Introduction

Why use R?
Why not use a spreadsheet?

Todays workshop

- A common scenario
- A friend has emailed you her data in a spreadsheet
- Todays workshop is about how to get started.
- It's not about impressing with R code

Why not use a spreadsheet?

- Data manipulation in Excel is VERY risk and time consuming
- A range of software packages are available for Excel
- Large data sets can exceed the size limits of standard programs
- Spreadsheets don't have the inherent understanding of statistics that R has
- For example handling of NA's
- R is hot!

Why use R?

Why use R?

- R is free
- R is available on most operating systems Windows, OS X, Linux
- There are huge numbers of packages available
- Its becoming the international standard for statistics

Getting Started
Some References

References

[1] James P. Howard. *R Cookbook*. O'Reilly Media, Inc, 2011.

[2] Phil Spector. *Data Manipulation with R.* Use R series Springer, 2008

Getting Started
Todays Files

Workshop files on Github
`https://github.com/pechang03/SizeDoesMatter`

- The slides. **main.pdf**
- The handouts **handout.pdf**
- The R code **SizeDoesMatterEg.R**
- The spreadsheets
  - `1_RWkshp_dummydata_OTUtable.xlsx`
  - `2_RWkshp_dummydata_EnvData_incl2outliersMK.xlsx`
  - `3_Followupdatafromcontaminatedsite_MK.xlsx`

Getting Started
Installing R!

Download it

- Open `http://www.r-project.org`
- Click CRAN (Under download on Top Left)
- Click `http://cran.ms.unimelb.edu.au/` University of Melbourne

Windows

- Select Windows
- Select Base
- Download R (suggest latest version)

OS X

- Select Select OS X
- Select R-3.2.2.pkg (or the version that matches your OS version)

Getting Started

How about RStudio

- https://www.rstudio.com/products/rstudio/download/
- Its also on your thumb drive

Getting Started

Basic steps

```
2+5

## [1] 7

# Create a sequence of numbers
X = 2:10

# Display basic statistical measures
summary(X)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2       4       6       6       8      10

# use q() to quit
```

Getting Started

Help Functions

To access the documentation type

```
help.start()
help(summary)
args(summary)
example(sd)
??package
```

Help Functions
Search the Web

To search R documentation

- RSiteSearch("key phrase")
- help(adf.test,package="tseries")
- To search for a tutorial for a package vignette(package="packagename")
- For an intro to vignettes see `https://cran.r-project.org/web/packages/sos/vignettes/sos.pdf`
- Examples on the web `http://shiny.rstudio.com/gallery/`

Custom Google search focused on R-specific websites
`http://rseek.org`

Coding Q&A site
`http://stackoverflow.com` `http://stats.stakexchange.com`

Some manners

Iterative development
Working Creatively

Research on how to work creatively based on case studies of successful R&D projects developed into Agile

- Keep the 'manager' away
- Work sustainably
- People over process
- Iterative development

Basic R Data types

R Data types
Lists, frames and tables

Vectors

- Vectors $l \leftarrow c(1, 3, 4, 7, 11)$
- Refer to elements using array $l[c(2, 5)]$ 2nd and 5th elements of l

Data Frames

```r
a <- c(35,23,24,65)
e <- c("Peter", "John", "Mark", NA)
f <- c(TRUE,TRUE,TRUE,FALSE)
team <- data.frame(a,e,f)
names(team) <- c("Age","Names","Passed") # variable names
str(team)

## 'data.frame': 4 obs. of  3 variables:
##  $ Age   : num  35 23 24 65
##  $ Names : Factor w/ 3 levels "John","Mark",..: 3 1 2 NA
##  $ Passed: logi  TRUE TRUE TRUE FALSE
```

Reading files

Let's read the first table

Check the current directory

Where are we

```r
getwd()
setwd("/Users/pcru/SizeDoesMatter1")
dir() #This lists the files
ls()  #This lists the variables
```

http://www.statmethods.net/input/contents.html

Reading a table from a file

Reading an excel table

To read a csv table as a table try

```r
tab1 <- as.matrix(read.csv(file="filetable.csv", sep=",", header=FALSE))
```

But our table is an excel file

- What about a package?
- http://www.thertrader.com/2014/02/11/a-million-ways-to-connect-r-and-excel/
- Installing the R package xlsx
- CRAN mirror http://cran.csiro.au
- Change in preferences

Getting help on packages

R Packages
CRAN

Where from

- install command
- *install.packages*(*pkgs*)

Citing Packages

- Citing packages
- Getting the bibtex entry into endnote
- http://www.lib.uts.edu.au/question/5955/how-can-i-import-bibliography-endnote-bibtex-latex

```
x<-citation()
x1<-citation(package="RSQLite")
toBibtex(x)

sessionInfo()
packages_in_use <- c( sessionInfo()$basePkgs, names( sessionInfo()$loadedOnly ) )
the_citations_list <- lapply( X=packages_in_use, FUN=citation)
the_citations_list
```

Reading an excel table
An example

```
table1<-read.xlsx2("1_R Wkshp_dummy data_OTU table.xlsx", sheetName =

"Sheet1",header=FALSE,rowNames=FALSE,transpose=TRUE,endRow=18)
```

Loading the xlsx package

```
## Loading required package: xlsx
## Warning: package 'xlsx' was built under R version 3.1.3
## Loading required package: rJava
## Warning: package 'rJava' was built under R version 3.1.3
## Loading required package: methods## Loading required package:
xlsxjars## Loading required package: xtable
```

Reading an excel table
The columns types are wrong

	X1	X2	X3	X4	X5	X6	X7
1	Group	Contaminated					
2	Site	1			2		
3	Sample ID	10000	10001	10002	10003	10004	10005
4	Rep	1	2	3	1	2	3
5	phormidiaceae	24872	24872	5822	7538	7201	7538
6	streptococcaceae	11	7	14	8	10	8

Reading an excel table
Transpose the table

Transposing
We need to transpose the table and set the column names correctly

```
table1t=setNames(data.frame(t(table1[,-1])),table1[,1])
```

http://rgm3.lab.nig.ac.jp/RGM/R_rdfile?f=Ecdat/man/read.transpose.Rd&d=R_CC http://stackoverflow.com/questions/17288197/reading-a-csv-file-organized-horizontally

