

# Data & model output visualisation: ICCB 2015

Olivia Burge

23 July 2015

Overview for this afternoon's session:

## ① Introduction to ggplot

- ① Basic plots
- ② Changing linetype; colour; fill by factor
- ③ Facetting
- ④ Customising legends and labels

## ② Visualising correlations - ggpairs

- ③ Plotting model output - ggplot and coefplot2()
- ④ More if we have time: mapping (or ordinations)

Please either install (`install.packages("package")`) or update (same command) the following packages:

- ① `ggplot2`
- ② `ggthemes`
- ③ `wesanderson`
- ④ `GGally` (for `ggpairs`)
- ⑤ `coefplot2()` (\*\*)
- ⑥ `effects`
- ⑦ `vegan`
- ⑧ `gridExtra`

\*\* `coefplot2` will probably only work if you use:

```
install.packages("coefplot2", repos =
"http://www.math.mcmaster.ca/bolker/R",
type="source")
```

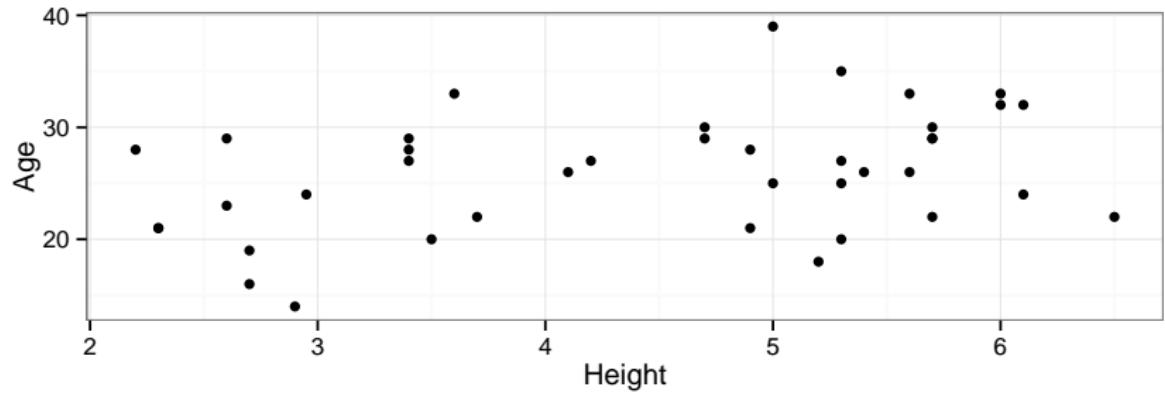
## The main graphics systems in R:

- ① Base graphics (built-in)
- ② Lattice (package)
- ③ ggplot2 (package)
- ④ ggviz (<http://ggvis.rstudio.com/ggplot2.html>)

```
getwd()  
# setwd()  
  
manuka_data <- read.csv("data/manuka.csv")  
weta_data <- read.csv("data/weta_270.csv")  
require(ggplot2)  
require(dplyr)  
theme_set(theme_bw())
```

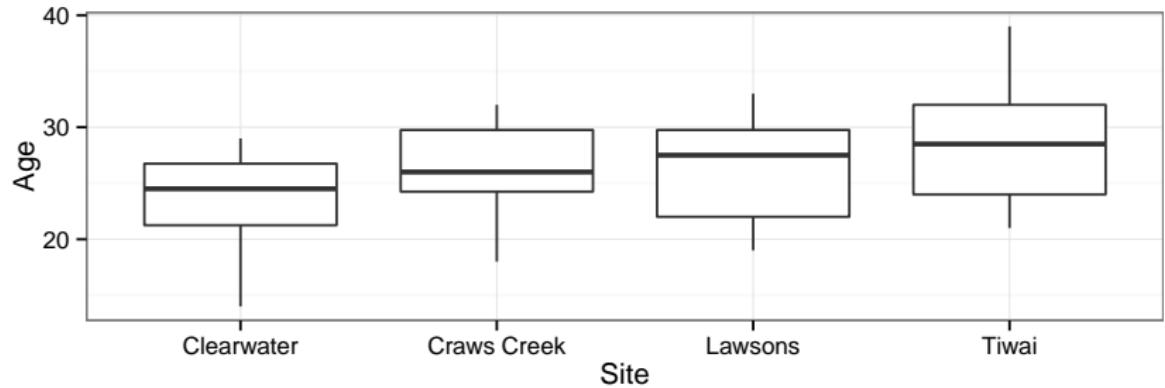
# Scatter plot

```
ggplot(data = manuka_data, aes(x = Height, y = Age)) +  
  geom_point()
```



