Introduction to the Bootstrap and Robust Statistics

PSY711/712

Winter Term 2009

1 Comparing Multiple Dependent Groups

1.1 adjusted degrees of freedom

All of the analyses described in the previous sections assume that the errors associated with each measurement are independent of each other. That assumption undoubtedly is too strong when multiple measurements are made on the same sampling unit. For instance, if we measure the behaviour of an animal on different days, then the data for each animal are likely to be correlated. In these types of within-subjects experimental designs (also called repeated measures), the independence assumption is likely to be invalid. Consequently, the p-values obtained in such cases are likely to be misleading.

To analyze data from within-subjects designs, the independence assumption is relaxed to allow for particular patterns of correlations to exist among the levels of within-subject variables. Specifically, the variance-covariance matrix of the within-subject measures is assumed to exhibit the property known as *sphericity*. Consider a set of J measures made on our group of animals: we calculate all possible pairwise differences among the J measures for each animal, and calculate the variance for each difference score. If the sphericity assumption is valid, then the variances of the difference scores will be equal. If the variances differ significantly, then the data are not spherical. Also, if the correlations among J dependent measures are equal – i.e, if the correlation between J_j and J_k are equal for all $j \neq k$ – then the data are spherical. This statement implies that sphericity necessarily is true when the within-subjects variable has only two levels. More generally, sphericity is only an issue with F tests that have more than 1 degree of freedom in the numerator.

Unfortunately, the assumption of sphericity often is not valid. In other words, the observed variance-covariance matrix of dependent measures often deviates significantly from sphericity, and therefore the p values for within-subjects F tests are invalid. For example, suppose we use a repeated measures design to measure the behaviour of animals in some task at 1-hour intervals. Generally, things that happen close together in time (or space) tend to be correlated more than things that happen far apart in time (or space). Therefore, we would expect to find that the correlations among our repeated measures in this hypothetical experiment to *not* be equal, and the variance-covariance matrix of our dependent variables would not be spherical. Consequently, the F calculated on our sample would not follow the theoretical F distribution, and the p-value for our significance test could be misleading. In turns out, however, that that the observed F is distributed *approximately* as an F variable with reduced degrees of freedom. The appropriate reduction in degrees of freedom depends on how close the variance-covariance matrix is to being spherical. Let's call our index of sphericity ϵ, which varies from 1 (perfect sphericity) to a minimum value of $1/(J-1)$: the adjusted degrees of freedom are $\epsilon(J-1)$ and $\epsilon(n-1)(J-1)$, where n is the number of samples/subjects per condition. In the worst case – when the data depart maximally from sphericity – the degrees of freedom are 1 and $n-1$. Evaluating F with these degrees of freedom is known as the **Geisser-Greenhouse conservative F test**: if the F is significant with these degrees of freedom, then it is significant regardless of the spherical nature of the variance-covariance matrix.

There are two ways of estimating ϵ that are in common use. The first is the Geisser-Greenhouse estimate, which often is denoted as $\hat{\epsilon}$. The second is the Huynh-Feldt estimate, often labelled as $\tilde{\epsilon}$. Generally, $\hat{\epsilon} \leq \tilde{\epsilon}$ for any given set of data. Consequently, $\hat{\epsilon}$ does a better job controlling Type I error rates, but $\tilde{\epsilon}$ provides

slightly more power. When reporting results in scientific papers, it is standard practice to report the normal, unadjusted degrees of freedom, the value of $\hat{\epsilon}$ or $\tilde{\epsilon}$, and the adjusted degrees of freedom. The following example is from Gaspar et al. [2]:

> Statistical analyses were done with R [5]. When appropriate, the Huynh-Feldt estimate of sphericity ($\tilde{\epsilon}$) was used to adjust p values of F tests conducted on within-subject variables [4]. The strength of association between the dependent and independent variables was expressed as partial omega-squared (ω_p^2), which was calculated using formulae described by Kirk [3]. ... The main effects of cutoff frequency, $F(4, 20) = 0.82, \tilde{\epsilon} = 0.74, p = 0.50, \omega_p^2 = 0$, and noise type, $F(1, 5) = 0.01, p = 0.92, \omega_p^2 = 0$, were not significant, nor was the cutoff x noise type interaction, $F(4, 20) = 0.4, \tilde{\epsilon} = 0.63, p = 0.72, \omega_p^2 = 0$.