Fields across many columns
Replicating first column

TDD – First do it the easy way first

```
ctridx<-which(table1t$Group=="Control")
table1t$Group[1:48]<-"Contaminated"
table1t$Group[(ctridx+1):48]<-"Control"
```

```
ttt←table1t$Site
for(i in c(2:length(table1t$Site)))
{
    temp←as.character(table1t$Site[i])
    tempb←as.character(ttt[i−1])
    if(table1t$Site[i]=="")
    {
        ttt[i]←tempb
    }
    if(!table1t$Site[(i)]=="")
    {
        ttt[i]←temp
    }
}
table1t$Site←ttt
```

```
## X3
##  1
## Levels:  1 2 3 4 FALSE TRUE
## X4
##  1
## Levels:  1 2 3 4 FALSE TRUE
## X5
##  2
## Levels:  1 2 3 4 FALSE TRUE
## X6
##  2
## Levels:  1 2 3 4 FALSE TRUE
## X7
```

```
##  2
## Levels:  1 2 3 4 FALSE TRUE
## X8
##  1
## Levels:  1 2 3 4 FALSE TRUE
## X9
##  1
## Levels:  1 2 3 4 FALSE TRUE
## X10
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X11
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X12
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X13
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X14
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X15
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X16
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X17
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X18
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X19
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X20
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X21
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X22
```

```
##   1
## Levels:  1 2 3 4 FALSE TRUE
## X23
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X24
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X25
##   2
## Levels:  1 2 3 4 FALSE TRUE
## X26
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X27
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X28
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X29
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X30
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X31
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X32
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X33
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X34
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X35
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X36
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X37
```

```
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X38
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X39
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X40
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X41
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X42
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X43
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X44
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X45
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X46
##   3
## Levels:  1 2 3 4 FALSE TRUE
## X47
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X48
##   4
## Levels:  1 2 3 4 FALSE TRUE
## X49
##   4
## Levels:  1 2 3 4 FALSE TRUE
## rowNames
##    FALSE
## Levels:  1 2 3 4 FALSE TRUE
## transpose
##      TRUE
## Levels:  1 2 3 4 FALSE TRUE
```

Strings

How to work with strings

stringr package

- `require(stringr)`

A look at the stringer package

- `stri_c(str1,str2)`

concatenates two string

- `str_len(str)`

```
require(stringr)

## Loading required package: stringr

table1t$Rep<-str_replace(table1t$Rep,"[rep]{3}?","\\1")
table1t$Rep<-str_replace(table1t$Rep,"A","1")
table1t$Rep<-str_replace(table1t$Rep,"B","2")
table1t$Rep<-str_replace(table1t$Rep,"C","3")
table1t$Rep<-as.factor(table1t$Rep)
```

Reading Tables

Reading a table of other types

- http://www.statmethods.net/input/importingdata.html
- http://stackoverflow.com/questions/17288197/reading-a-csv-file-organized-horizontally
- http://rgm3.lab.nig.ac.jp/RGM/R_rdfile?f=Ecdat/man/read.transpose.Rd&d=R_CC
- Input files from Stata

```
library(foreign)
mydata ← read.dta("c:/mydata.dta")
```

Morning Tea Time

Back in 20min

Need coffee !!

Types

Let's read the next table

Reading a table using xlxs

```
setwd("/Users/pcru/SizeDoesMatter1")
#dir()
table2<-read.xlsx2("2_R Wkshp_dummy data_Env Data_incl2outliersMK.xlsx", sheetName ="Sheet2",head
```

	Group	Site	Sample.ID	Rep	Spill.date	Sample.collection.date
1	Contaminated	1	10000	1	14-May-14	15.5.14
2	Contaminated	1	10001	2	14-May-14	15.5.14
3	Contaminated	1	10002	3	14-May-14	15.5.14
4	Contaminated	2	10003	1	14-May-14	15.5.14
5	Contaminated	2	10004	2	14-May-14	15.5.14
6	Contaminated	2	10005	3	14-May-14	15.5.14

Reading the next table

Reading a table I

Oh NO

- All columns have been set to factors
- Dates have different formats

```
str(table2[,1:11])
```

```
## 'data.frame': 48 obs. of  11 variables:
##  $ Group              : Factor w/ 2 levels "Contaminated",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Site               : Factor w/ 4 levels "1","2","3","4": 1 1 1 2 2 2 1 1 1 2 ...
##  $ Sample.ID           : Factor w/ 18 levels "10000","10001",..: 1 2 3 4 5 6 7 8 9 1 ...
##  $ Rep                : Factor w/ 9 levels "1","2","3","A",..: 1 2 3 1 2 3 7 8 9 7 ...
##  $ Spill.date          : Factor w/ 2 levels "14-May-14","N/A": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Sample.collection.date: Factor w/ 4 levels "15.5.14","17/5/14",..: 1 1 1 1 1 1 2 2 2 2 ...
##  $ labnum             : Factor w/ 36 levels "2000","2001",..: 1 2 3 4 5 6 7 8 9 19 ...
##  $ phosphate..ppb.      : Factor w/ 39 levels "10","105","108",..: 27 30 28 26 25 27 12 15 13 7 .
##  $ ammonia..ppb.        : Factor w/ 41 levels "10","103","1042",..: 10 14 15 6 7 4 31 34 32 28 ...
##  $ chlorophyll..ug.L.    : Factor w/ 38 levels "1","10","11",..: 20 23 21 25 17 18 16 14 15 12 ...
##  $ DO....             : Factor w/ 31 levels "100","120","31",..: 5 4 3 7 6 5 8 7 9 11 ...
```

Reading the next table

Reading a table II

Break it down

First read a few rows only

```
table2 <- read.xlsx2("2_R Wkshp_dummy data_Env Data_incl2outliersMK.xlsx", sheetName = "Sheet2",
   header = TRUE, rowNames = FALSE, as.Data.frame = FALSE, colIndex = c(1:5),
   stringsAsFactors = FALSE, colClasses = c("character", "numeric", "numeric",
        rep("character", 2)), endRow = 4)
sapply(table2, mode)
```

```
##         Group          Site     Sample.ID           Rep    Spill.date
##   "character"     "numeric"     "numeric"   "character"   "character"
##      rowNames as.Data.frame
##     "logical"     "logical"
```

```
sapply(table2, class)
```

```
##         Group          Site     Sample.ID           Rep    Spill.date
##   "character"     "numeric"     "numeric"   "character"   "character"
##      rowNames as.Data.frame
##     "logical"     "logical"
```

Reading the next table
Setting the data types

colClasses

- The variable colClasses can be used to specify the row types.
- We need to set **stringsAsFactor=FALSE** or all columns with be loaded as factors
- The dates are in a non-standard format so we need to read them as chars first

```
table2b<-read.xlsx2("2_R Wkshp_dummy data_Env Data_incl2outliersMK.xlsx",
sheetName = "Sheet2",header=TRUE,rowNames=FALSE,as.Data.frame=FALSE,
colIndex=c(1:11),stringsAsFactors=FALSE,
colClasses=c("character",rep("numeric",2),"character",rep("character",2),rep("numeric",6)))
sapply(table2,class)

##       Group        Site    Sample.ID         Rep   Spill.date
##   "character"    "numeric"    "numeric"   "character"   "character"
##      rowNames as.Data.frame
##     "logical"     "logical"
```