# Boxplot

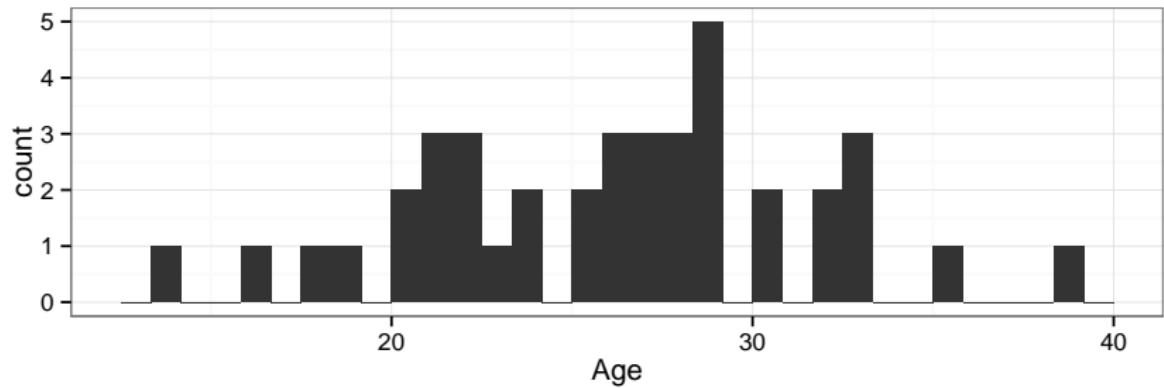
```
ggplot(data = manuka_data, aes(x = Site, y = Age)) +  
  geom_boxplot()
```



# Histogram

```
ggplot(data = manuka_data, aes(x = Age)) +  
  geom_histogram()
```

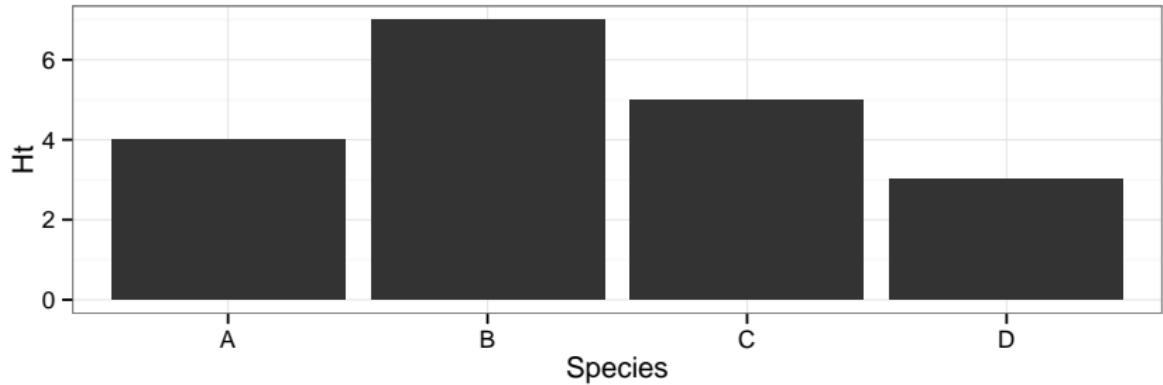
## stat\_bin: binwidth defaulted to range/30. Use 'binwidth'



# Bar Plot

```
#generate data
bar_data <- data.frame("Ht" = rpois(4, 5),
                       'Species'= c('A','B','C','D'))

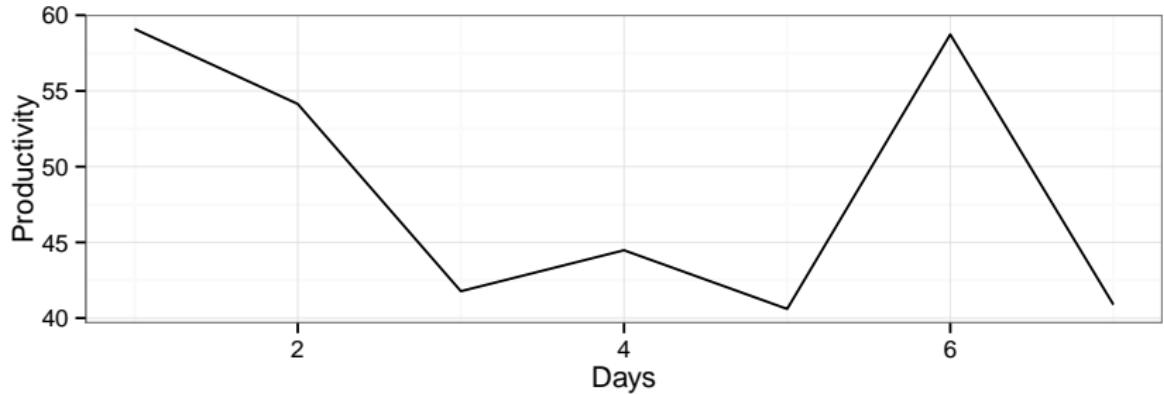
ggplot(data = bar_data, aes(x = Species, y = Ht)) +
  geom_bar(stat="identity")
```



## Line Plot

```
#generate data
line_data <- data.frame('Days'=1:7,
                        'Productivity'= rnorm(7, 50, 10))

ggplot(data = line_data,
       aes(x = Days, y = Productivity)) +
  geom_line()
```



