Bow River Microbiome Analysis 2022

Emilie Diesbourg

01/04/2023

##Introduction

This data comes from a study done in 2022 across 5 sites on the Bow River, Calgary AB and 3 microcosm streams from Pine Creek wastewater treatment plant with varying municipal wastewater effluent exposures. The objective of this study is to determine whether there are changes to the microbiome of freshwater macroinvertebrates with increased exposure to wastewater.

#setwd("Microbiome\_ED")  
library(phyloseq) #phyloseq will be the main package used for structuring microbiome data and diversity comparisons  
library(ggplot2) #For creating graphs

## Warning: package 'ggplot2' was built under R version 4.1.3

library(dplyr) #Helps with data wrangling

## Warning: package 'dplyr' was built under R version 4.1.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(cluster) #Used for cluster analyses  
library(grid)  
library(FSA)

## Warning: package 'FSA' was built under R version 4.1.3

## ## FSA v0.9.3. See citation('FSA') if used in publication.  
## ## Run fishR() for related website and fishR('IFAR') for related book.

library(multcomp)

## Warning: package 'multcomp' was built under R version 4.1.3

## Loading required package: mvtnorm

## Loading required package: survival

## Loading required package: TH.data

## Warning: package 'TH.data' was built under R version 4.1.3

## Loading required package: MASS

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

##   
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':  
##   
## geyser

library(MASS)  
library(glmmTMB)

## Warning: package 'glmmTMB' was built under R version 4.1.3

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.1.3

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

library(vegan)

## Warning: package 'vegan' was built under R version 4.1.3

## Loading required package: permute

## Warning: package 'permute' was built under R version 4.1.3

## Loading required package: lattice

## This is vegan 2.6-4

library(tidyverse) #data wrangling

## Warning: package 'tidyverse' was built under R version 4.1.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --

## v tibble 3.1.8 v purrr 1.0.1  
## v tidyr 1.2.1 v stringr 1.5.0  
## v readr 2.1.3 v forcats 0.5.2

## Warning: package 'tibble' was built under R version 4.1.3

## Warning: package 'tidyr' was built under R version 4.1.3

## Warning: package 'readr' was built under R version 4.1.3

## Warning: package 'purrr' was built under R version 4.1.3

## Warning: package 'stringr' was built under R version 4.1.3

## Warning: package 'forcats' was built under R version 4.1.3

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x gridExtra::combine() masks dplyr::combine()  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()  
## x MASS::select() masks dplyr::select()

library(ape)

## Warning: package 'ape' was built under R version 4.1.3

library(microbiome) #For microbiome analyses

##   
## microbiome R package (microbiome.github.com)  
##   
##   
##   
## Copyright (C) 2011-2021 Leo Lahti,   
## Sudarshan Shetty et al. <microbiome.github.io>  
##   
##   
## Attaching package: 'microbiome'  
##   
## The following object is masked from 'package:vegan':  
##   
## diversity  
##   
## The following object is masked from 'package:ggplot2':  
##   
## alpha  
##   
## The following object is masked from 'package:base':  
##   
## transform

#if (!require("BiocManager", quietly = TRUE)) #Installing microbiome package  
 #install.packages("BiocManager")  
#BiocManager::install("microbiome")  
library(PERFect) #Permutation filtration for microbiome data. Used when comparing beta diversities across sites/locations.

## Loading required package: sn

## Warning: package 'sn' was built under R version 4.1.3

## Loading required package: stats4  
##   
## Attaching package: 'sn'  
##   
## The following object is masked from 'package:stats':  
##   
## sd

#if(!requireNamespace("BiocManager", quietly = TRUE))  
 #install.packages("BiocManager")  
#BiocManager::install("PERFect")  
library(knitr) #For R Markdown knitting

## Warning: package 'knitr' was built under R version 4.1.3

library(kableExtra)

## Warning: package 'kableExtra' was built under R version 4.1.3

##   
## Attaching package: 'kableExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## group\_rows

library(speedyseq)

##   
## Attaching package: 'speedyseq'  
##   
## The following objects are masked from 'package:phyloseq':  
##   
## filter\_taxa, plot\_bar, plot\_heatmap, plot\_tree, psmelt, tax\_glom,  
## tip\_glom, transform\_sample\_counts