Reading table 2

Setting the Date Type

```
table2f <- table2
table2f$Spill.date <- as.Date(table2f$Spill.date, "%d-%b-%y")
table2f$Sample.collection.date <- as.Date(table2f$Sample.collection.date, "%d.%m.%y")

## Error in as.Date.default(table2f$Sample.collection.date, "%d.%m.%y"):
do not know how to convert 'table2f$Sample.collection.date' to
class "Date"

# sapply(table2f,mode)
sapply(table2f, class)

##       Group        Site    Sample.ID         Rep    Spill.date
##   "character"    "numeric"    "numeric"   "character"       "Date"
##      rowNames as.Data.frame
##     "logical"     "logical"
```

Reading table 2
Setting the Date Type Correctly

colClasses

- The as.Data method can take a format string as the second variable
- The format strings are described in help on strptime
- But Spill.data has **two formats**
- We can use the if else function to combine them

```
table2bf<-table2b
table2bf$Spill.date<-as.Date(table2bf$Spill.date,"%d-%b-%y")
cdate1<-as.Date(table2bf$Sample.collection.date,"%d.%m.%y")
cdate2<-as.Date(table2bf$Sample.collection.date,"%d/%m/%y")
table2bf$Sample.collection.date<-as.Date(ifelse
(!is.na(cdate1),as.Date(cdate1),as.Date(cdate2)), origin="1970-01-01")
table2bf$Group<-as.factor(table2bf$Group)
table2bf$Rep<-as.factor(table2bf$Rep)
dated<-table2bf$Sample.collection.date-table2bf$Spill.date
```

Reading table 2
Setting the Date Type Correctly

Count the NAs

```
na_count <-sapply(table2bf, function(y) sum(length(which(is.na(y)))))
na_count
```

```
##                Group                 Site            Sample.ID
##                    0                    0                    0
##                  Rep           Spill.date Sample.collection.date
##                    0                   24                    0
##              labnum       phosphate..ppb.        ammonia..ppb.
##                    0                    0                    0
##    chlorophyll..ug.L.              DO....            rowNames
##                    0                    0                    0
##          as.Data.frame
##                    0
```

Strings

Reading table 2

Just fix the Rep column using the stringer package again

```
require(stringr)
table2bf$Rep<-str_replace(table2bf$Rep,"[rep]{3}?","\\1")
table2bf$Rep<-str_replace(table2bf$Rep,"A","1")
table2bf$Rep<-str_replace(table2bf$Rep,"B","2")
table2bf$Rep<-str_replace(table2bf$Rep,"C","3")
table2bf$Rep<-as.factor(table2bf$Rep)
str(table2bf)
```

```
## 'data.frame': 48 obs. of  13 variables:
##  $ Group              : Factor w/ 2 levels "Contaminated",..: 1 1 1 1 1 1 1 1 1 1 ...
##   $ Site                  : num  1 1 1 2 2 2 1 1 1 2 ...
##  $ Sample.ID           : num  10000 10001 10002 10003 10004 ...
##  $ Rep                : Factor w/ 3 levels "1","2","3": 1 2 3 1 2 3 1 2 3 1 ...
##  $ Spill.date          : Date, format: "2014-05-14" "2014-05-14" ...
##  $ Sample.collection.date: Date, format: "2014-05-15" "2014-05-15" ...
##   $ labnum                : num  2000 2001 2002 2003 2004 ...
##   $ phosphate..ppb.       : num  3020 3253 3169 2999 2879 ...
##  $ ammonia..ppb.        : num  13880 14598 14676 10984 11657 ...
##  $ chlorophyll..ug.L.    : num  302 323 315 352 289 296 254 248 250 220 ...
##  $ DO....               : num  34 33 31 38 36 34 40 38 41 45 ...
##  $ rowNames           : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ as.Data.frame       : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

Merging Tables

How to merge two data sets?

Using the merge command

The inbuilt command merge

- R has a command merge
- To begin, start looking at the first 9 lines of the tables and merge them
- Need to use Group, Site, Sample.ID because otherwise it's not unique

```
merge(x, y, by = intersect(names(x), names(y)),
      by.x = by, by.y = by, all = FALSE, all.x = all, all.y = all,
      sort = TRUE, suffixes = c(".x",".y"),
      incomparables = NULL, ...)
```

```
tab1c<-table1t[1:9,]
tab2c<-table2b[1:9,]
m1<-merge(tab1c,tab2c,by.x="Sample ID",by.y="Sample.ID")
m2<-merge(table1t,table2bf,by.x=c("Group","Site","Sample ID"),
by.y=c("Group","Site","Sample.ID"))
m3<-merge(table1t,table2bf,by.x=c("Group","Site","Sample ID","Rep"),
by.y=c("Group","Site","Sample.ID","Rep"))
```

Lunch Time
Back in 30 min
Provided

How do I append two data sets?
To begin load the third data set

Follow up data from contaminated site

```
table3<-read.xlsx2("3_Follow up data from contaminated site_MK.xlsx",
 sheetName ="Sheet1",header=TRUE,rowNames=FALSE,
 colClasses=c(rep("character",3),

rep("character",2),rep("numeric",18)))
table3f<-table3
table3f$Spill.date<-as.Date(table3f$Spill.date,"%d.%m.%y")
table3f$Sample.collection.date<-as.Date(table3f$Sample.collection.date,"%d.%m.%y")
sapply(table3f,mode)
sapply(table3f,class)
```

How do I append two data sets?
Loading the third data set

Joining table 3 to the other merged tables

- We need to be careful to match everything
- Install the **plyr** package This has lots of useful functions for renaming var etc
- This means we need columns for corynebacteriaceae and porphyromondaceae
- Should these values be NA or 0?
- We will do one of each.
- Generally we would use NA but in this case 0 is better as its likely the rows were missing as none were detected

How do I append two data sets?

Appending the third set

```
require(plyr)
Sample.ID<-rep(20000,3)
table3fi<-cbind(table3f,Sample.ID)
#how many columns I can't count
ncol(table3fi)
ncol(m3)
#now get the cols all right
table3fii<-table3fi[c(1,2,24,3,4:23)]
m3i<-m3[c(1:4,19:20,5:18,21:26)]
setdiff(names(m3i),names(table3fii))
m3ii<-rename(m3i,c("Sample ID"="Sample.ID"))
corynebacteriaceae<-rep(0,nrow(table3fii))
porphyromondaceae<-rep(NA,nrow(table3fii))
table3fiii<-cbind(table3fii, corynebacteriaceae, porphyromondaceae)
setdiff(names(m3ii),names(table3fiii))

m3ii[,c(7:24)] <- sapply(m3ii[,c(7:24)],as.numeric)
m3ii[,c(1:4)] <-sapply(m3ii[,c(1:4)],as.character)
#m3ii[,c("Site")] <-sapply(m3ii[,c("Site")],as.character)

table3fiii[,c(1:4)] <- sapply(table3fiii[,c(1:4)],as.character)
table3fiii[,c(7:24)] <- sapply(table3fiii[,c(7:24)],as.numeric)
table4<-rbind(m3ii,table3fiii)
table4[,1] <- sapply(table4[,1],as.factor)
```