## Basic plots: individual exercise

- ① Please create a scatter plot of soil moisture (MST) and conductivity (MV)
- ② Please create a boxplot of manuka height (Height) by site
- ③ Please create a bar graph of mean manuka age (Age) by site

## Basic plots: individual answers

```
# scatter plot MST and MV
ggplot(manuka_data, aes(x = MST, y = MV)) +
  geom_point()

# boxplot height and site
ggplot(manuka_data, aes(x = Site, y = Height)) +
  geom_boxplot()
```

## Bar graph answer & a quick note about 'stat ='

```
ggplot(manuka_data, aes(x = Age)) +  
  geom_bar(stat = "bin")
```

```
ggplot(manuka_data, aes(x = Site, y = Age)) +  
  geom_bar(stat = "identity")
```

```
ggplot(manuka_data, aes(x = Site, y = Age)) +  
  stat_summary(fun.y = "mean", geom = "bar")
```

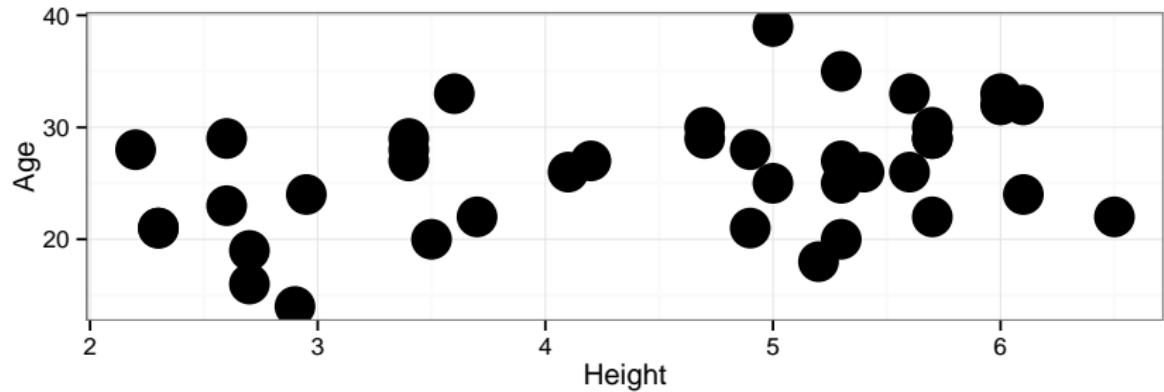
## Bar graph answer & a quick note about 'stat ='

```
manuka_means <- manuka_data %>%
  group_by(Site) %>%
  summarise(mean_age = mean(Age),
            sd_age = sd(Age))

ggplot(manuka_means, aes(x = Site, y = mean_age)) +
  geom_bar(stat = "identity",
            fill = "white",
            colour = "black") +
  geom_errorbar(aes(ymin = mean_age - sd_age,
                     ymax = mean_age + sd_age),
                width = 0.2, colour = "cornflowerblue") +
  scale_y_continuous(expand=c(0, 0),
                     limits = c(0, 35)) +
  theme_classic()
```

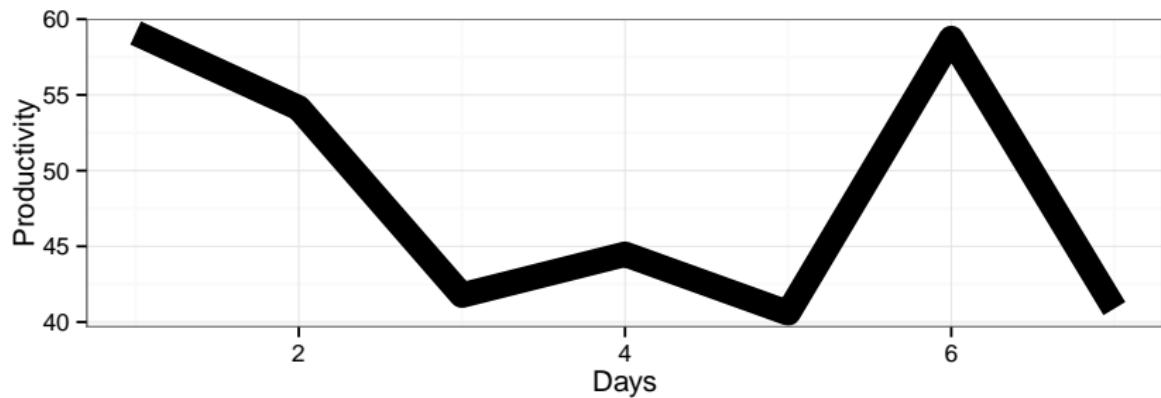
# Changing size of geoms (1)

```
ggplot(data = manuka_data, aes(x = Height, y = Age)) +  
  geom_point(size = 8)
```