#if (!requireNamespace("remotes", quietly = TRUE))  
 #install.packages("remotes")  
#remotes::install\_github("mikemc/speedyseq")  
library(colorspace)

## Warning: package 'colorspace' was built under R version 4.1.3

library(RColorBrewer)

## Warning: package 'RColorBrewer' was built under R version 4.1.3

library(picante)

## Warning: package 'picante' was built under R version 4.1.3

## Loading required package: nlme  
##   
## Attaching package: 'nlme'  
##   
## The following object is masked from 'package:dplyr':  
##   
## collapse  
##   
##   
## Attaching package: 'picante'  
##   
## The following object is masked from 'package:sn':  
##   
## psc

library(ggpubr)

## Warning: package 'ggpubr' was built under R version 4.1.3

## Registered S3 methods overwritten by 'car':  
## method from  
## hist.boot FSA   
## confint.boot FSA   
##   
## Attaching package: 'ggpubr'  
##   
## The following object is masked from 'package:ape':  
##   
## rotate

library(data.table)

## Warning: package 'data.table' was built under R version 4.1.3

##   
## Attaching package: 'data.table'  
##   
## The following object is masked from 'package:purrr':  
##   
## transpose  
##   
## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

#devtools::install\_github("microsud/microbiomeutilities")  
library(microbiomeutilities)

## Warning: replacing previous import 'ggplot2::alpha' by 'microbiome::alpha' when  
## loading 'microbiomeutilities'

library(imsig)

## Warning: package 'imsig' was built under R version 4.1.3

##   
## Attaching package: 'imsig'  
##   
## The following object is masked from 'package:phyloseq':  
##   
## plot\_network

library(phangorn) #For phylogenetic tree

## Warning: package 'phangorn' was built under R version 4.1.3

##   
## Attaching package: 'phangorn'  
##   
## The following object is masked from 'package:colorspace':  
##   
## coords  
##   
## The following object is masked from 'package:microbiome':  
##   
## diversity  
##   
## The following objects are masked from 'package:vegan':  
##   
## diversity, treedist

library(metacoder)

## Warning: package 'metacoder' was built under R version 4.1.3

## This is metacoder verison 0.3.5 (stable)  
##   
## Attaching package: 'metacoder'  
##   
## The following object is masked from 'package:speedyseq':  
##   
## filter\_taxa  
##   
## The following object is masked from 'package:microbiome':  
##   
## taxa  
##   
## The following object is masked from 'package:ape':  
##   
## complement  
##   
## The following object is masked from 'package:ggplot2':  
##   
## map\_data  
##   
## The following object is masked from 'package:phyloseq':  
##   
## filter\_taxa

library(tibble)  
#if (!require("BiocManager", quietly = TRUE))  
 #install.packages("BiocManager")  
  
#BiocManager::install("DESeq2")  
library(DESeq2)

## Loading required package: S4Vectors

## Warning: package 'S4Vectors' was built under R version 4.1.3

## Loading required package: BiocGenerics  
##   
## Attaching package: 'BiocGenerics'  
##   
## The following object is masked from 'package:sn':  
##   
## sd  
##   
## The following object is masked from 'package:gridExtra':  
##   
## combine  
##   
## The following objects are masked from 'package:dplyr':  
##   
## combine, intersect, setdiff, union  
##   
## The following objects are masked from 'package:stats':  
##   
## IQR, mad, sd, var, xtabs  
##   
## The following objects are masked from 'package:base':  
##   
## anyDuplicated, append, as.data.frame, basename, cbind, colnames,  
## dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,  
## grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,  
## order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,  
## rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,  
## union, unique, unsplit, which.max, which.min  
##   
##   
## Attaching package: 'S4Vectors'  
##   
## The following objects are masked from 'package:data.table':  
##   
## first, second  
##   
## The following object is masked from 'package:tidyr':  
##   
## expand  
##   
## The following objects are masked from 'package:dplyr':  
##   
## first, rename  
##   
## The following objects are masked from 'package:base':  
##   
## expand.grid, I, unname  
##   
## Loading required package: IRanges  
##   
## Attaching package: 'IRanges'  
##   
## The following object is masked from 'package:metacoder':  
##   
## reverse  
##   
## The following object is masked from 'package:data.table':  
##   
## shift  
##   
## The following object is masked from 'package:nlme':  
##   
## collapse  
##   
## The following object is masked from 'package:microbiome':  
##   
## coverage  
##   
## The following object is masked from 'package:purrr':  
##   
## reduce  
##   
## The following objects are masked from 'package:dplyr':  
##   
## collapse, desc, slice  
##   
## The following object is masked from 'package:phyloseq':  
##   
## distance  
##   
## The following object is masked from 'package:grDevices':  
##   
## windows  
##   
## Loading required package: GenomicRanges