```
## Loading required package: plyr

## [1] 24
## [1] 27
## [1] "Sample ID"        "corynebacteriaceae" "porphyromondaceae"
## character(0)
```

Another Break

Fat or wide

Reshaping Tables

reshape2

reshape2

- vignette(reshape) doesn't work
- try `http://had.co.nz/reshape/`
- and `http://seananderson.ca/2013/10/19/reshape.html`

A small example for melt

- Suppose we what a box plot to see if there are outliers

- We will use ggplot2 box plot
- The box plot needs data in long format.
- To use this first **melt** the data
- We need to specify the unique key, the variable name and the value name
- The key is not unique.
- Then plot it

Reshaping Tables
melt and boxplot

The code

```
matable4←melt(table4[,c(1:4,6:25)],variable.name = "microbe",
value.name ="abundance", id=c("Group","Site","Sample.ID","Rep"),
factorsAsStrings=FALSE,rm.na=TRUE)
```

```
require(reshape2)

## Loading required package: reshape2

matable4<-melt(table4[,c(1:4,7:25)],variable.name = "microbe",
value.name ="abundance", id=c("Group","Site","Sample.ID","Rep"),
factorsAsStrings=FALSE,rm.na=TRUE)
```

Reshaping Tables
Boxplot cont

Using ggplot

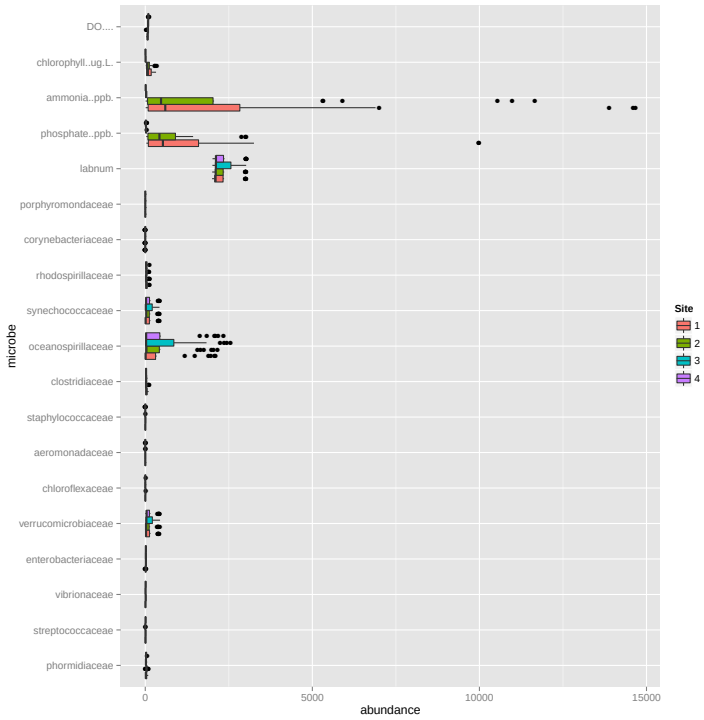
- As we have keys we need to specify the x and y
- Let's make the sites different colors
- The variable names are long so flip it with *coord_flip*()
- Looks like we have outliers...hmm

```
require(ggplot2)

## Loading required package: ggplot2

 ggplot(matable4,aes(x=microbe,y=abundance,fill=Site)) + geom_boxplot() +  coord_flip()

## Warning: Removed 24 rows containing non-finite values (stat_boxplot).
```



Finding Outliers
Interquartile range

Finding Outliers

- Outliers are defined 1.5 times the interquartile range above the upper quartile
- Assume that rows 12 and 14 in phosphate are errors as the 9 is typed twice
- Still issues with ammonia to explore

```
phosphate<-table4[,"phosphate..ppb."]
upper.limit <- quantile(phosphate)[4] + 1.5*IQR(phosphate)
lower.limit <- quantile(phosphate)[2] - 1.5*IQR(phosphate)
#table4[phosphate> upper.limit,c("Site","phosphate..ppb.")]
```

Reshaping Tables
Finding Outliers

Removing Outliers

	Site	phosphate..ppb.
1	1	3020.00
2	1	3253.00
3	1	3169.00
12	1	9982.00
14	1	9982.00
16	1	1542.00

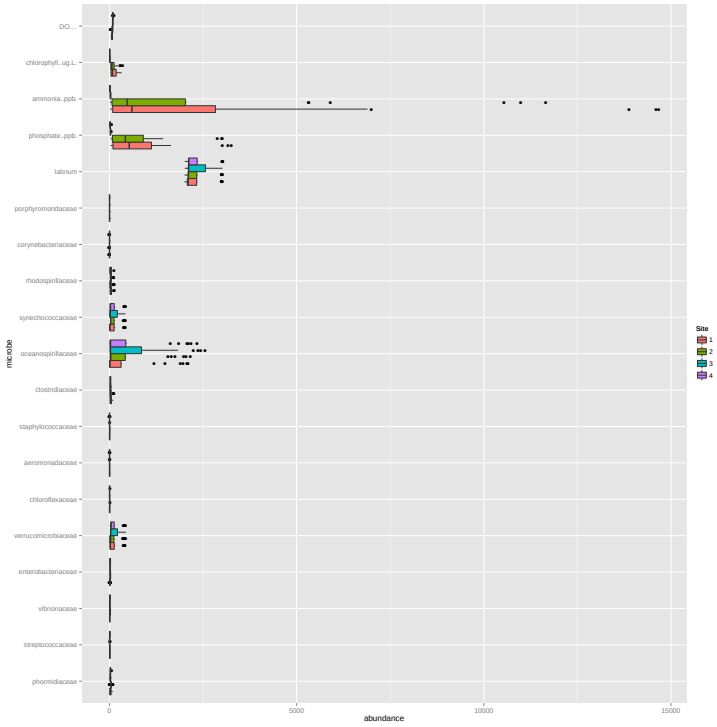
```
table4[12,"phosphate..ppb."]<-982
table4[14,"phosphate..ppb."]<-982
```

Outliers check

Redo the boxplot

Look again ggplot

```
## Warning: Removed 24 rows containing non-finite values (stat_boxplot).
```



R package

RQLlite

RSQLite

- Suppose merge is not enough? I know about SQL and want to do joins
- Install RSQLite
- We also need to install DBI

```
## Loading required package: RSQLite
```

```
 db <- dbConnect(SQLite(), dbname="Test.sqlite")
#getConfig()$staged.queries
# sqldf(attach "Test1.sqlite" as new)
dbBegin(db)

