

## MD1 - breed figure

```
knitr::opts_chunk$set(echo = TRUE)
```

```
library(tidyverse)
library(phytools)
library(here)
library(plotrix)
library(RColorBrewer)
```

### Preregistered data

#### Read data

```
prereg.data <- read.csv(here('data', 'md1_data_vienna_long.csv')) %>%
  filter(preregistered_data=='after_prereg') #only data that were collected after preregistration
```

#### Aggregate breed level data

```
breed.means <- prereg.data %>%
  filter(condition != "odour", condition != "w_2cp_vd") %>% #only test conditions
  group_by(subject_ID, breed, condition) %>%
  summarise(mean_resp = mean(response)) %>%
  ungroup() %>%
  group_by(condition, breed) %>%
  summarise(mean = mean(mean_resp, na.rm = T), sem = std.error(mean_resp, na.rm = T)) %>%
  pivot_wider(id_cols = breed, names_from = condition, values_from = c(mean, sem))
```

#### Load cladogram file and align names for plot

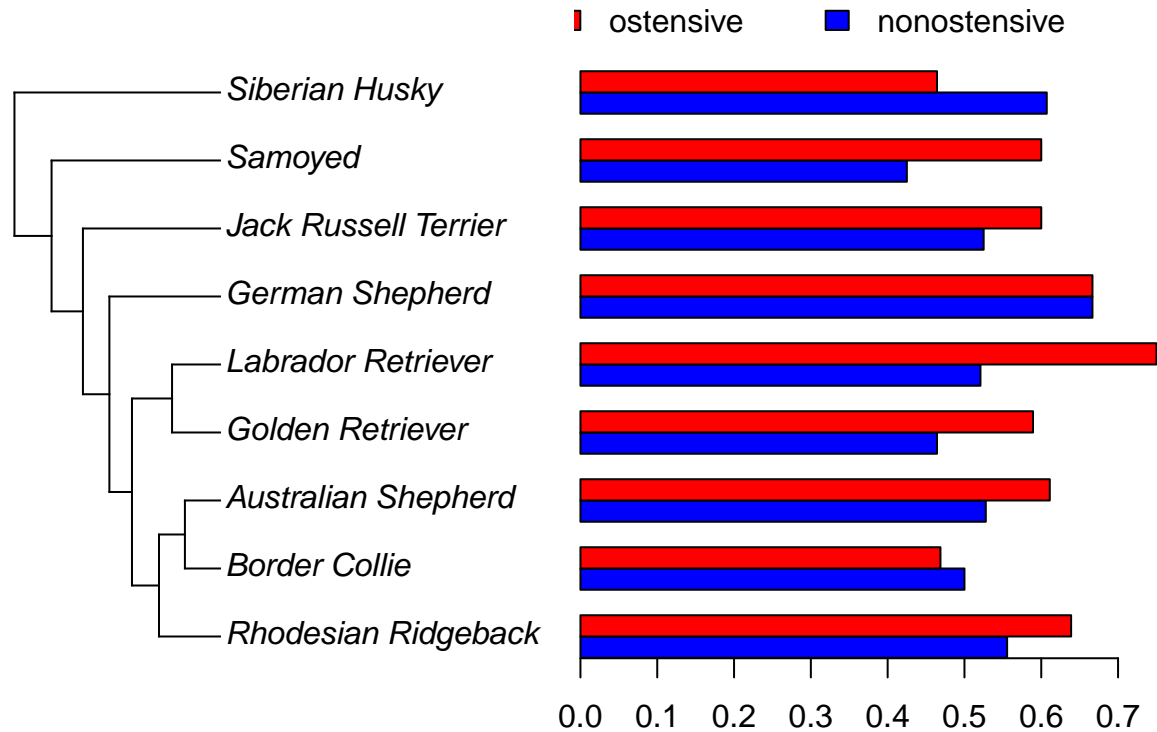
White Swiss Shepherd Dogs descend from white colored GSD and for purposes of the cladogram will be treated as GSD

```
load(here('data', 'dog_tree.RData')) #cladogram
my.names <- read_csv(here('data', 'tree_names.csv'))
# Obviously for some of the plot functions I can't have spaces in the names
my.names$tree.name <- gsub(" ", "_", my.names$tree.name)
dog.tree$tip.label <- gsub(" ", "_", dog.tree$tip.label)
breed.means$tip <- my.names$tree.name[match(breed.means$breed, my.names$data.name)]
to.drop <- c(setdiff(breed.means$tip, dog.tree$tip.label), setdiff(dog.tree$tip.label, breed.means$tip))
dog.tree <- drop.tip(dog.tree, to.drop)
dog.tree <- compute.brlen(dog.tree, 1)
breed.means <- breed.means[!breed.means$tip %in% to.drop,]
# align order
breed.means <- breed.means[match(dog.tree$tip.label, breed.means$tip),]
```

## Make plots on tree

Here's a barplot option. Getting error bars is not easy and i can spend some time trying to figure it out if people like this approach

```
plot.data <- breed.means %>% select(mean_non, mean_ost, tip) %>% column_to_rownames(var = "tip") %>% reshape_wider(varnames = c("non", "ost"))
dog.chronos <- chronos(dog.tree)
plotTree.barplot(dog.chronos, plot.data, args.barplot = list(beside=TRUE, space = c(0,1.2), col = c('blue', 'red')))
```



Or alternatively, here is a heatmap

```
my.colors = colorRampPalette(brewer.pal(9, "YlOrRd"))(500)
phylo.heatmap(dog.tree, plot.data, split = c(0.5, 0.3), fsize = c(1, 1, 0.75), colors = my.colors, grid = TRUE)
```