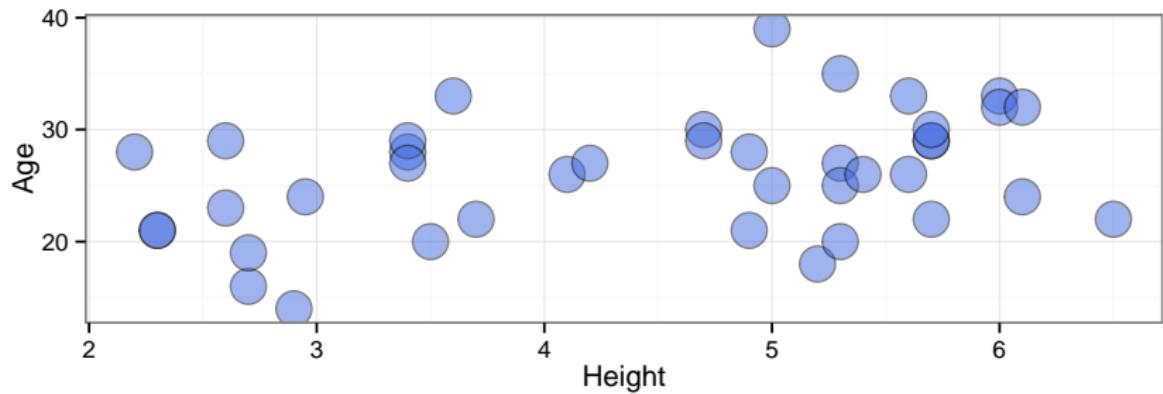
## Changing size of geoms (2)

```
ggplot(data = line_data,  
       aes(x = Days, y = Productivity)) +  
  geom_line(size = 5)
```



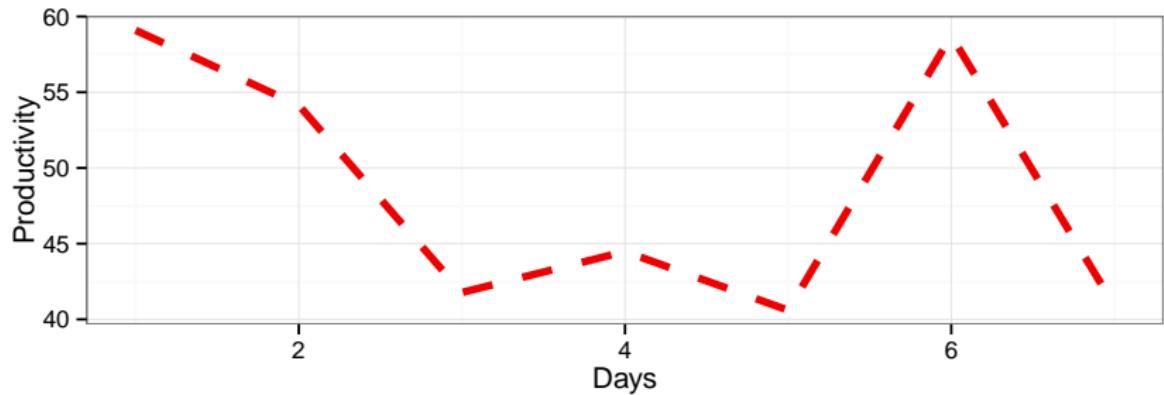
## Changing other variables of geoms (1)

```
ggplot(data = manuka_data, aes(x = Height, y = Age)) +  
  geom_point(size = 7, alpha = 0.5,  
             colour = "black", fill = "royalblue",  
             shape = 21)
```



## Changing other variables of geoms (2)

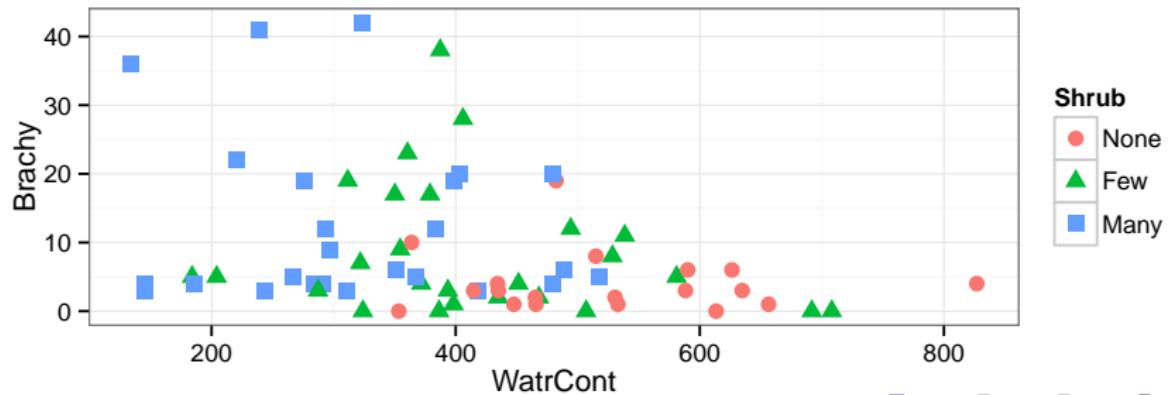
```
ggplot(data = line_data,  
       aes(x = Days, y = Productivity)) +  
  geom_line(size = 1.5, linetype = "dashed",  
            colour = "red2")
```



