Lab 6

MANOVA with Distances: MRPP, perMANOVA

Multivariate ANOVA, which we covered in Lab 4 works by rotating the coordinate system so that groups are optimally separated. This lab covers a clever distance-based alternative. Using distances reduces multiple response variables into a quasi-univariate response metric, and that means you are not running out of degrees of freedom easily. This is the biggest advantage of perMANOVA over MANOVA (see section 6.4. in this lab).

MRPP and perMANOVA are similar to univariate ANOVA in another way. ANOVA calculates a signal-to-noise ratio (F-value) as variance between groups (signal) divided by variance within groups (noise). The bigger the F-value, the smaller the p-value (calculated as area under the curve of an F-distribution). For more details, see the video: http://tinyurl.com/mv690/v/inferential.

This also works with distances: MRPP and perMANOVA rely on a signal-to-noise ratio from among group distances (signal) divided by within group distances (noise), P-values can then be found in a surprisingly straight-forward way. See video: http://tinyurl.com/mv690/v/permanova.

6.1. Multiple Response Permutation Procedure (MRPP)

- Download the dataset "reclaim.csv" from http://tinyurl.com/mv690/lab/6/data. This data represents species counts on reclamation sites that received different soil amendments and fertilizers to enhance the revegetation success:

    ```
    rec = read.csv("reclaim.csv")
    View(rec)
    ```

- For analysis, we have to break the dataset up into separate objects for treatments, which must be defined as a "factor" or class variable, and species frequency measurements, which must be defined as a matrix that contains only numbers:

    ```
    soils = as.factor(rec[,1])
    fertilizer = as.factor(rec[,2])
    measurements = as.matrix(rec[,3:32])
    ```

- If you look at histograms of the species data, you see typical zero-inflated community ecology data. Thus, a Bray-Curtis distance is a good distance metric for this study. If the variables were normally distributed, then a Euclidean distance of standardized variables should be used, or a Mahalanobis distance for highly autocorrelated variables (as discussed in Lab 5):

    ```
    hist(rec$Achmil)
    hist(rec$Brarut)
    hist(rec$Calcus)
    hist(rec$Elepal)
    ```

- So, let's go ahead with an MRPP using the Bray-Curtis distance. Unfortunately, MRPP can only analyze one factor at a time, so you have to separately test the effect of soils and fertilizer treatments:

    ```
    library(vegan)
    mrpp(measurements, group=soils, distance="bray")
    mrpp(measurements, group=fertilizer, distance="bray")
    ```

6.2. Permutational Multivariate Analysis of Variance (perMANOVA)

- This is a newer and better method, based on the same general principle as MRPP, but allowing for multiple factors and interactions. Tha adonis2() function of the latest "vegan" package implements a permutational Multivariate Analysis of Variance that will give you an output exactly equivalent to a MANOVA or to a univariate ANOVA:

```
adonis2(measurements~soils, distance="bray")
adonis2(measurements~fertilizer, distance="bray")
adonis2(measurements~fertilizer*soils, distance="bray")
adonis2(measurements~fertilizer+soils, distance="bray")
```

- Note that your p-value will be limited by the number of permutations. The default is 999 permutations, so the smallest possible p-value you can get is 0.001. If you set the permutations higher, e.g. 9999, you will get smaller p-values:

```
adonis2(measurements~fertilizer+soils, distance="bray",
        permutations=9999)
```

- You can follow this up with pairwise comparisons adjusted for multiple inference just like a regular ANOVA, using the "RVAideMemoire" package:

```
adonis2(measurements~soils, distance="bray")
library(RVAideMemoire)
library(ecodist)
bray=bcdist(measurements)
pairwise.perm.manova(bray, soils, p.method="holm", nperm=9999)
```

- The matrix of p-values shows that random chance is an unlikely explanation for the difference observed between LFH and PEAT (p=0.003), and for the difference observed between MIX and PEAT (p=0.001). In contrast, it is quite likely that a difference as big or bigger than the observed distance between the groups LFH and MIX could arise due to random chance alone (one out of ~3 times, if you repeated an experiment like this over and over).

