the next section.) And compare this to the right panel, which is an estimate of the regression surface using LOESS (created by the R function `lplot`). This suggests that using the usual interaction model is unsatisfactory for the situation at hand. The R function `adtest` returns a p-value less than .01 indicating that an interaction exists.

25.1 R Functions kercon, runsm2g, regi, ols.plot.inter and reg.plot.inter

The R function

```
kercon(x,y,cval=NA,eout=F,xout=F, outfun=out,xlab="X",ylab="Y"),
```

creates the first of the plots mentioned in the previous section. The argument `x` is assumed to be a matrix with two columns. By default, three plots are created: a smooth of Y and X_1, given that X_2 is equal to its lower quartile, its median, and its upper quartile. Different choices for the X_2 values can be specified via the argument `cval`.

The R function

```
runsm2g(x1,y,x2, val = median(x2), est = tmean, sm = F, xlab = "X", ylab =
                                    "Y", ...)
```

splits the data in `x1` and `y` into two groups based on the value in the argument `val` and the data stored in the argument `x2`. By default, a median split is used. The function returns a smooth for both groups. If there is no interaction, these two smooths should be approximately parallel. The smooths are based on the goal of estimating the trimmed mean of the outcome variable. But other measures of location can be used via the argument `est`.

The R function

```
regi(x,y,z,pt=median(z),est=onestep,regfun=tsreg,
```

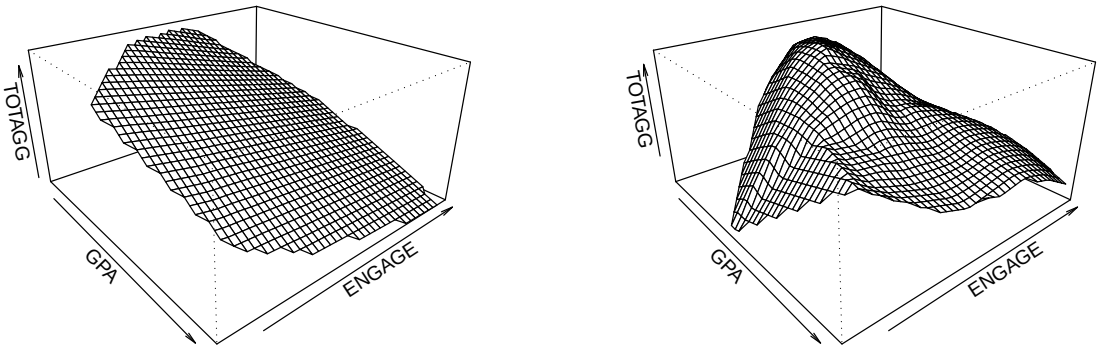

Figure 32: The left panel shows the plot created by `ols.plot.inter`, which assumes that an interaction can be modeled with $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + e$ and where the least squares estimate of the parameters is used. The right panel shows an approximation of the regression surface based on the R function `lplot`

testit=F,...)

creates two smooths in a manner similar to the function `runsm2g`. In fact this function simply calls the function `runsm2g`, only it uses by default a one-step M-estimator rather than a 20% trimmed mean. An advantage of this function is that it also has an option for replacing the smoother with a robust regression estimator, via the argument `regfun`, which by default is the Theil - Sen estimator. This is done by setting the argument `testit=T`, in which case the function also tests the hypothesis that the two regression lines have equal slopes. Rejecting the hypothesis of equal slopes indicates an interaction.

The R function

ols.plot.inter(x, y, pyhat = F, eout = F, xout = F, outfun = out, plotit = T, expand = 0.5, scale = F, xlab = "X", ylab = "Y", zlab = "", theta = 50, phi = 25, family = "gaussian", duplicate = "error")

plots the regression surface assuming that the commonly used interaction model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + e$$

is true and that the least squares estimate of the parameters is used.

The R function

reg.plot.inter(x, y, regfun=tsreg, pyhat = F, eout = F, xout = F, outfun = out, plotit = T, expand = 0.5, scale = F, xlab = "X", ylab = "Y", zlab = "", theta = 50, phi = 25, family = "gaussian", duplicate = "error")

is exactly like the function ols.plot.inter, only it can be used with any regression estimator that returns the residuals in $residuals. By default, the Theil - Sen estimator is used.