Wilcox [6] provides a simple way of using adjusted degrees of freedom for evaluating the hypothesis of no differences among dependent group means. The function `rmanova` calculates both $\hat{\epsilon}$ and $\tilde{\epsilon}$, but the Huynh-Feld correction, $\tilde{\epsilon}$, is the one used to adjust the degrees of freedom and calculate a p-value. Here is an example of using `rmanova`:

```
> source(url("http://www-rcf.usc.edu/~rwilcox/Rallfun-v9_2"))
> Y <- list()
> set.seed(715)
> Y[[1]] <- rnorm(20)
> Y[[2]] <- rnorm(20, m = 1)
> Y[[3]] <- rnorm(20, m = -0, sd = 4)
> rmanova(Y, tr = 0)

[1] "The number of groups to be compared is"
[1] 3
$test
[1] 6.691624

$df
[1]  1.229805 23.366303

$siglevel
[1] 0.01214735

$tmeans
[1]  0.1014829  1.2757470 -1.4299530

$ehat
[1] 0.597422

$etil
[1] 0.6149027
```

Notice that the trimming parameter was set to zero to force `rmanova` to evaluate the null hypothesis of no difference among group means. Evaluating hypotheses about trimmed means is done simply by using a different amount of trimming. The default amount is 20%:

```
> rmanova(Y, tr = 0.2)

[1] "The number of groups to be compared is"
[1] 3
```

```
$test
[1] 5.376855

$df
[1]  1.062677 11.689445

$siglevel
[1] 0.03775676

$tmeans
[1]  0.1383579  1.3157354 -1.6296190

$ehat
[1] 0.5267816

$etil
[1] 0.5313384
```

The function returns the group means or trimmed means, test statistic (F or F_t), adjusted degrees of freedom, and the values of $\hat{\epsilon}$ and $\tilde{\epsilon}$.

1.2 bootstrap methods

1.2.1 `rmanovab`

Adjustment of the degrees of freedom to correct for deviations from sphericity yield a statistic that is distributed approximately as F when the data are drawn from normal populations. When this is not true, then (as always) the p-values obtained in our analysis may be misleading. In cases where we are unwilling to make the normality assumption, bootstrap procedures can be used to generate estimates of the sampling distribution of F. The function `rmanovab` uses a bootstrap to estimate the sampling distribution for F_t *when the null hypothesis is true*:

```
> rmanovab(Y, tr = 0, alpha = 0.05, nboot = 999)

[1] "Taking bootstrap samples. Please wait."
[1] "The number of groups to be compared is"
[1] 3
$teststat
[1] 6.691624

$crit
[1] 3.796138

> rmanovab(Y, tr = 0.2, alpha = 0.05, nboot = 999)

[1] "Taking bootstrap samples. Please wait."
[1] "The number of groups to be compared is"
[1] 3
$teststat
[1] 5.376855

$crit
[1] 4.855839
```

The function returns the observed value of F_t (which is equivalent to F when trimming is zero) calculated on the sample, as well as the **critical value** of F_t. When the observed value is greater than the critical value, then the null hypothesis of no difference among trimmed means is rejected at $p < \alpha$.

1.2.2 bootstrapping subjects, not scores

In a within-subjects design, each subject contributes J scores. These scores are likely to be correlated, and the bootstrap procedure should preserve this aspect of the data. Therefore, when using a bootstrap to analyze data collected with within-subjects designs, we generate bootstrapped samples of *subjects* rather than bootstrapped samples of the J conditions. So, if our experiment consists of 20 subjects and four conditions, then we would create a bootstrapped sample of 20 subjects (i.e., sample 1-20 randomly with replacement), and then use all 4 scores from each subject. Here's another way of thinking about the bootstrapping procedure: we could represent the data from our imaginary experiment as a matrix containing 20 rows and four columns of numbers. The bootstrapping procedure selects *rows* of data randomly (with replacement) from the original sample.

1.2.3 `bd1way`

`bd1way` – part of Wilcox's collection of R functions – is a general function that evaluates the null hypothesis of no group differences using the statistic

$$Q = \sum_{j=1}^{J} (\hat{\theta}_j - \bar{\theta})^2$$

where $\bar{\theta} = \sum \hat{\theta}_j / J$, and θ_j is the statistic of interest (e.g., the mean, MOM, median, etc.). Note that Q is not the same as F – even if we are calculating differences among means – because it is not divided by an estimate of error. Nevertheless, Q is a reasonable statistic in the sense that it will be small when θ is approximately equal in all groups.