# Changing geoms by factor (1)

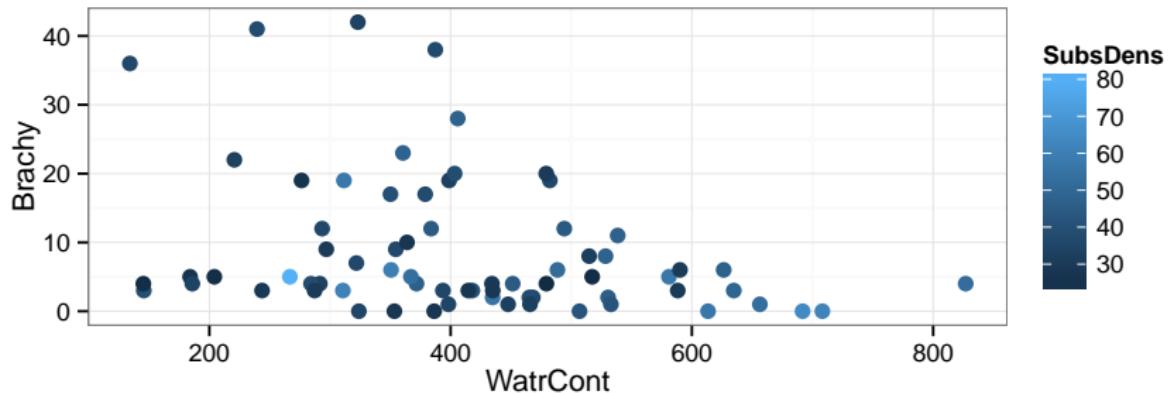
```
require(vegan)
data(mite)
data(mite.env)
mite_all <- cbind(mite, mite.env)

ggplot(data = mite_all, aes(y = Brachy, x = WatrCont)) +
  geom_point(aes(colour = Shrub, shape = Shrub),
             size = 3)
```



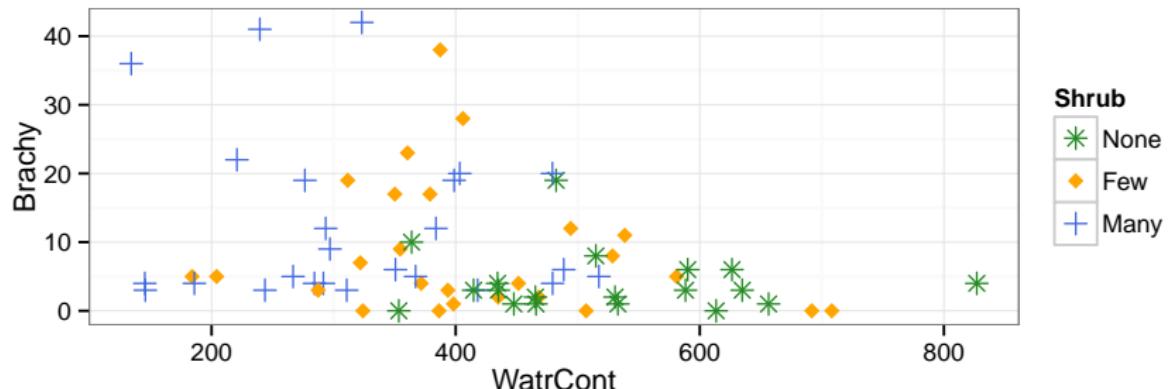
## Changing geoms by factor (2)

```
ggplot(data = mite_all, aes(y = Brachy, x = WatrCont)) +  
  geom_point(aes(colour = SubsDens), size = 3)
```



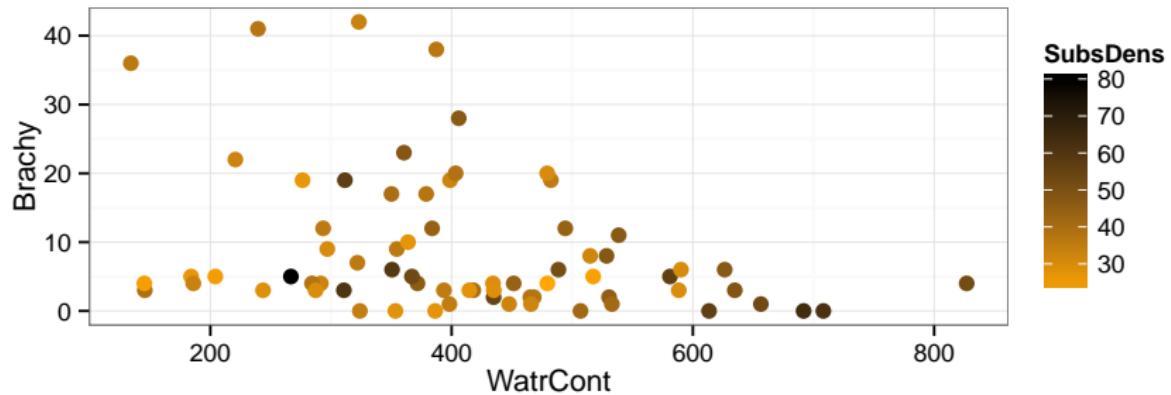
## Changing geoms by factor (3)

```
ggplot(data = mite_all, aes(y = Brachy, x = WatrCont)) +  
  geom_point(aes(colour = Shrub, shape = Shrub),  
             size = 3) +  
  scale_colour_manual(values = c("forestgreen",  
                                "orange",  
                                "royalblue")) +  
  scale_shape_manual(values = c(8, 18, 3))
```