6.3. Follow-up with ordination

Like with an ANOVA, you definitely want a visual to go with your MANOVA or perMANOVA results. In ANOVA you would show a bar chart or box plot for a single response variable. The equivalent for MANOVA would be an ordination.

As an informal assignment to test your skills, use the Reclaim dataset to generate an NMDS as covered in Lab 5 (but skip it for the moment and do this later). Color and label your plots by treatment and add ellipses. Can you visually confirm the significant and non-significant treatment effects that you inferred with PERMANOVA?
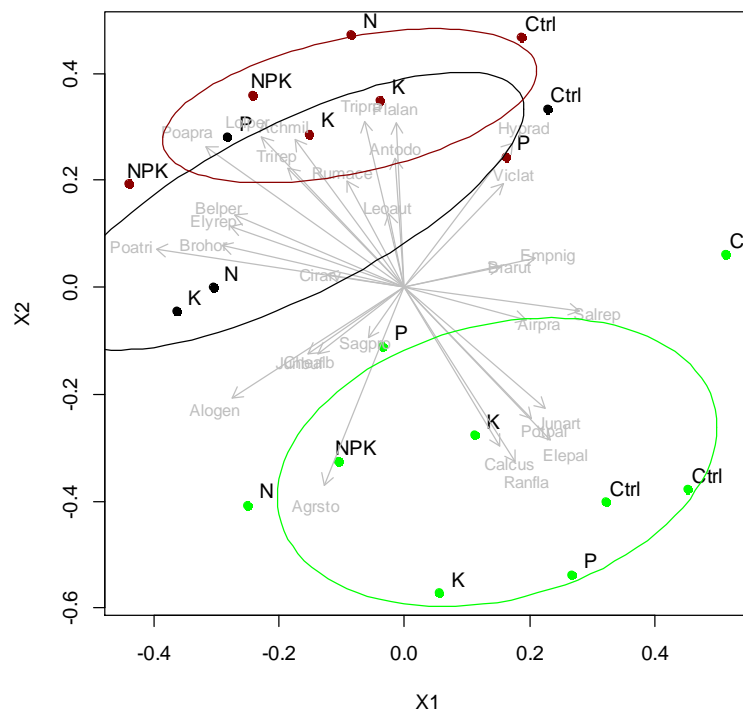
6.4. Why MANOVA is generally less useful than perMANOVA

- Standard multivariate analysis of variance has a problem that we can explore here. Let's assume that the data was normally distributed, try to run the code that we covered in Lab 3:

```
output = manova(measurements~soils)
summary(output, test="Wilks")
```

You see that it doesn't work here, because you are running out of degrees of freedom. The data table is wider than long. You have too many variables and too few observations to do any valid statistical tests. Converting all species measurements into a distance matrix elegantly solves this common problem.

- MANOVA does work with fewer variables. Try the code with just the first 5 species:

```
output = manova(measurements[,1:5]~soils)
summary(output,test="Wilks")
```

6.5. Univariate follow-up: ANOVA or permutational ANOVA

Once you know that you have a significant multivariate response, you may be interested in following-up with univariate statistics. MANOVA will, for example, tell you that you have a significant species community response, and a follow-up ordination will provide a visual how the community response looks like. The figure above is really a nice result how the community composition shifts as a function of treatments, but it is still a somewhat vague result in terms of actionable recommendations, or even for a good statement in an abstract that would be cited.

For stronger statements, go univariate. You may have a particular variable of special interest that you want to investigate in more detail. To give some examples, you may be particularly interested in (1) a specific endangered species, (2) the proportion of IUCN red-listed species, (3) the overall abundance of lichen, (4) an invasive species of concern, (5) a particular concerning pollutant, (6) overall heavy metal concentration, (7) species diversity as measured by Shannon's index, (8) overall species richness, (9) overall tree cover (10) overall re-vegetation success …

You see, there are endless possibilities to generate univariate indices or select important univariate variables from a larger multivariate dataset. This usually adds much value to your analysis. As a rule of thumb, start with multivariate ordinations: this is very useful to make sure you don't miss anything important. But when it comes to effectively conveying important scientific findings or delivering practical advice, a follow-up with univariate indices, univariate summaries, or a human-curated selection of specific variables is essential.