R Example:

First I'll use `bd1way` to evaluate differences among means:

```
> bd1way(Y, est = mean, nboot = 999, alpha = 0.05)

[1] "Taking bootstrap samples. Please wait."
$test
[1] 3.681668

$crit
[1] 1.733820
```

Notice that the test statistic is not the same as the F value computed previously. Here's where the value of 3.68 comes from:

```
> t1 <- mean(Y[[1]])
> t2 <- mean(Y[[2]])
> t3 <- mean(Y[[3]])
> t.bar <- (t1 + t2 + t3)/3
> (Q <- ((t1 - t.bar)^2 + (t2 - t.bar)^2 + (t3 - t.bar)^2))

[1] 3.681668
```

The observed value of Q exceeds the critical value, and therefore we reject the null hypothesis of no difference among group means.

Next I'll use `bd1way` to evaluate differences among 20% trimmed means:

```
> bd1way(Y, nboot = 999, alpha = 0.05, est = tmean, tr = 0.2)

[1] "Taking bootstrap samples. Please wait."
$test
[1] 4.395691

$crit
[1] 2.576135
```

And, finally, we'll use it to evaluate difference among Modified One-step M-estimators:

```
> bd1way(Y, nboot = 999, alpha = 0.05, est = mom)

[1] "Taking bootstrap samples. Please wait."
$test
[1] 4.024229

$crit
[1] 2.784723
```

2 Correlations

This example is taken from R's help page for `cor.test`. The data are from a paper by Hollander & Wolfe (1973). The data frame `tuna` contains data from an experiment that measured the lightness of tuna and perceived quality. The quality scores were averages of consumer panel scores which ranged from 1 to 6 (with 6 being highest quality). The two variables in `tuna` – `lightness` and `quality` – are accessed using the $ command.

We should start by plotting the data, but my figure is going to include the regression line. Therefore, I will start by doing the linear regression:

```
> load(url("http://psycserv.mcmaster.ca/bennett/rdata/tuna.Rdata"))
> # compute regression line:
> tuna.lm <- lm(quality~lightness, data=tuna)
> # show regression table:
> summary(tuna.lm)

Call:
lm(formula = quality ~ lightness, data = tuna)

Residuals:
    Min      1Q  Median      3Q     Max
-1.0165 -0.5685 -0.3365  0.7384  1.2970

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.02411    2.54020  -0.403    0.699
lightness    0.09718    0.05278   1.841    0.108

Residual standard error: 0.8673 on 7 degrees of freedom
Multiple R-squared: 0.3262,        Adjusted R-squared:  0.23
F-statistic:  3.39 on 1 and 7 DF,  p-value: 0.1082
```

Next, I'm going to plot the data, including the regression line:

```
> plot(tuna$lightness,tuna$quality,"p",xlab="Lightness",ylab="Quality",main="Tuna Quality vs. Ligh
> abline(tuna.lm) # add regression line
```

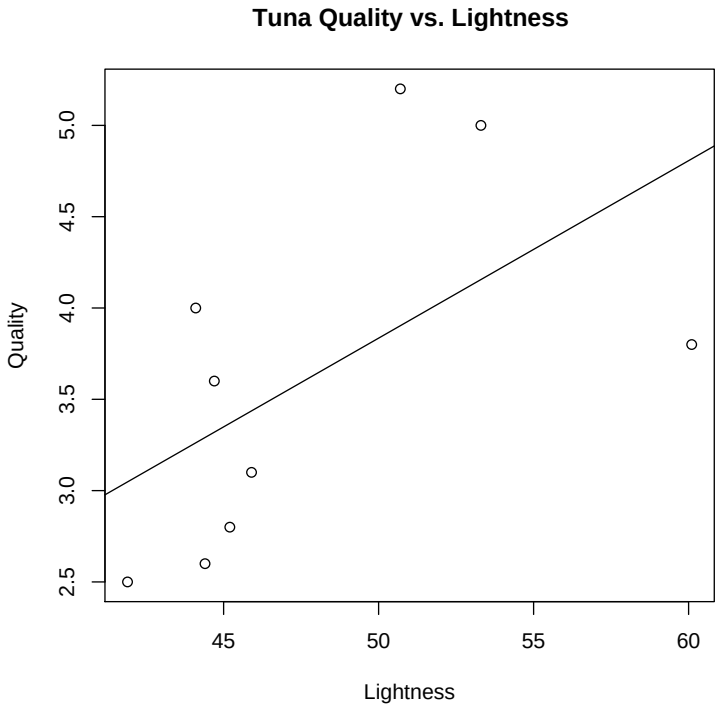

Figure 1: Tuna data and best-fitting (least-squares) regression line.