## Changing geoms by factor (4)

```
ggplot(data = mite_all, aes(y = Brachy, x = WatrCont)) +  
  geom_point(aes(colour = SubsDens), size = 3) +  
  scale_colour_gradient(low = "orange", high = "black")
```



- ① Please create a histogram of weta abundance (`total.weta`), with a black outline and grey fill.
- ② Please create a scatter plot the abundance of NCOR mites (“NCOR”) against substrate density (`SubsDens`). Please colour by topo type (“Topo”) and change the size by shrub type (“Shrub”). Please use the classic theme (`theme_classic()`).

- histogram of weta abundance (total.weta) with black outline and grey fill

```
ggplot(weta_data, aes(x = total.weta)) +  
  geom_histogram(colour = "black",  
                 fill = "grey")
```

- scatter plot of NCOR against substrate density, colour by topo, size by shrub, theme\_classic

```
ggplot(mite_all, aes(x = SubsDens, y = NCOR)) +  
  geom_point(aes(colour = Topo,  
                 size = Shrub)) +  
  theme_grey()
```

## Questions so far?



Photo credit: [http://www.senckenberg.de/root/index.php?page\\_id=14691](http://www.senckenberg.de/root/index.php?page_id=14691)

- Useful for visualising correlations
- Not as intuitive to customise
- Slower than `pairs()` but more features

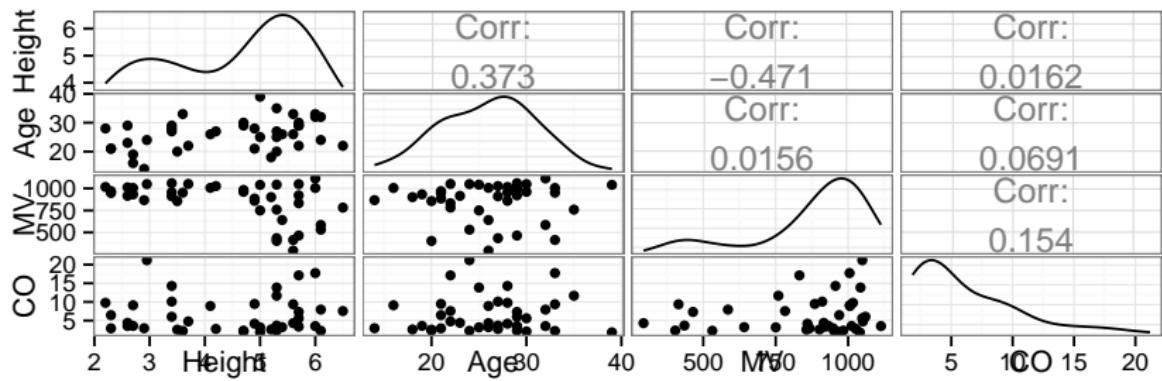
```
require(GGally)
```

## GGPAIRS: example

```
manuka_data %>%
  select(Height, Age, MV, CO) %>%
  ggpairs()

ggpairs(manuka_data[ , c("Height", "Age", "MV", "CO")])
```

# GGPAIRS: example

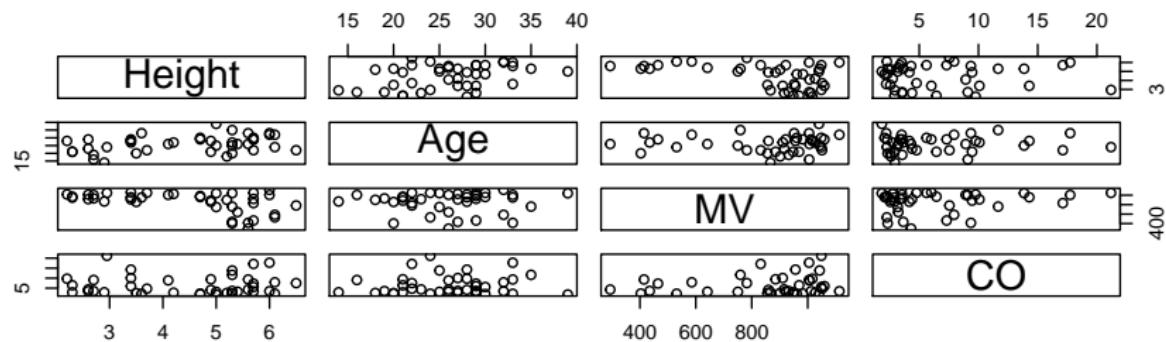


## Compared to pairs: example

```
manuka_data %>%
  select(Height, Age, MV, CO) %>%
  pairs()

pairs(manuka_data[ , c("Height", "Age", "MV", "CO")])
```