## [1] TRUE

dbWriteTable(db,"table1",table1t,overwrite=TRUE)

## [1] TRUE

dbReadTable(db,"table1")

##                  Group  Site Sample.ID   Rep phormidiaceae
## X2        Contaminated     1     10000     1         24872
## X3        Contaminated     1     10001     2         24872
## X4        Contaminated     1     10002     3          5822
## X5        Contaminated     2     10003     1          7538
## X6        Contaminated     2     10004     2          7201
## X7        Contaminated     2     10005     3          7538
## X8        Contaminated     1     10006     1          8467
## X9        Contaminated     1     10007     2          7340
## X10       Contaminated     1     10008     3          8467
## X11       Contaminated     2     10000     1          2000
## X12       Contaminated     2     10001     2          2083
## X13       Contaminated     2     10002     3          1899
## X14       Contaminated     1     10003     1          1947
## X15       Contaminated     1     10004     2          2733
## X16       Contaminated     1     10005     3          2385
## X17       Contaminated     2     10006     1           800
## X18       Contaminated     2     10007     2           738
## X19       Contaminated     2     10008     3           800
## X20       Contaminated     1     10003     1           200
```

```
## X21       Contaminated     1     10004     2           189
## X22       Contaminated     1     10005     3           271
## X23       Contaminated     2     10006     1            46
## X24       Contaminated     2     10007     2            62
## X25       Contaminated     2     10008     3            94
## X26       Contaminated     3     10009     1            24
## X27            Control     3     10010     2            64
## X28            Control     3     10011     3            21
## X29            Control     4     10012     1            56
## X30            Control     4     10013     2            27
## X31            Control     4     10014     3            53
## X32            Control     3     10015     1           115
## X33            Control     3     10016     2            97
## X34            Control     3     10017     3            45
## X35            Control     4     10009     1            33
## X36            Control     4     10010     2            51
## X37            Control     4     10011     3            47
## X38            Control     3     10012     1           105
## X39            Control     3     10013     2            72
## X40            Control     3     10014     3           115
## X41            Control     4     10015     1            18
## X42            Control     4     10016     2            54
## X43            Control     4     10017     3            33
## X44            Control     3     10012     1            36
## X45            Control     3     10013     2            58
## X46            Control     3     10014     3            36
## X47            Control     4     10015     1            60
## X48            Control     4     10016     2           164
## X49            Control     4     10017     3            79
## rowNames         FALSE FALSE     FALSE F1LSE         FALSE
## transpose         TRUE  TRUE      TRUE  TRUE          TRUE
##           streptococcaceae vibrionaceae enterobacteriaceae
## X2                      11           33                131
## X3                       7           40                200
## X4                      14           40                200
## X5                       8           95                151
## X6                      10           83                140
## X7                       8           95                151
## X8                       5           29                132
## X9                       5           51                168
## X10                      5           29                132
## X11                     10           34                 97
## X12                     17           38                 91
## X13                     27           31                 51
## X14                      0            0                  2
```

```
## X15                     1            0                 1
## X16                     0            0                 2
## X17                    26           33                34
## X18                    22           58                42
## X19                    26           33                34
## X20                     6            5                39
## X21                     2            3                23
## X22                     5            1                39
## X23                     3            9                55
## X24                     1            5                95
## X25                     3            0                55
## X26                     2            6                36
## X27                     3            3                36
## X28                     0            4                30
## X29                     1           30                79
## X30                     5            9               129
## X31                     1            1               124
## X32                     0           10                52
## X33                     4            6                13
## X34                     0            2                 9
## X35                     4            4                11
## X36                     0           10                41
## X37                     0            3                29
## X38                    10           39               288
## X39                    10           29               413
## X40                    12           34               481
## X41                     5            2                43
## X42                     3            5                50
## X43                     4            5                86
## X44                    10            4                28
## X45                    12            1                28
## X46                     1            0                44
## X47                     4            7                48
## X48                     2           11               111
## X49                     3            5                88
## rowNames            FALSE        FALSE             FALSE
## transpose            TRUE         TRUE              TRUE
##           verrucomicrobiaceae chloroflexaceae aeromonadaceae
## X2                        977            351             20
## X3                       1500            246             76
## X4                        844            246             76
## X5                       1006             41              1
## X6                       1112             83              6
## X7                       1195             41              0
## X8                       1805             23              0
```

```
## X9                      1906              28               0
## X10                     1902              23               0
## X11                     1244              40               1
## X12                     1933              40               1
## X13                     1244              80               0
## X14                      251               1               0
## X15                      271               0               1
## X16                      299               0               0
## X17                     1348             209               1
## X18                     3612             205               0
## X19                     1348             209               3
## X20                      176               1               0
## X21                      211               0               3
## X22                      183               1               0
## X23                      544               0               9
## X24                      611               1               0
## X25                      544               0               1
## X26                      471               0               0
## X27                      500               0               1
## X28                      541               0               0
## X29                     1405               3               1
## X30                     1678               1               1
## X31                     1360               0               0
## X32                     1590               0               0
## X33                      398               0               0
## X34                      195               1               0
## X35                     1213               7               2
## X36                     3461              12               1
## X37                     1688               3               0
## X38                      590               1               1
## X39                      598               0               0
## X40                      639               1               0
## X41                      949               0               0
## X42                      974               0               0
## X43                      662               0               0
## X44                      267               0               0
## X45                      249               2               0
## X46                      337               0               0
## X47                      625               0               0
## X48                      886               0               0
## X49                      791               0               0
## rowNames                FALSE           FALSE           FALSE
## transpose                TRUE            TRUE            TRUE
##           staphylococcaceae clostridiaceae oceanospirillaceae
## X2                      115            274               1438
```

```
## X3                342              288              1789
## X4                342              258              1789
## X5                  4              365                 5
## X6                  9              365                 2
## X7                  4              365                 9
## X8                  1              643                14
## X9                  1              941                14
## X10                 1              711                14
## X11                 0              204                93
## X12                 0              229                72
## X13                 0              285                93
## X14                 0                8              1080
## X15                 0               12              1633
## X16                 1               12              1080
## X17                 4              400               747
## X18                 7              733               636
## X19                 2              299               747
## X20                 1               76               256
## X21                 0               63               263
## X22                 2               85               256
## X23                 1              136               284
## X24                20              643               293
## X25                 1              136               293
## X26                 0               31               189
## X27                 1               59               510
## X28                 0               42               215
## X29                 0              143              2096
## X30                 0              124               834
## X31                 0              100               426
## X32                 0               34               263
## X33                 0               33              1095
## X34                 0               18               523
## X35                 0               96               273
## X36                 0              100              1432
## X37                 0               74               330
## X38                 1              119              4584
## X39                 1              181              3811
## X40                 0              202              3165
## X41                 0               38               380
## X42                 0               29               548
## X43                 0               56               403
## X44                 0               62               394
## X45                 0               58               311
## X46                 0               66               376
## X47                 0               66               773
```