2.1 Pearson's r

Now I compute the correlation (i.e., Pearson's r). The function `cor` merely returns Pearson's r, whereas `cor.test` returns r, a p-value for the null hypothesis that $r = 0$, and a confidence interval. The p-value is based on the assumption that r follows t distribution with $n - 2$ degrees of freedom. The confidence interval is based on Fishers r-to-Z transformation.

```
> # pearson's r:
> (r <- cor(tuna$lightness,tuna$quality,method="pearson") )

[1] 0.5711816

> # compare r^2 to multiple R-squared from regression:
> r^2

[1] 0.3262484

> summary(tuna.lm)$r.squared # multiple R-squared

[1] 0.3262484

> cor.test(tuna$lightness,tuna$quality,method="pearson") # r plus confidence interval and p-value
```

```
        Pearson's product-moment correlation

data:  tuna$lightness and tuna$quality
t = 1.8411, df = 7, p-value = 0.1082
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1497426  0.8955795
sample estimates:
      cor
0.5711816
```

The default method for `cor` and `cor.test` is `pearson`, so calling the functions without specifying the method would yield the same results.

Wilcox's `pcorb` function uses a modified percentile bootstrap to compute 95% confidence intervals for r which is not based on any distributional assumption. Note that the default number of bootstrap iterations, 599, cannot be changed:

```
> source(url("http://www-rcf.usc.edu/~rwilcox/Rallfun-v9_2"))
> pcorb(tuna$lightness,tuna$quality)

$r
[1] 0.5711816

$ci
[1] 0.1440654 0.9653256
```

2.1.1 bootstrapping *pairs* of scores

In a correlational study, each sampling unit – e.g., subject – selected from the population brings a pair of scores, and our bootstrap procedure must preserve this structure in our data. Therefore, bootstrapping methods applied to correlations generally select *pairs* of scores randomly with replacement from the original sample.

2.2 Ranked-based Correlations

This section describes two well-known correlations that are based on *ranked* data. Because they are based on ranks, rather than raw scores, they both are more robust than r. Specifically, they tend to be less influenced by outliers in X, ignoring Y, and by outliers in Y, ignoring X. However, multivariate outliers – i.e., the right combination of outliers on X *and* Y – can strongly alter these correlations.

Spearman's rho (ρ) is Pearson's r calculated on two sets of ranked data. In other words, Spearman's ρ treats the ranks as the raw data. `cor.test` calculates ρ and returns a p-value, but does not return a confidence interval.

```
> cor.test(tuna$lightness,tuna$quality,method="spearman")

        Spearman's rank correlation rho

data:  tuna$lightness and tuna$quality
S = 48, p-value = 0.0968
alternative hypothesis: true rho is not equal to 0
sample estimates:
rho
0.6
```

Wilcox's `corb` function can be used to calculate a confidence interval based on the percentile bootstrap. Note the inclusion of `spear` in the function:

```
> corb(tuna$lightness,tuna$quality,spear)

$cor.ci
[1] -0.1559633  0.9824561

$p.value
[1] 0.1001669

$cor.est
[1] 0.6
```

Kendall's tau is based on the number of inversions/reversals of ranked scores:

```
> cor.test(tuna$lightness,tuna$quality,method="kendall")

        Kendall's rank correlation tau

data:  tuna$lightness and tuna$quality
T = 26, p-value = 0.1194
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.4444444

> corb(tuna$lightness,tuna$quality,tau) # bootstrapped confidence intervals:

$cor.ci
[1] -0.08333333  0.77777778

$p.value
[1] 0.09348915

$cor.est
[1] 0.4444444
```

2.3 Percentage-bend Correlation

The percentage-bend correlation, r_{pb} is based on the percentage-bend M-estimator of location.

```
> pbcor(tuna$lightness,tuna$quality)

$cor
[1] 0.7556884

$test
[1] 3.052784

$siglevel
[1] 0.01851051
```

The p-value for r_{pb}, which is used to evaluate the null hypothesis that $r_{pb} = 0$, is based on the statistic

$$T_{pb} = r_{pb}\sqrt{\frac{n-2}{1-r_{pb}^2}}$$

which is assumed to follow a t distribution with $n-2$ degrees of freedom. The function `corb` can be used to calculate percentile bootstrap confidence that do not require the distributional assumption:

```
> corb(tuna$lightness,tuna$quality,pbcor)

$cor.ci
[1] -0.1096950  0.9708319

$p.value
[1] 0.08347245

$cor.est
[1] 0.7556884
```