## Warning: package 'GenomicRanges' was built under R version 4.1.2

## Loading required package: GenomeInfoDb

## Warning: package 'GenomeInfoDb' was built under R version 4.1.2

## Loading required package: SummarizedExperiment  
## Loading required package: MatrixGenerics  
## Loading required package: matrixStats

## Warning: package 'matrixStats' was built under R version 4.1.3

##   
## Attaching package: 'matrixStats'  
##   
## The following object is masked from 'package:dplyr':  
##   
## count  
##   
##   
## Attaching package: 'MatrixGenerics'  
##   
## The following objects are masked from 'package:matrixStats':  
##   
## colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,  
## colCounts, colCummaxs, colCummins, colCumprods, colCumsums,  
## colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,  
## colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,  
## colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,  
## colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,  
## colWeightedMeans, colWeightedMedians, colWeightedSds,  
## colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,  
## rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,  
## rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,  
## rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,  
## rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,  
## rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,  
## rowWeightedMads, rowWeightedMeans, rowWeightedMedians,  
## rowWeightedSds, rowWeightedVars  
##   
## Loading required package: Biobase  
## Welcome to Bioconductor  
##   
## Vignettes contain introductory material; view with  
## 'browseVignettes()'. To cite Bioconductor, see  
## 'citation("Biobase")', and for packages 'citation("pkgname")'.  
##   
##   
## Attaching package: 'Biobase'  
##   
## The following object is masked from 'package:MatrixGenerics':  
##   
## rowMedians  
##   
## The following objects are masked from 'package:matrixStats':  
##   
## anyMissing, rowMedians  
##   
## The following object is masked from 'package:phyloseq':  
##   
## sampleNames

##Phyloseq object

For a phyloseq object you need 3 pieces of data:

1. The ASV table with the genetic barcode identifier (the ASV) of each identified bacterial taxa within each insect sample ran (ex. KK100)
2. The taxa table with the genetic barcode of each unique ASV and the respective bacterial classification “Kingdom”, “Phylum”, “Class”, “Order”, “Family”, “Genus”.
3. A metadata file with the sample names (ex. KK100) and any other variables that were measured. Ex. the family of insect or fish organism collected, the site the organism was collected at, the water temperature, the distance to waste water treatment plant, the weight and length of organism, etc.

The phyloseq object will merge all this data together in one object that can be called in by various functions in the phyloseq and microbiome packages.

#Importing ASV table  
ASVseq<-read.csv("ASV\_data\_2022\_ED.csv")  
  
#The row names have to be the ASV identifier (genetic barcode) in order for it to be read. The following lines of code are to re-organize the row names  
n1 <- ASVseq$X #Currently the ASVs are being called "X" in the first column. We want them to be unnamed in the row names column  
ASVseq <- ASVseq[,-1] #This removes the first column of the ASV dataframe  
rownames(ASVseq) <- n1 #This moves the ASVs to the rownames column of the dataframe  
  
ASVtable <- as.matrix(ASVseq) #must be a matrix so we convert to a matrix  
#View(ASVtable) #The ASVs are now the row names  
  
#Importing taxa table  
taxonomy <-read.csv ("Taxonomy\_ED.csv")  
  
#Using the same steps as above with the ASV table to make the genetic barcode the rowname  
n2 <- taxonomy$X  
taxonomy <- taxonomy[,-1]  
rownames(taxonomy) <- n2  
  
taxa\_table<-as.matrix(taxonomy) #Turn into a matrix  
#View(taxa\_table) #ASVs are now the row names  
  
#Importing metadata file  
metadata<-read.csv("metadata\_ED.csv")  
  
metadata$Site<-factor(metadata$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3")) #Order sites  
  
#Need to make sample names as row names so we follow the same steps as above.   
n3 <- metadata$Study\_ID  
  