- For our re-vegetation experiment, we may want to ask: what is the overall re-vegetation success (across all species) for different soil amendments. This is a univariate problem. Let's create a univariate dataset:

```
dat1 = read.csv("reclaim.csv")
data.frame(names(dat1))
dat2=cbind(dat1[,1:2],FREQ=rowSums(dat1[,3:32]))
View(dat2)
attach(dat2)

boxplot(FREQ~SOIL)
boxplot(FREQ~FERT)
```

- We can see that LFH (a standard top-soil amendment) and MIX (a mix of LFH and PEAT) yield better results than PEAT alone, while the fertilizer data shows some effects that seem to

make sense (NPK yields the highest, CTRL (=no fertilizer) the lowest number of plants per hectare).

- If the distributions appear reasonably normal as in this boxplot, we may test for statistical significance of treatment effects with an ANOVA. Otherwise, you can use permutational ANOVA if you wish, but it pretty much always yields a near identical result. Compare:

```
anova(lm(FREQ~SOIL))

library(lmPerm)
anova(lmp(FREQ~SOIL, seqs=T))
```

A standard follow-up to a significant ANOVA effect (SOIL) are pairwise comparisons or contrasts of different treatment levels. If you have a lot of comparisons, you can use the compact matrix from the pwpm() function, which reports the p-values in the upper right, the means in the diagonal and the differences in the lower left:

```
library(emmeans)
out1 = lm(FREQ~SOIL)
out2 = emmeans(out1, ~SOIL, level=0.95)
anova(out1)
summary(out2)
pairs(out2, adjust="sidak")
pwpm(out2, adjust="sidak")
```

Note that you need to carry out an adjustment for multiple inference (using Dunn–Šidák adjustment method here). If you are unfamiliar with adjustments for multiple inference, check: http://tinyurl.com/rr480/videos/pairwisetests.

- Normally, you are not interested in all pairwise comparisons. For example, you may simply want to test a control (i.e., a null or business-as-usual treatment) versus alternative treatments. Let's say LFH is our control treatment that we cmpare to the others:

```
contrast(out2, "trt.vs.ctrl", ref="LFH", adjust="sidak")
```

Even for the switch from two to three comparisons the statistical power increases: the p-value for PEAT vs. LFH was reduced from 0.018 to 0.012. This is because the adjustment for multiple inference is only carried out for the comparisons you request. If you have many treatment levels, the increase in power to detect differences can be very large.

6.6. Single-factor effect size statistics

The above standard statistical null-hypothesis testing is usually not that interesting or useful. In most cases, this type of testing does not answer any relevant question. Real questions usually revolve around *superiority*, *inferiority*, *non-superiority*, or *non-inferiority* of treatments, as well as treatments meeting economic or regulatory *minimum requirements*.

The only question that standard statistical null-hypothesis testing answers is: "*Could the observed difference between treatments be due to random chance?*". The answer is always "*Yes, with a probability of p*". This can establish that there is a difference >0, but nothing else.

Effect size statistics tell you the *practical or scientific relevance* of a research outcome. It answers the question if the effect is large enough (or, notably, small enough) to be meaningful in the context of real world applications. Alternatively, quantifying the effect size is usually also essential to testing a scientific hypothesis (which is not the same as a statistical hypothesis).

- In our re-vetation experiement, ***minimum requirements*** for treatment outcomes are a very likely context: Perhaps government regulations require a minimum of 30 plants/m² for re-vegetation of mine sites, or the mining company wants at least a 30 plants/m² re-vegetation success for to justify a large investment into ecological restoration.