3 Regression

Let's reprint the regression table:

```
> summary(tuna.lm)

Call:
lm(formula = quality ~ lightness, data = tuna)

Residuals:
    Min      1Q  Median      3Q     Max
-1.0165 -0.5685 -0.3365  0.7384  1.2970

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.02411    2.54020  -0.403    0.699
lightness    0.09718    0.05278   1.841    0.108

Residual standard error: 0.8673 on 7 degrees of freedom
Multiple R-squared: 0.3262,        Adjusted R-squared:  0.23
F-statistic:  3.39 on 1 and 7 DF,  p-value: 0.1082
```

The table lists the estimated intercept and slope (i.e., the regression coefficient for lightness) for the regression line, as well as their standard errors. The standard errors can be used to calculate 95% confidence intervals:

```
> df.error <- 7;
> alpha <- .05;
> t.low <- qt(alpha/2,df=df.error)
> t.high <- qt(1-alpha/2,df=df.error)
> ( ci.intercept <- c( -1.024-t.high*2.54, -1.024-t.low*2.54) )

[1] -7.030146  4.982146

> ( ci.slope <- c( 0.09718-t.high*0.05278, -1.024-t.low*0.05278) )
```

```
[1] -0.02762487 -0.89919513
```

Both intervals contain zero, so there is insufficient evidence to reject the null hypotheses that the intercept and slope are zero. The two-tailed t tests shown in the regression table lead to the same conclusion. It is important to note the procedure used to estimate the parameters of the line, as well as the standard errors of the parameters, assumes that the error associated with each value of Y is drawn from a normal distribution that has a mean of zero and a variance that is independent of X. If this assumption is false – for example, if the variance of the error distribution grows with increasing X – the confidence interval and p-value for each parameter may be misleading.

Wilcox's function `lsfitci` uses a modified percentile bootstrap procedure to estimate a 95% confidence interval for each predictor variable in a regression model:

```
> lsfitci(tuna$lightness,tuna$quality)

[1] "Taking bootstrap samples; please wait"
$intercept.ci
[1] -10.337685   2.503083

$slope.ci
           [,1]      [,2]
[1,] 0.01626645 0.3361389

$crit.level
[1] NA

$p.values
[1] NA
```

Note that the bootstrapped confidence interval for the slope does not include zero.

3.0.1 bootstrapping cases vs. residuals

The function `lsfitci` works in a manner that is similar to the way we computed bootstrapped confidence intervals for correlations: the sampling distributions of the regression parameters were estimated by sampling *cases* (i.e., the criterion (Y) and predictor (X) variables for each subject, or sampling unit) randomly with replacement from the original sample.

There is, however, another way of applying the bootstrap to regression models. Multiple regression models can be expressed as $Y_i = \hat{Y}_i + \epsilon_i$ where Y_i is the observed score for case i, $\hat{Y}_i$ is the predicted score for case i (where the prediction comes from the regression model), and ϵ_i is the residual for case i. In standard regression models, the residuals are assumed to be drawn randomly from a normal distribution that has a mean of zero and a variance, σ^2_ϵ, that is independent of the predictor variables. Note that the ϵ's are the only random component in the model. Another way of approaching the bootstrap would be to create bootstrapped samples of the *residuals*, add them to the $\hat{Y}_i$'s, and then recompute the regression line. This method – often referred to as bootstrapping the residuals – assumes that the statistical model in Eq 3.0.1 is correct. More specifically, it assumes that the residuals for all cases really are drawn from a *single*, though not necessarily normal, distribution. Note that the method of bootstrapping the cases does not make this assumption, and for that reason Wilcox [6] favors bootstrapping cases over bootstrapping residuals. These issues are discussed further in standard bootstrapping texts [1].

References

[1] Bradley Efron and Robert Tibshirani. *An introduction to the bootstrap*. Monographs on statistics and applied probability ; 57. Chapman and Hall, New York, 1993.

[2] Carl Gaspar, Allison B Sekuler, and Patrick J Bennett. Spatial frequency tuning of upright and inverted face identification. *Vision Res*, 48(28):2817–2826, 2008.

[3] Roger E Kirk. *Experimental design: procedures for the behavioral sciences*. Brooks/Cole, Pacific Grove, Calif., 3rd ed edition, 1995.

[4] Scott E Maxwell and Harold D Delaney. *Designing experiments and analyzing data: a model comparison perspective*. Lawrence Erlbaum Associates, Mahwah, N.J., 2nd ed edition, 2004.

[5] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria (http://www.R-project.org), 2007.

[6] Rand R. Wilcox. *Introduction to robust estimation and hypothesis testing*. Elsevier Academic Press, 2005.