# move names to row headings  
metadata <- metadata[,-1]  
rownames(metadata) <- n3  
  
#View(metadata) #sample names are now row names  
  
#Combining phyloseq object  
ASVtable = otu\_table(ASVtable, taxa\_are\_rows = TRUE) #Taxa are rows means that the ASVs of each identified microbe are the row names which we made sure was true  
taxa\_table = tax\_table(taxa\_table)  
  
#Make sure the ASV table and taxa table are both matricies before trying to turn into phyloseq object  
ps <- phyloseq(ASVtable, sample\_data(metadata), taxa\_table)

##Filtering/pre-processing steps

Experimental samples:

Removing low confidence data: I removed all ASVs identified as Eukaryota and phyla that were identified as . This removed 39 and 393 taxa respectively.

Removing low abundance taxa (including singletons and doubletons): I removed taxa whose sums of reads were less than or equal to 5 across all samples. This removed 5533 ASVs.

Removing low read samples: I removed all samples whose total reads were less than 2000. This removed 10 samples.

Rarefying: Before statistical analysis the reads will be rarefied to the minimum sampling depth to ensure even sampling depth. The minimum even sampling depth is 2259 reads.

In total, 30.5% of the taxa (ASVs) were filtered out but only 0.52% of the entire dataset (reads) were filtered out. This may have consequences for diversity measures.

Blank Samples:

Removing low abundance taxa (including singletons and doubletons): I removed taxa whose sums of reads were less than or equal to 10 across all samples. This removed 90 ASVs.

Rarefying: Before statistical analysis the reads will be rarefied to the minimum sampling depth to ensure even sampling depth. The minimum even sampling depth is 7946 reads.

In total, 19.1% of the taxa (ASVs) were filtered out but only 0.51% of the entire dataset (reads) were filtered out.

Exploratory data techniques:

Graphing reads per sample versus sample counts and abundance across samples distributions

#Subsetting experimental samples only from the original phyloseq object  
psexp<-subset\_samples(ps, Sample\_Type == "Sample") #Removing ASVs where the abundance is 0 (after removing the blank samples)  
psexp<-prune\_taxa(taxa\_sums(psexp) > 0, psexp)  
psexp #30,221 taxa and 350 samples

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 30221 taxa and 350 samples ]:  
## sample\_data() Sample Data: [ 350 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 30221 taxa by 7 taxonomic ranks ]:  
## taxa are rows

#Removing Eukaryotes  
ps0 <- subset\_taxa(psexp, Kingdom != "Eukaryota")  
ntaxa(psexp)-ntaxa(ps0) #Got rid of 39 ASVs

## [1] 39

#Removing low confidence data (where phylum could not be assigned)  
table(tax\_table(ps0)[,"Phylum"],exclude=NULL)

##   
## Abditibacteriota Acidobacteriota   
## 50 930   
## Actinobacteriota Armatimonadota   
## 2215 7   
## Bacteroidota Bdellovibrionota   
## 7802 309   
## Caldatribacteriota Caldisericota   
## 1 5   
## Calditrichota Campilobacterota   
## 3 29   
## Chloroflexi Cloacimonadota   
## 551 4   
## Cyanobacteria Deferribacterota   
## 1342 25   
## Deferrisomatota Deinococcota   
## 2 53   
## Dependentiae Desulfobacterota   
## 94 596   
## Elusimicrobiota Entotheonellaeota   
## 71 10   
## FCPU426 Fibrobacterota   
## 1 31   
## Firmicutes Fusobacteriota   
## 2712 27   
## Gemmatimonadota Halanaerobiaeota   
## 203 1   
## Hydrogenedentes Latescibacterota   
## 5 50   
## LCP-89 Margulisbacteria   
## 5 5   
## MBNT15 Methylomirabilota   
## 20 21   
## Myxococcota NB1-j   
## 615 18   
## Nitrospinota Nitrospirota   
## 7 61   
## NKB15 Patescibacteria   
## 1 208   
## Planctomycetota Proteobacteria   
## 908 10261   
## RCP2-54 Rs-K70 termite group   
## 12 16   
## SAR324 clade(Marine group B) Spirochaetota   
## 30 29   
## Sumerlaeota Sva0485   
## 13 22   
## Synergistota Thermotogota   
## 14 1   
## Verrucomicrobiota WPS-2   
## 368 18   
## WS2 Zixibacteria   
## 1 6   
## <NA>   
## 393

ps1<-subset\_taxa(ps0, !is.na(Phylum) & !Phylum %in% c("","uncharacterized")) #Removes the phyla characterized as NA  
ntaxa(ps0) - ntaxa(ps1) #Got rid of 393 ASVs

## [1] 393

#Getting rid of singletons and doubletons  
any(taxa\_sums(ps1)>=2) #TRUE. Means there are ASVs with a sum of sequences less than 2 across all samples.