- This is actually a real dataset from an experiment supporting a re-vegetation of a mine site. The real objective that the company had was to reduce their costs and still meet minimum governmental re-vegetation requirements. LFH is a soil amendment that uses top-soil saved before mining started (this is the normal practice and therefore the control treatment). PEAT is a much cheaper amendment, and MIX is a 1:1 mix of topsoil and peat, also reducing cost.

- Let's see what we can recommend as applied scientists with our experiment. We'll do a one sided test with the alternative hypothesis that the treatment mean is greater than 30:

  ```
  test(out2, null=30, side=">", adjust="sidak")
  ```

- Let's also visualize the results. The standard graphical representation for effect size statistics are horizontal dot-charts with 95% confidence intervals:

  ```
  library(ggplot2)
  ggplot(as.data.frame(out2), aes(x=SOIL, y=emmean)) +
  geom_pointrange(aes(ymin=lower.CL, ymax=upper.CL)) +
  geom_hline(yintercept=30, linetype="dashed", color = "red") +
  xlab("Soil amendment") + ylab("Plants / m²") + coord_flip()
  ```
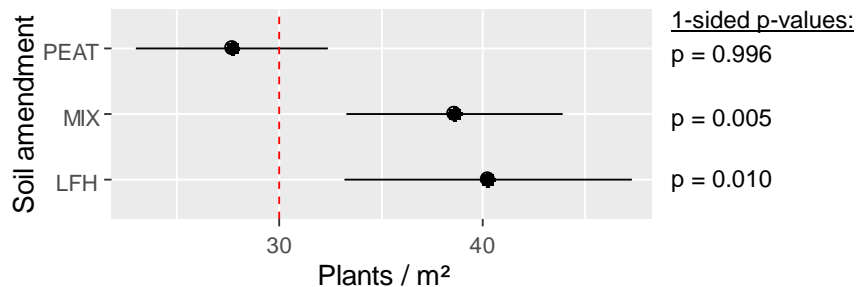

  The confidence in the success of your treatments is 1-p. Therefore you can conclude:
  "*LFH meets the 30 plans/m² requirement with 99% confidence (H_0: $\mu \leq 30$ with Dunn–Šidák correction for multiple inference). The cheaper MIX amendment meets the threshold as well with an even higher (99.5%) confidence, and is therefore the preferred option. The PEAT treatment will not meet the re-vegetation requirement.*"

Contrast this to what most people would report: "*I found a significant difference between PEAT and LFH and between PEAT and MIX ($\alpha = 0.05$)*", but that has no meaning. Testing the default null hypothesis (H_0: $\mu_1 = \mu_2$) just does not answer the question. Further, non-significance for LFH vs. MIX just means that the observed difference could be due to random chance (it does not mean that there is no difference). This is also profoundly unhelpful. With the use of effect size statistics, you instead generated a valuable and publishable recommendation, fully supported by meaningful and rigorous statistics. Do that instead!