```
## X48                       0            167               1778
## X49                       0             40               1289
## rowNames              FALSE          FALSE              FALSE
## transpose              TRUE           TRUE               TRUE
##          synechococcaceae rhodospirillaceae corynebacteriaceae
## X2                    471              1267                  0
## X3                    498              1597                  0
## X4                    692              1844                  0
## X5                     20                70                  0
## X6                     20                82                  0
## X7                     48                70                  0
## X8                     27                97                  0
## X9                     83                97                  0
## X10                    27                97                  0
## X11                    61               579                  0
## X12                    61               603                  0
## X13                    61               579                  0
## X14                   245              2245                  0
## X15                   245              2001                  0
## X16                   142              2834                  0
## X17                    95              1432                  0
## X18                    70              1834                  0
## X19                    95              1432                  0
## X20                   101               786                  0
## X21                   104               844                  0
## X22                   101               826                  0
## X23                    65              1833                  0
## X24                    53              2528                  0
## X25                    65              2999                  0
## X26                    67               568                  0
## X27                   128              1877                  0
## X28                   152               582                  0
## X29                   769              1699                  0
## X30                   954              3145                  0
## X31                   555              1171                  0
## X32                    45               323                  0
## X33                   164               911                  0
## X34                   513               485                  0
## X35                    75               732                  0
## X36                   414              3101                  0
## X37                   298              1262                  0
## X38                   807              3586                  0
## X39                  1916              5757                  0
## X40                  1120              4168                  0
## X41                   276               821                  0
```

```
## X42                   394                489                   0
## X43                   498                611                   0
## X44                   212               1001                   0
## X45                   301                889                   0
## X46                   330                943                   0
## X47                   521               1300                   0
## X48                  1220               3013                   0
## X49                   383               1255                   0
## rowNames            FALSE              FALSE               FALSE
## transpose            TRUE               TRUE                TRUE
##           porphyromondaceae
## X2                        0
## X3                        0
## X4                        0
## X5                        0
## X6                        0
## X7                        0
## X8                        0
## X9                        0
## X10                       0
## X11                       0
## X12                       0
## X13                       0
## X14                       0
## X15                       0
## X16                       0
## X17                       0
## X18                       0
## X19                       0
## X20                       0
## X21                       0
## X22                       0
## X23                       0
## X24                       0
## X25                       0
## X26                       0
## X27                       0
## X28                       0
## X29                       0
## X30                       0
## X31                       0
## X32                       0
## X33                       0
## X34                       0
## X35                       0
```

```
## X36                       0
## X37                       0
## X38                       0
## X39                       0
## X40                       0
## X41                       0
## X42                       0
## X43                       0
## X44                       0
## X45                       0
## X46                       0
## X47                       0
## X48                       0
## X49                       0
## rowNames              FALSE
## transpose              TRUE
```

```
dbListFields(db,"table1")
```

```
##  [1] "row_names"           "Group"               "Site"
##  [4] "Sample ID"           "Rep"                 "phormidiaceae"
##  [7] "streptococcaceae"    "vibrionaceae"        "enterobacteriaceae"
## [10] "verrucomicrobiaceae" "chloroflexaceae"     "aeromonadaceae"
## [13] "staphylococcaceae"   "clostridiaceae"      "oceanospirillaceae"
## [16] "synechococcaceae"    "rhodospirillaceae"   "corynebacteriaceae"
## [19] "porphyromondaceae"
```

```
dbListTables(db)
```

```
## [1] "table1"
```

```
dbGetQuery(db, "SELECT * from table1")
```

```
##    row_names        Group  Site Sample ID   Rep phormidiaceae
## 1         X2 Contaminated     1     10000     1         24872
## 2         X3 Contaminated     1     10001     2         24872
## 3         X4 Contaminated     1     10002     3          5822
## 4         X5 Contaminated     2     10003     1          7538
## 5         X6 Contaminated     2     10004     2          7201
## 6         X7 Contaminated     2     10005     3          7538
## 7         X8 Contaminated     1     10006     1          8467
## 8         X9 Contaminated     1     10007     2          7340
## 9        X10 Contaminated     1     10008     3          8467
## 10       X11 Contaminated     2     10000     1          2000
## 11       X12 Contaminated     2     10001     2          2083
## 12       X13 Contaminated     2     10002     3          1899
## 13       X14 Contaminated     1     10003     1          1947
```

```
## 14       X15 Contaminated     1     10004     2          2733
## 15       X16 Contaminated     1     10005     3          2385
## 16       X17 Contaminated     2     10006     1           800
## 17       X18 Contaminated     2     10007     2           738
## 18       X19 Contaminated     2     10008     3           800
## 19       X20 Contaminated     1     10003     1           200
## 20       X21 Contaminated     1     10004     2           189
## 21       X22 Contaminated     1     10005     3           271
## 22       X23 Contaminated     2     10006     1            46
## 23       X24 Contaminated     2     10007     2            62
## 24       X25 Contaminated     2     10008     3            94
## 25       X26 Contaminated     3     10009     1            24
## 26       X27      Control     3     10010     2            64
## 27       X28      Control     3     10011     3            21
## 28       X29      Control     4     10012     1            56
## 29       X30      Control     4     10013     2            27
## 30       X31      Control     4     10014     3            53
## 31       X32      Control     3     10015     1           115
## 32       X33      Control     3     10016     2            97
## 33       X34      Control     3     10017     3            45
## 34       X35      Control     4     10009     1            33
## 35       X36      Control     4     10010     2            51
## 36       X37      Control     4     10011     3            47
## 37       X38      Control     3     10012     1           105
## 38       X39      Control     3     10013     2            72
## 39       X40      Control     3     10014     3           115
## 40       X41      Control     4     10015     1            18
## 41       X42      Control     4     10016     2            54
## 42       X43      Control     4     10017     3            33
## 43       X44      Control     3     10012     1            36
## 44       X45      Control     3     10013     2            58
## 45       X46      Control     3     10014     3            36
## 46       X47      Control     4     10015     1            60
## 47       X48      Control     4     10016     2           164
## 48       X49      Control     4     10017     3            79
## 49  rowNames        FALSE FALSE     FALSE F1LSE         FALSE
## 50 transpose         TRUE  TRUE      TRUE  TRUE          TRUE
##    streptococcaceae vibrionaceae enterobacteriaceae verrucomicrobiaceae
## 1                11          33               131                977
## 2                 7          40               200               1500
## 3                14          40               200                844
## 4                 8          95               151               1006
## 5                10          83               140               1112
## 6                 8          95               151               1195
## 7                 5          29               132               1805
```