## [1] TRUE

ps2 <- prune\_taxa(taxa\_sums(ps1) > 2, ps1) #Getting rid of singletons and doubletons. (Taxa that only showed up in one or two sequences across all samples)  
any(taxa\_sums(ps2) <2) #FALSE. Means they were removed.

## [1] FALSE

ntaxa(ps1)- ntaxa(ps2) #Got rid of 3205 ASVs.

## [1] 3205

#How many bacterial taxa do we have/did we remove  
ntaxa(psexp) #30,221 before any filtering was done.

## [1] 30221

ntaxa(ps0) #30,182 after removing Eukaryotes (39 removed)

## [1] 30182

ntaxa(ps1) #29,789 after phyla with NA were removed (393 removed).

## [1] 29789

ntaxa(ps2) #26,584 after the samples with a read sum less than 2 were removed. (3205 removed).

## [1] 26584

# Compute prevalence of each feature, store as data.frame  
prevdf=apply(X=otu\_table(ps2),MARGIN=ifelse(taxa\_are\_rows(ps2),yes=1,no=2),FUN = function(x){sum(x > 0)})  
# Add taxonomy and total read counts to this data.frame  
prevdf= data.frame(Prevalence= prevdf,TotalAbundance=taxa\_sums(ps2),tax\_table(ps2))  
  
plyr::ddply(prevdf,"Phylum",function(df1){cbind(mean(df1$Prevalence),sum(df1$Prevalence))})

## Phylum 1 2  
## 1 Abditibacteriota 1.255319 59  
## 2 Acidobacteriota 1.887059 1604  
## 3 Actinobacteriota 2.368753 4882  
## 4 Armatimonadota 1.285714 9  
## 5 Bacteroidota 4.054137 27633  
## 6 Bdellovibrionota 1.431818 378  
## 7 Caldatribacteriota 1.000000 1  
## 8 Caldisericota 1.000000 4  
## 9 Calditrichota 1.000000 3  
## 10 Campilobacterota 3.137931 91  
## 11 Chloroflexi 1.954000 977  
## 12 Cloacimonadota 1.000000 2  
## 13 Cyanobacteria 4.576923 5831  
## 14 Deferribacterota 19.863636 437  
## 15 Deferrisomatota 1.000000 2  
## 16 Deinococcota 2.041667 98  
## 17 Dependentiae 1.256757 93  
## 18 Desulfobacterota 2.640449 1410  
## 19 Elusimicrobiota 1.327586 77  
## 20 Entotheonellaeota 1.000000 10  
## 21 FCPU426 1.000000 1  
## 22 Fibrobacterota 1.192308 31  
## 23 Firmicutes 3.723179 9509  
## 24 Fusobacteriota 1.800000 45  
## 25 Gemmatimonadota 1.436464 260  
## 26 Halanaerobiaeota 1.000000 1  
## 27 Hydrogenedentes 1.000000 5  
## 28 Latescibacterota 1.210526 46  
## 29 LCP-89 1.000000 5  
## 30 Margulisbacteria 1.400000 7  
## 31 MBNT15 1.117647 19  
## 32 Methylomirabilota 1.277778 23  
## 33 Myxococcota 2.078947 1027  
## 34 NB1-j 1.214286 17  
## 35 Nitrospinota 3.000000 21  
## 36 Nitrospirota 4.035088 230  
## 37 NKB15 1.000000 1  
## 38 Patescibacteria 1.664804 298  
## 39 Planctomycetota 1.860553 1481  
## 40 Proteobacteria 4.005060 36410  
## 41 RCP2-54 2.333333 28  
## 42 Rs-K70 termite group 10.500000 168  
## 43 SAR324 clade(Marine group B) 1.360000 34  
## 44 Spirochaetota 1.269231 33  
## 45 Sumerlaeota 1.000000 8  
## 46 Sva0485 1.900000 38  
## 47 Synergistota 9.642857 135  
## 48 Thermotogota 1.000000 1  
## 49 Verrucomicrobiota 1.622642 516  
## 50 WPS-2 1.250000 20  
## 51 WS2 1.000000 1  
## 52 Zixibacteria 2.166667 13