Examples for other valuable statements of this kind may be "*This conservation strategy maintains at least 90% of biodiversity*" or "*This preventative measure keeps pollution below 20 ppm*" or "*This silvicultural treatment improves forest growth by at least 25%*". That's your ticket!
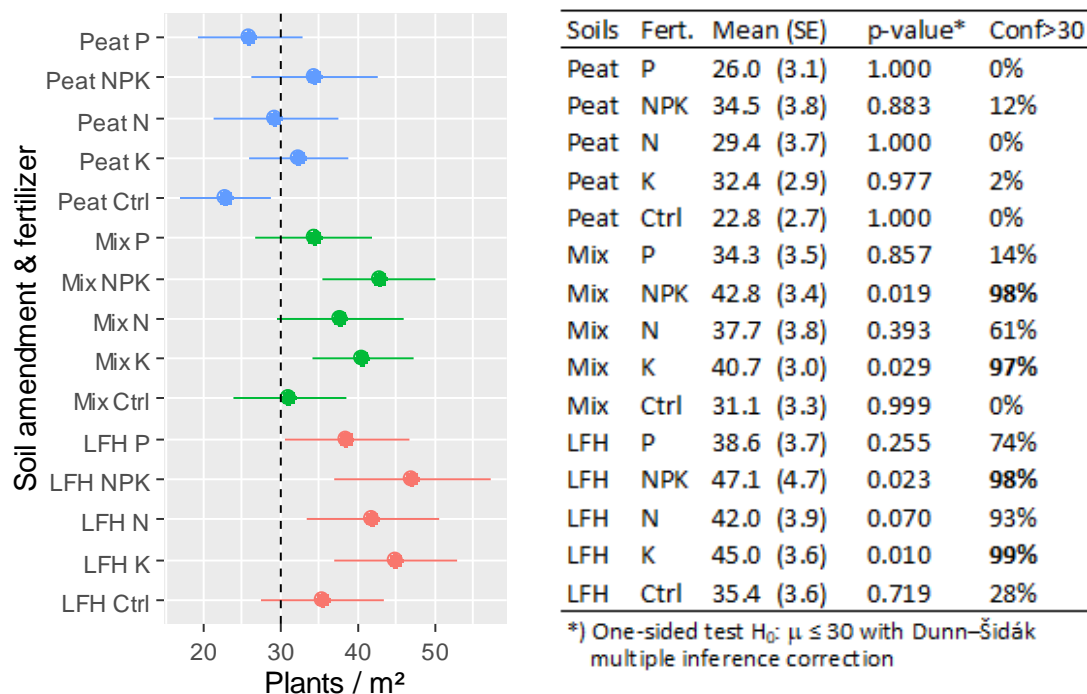
6.7. Multi-factor effect size statistics

Now let's expand the analysis and visualization to our multi-factor experiment, which also includes a second factor: different fertilizer treatments. Everything is done in the same way. We use `SOIL+FERT` here, because it is a small experiment and we don't have enough replications to estimate all interactions, but you can do `SOIL*FERT` if you have sufficient replications in your own data. We will also write the mean estimates, standard errors, confidence intervals and p-values out to a CSV file, so you have the option to present the results as a table as well, (typically a more actionable and citable result).

```
out1 = lm(FREQ~SOIL+FERT)
out2 = emmeans(out1, ~SOIL+FERT, level=0.95)
out2; write.csv(data.frame(out2), "Means & SE.csv")

out3 = test(out2, null=30, side=">", adjust="sidak")
out3; write.csv(out3, "Effect size statistics.csv")

library(ggplot2)
ggplot(as.data.frame(out2), aes(x=paste(SOIL,FERT),
        y=emmean, group=SOIL, color=SOIL)) +
geom_pointrange(aes(ymin=lower.CL, ymax=upper.CL)) +
geom_hline(yintercept=30, linetype="dashed", color = "black") +
xlab("Soil amendment & fertilizer") + ylab("Plants / m²") + coord_flip()
```


Soils	Fert.	Mean (SE)	p-value*	Conf>30
Peat	P	26.0 (3.1)	1.000	0%
Peat	NPK	34.5 (3.8)	0.883	12%
Peat	N	29.4 (3.7)	1.000	0%
Peat	K	32.4 (2.9)	0.977	2%
Peat	Ctrl	22.8 (2.7)	1.000	0%
Mix	P	34.3 (3.5)	0.857	14%
Mix	NPK	42.8 (3.4)	0.019	**98%**
Mix	N	37.7 (3.8)	0.393	61%
Mix	K	40.7 (3.0)	0.029	**97%**
Mix	Ctrl	31.1 (3.3)	0.999	0%
LFH	P	38.6 (3.7)	0.255	74%
LFH	NPK	47.1 (4.7)	0.023	**98%**
LFH	N	42.0 (3.9)	0.070	93%
LFH	K	45.0 (3.6)	0.010	**99%**
LFH	Ctrl	35.4 (3.6)	0.719	28%