```
## 8                5          51               168              1906
## 9                5          29               132              1902
## 10              10          34                97              1244
## 11              17          38                91              1933
## 12              27          31                51              1244
## 13               0           0                 2               251
## 14               1           0                 1               271
## 15               0           0                 2               299
## 16              26          33                34              1348
## 17              22          58                42              3612
## 18              26          33                34              1348
## 19               6           5                39               176
## 20               2           3                23               211
## 21               5           1                39               183
## 22               3           9                55               544
## 23               1           5                95               611
## 24               3           0                55               544
## 25               2           6                36               471
## 26               3           3                36               500
## 27               0           4                30               541
## 28               1          30                79              1405
## 29               5           9               129              1678
## 30               1           1               124              1360
## 31               0          10                52              1590
## 32               4           6                13               398
## 33               0           2                 9               195
## 34               4           4                11              1213
## 35               0          10                41              3461
## 36               0           3                29              1688
## 37              10          39               288               590
## 38              10          29               413               598
## 39              12          34               481               639
## 40               5           2                43               949
## 41               3           5                50               974
## 42               4           5                86               662
## 43              10           4                28               267
## 44              12           1                28               249
## 45               1           0                44               337
## 46               4           7                48               625
## 47               2          11               111               886
## 48               3           5                88               791
## 49           FALSE       FALSE            FALSE             FALSE
## 50            TRUE        TRUE             TRUE              TRUE
##    chloroflexaceae aeromonadaceae staphylococcaceae clostridiaceae
## 1              351            20              115            274
```

```
## 2              246              76              342              288
## 3              246              76              342              258
## 4               41               1                4              365
## 5               83               6                9              365
## 6               41               0                4              365
## 7               23               0                1              643
## 8               28               0                1              941
## 9               23               0                1              711
## 10              40               1                0              204
## 11              40               1                0              229
## 12              80               0                0              285
## 13               1               0                0                8
## 14               0               1                0               12
## 15               0               0                1               12
## 16             209               1                4              400
## 17             205               0                7              733
## 18             209               3                2              299
## 19               1               0                1               76
## 20               0               3                0               63
## 21               1               0                2               85
## 22               0               9                1              136
## 23               1               0               20              643
## 24               0               1                1              136
## 25               0               0                0               31
## 26               0               1                1               59
## 27               0               0                0               42
## 28               3               1                0              143
## 29               1               1                0              124
## 30               0               0                0              100
## 31               0               0                0               34
## 32               0               0                0               33
## 33               1               0                0               18
## 34               7               2                0               96
## 35              12               1                0              100
## 36               3               0                0               74
## 37               1               1                1              119
## 38               0               0                1              181
## 39               1               0                0              202
## 40               0               0                0               38
## 41               0               0                0               29
## 42               0               0                0               56
## 43               0               0                0               62
## 44               2               0                0               58
## 45               0               0                0               66
## 46               0               0                0               66
```

```
## 47              0             0               0           167
## 48              0             0               0            40
## 49          FALSE         FALSE           FALSE         FALSE
## 50           TRUE          TRUE            TRUE          TRUE
##    oceanospirillaceae synechococcaceae rhodospirillaceae
## 1                1438              471              1267
## 2                1789              498              1597
## 3                1789              692              1844
## 4                   5               20                70
## 5                   2               20                82
## 6                   9               48                70
## 7                  14               27                97
## 8                  14               83                97
## 9                  14               27                97
## 10                 93               61               579
## 11                 72               61               603
## 12                 93               61               579
## 13               1080              245              2245
## 14               1633              245              2001
## 15               1080              142              2834
## 16                747               95              1432
## 17                636               70              1834
## 18                747               95              1432
## 19                256              101               786
## 20                263              104               844
## 21                256              101               826
## 22                284               65              1833
## 23                293               53              2528
## 24                293               65              2999
## 25                189               67               568
## 26                510              128              1877
## 27                215              152               582
## 28               2096              769              1699
## 29                834              954              3145
## 30                426              555              1171
## 31                263               45               323
## 32               1095              164               911
## 33                523              513               485
## 34                273               75               732
## 35               1432              414              3101
## 36                330              298              1262
## 37               4584              807              3586
## 38               3811             1916              5757
## 39               3165             1120              4168
## 40                380              276               821
```

```
## 41                548               394                489
## 42                403               498                611
## 43                394               212               1001
## 44                311               301                889
## 45                376               330                943
## 46                773               521               1300
## 47               1778              1220               3013
## 48               1289               383               1255
## 49              FALSE             FALSE              FALSE
## 50               TRUE              TRUE               TRUE
##    corynebacteriaceae porphyromondaceae
## 1                   0                 0
## 2                   0                 0
## 3                   0                 0
## 4                   0                 0
## 5                   0                 0
## 6                   0                 0
## 7                   0                 0
## 8                   0                 0
## 9                   0                 0
## 10                  0                 0
## 11                  0                 0
## 12                  0                 0
## 13                  0                 0
## 14                  0                 0
## 15                  0                 0
## 16                  0                 0
## 17                  0                 0
## 18                  0                 0
## 19                  0                 0
## 20                  0                 0
## 21                  0                 0
## 22                  0                 0
## 23                  0                 0
## 24                  0                 0
## 25                  0                 0
## 26                  0                 0
## 27                  0                 0
## 28                  0                 0
## 29                  0                 0
## 30                  0                 0
## 31                  0                 0
## 32                  0                 0
## 33                  0                 0
## 34                  0                 0
```

```
## 35                  0                  0
## 36                  0                  0
## 37                  0                  0
## 38                  0                  0
## 39                  0                  0
## 40                  0                  0
## 41                  0                  0
## 42                  0                  0
## 43                  0                  0
## 44                  0                  0
## 45                  0                  0
## 46                  0                  0
## 47                  0                  0
## 48                  0                  0
## 49              FALSE              FALSE
## 50               TRUE               TRUE

#dbDisconnect(db)
```

R package
RQLlite

RSQLite

- Some links to RSQL ideas
- http://stackoverflow.com/questions/12307685/join-more-than-2-tables-in-r-using-rsqlite
- https://support.rstudio.com/hc/en-us/articles/201057987-Quick-list-of-useful-R-packages
- https://cran.rstudio.com/web/packages/dplyr/vignettes/introduction.
  html

```
select coalesce(fileA,fileB),valA,valB
                from t1 LEFT OUTER JOIN t2 On t1.fileA= t2.fileB
UNION select coalesce(fileA,fileB),valA,valB
                from t2 LEFT OUTER JOIN t1 ON t1.fileA= t2.fileB
(CREATE TABLE all_files AS SELECT fileA FROM t1 UNION SELECT fileB from t2 UNION ...).
```

R package
svUnit

Another important component of TDD is refactoring and unit tests

- Refactoring http://refactoring.com/
- http://www.r-bloggers.com/my-experience-of-learning-r-from-basic-graphs-to-performance-tun
- TDD in R http://www.slideserve.com/andrew/test-driven-development-in-r
- Version Control tortiseSVN ttp://tortoisesvn.net/}\itemGitHub\
  url{ttps://github.com/

Cleaning things up

Dropping row and columns

Dropping selected variables

Dropping Row and Columns with too many NAs