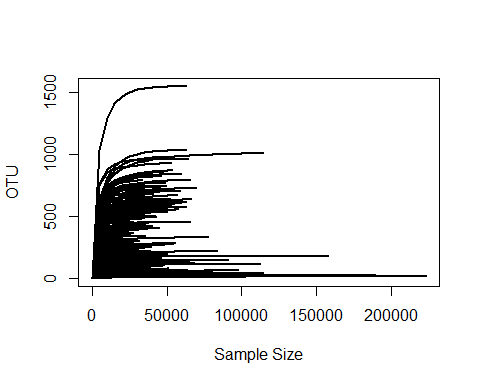
#Shows mean and total prevalence of taxa within each phylum. The ones that are only prevalent once in each phylum might be worth removing. This means they only showed up in one sample across all samples but might have a high abundance (total number of reads).   
  
#There are a couple of phyla with only a prevalence sum of one or two so these might be worth removing. (these include Caldatribacteriota, Deferrisomatota, FCPU426, Halanaerobiaeota, NKB15, Thermotogota, WS2)  
  
plyr::ddply(prevdf, "Phylum", function(df1){  
 data.frame(mean\_prevalence=mean(df1$Prevalence),total\_abundance=sum(df1$TotalAbundance,na.rm = T),stringsAsFactors = F)  
 })

## Phylum mean\_prevalence total\_abundance  
## 1 Abditibacteriota 1.255319 1105  
## 2 Acidobacteriota 1.887059 32957  
## 3 Actinobacteriota 2.368753 169503  
## 4 Armatimonadota 1.285714 42  
## 5 Bacteroidota 4.054137 2314994  
## 6 Bdellovibrionota 1.431818 4418  
## 7 Caldatribacteriota 1.000000 5  
## 8 Caldisericota 1.000000 19  
## 9 Calditrichota 1.000000 21  
## 10 Campilobacterota 3.137931 10745  
## 11 Chloroflexi 1.954000 12794  
## 12 Cloacimonadota 1.000000 8  
## 13 Cyanobacteria 4.576923 1983528  
## 14 Deferribacterota 19.863636 49748  
## 15 Deferrisomatota 1.000000 9  
## 16 Deinococcota 2.041667 970  
## 17 Dependentiae 1.256757 1177  
## 18 Desulfobacterota 2.640449 21476  
## 19 Elusimicrobiota 1.327586 928  
## 20 Entotheonellaeota 1.000000 69  
## 21 FCPU426 1.000000 3  
## 22 Fibrobacterota 1.192308 307  
## 23 Firmicutes 3.723179 1393513  
## 24 Fusobacteriota 1.800000 741  
## 25 Gemmatimonadota 1.436464 2840  
## 26 Halanaerobiaeota 1.000000 3  
## 27 Hydrogenedentes 1.000000 16  
## 28 Latescibacterota 1.210526 479  
## 29 LCP-89 1.000000 120  
## 30 Margulisbacteria 1.400000 42  
## 31 MBNT15 1.117647 142  
## 32 Methylomirabilota 1.277778 110  
## 33 Myxococcota 2.078947 30870  
## 34 NB1-j 1.214286 102  
## 35 Nitrospinota 3.000000 438  
## 36 Nitrospirota 4.035088 8737  
## 37 NKB15 1.000000 26  
## 38 Patescibacteria 1.664804 3482  
## 39 Planctomycetota 1.860553 19538  
## 40 Proteobacteria 4.005060 7025495  
## 41 RCP2-54 2.333333 876  
## 42 Rs-K70 termite group 10.500000 12266  
## 43 SAR324 clade(Marine group B) 1.360000 285  
## 44 Spirochaetota 1.269231 597  
## 45 Sumerlaeota 1.000000 63  
## 46 Sva0485 1.900000 304  
## 47 Synergistota 9.642857 11992  
## 48 Thermotogota 1.000000 16  
## 49 Verrucomicrobiota 1.622642 4728  
## 50 WPS-2 1.250000 179  
## 51 WS2 1.000000 4  
## 52 Zixibacteria 2.166667 133