26 Mediation Analysis

Mediation analysis is similar to a moderator analysis in the sense that the goal is to understand how the association between two variables is related to a third (mediating) variable. In the parlance of researchers working on this problem, an *indirect effect*, also known as a *mediation effect*, refers to a situation where two variables of interest are associated via a

third variable. For example, stress and obesity are believed to be associated through cortisol secretion (Rosmond et al., 1998). The strategy behind a mediation analysis is to assume that the three variables of interest satisfy three linear models. The first is that two primary variables of interest x and y (e.g., stress and obesity) are related via the usual linear model

$$y = \beta_{01} + \beta_{11}x + \epsilon_1. \tag{35}$$

The second assumption is that the mediating variable (cortisol in the example), which here is labeled x_m, is related to x via

$$x_m = \beta_{02} + \beta_{12}x + \epsilon_2. \tag{36}$$

And finally, it is assumed that

$$y = \beta_{03} + \beta_{13}x + \beta_{23}x_m + \epsilon_3. \tag{37}$$

Roughly, if $\beta_{13} = 0$, this is said to constitute full mediation (Judd and Kenny, 1981a, 1981b). If the strength of the association between of x and y is reduced when the mediator is included, in the sense that $\beta_{13} < \beta_{11}$, there is said to be partial mediation.

Various strategies have been proposed for assessing whether x_m mediates the association between y and x. One is to focus on $\beta_{11} - \beta_{13}$. Another is to focus on the product $\beta_{12}\beta_{23}$, which has been called the *mediated effect* or *indirect effect*. This approach arises by noting that the total effect represented by the slope in Eq. (35) satisfies $\beta_{11} = \beta_{12}\beta_{23} + \beta_{13}$. Consequently, a common goal is testing

$$H_0 : \beta_{12}\beta_{23} = 0. \tag{38}$$

Under normality and homoscedasticity, a bootstrap method for testing this hypothesis, using the least squares estimator, has been found to perform reasonably well in simulations. But under nonnormality, or when there is heteroscedasticity, this is no longer the case (Ng, 2009a). Replacing the least squares estimator with the Theil - Sen estimator, Ng (2009a) found that a percentile bootstrap method performs well in simulations when $\beta_{12} = \beta_{23} = 0$. But otherwise, control over the probability of a Type I error can be unsatisfactory in some situations. It appears that testing

$$H_0 : \beta_{11} - \beta_{13} = 0 \tag{39}$$

via the Theil-Sen estimator (using a percentile bootstrap method) avoids Type I error probabilities greater than the nominal level, but for $n = 40$, the actual level can drop well below the nominal level. When testing at the .05 level, simulations suggest that with $n = 80$, this problem is avoided.

Recently, Zu and Yuan (2010) derived an approach to testing Eq. (38) based on a Huber-type M-estimator. One of the methods they considered uses a percentile bootstrap technique, which has the advantage of allowing heteroscedasticity. The method appears to perform well in terms of controlling the probability of a Type I error when there is homoscedasticity. But with $n = 60$ and when all variables have normal distributions, only two outliers can result in a Type I error probability of .09 when testing at the .05 level. And there are situations where the method is unsatisfactory when there is heteroscedasticity. Currently, it seems that a simple modification of their method avoids these problems even with $n = 40$: eliminate any (X, X_m, Y) value for which X is an outlier. Here the MAD-median rule is used. If instead (X, X_m, Y) is eliminated when (X, X_m) is an outlier, control over the probability of a Type I error can be poor. (Biesanz et al., 2010, compared several methods but the results relevant to nonnormality are limited to a single nonnormal distribution that is skewed with a relatively light tail.)

It should be noted that Green et al. (2010) argue that mediation analyses have been based on regression models that rest on naive assumptions. The stated goal in the abstract of their paper is "to puncture the widely held view that it is a relatively simple matter to establish the mechanism by which causality is transmitted. This means puncturing the faith that has been placed in commonly used statistical methods of establishing mediation."

26.1 R Functions ZYmediate and regmediate

The R function

```
ZYmediate(x, y, nboot = 2000, alpha = 0.05, kappa = 0.05, SEED = T,xout=T)
```

tests the hypothesis given by Eq. (38) using a slight modification of the method derived by Zu and Yuan (2010), which was outlined in the previous section. Currently, it seems to be one of the better methods when the sample size is small.

The R function

```
regmediate(x,y,regfun=tsreg,nboot=400,alpha=.05,xout=F,outfun=out,MC=F,SEED=T,...)
```

computes a confidence interval for $\beta_{11} - \beta_{13}$, and a p-value when testing H_0: $\beta_{11} = \beta_{13}$ is returned as well. The Theil- Sen estimator is used by default. Very limited results suggest that ZYmediate is generally better in terms of power, but this issue is in need of further study.

27 ANCOVA

Let m(x) be some measure of location associated with Y, given X. For two independent groups, one general goal is to test

$$H_0 : m_1(X) = m_2(X), \text{ for all } X \tag{40}$$

That is, the regression lines do not differ in any manner.