## Compared to pairs: example

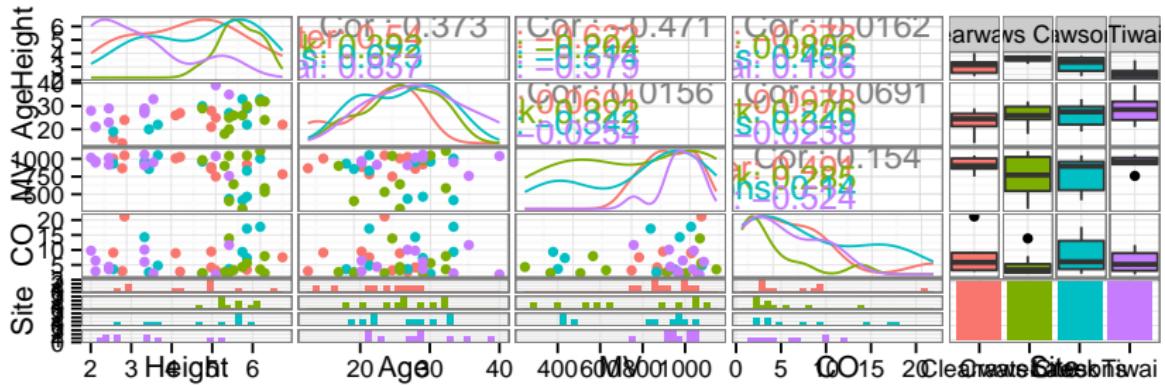


## GGPAIRS: example with factor

```
manuka_data %>%
  select(Height, Age, MV, CO, Site) %>%
  ggpairs(colour = "Site")

ggpairs(manuka_data[ , c("Height", "Age", "MV",
                         "CO", "Site")],
        colour = "Site")
```

# GGPAIRS: example with factor



- ① Please use the mite.env data and create a ggpairs plot of SubsDens, WatrCont, coloured by Topo type.

## GGPAIRS: individual exercise - answer

```
mite.env %>%
  select(SubsDens, WatrCont, Topo) %>%
  ggpairs(colour = "Topo")

ggpairs(mite.env[ , c("SubsDens", "WatrCont", "Topo")],
        colour = "Topo")
```

Questions?



Model done - now what?

Assuming you have completed all assumption checks, overdispersion, etc (the model we will use is in fact overdispersed).

Two options:

- ① Plot the coefficients; or
- ② Plot the predicted values for the treatment effects

```
mite_brachy <- mite_all %>%
  select(Brachy, SubsDens, Shrub)

mod1 <- glm(data = mite_brachy,
            formula = Brachy ~ SubsDens * Shrub,
            family = poisson)
```

```
require(coefplot2)
coefplot2(mod1)
```

```
mite_new1 <- expand.grid(
  SubsDens = seq(from = min(mite_brachy$SubsDens),
                 to = max(mite_brachy$SubsDens),
                 by = 1),
  Shrub = levels(mite_brachy$Shrub))

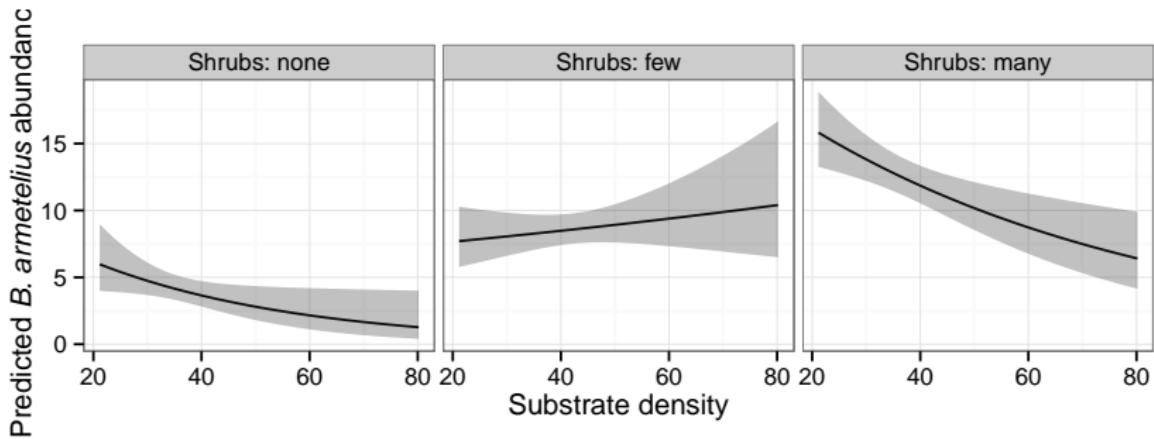
mite_new2 <- cbind(mite_new1,
                     predict(mod1,
                             newdata = mite_new1,
                             type = "link", se = TRUE))
# the se.fit (untransformed se) and
# the residual scale (we are not using this).
```

```
# next, within the dataframe we just created,  
# we do create new columns:  
mite_new2 <- within(mite_new2, {  
    #back tranformation to the response scale  
transformed_response <- exp(fit)  
    #backtransforming the SE  
lower_CI <- exp(fit - (1.96 * se.fit))  
upper_CI <- exp(fit + (1.96 * se.fit))  
})
```

```
# to plot the results:  
mite_new2$Shrub2 <- factor(  
  mite_new2$Shrub,  
  levels = levels(mite_new2$Shrub),  
  labels = paste("Shrubs",  
    tolower(levels(mite_new2$Shrub)),  
    sep = ": "))
```

```
predicted_brachy <- ggplot(mite_new2,
                           aes(x = SubsDens,
                               y = transformed_response))+  
  geom_line() +  
  geom_ribbon(aes(ymin = lower_CI,
                  ymax = upper_CI), alpha = 0.3)+  
  labs(y = expression(paste("Predicted ",
                            italic("B. armatulus "),
                            "abundance")),  
        x = "Substrate density") +  
  facet_wrap(~Shrub2) +  
  theme_bw()
```