#There are a few that only show up in one sample and have an abundance less than 10...these might need to be removed but I will keep them for now  
  
#Creating a rarefaction curve  
tab <- otu\_table(ps2)  
class(tab) <- "matrix" # as.matrix() will do nothing

## Warning in class(tab) <- "matrix": Setting class(x) to "matrix" sets attribute  
## to NULL; result will no longer be an S4 object

## you get a warning here, but this is what we need to have  
tab <- t(tab) # transpose observations to rows  
library(vegan)  
rare <- rarecurve(tab, step=5000, lwd=2, ylab="OTU", label=F)



#The rarefaction curves are very spread out. This means that there are different sequencing depths and standardizing the data will be required for analysis.   
  
sort(sample\_sums(ps2)) #minimum is 57, maximum is 222,954

## KK1926 KK2026 KK1922 KK1929 KK2081 KK2012 KK1925 KK2115 KK1910 KK1918 KK1924   
## 57 138 663 694 1149 1286 1345 1764 1816 1962 2259   
## KK1911 KK1944 KK1936 KK2079 KK1937 KK2136 KK2171 KK1920 KK1935 KK1948 KK1912   
## 2818 3673 4056 4302 5077 5223 5272 5677 5873 6256 6826   
## KK2170 KK2027 KK1950 KK1932 KK2153 KK2091 KK1880 KK2111 KK1923 KK1949 KK1978   
## 6833 7467 7734 8140 8164 8227 8344 8380 9025 9115 9154   
## KK2137 KK1979 KK1916 KK1951 KK2132 KK2006 KK1909 KK2017 KK1919 KK1913 KK2135   
## 9238 9256 9349 9591 10240 10431 10591 10708 10816 10894 11006   
## KK2032 KK2086 KK2010 KK2018 KK1930 KK2084 KK2092 KK2155 KK1890 KK1980 KK2168   
## 11076 11218 11602 11731 11984 12342 12964 13067 13212 13293 13462   
## KK1946 KK2098 KK2133 KK2107 KK1818 KK1934 KK1917 KK2146 KK2110 KK1931 KK2022   
## 13660 13982 13990 14064 14176 14267 14300 14324 14455 14509 15062   
## KK1988 KK1927 KK2106 KK2158 KK1826 KK2126 KK1887 KK1888 KK1901 KK2105 KK2008   
## 15293 15383 15679 16022 16237 16258 16382 16717 16932 16962 17221   
## KK2116 KK1828 KK2016 KK2121 KK1829 KK1827 KK1987 KK1907 KK2076 KK2169 KK1881   
## 17277 17433 17823 17929 17988 18005 18062 18130 18365 18367 18417   
## KK2104 KK1945 KK1811 KK1830 KK1825 KK2082 KK2156 KK2095 KK1824 KK1914 KK1982   
## 18561 18601 18620 18663 18832 18889 18985 19118 19533 19658 19845   
## KK1993 KK1823 KK1874 KK1933 KK1822 KK1820 KK1814 KK2009 KK2167 KK2103 KK2031   
## 19857 20265 20327 20390 20637 21012 21083 21197 21288 21392 21419   
## KK2102 KK1843 KK1991 KK2154 KK2108 KK2150 KK2019 KK2088 KK1821 KK1813 KK1872   
## 21517 21552 21755 22124 22266 22498 22696 22945 23187 23226 23233   
## KK2024 KK1895 KK1810 KK1986 KK2166 KK1819 KK2113 KK1841 KK2077 KK1886 KK2128   
## 23352 23654 23854 23877 24000 24116 24154 24407 24479 24542 24565   
## KK1816 KK2163 KK2165 KK1809 KK1882 KK2090 KK1817 KK1812 KK2003 KK1990 KK2112   
## 24717 24748 24923 25075 25087 25092 25165 25473 25727 26069 26116   
## KK2157 KK1815 KK1836 KK2101 KK1877 KK1849 KK1915 KK1938 KK2130 KK2122 KK2148   
## 26398 26779 27130 27327 27373 27378 27477 27502 27732 27848 27958   
## KK2001 KK1854 KK1891 KK1995 KK1992 KK1878 KK1985 KK2085 KK1870 