One-way ANOVA and Contrast Analysis in R

Use Expectation data set. The experimenters' were the actual subjects of the study. They collected ratings of the apparent successfulness of people in pictures who were pre-selected for their average appearance. The experimenters were told prior to collecting data that the pictures were either high or low in their appearance of success, and were instructed to get good data, scientific data, or were given no such instruction. Each experimenter collected ratings from 18 randomly assigned respondents; a few subjects were deleted at random to produce an unbalanced design. (from RDocumentation for the car package)

1. Compute descriptive stats per group
   a. Several descriptive statistics: **tapply(outcome.variable, grouping.variable, stat.desc)**
      i. Stat.desc is in the pastecs package,
      ii. Output: number of valid cases, min, max, range, sum, median, mean, SE of the mean, value to add and subtract from mean to get the 95% CI, variance, standard deviation, and the variation coefficient [*SD divided by the mean*])
   b. Mean per group: **tapply(rating, instruction, mean, na.rm=TRUE)**
   c. Standard deviation per group **tapply(rating, instruction, sd, na.rm=TRUE)**
   d. Sample size per group: **table(instruction)**

2. Check for homogeneity of variance
   a. Output variances for each group of instruction: **tapply (rating, instruction, var, na.rm=TRUE)**
   b. Compute Bartlett's test: **bartlett.test (rating ~ instruction, data=mydata)**

3. One-way ANOVA with equal variances
   a. Specify the model: **Anova.1 <- aov(outcome ~ predictor, data=dataframe, na.action=na.exclude)**
      i. na.action=na.exclude will exclude all cases with missing data; not needed if you have complete data
   b. Output the source table: **summary(Anova.1)**
      i. Between groups variance: name of grouping variable
      ii. Within groups variance: Residuals

4. One-way ANOVA that does not assume equal variances (Welch)
   a. oneway.test(rating~instruction)
      i. There is not an associated post hoc test

5. Use figures to check other assumptions
   a. plot(Anova.1)
   b. Residuals vs Fitted: check for homogeneity of variance; there should not be a pattern or consistent change in residuals across the x-axis
   c. Normal Q-Q: points should be on the line
   d. Scale-location: similar to Residuals vs Fitted
   e. Residuals vs. Leverage: tries to identify points that might have more influence than others (and may be outliers); further investigate points with labels

6. Conduct post hoc tests
   a. Bonferroni: **pairwise.t.test (outcome, predictor, paired=FALSE, p.adjust.method= "method")**
      i. paired=FALSE because the data are independent
      ii. p.adjust.method: use to specify with correction to apply; one option is bonferroni
      iii. output: p-values

b. Tukey: use glht (general linear hypotheses) function from the multcomp package
   i. **newModel<-glht(Anova.1, linfct = mcp(predictor= “method”))**
   ii. Anova.l – model that has already been created
   iii. replace “method” with “Tukey” to get the Tukey test
   iv. to output results: **summary(newModel)**
      1. Estimate = difference between the means
   v. to output confidence intervals: **confint(newModel)**
   vi. to create plot of confidence intervals: **plot(confint(newModel))**

7. Compute effect sizes
   a. Eta-squared: Multiple R-squared from **summary.lm(Anova.1)**
   b. For pairs: use mes() function from the compute.es package
      i. mes(mean1, mean2, sd1, sd2, n1, n2)
      ii. output: estimate and confidence intervals for d, g, r, z, odds ratio, and log odds ratio

8. Power analysis with the pwr package
   a. pwr.anova.test(k = NULL, n = NULL, f = NULL, sig.level = 0.05, power = NULL)
      i. k=number of groups, n=number of observations per group, f=effect size similar to Cohen's d
   b. To determine sample size, provide values for all other elements
   c. Example: for 3 groups with effect size of .25 and power of .80 use: **pwr.anova.test(k = 3, n = NULL, f = .25, sig.level = 0.05, power = .80)**

9. Contrast analysis
   a. From the gmodels package
   b. **fit.contrast (Anova.1, predictor.variable, c(contrast.weights separated by commas), df=T)**