## predicted\_brachy





Some packages required

```
require(mapdata)
require(maps)
require(wesanderson)
require(ggplot2)
require(ggthemes)
```

```
nz_data <- map_data("nzHires")
nz_map <- fortify(nz_data)
nz_data <- nz_data[nz_data$region %in% c(
  "North Island", "South Island"),]
```

## Data - species occurrences

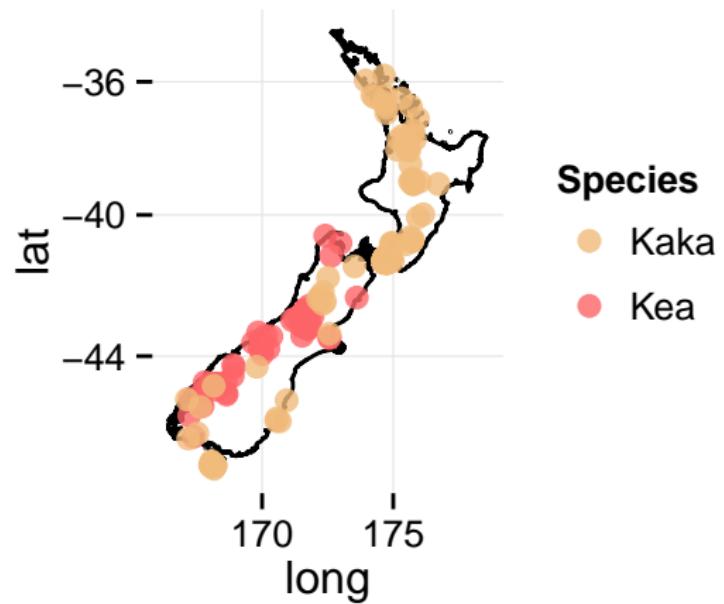
```
kaka_data <- read.csv("data/kaka.csv")
parrot_data <- read.csv("data/parrot.csv")
```

```
ggplot() +
  geom_path(data = nz_data,
            aes(x = long,
                y = lat,
                group = group)) +
  geom_point(data = kaka_data[ !is.na(kaka_data$latitude),
                               ],
             aes(x = longitude,
                 y = latitude,
                 colour = quality_grade),
             size = 3,
             alpha = 0.8) +
  coord_map() +
  theme_minimal()
```

```
parrotmap <- ggplot() +  
  geom_path(data = nz_data,  
            aes(x = long, y = lat,  
                 group = group)) +  
  geom_point(data = parrot_data,  
             aes(x = longitude, y = latitude,  
                  colour = species),  
             size = 3, alpha = 0.8) +  
  scale_colour_manual(name = "Species",  
                      values = wes_palette(  
                        "GrandBudapest")) +  
  coord_map() +  
  theme_minimal() +  
  theme(legend.key = element_blank())
```

# Parrot map

parrotmap



Just discussed in workshop as an extra. See the extra materials folder, ggplot2 intermediate handout for code and discussion.

# Thanks very much!

@oliviarata (twitter)

oliviarata.wordpress.com

<http://github.com/orb16>

Or come to the poster session on Tuesday :)

Want to share/use/improve?

Please do!

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc-sa/4.0/>