Another goal is determining where the regression lines differ and by how much.

The methods used here allow both heteroscedasticity and curvature. (Smoothers are used to approximate the regression lines.)

27.1 R Functions ancsm, Qancsm, ancova, ancpb, ancbbpb and ancboot

The R function

```
ancova(x1,y1,x2,y2,fr1=1,fr2=1,tr=0.2,alpha=0.05,plotit=T,pts = NA)
```

performs a local nonparametric ANCOVA analysis based on the running interval smoother, which defaults to estimating the 20% trimmed of Y given X. The arguments `x1, y1, x2, y2, tr` and `alpha` have their usual meanings. The arguments `fr1` and `fr2` are the spans used for groups one and two respectively. The argument `pts` can be used to specify the X values at which the two groups are to be compared. For example, `pts=12` will result in comparing the trimmed mean for group 1 (based on the y1 values) to the trimmed mean of group 2 given that $X = 12$. If there is no trimming, the null hypothesis is $H_0 : E(Y_1|X = 12) = E(Y_2|X = 12)$, where Y_1 and Y_2 are the outcome variables of interest corresponding to the two groups. Using pts=c(22,36) will result in testing two hypotheses. The first is $H_0 : m_1(22) = m_2(22)$ and the second is $H_0 : m_1(36) = m_2(36)$. If no values for `pts` are specified, then the function picks five X values and performs the appropriate tests. The values that it picks are reported in the output as illustrated below. Generally, this function controls FWE using the method in Section 13.1.8. If plotit=T is used, the function also creates a scatterplot and smooth for both groups with a + and a dashed line indicating the points and the smooth, respectively, for group 2.

The function

ancpb(x1,y1,x2,y2,est=onestep,pts=NA,fr1=1,fr2=1,
nboot=599,plotit=T,...)

is like the R function ancova only a percentile bootstrap method is used to test hypotheses and by default the measure of location is the one-step M-estimator. Now FWE is controlled as described in Section 13.1.11. In essence, the function creates groups based on the values in `pts`, in conjunction with the strategy behind the smooth, it creates the appropriate set of linear contrasts, and then it calls the function pbmcp described in Section 13.1.12.

The function

ancboot(x1,y1,x2,y2,fr1=1,fr2=1,tr=0.2,nboot=599,pts=NA,plotit = T)

compares trimmed means using a bootstrap-t method. Now FWE is controlled as described in Section 13.1.13.

The function

ancbbpb(x1,y1,x2,y2, fr1 = 1, fr2 = 1, nboot = 200, pts = NA, plotit = T, SEED = T,
alpha = 0.05)

is the same as `ancova`, only a bootstrap bagging method is used to estimate the regression line. Roughly, *bootstrap bagging* means that B bootstrap samples are taken, for each bootstrap sample a smoother is applied, and the final estimate of the regression line is based on the average of the B bootstrap estimates. (See Wilcox, 2009b, for more details.) The practical advantage is that it might provide higher power than `ancova`. A negative feature is that execution time can be high.

ancsm(x1, y1, x2, y2, nboot = 200, SEED = T, est = tmean, fr = NULL, plotit = T, sm =
F, tr = 0.2, xout=F, outfun=out,...)

applies a global test based on the notion of regression depth.

EXAMPLE

The ANCOVA methods described in this section are illustrated with the Pygmalion data described at the end of Section 15.7.1. The goal is to compare posttest scores for the two

groups taking into account the pretest scores. If the data for the experimental group are stored in the R matrix pyge, with the pretest scores in column 1, and the data for the control group are stored in pygc, the command

ancsm(pyge[,1],pyge[,2],pygc[,1],pygc[,2])

returns p-value of 0.025, indicating that the regression lines differ for some values of the covariate.

ancova(pyge[,1],pyge[,2],pygc[,1],pygc[,2])

returns

```
   X n1 n2      DIF     TEST        se     ci.low    ci.hi
  72 12 63 13.39103 1.848819  7.243016  -9.015851 35.79790
  82 16 68 14.79524 1.926801  7.678655  -8.211174 37.80165
 101 14 59 22.43243 1.431114 15.674806 -26.244186 71.10905
 111 12 47 23.78879 1.321946 17.995286 -35.644021 83.22161
 114 12 43 21.59722 1.198906 18.014112 -37.832791 81.02724
```

The first column headed by X says that posttest scores are being compared given that pretest scores (X) have the values 72, 82, 101, 111 and 114. The sample sizes used to make the comparisons are given in the next two columns. For example, when $X = 72$, there are twelve observations being used from the experimental group and sixty-three from the control. That is, there are twelve pretest scores in the experimental group and sixty-three values in the control group that are close to $X = 72$. The column headed by DIF contains the estimated difference between the trimmed means. For example, the estimated difference between the posttest scores, given that $X = 72$, is 13.39. The last two columns indicate the ends of the confidence intervals. These confidence intervals are designed so that FWE is approximately α. The critical value is also reported and is 3.33 for the situation here. All of the confidence intervals contain zero, none of the tests statistics exceeds the critical value, so we fail to detect any differences between posttest scores taking into account the pretest scores of these individuals.