```
numNAs_inData4_rows ← apply(rawData4, 1, function(z) sum(is.na(z)))
numNAs_inData4_col ← apply(table4, 2, function(z) sum(is.na(z))) # count NAs in Data4
lessThan20 ← table4[!(numNAs_inData4_rows > 20),]      #only select the rows contain less Than 20 N
lessThan20col ← table4[,!(numNAs_inData4_col > 20)]
```

Dropping row and columns

Dropping selected variables

Tidy Data

In tidy data:

- Each variable forms a column.
- Each observation forms a row.
- Each type of observational unit forms a table.
- https://cran.r-project.org/web/packages/tidyr/vignettes/tidy-data.html
- http://pj.freefaculty.org/R/Rtips.html#toc-Subsection-1.11

Spit out the dates and numbers

```
dates4←table4[,c(5,6)]
abundance←table4[,c(7:25)]
```

Adding a new column

Calculating the number of days

Calculating the number of days

We can just subtract as.Date fields

```
dates4<-table4[,c(5,6)]
abundance<-table4[,c(7:25)]
 days<-dates4[,2]-dates4[,1]
```

Setting the relative abundance

Setting the relative abundance

Normalizing data

sapply

- Also known as centring the data
- Ecological percentage of the sum of the variables
- We an use sweep to centre the data
- $options(digits = 1)$ Just to make things pretty

```
sweepOutContinu<-sweep(abundance,2,apply(abundance,2,min,na.rm=TRUE))
afterSweepContinu<-sweep(sweepOutContinu,2,apply(sweepOutContinu,2,max,na.rm=TRUE),"/")
table5<-cbind(table4[,c(1:6)],afterSweepContinu,days)
options(digits=1)
sweep(abundance, 2, colSums(abundance), FUN="/")
scale(abundance, center=FALSE, scale=colSums(abundance))
```

Now let's have some fun

Graphics in R

R has nice graphs

- A graphical output
- `http://rcharts.io/gallery/`
- R Graph gallery currently down try `http://rgraphgallery.blogspot.com/`
- A reference on where to go R thumbnails
- ggplot2 (scatter plot of 2 var and then 3 plots)
- To create a correlation heat map

```
library(corrplot)
abuncor<-cor(t5lessThan20col[,c(6:22)])
require(corrplot)
corrplot(abuncor, method = "circle")
```

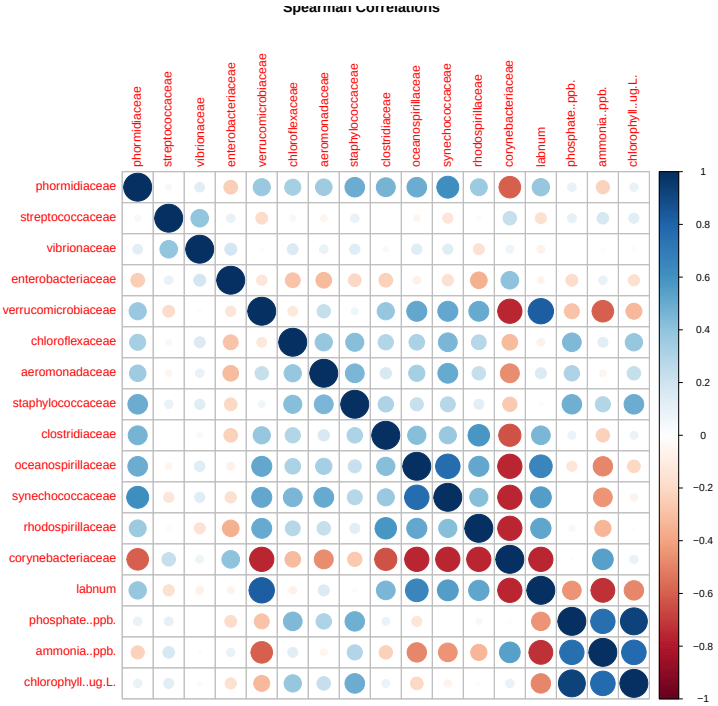
```
## [1] 23

## Loading required package: corrplot
```

Now let's have some fun

Making a heat map

A heat map



Simple Tests

ttests

t tests

There are many $t - tests$ available in R `http://www.statmethods.net/stats/ttest.html`

```
# independent 2-group t-test
t.test(t5lessThan20col[,12],t5lessThan20col[,8])

##
##  Welch Two Sample t-test
##
## data:  t5lessThan20col[, 12] and t5lessThan20col[, 8]
## t = -3.4052, df = 180.441, p-value = 0.0008149
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -0.20589367 -0.05481872
## sample estimates:
## mean of x mean of y
## 0.2725146 0.4028708
```

What next
Proposed future talks

Help is on the way

- Parameterized Complexity Research Unit (PCRU) PhD students
- PhD student in Bioinformatics from Central South Uni

Your feedback on some ideas

- Using Sweave or Knitr
- Advanced Data Cleaning
- Network Centric data analysis

Resources
If you want to improve this style

References

[1] LaTeX Beamer `http://latex-beamer.sourceforge.net/`

[2] Sharelatex Site `https://www.sharelatex.com`

[3] A Data Cleaning Mooc `https://www.sharelatex.com`

R Packages Used
Session Info

Output of sessionInfo

```
sessionInfo()

## R version 3.1.2 (2014-10-31)
## Platform: x86_64-apple-darwin13.4.0 (64-bit)
##
## locale:
## [1] C
```

```
##
## attached base packages:
## [1] methods   stats     graphics  grDevices utils     datasets  base
##
## other attached packages:
##  [1] corrplot_0.73  RSQLite_1.0.0  DBI_0.3.1      ggplot2_1.0.0
##  [5] reshape2_1.4.1 plyr_1.8.1     stringr_0.6.2  xtable_1.7-4
##  [9] xlsx_0.5.7     xlsxjars_0.6.1 rJava_0.9-7    knitr_1.11
##
## loaded via a namespace (and not attached):
##  [1] MASS_7.3-39     Rcpp_0.11.5     colorspace_1.2-6 digest_0.6.8
##  [5] evaluate_0.7.2  formatR_1.2     grid_3.1.2       gtable_0.1.2
##  [9] highr_0.5       labeling_0.3    munsell_0.4.2    proto_0.3-10
## [13] scales_0.2.4     tools_3.1.2

#  packages_in_use <- c( sessionInfo()$basePkgs, names( sessionInfo()$loadedOnly ) )
#the_citations_list <- lapply( X=packages_in_use, FUN=citation)
#the_citations_list
```