*) One-sided test H₀: $\mu \le 30$ with Dunn–Šidák multiple inference correction
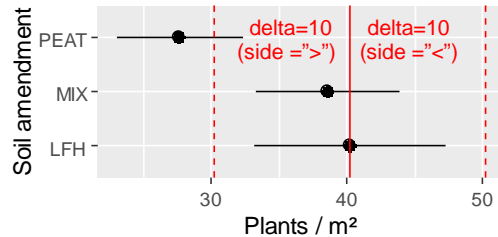
Either a figure or a table is appropriate for the presentation of results. If you use a figure, then you can report the statistics in your narrative: "*We can achieve ≥ 30 plants/m² with at least 95% certainty as follows: soil amendment "LFH" or "MIX" plus fertilizer K or NPK (based on one-sided tests with H₀: $\mu \le 30$ and a Dunn–Šidák correction for multiple inference). Since the MIX amendment is considerably less expensive, and potassium is a limiting factor, our recommendation is: MIX amendment with K fertilizer*".

Note 1: You might expect that if the 95% confidence interval just touches the dashed line (e.g., LFH-P), then the p-value should be 0.05 and the confidence of exceeding 30 plants/m² should be 95%. However, with adjustment for multiple inferences, your actual confidence is lower (74% for LFH-P). This is as it should be. The adjustment protects against false conclusions across multiple comparisons. The concept is explained here: http://tinyurl.com/rr480/videos/pairwisetests.

6.8. Tests of superiority, inferiority, non-superiority, non-inferiority

If you don't have regulatory or economically motivated fixed minimum requirements, you can also test *relative to another treatment or a control*. However, rather than testing for a non-zero difference, you specify a practically meaningful difference (delta value), and then test if a treatment is (1) larger, (2) not larger, (3) smaller, or (4) not smaller than the control, by a practically meaningful effect size. Which of those 4 questions is applicable depends 100% *on the context* of the research!

- A test of **non-inferiority** is, in fact, exactly what our original researchers needed: they wanted to know if the cheaper PEAT and MIX amendments are non-inferior in re-vegetation outcomes to the more expensive LFH amendment. We always have to define what we practically mean by non-inferior: Let's say we tolerate no more than 10 plants/m² less than LFH. This test makes most sense with contrast() outputs that define your reference treatment to which the delta value is applied, but it technically also works with pairs() outputs. For simplicity, let's go back to our single factor model:


  ```
  out1 = lm(FREQ~SOIL)
  out2 = emmeans(out1, ~SOIL, level=0.95)
  out3 = contrast(out2, "trt.vs.ctrl", ref="LFH")
  test(out3, delta=10, side=">", adjust="sidak")  # non-inferior/inferior
  ```

  The answer is: "*MIX (p=0.0622) is no more than 10 plants/m² worse than LFH with 94% certainty (1-p). For PEAT (p=0.9302) the certainty of non-inferiority is only 7%, so we conclude that PEAT is, in fact, inferior to MIX with respect to supporting re-vegetation.*"

- Now let's do a test of **non-superiority**. For this test to make sense with our data, we have to change the research context: let's pretend we work on invasive species control, and our dependent variable represents the frequency of invasive species. LFH is the expensive reference treatment, and we are looking for cheaper PEAT and MIX alternatives hoping they do the job sufficiently well:

  ```
  test(out3, delta=10, side="<", adjust="sidak")  # non-superior/superior
  ```

  The results is: "*We can be sure that neither MIX nor PEAT will increase the abundance of invasives by more than 10 plants/m² compared to the LFH control (p-values of 0.0125 and <.0001, respectively).*"

- A test of **inferiority** is just the inverse of the test of non-inferiority above. I.e., you simply take 1-p to get the p-value for an inferiority test. We again change the research context to make sense: We are still working with invasives, but now we have a cheap LFH as status-quo reference, and we test more expensive PEAT and MIX invasive control measures. Therefore, we want to be sure that they reduce invasive species by a practically relevant amount: at least 10 plants/m² compared to LFH (i.e. we want fewer invasive species = test of inferiority)

  The results is: "*We can be 93% sure that PEAT will <u>decrease</u> the abundance of invasives by at least 10 plants/m² compared to the LFH control (p = 1–0.9302 = 0.0698). Mix offers no advantage over LFH (p = 0.9378)*"