If we apply the function `ancpb` with the argument `est=mom`, a portion of the output is

```
$mat
       X n1 n2
```

```
[1,]  72 12 63
[2,]  82 16 68
[3,] 101 14 59
[4,] 111 12 47
[5,] 114 12 43

     con.num    psihat p.value p.crit  ci.lower ci.upper
[1,]       1 12.334699  0.0604 0.0204 -3.920635 33.19209
[2,]       2  7.907925  0.1432 0.0338 -5.643923 54.38899
[3,]       3  8.092476  0.1168 0.0254 -5.228571 57.32143
[4,]       4  6.917874  0.1688 0.0500 -7.111111 97.44099
[5,]       5  5.388889  0.2558 0.0500 -9.784884 94.97619
```

Again we fail to find any differences. However, using the function `ancbbpb`, which uses bootstrap bagging, the results are

```
$output
       X n1 n2      DIF     ci.low    ci.hi    p.value
[1,]  72 12 63 12.03672 -2.7430218 22.37841 0.16528926
[2,]  82 16 68 16.24183  0.8489948 24.69427 0.03007519
[3,] 101 14 59 28.32713  3.5099184 48.23010 0.01398601
[4,] 111 12 47 31.94660  8.9667976 70.64249 0.01273885
[5,] 114 12 43 34.23546  7.4661331 71.24803 0.01149425
```

So now we reject at the 0.05 level for four of the five designs points. The R function `ancbbpb` might provide more power than `ancova`, but this comes at the cost of higher execution time.

27.2 Multiple Covariates

There are various ways multiple covariates might be handled. Momentarily focus on the ith value of the covariate in the first group, $\mathbf{x}_{i1}$. Then it is a simple matter to determine the set of observed points close to $\mathbf{x}_{i1}$. The same can be done for the second group, in which case we proceed as done by the R function ancova.

The R function

ancovamp(x1,y1,x2,y2,fr1=1,fr2=1,tr=.2,alpha=.05,pts=NA)

compares two groups based on trimmed means and takes into account multiple covariates.

The function

ancmppb(x1,y1,x2,y2,fr1=1,fr2=1,tr=.2,alpha=.05,pts=NA,est=tmean,
nboot=NA,bhop=F,...)

is like ancovamp, only a percentile bootstrap method is used and any measure of location can be employed.

27.3 Some Global Tests

The R function

ancom(x1,y1,x2,y2,dchk=F,plotit=T,plotfun=rplot,nboot=500,alpha=0.05,
SEED=T,PARTEST=F,...)

tests the hypothesis that two regression lines are identical. If plotit=T and $p = 1$ or 2, a plot of the pooled data is created along with a smooth indicated by the value of the argument plotfun. So, for example, if $p = 2$ and a bagged version of the running interval smooth is desired, use the command ancom(x1,y1,x2,y2,plotfun=rplotsm).

Two other R functions are

ancsm(x1, y1, x2, y2, nboot = 200, SEED = T, est = tmean, fr = NULL, plotit = T, sm =
F, tr = 0.2),

which defaults to using a running interval smoother with a 20% trimmed mean. Setting sm=T, bootstrap bagging will be used, which might increase power.

The function

Qancsm(x1, y1, x2, y2, nboot = 200, SEED = T, qval = 0.5, xlab = "X", ylab = "Y",
plotit = T).

is like ancsm, only COBS is used to estimate the quantile regression lines. The argument qval determines the quantile that is used and defaults to .5, the median.

There is a feature of the global tests that should be stressed that is relevant to the classic ANCOVA method as well. Imagine that for the first group, the range of the covariate values, x, is 0-10, and for the second group the range is 30-40. Classic ANCOVA would assume that the regression lines are parallel and compare the intercepts. In the even the regression lines are indeed straight, the global tests applied via the functions in this section are aimed at testing the hypothesis that the slopes, as well as the intercepts, are equal, assuming the usual linear model holds. But based on the local ANCOVA methods in the previous section, comparisons would not be made. For instance, it might be of interest to determine whether the groups differ when the covariate $x = 5$. Because there are no results for the second group when $x = 5$, comparisons could not be made. And perhaps comparisons should not be made by imposing assumptions such as those made by the classic ANCOVA model.