- Similarly, you do a test of **superiority** as the inverse (1-p) of the test of non-superiority. Research context change: back to the re-vegetation example, but now we have a cheap LFH amendment as status-quo reference, and we want to improve re-vegetation with more expensive PEAT and MIX amendments. Therefore, we want to be sure that they increase re-vegetation success by a practically relevant amount: at least 10 plants/m² more, compared to LFH (i.e. test of superiority)

  The results is: "*We found no evidence that either MIX nor PEAT will improve re-vegetation by at least 10 plants/m² compared to the LFH (p-values of 0.9875 and >0.9999, respectively). In fact, re-vegetation success was reduced. Therefore, LFH remains the best choice.*"

- All these tests can also be done for multi-factor designs

```
out1 = lm(FREQ~SOIL+FERT)
out2 = emmeans(out1, ~SOIL+FERT, level=0.95)
out3 = contrast(out2, "trt.vs.ctrl", ref="LFH CTRL")
test(out3, delta=10, side=">", adjust="sidak") # non-inferior/inferior
test(out3, delta=10, side="<", adjust="sidak") # non-superior/superior
```

- For clarity, when describing your results, you need a visual with the control treatment (solid line) and the delta (dashed line) indicated. This would be for a non-inferiority test, matching the original objective of the experimenters:
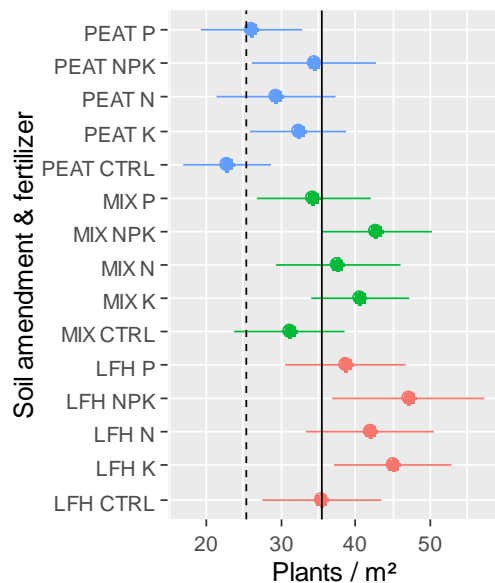
At a 95% confidence level, the non-inferior treatments are:

LFH K (p = 0.0009)
LFH N (p = 0.0121)
LFH P (p = 0.0295)
LFH NPK (p = 0.0018)
MIX NPK (p = 0.0270)

With less stringent 90% confidence level, this additional treatment would also be judged non-inferior:

MIX K (p = 0.0644)

- However, we shot ourselves in the foot by including treatments, that we would not really consider as cheaper alternatives (i.e. all LFH plus fertilizer), wasting statistical power by asking a question without thinking first. Let's exclude those:

```
out3 = contrast(out2, "trt.vs.ctrl", ref="LFH CTRL",
      exclude=c("LFH P","LFH N","LFH K","LFH NPK"))
test(out3, delta=10, side=">", adjust="sidak")
```

Now, MIX with NPK (p = 0.0193) and MIX with K (p = 0.0465) pass the non-inferiority test at the 95% confidence level due to a less aggressive adjustment for multiple inference.

- Lastly, a note that you can do these effect-size statistics not just in conjunction with the linear model lm() function of the R base package. The emmenas library with its contrast and test functions will play ball with all kinds of other advanced models, such as mixed models and non-parametric models:

  o Mixed models: e.g. lmer, lmerMod, lme, gls (packages lme4, lmerTest, nlme)
  o Generalized linear models: e.g. glm, glm.nb, glmerMod (packages stats, Mass, lme4)
  o Generalized additive models: e.g.: gam, Gam, gamm, (packages: mgcv, gam)
  o Ordinal scale models: e.g.: clm, clmm, (package: ordinal)
  o Count models: e.g.: hurdle, zerionfl (package: pcsl)