KK1876 KK1844   
## 28096 28106 28436 28646 28710 28726 28841 29151 29163 29433 29773   
## KK1989 KK1871 KK1892 KK1853 KK1889 KK2028 KK1852 KK1983 KK2160 KK1866 KK2054   
## 30019 30074 30166 30243 30669 30803 31024 31174 31197 31274 31275   
## KK2145 KK1972 KK2099 KK2087 KK1867 KK1869 KK1947 KK2109 KK2139 KK1837 KK2007   
## 31342 31425 31642 31676 31922 32109 32119 32222 32416 32708 32727   
## KK2048 KK1838 KK2052 KK1834 KK1899 KK1868 KK2161 KK2000 KK1835 KK1842 KK1875   
## 32771 32807 32862 32936 32966 32968 33163 33267 33275 33414 33462   
## KK1858 KK1846 KK1859 KK1897 KK2138 KK2093 KK2117 KK1883 KK1833 KK1865 KK1832   
## 33493 33750 33989 34095 34359 34824 34925 34951 35287 35362 35410   
## KK1973 KK1873 KK2023 KK1840 KK2152 KK1851 KK1996 KK2025 KK1862 KK1845 KK2071   
## 35603 35703 35733 35871 35895 36021 36661 36853 37325 37326 37682   
## KK1994 KK2083 KK2159 KK1857 KK1981 KK2172 KK1905 KK2011 KK1847 KK2002 KK2005   
## 37850 37939 38225 38349 38379 38623 38936 39073 39123 39142 39146   
## KK2100 KK1860 KK1997 KK2042 KK1864 KK2127 KK1884 KK1971 KK1898 KK1896 KK1921   
## 39189 39239 39503 39593 39739 39842 40203 40344 40825 41536 42438   
## KK2114 KK1850 KK2004 KK2034 KK1856 KK1863 KK1885 KK1893 KK2070 KK1906 KK2162   
## 42843 42946 43391 43394 45006 45612 45870 46261 46374 46493 46702   
## KK1969 KK1855 KK2029 KK1848 KK2013 KK2030 KK2033 KK1940 KK1977 KK1928 KK2129   
## 46745 47699 47852 48031 48236 48613 48640 48960 49722 49969 50409   
## KK1879 KK1831 KK2080 KK2089 KK2069 KK1861 KK1975 KK1904 KK1970 KK2060 KK2036   
## 50540 50610 50671 50991 51609 52380 52465 52648 52691 53049 53224   
## KK1902 KK1984 KK2014 KK2065 KK2134 KK1967 KK2015 KK2149 KK1965 KK1900 KK2059   
## 53932 54085 54458 55653 55719 55790 55965 56136 56768 56946 57459   
## KK1966 KK1943 KK2164 KK1939 KK1968 KK1976 KK2035 KK1894 KK1941 KK1961 KK2144   
## 57885 58743 58892 59357 59518 60788 61528 62676 62726 62745 62756   
## KK1908 KK2147 KK2123 KK2141 KK2094 KK2020 KK1903 KK1962 KK2131 KK2021 KK2120   
## 62949 63238 63841 64064 64443 65417 65888 66105 68067 68551 68635   
## KK1942 KK2058 KK2041 KK2151 KK2050 KK2125 KK2062 KK2097 KK2142 KK1998 KK1963   
## 69433 70437 70538 71860 72081 72093 73495 73520 75073 77466 78150   
## KK2056 KK2047 KK2096 KK2039 KK2140 KK2143 KK1964 KK2055 KK2124 KK2051 KK2040   
## 78716 79545 80237 81701 83329 83812 83869 84110 85135 85450 86063   
## KK2064 KK2044 KK2066 KK2049 KK1974 KK2057 KK2119 KK2068 KK2063 KK2043 KK2046   
## 86981 88322 89037 89043 91191 92575 97714 98630 101224 102205 108927   
## KK2053 KK2118 KK2038 KK1839 KK2061 KK1999 KK2037 KK2045 KK2067   
## 110329 112187 114369 114603 122037 126300 157636 188889 222954

mean(sample\_sums(ps2)) #37,494

## [1] 37494.18

ps3 <- subset\_samples(ps2, sample\_sums(ps2) > 2000)   
sort(sample\_sums(ps3)) #Minimum sampling depth is now 2259 reads. Got rid of 10 samples.

## KK1924 KK1911 KK1944 KK1936 KK2079 KK1937 KK2136 KK2171 KK1920 KK1935 KK1948   
## 2259 2818 3673 4056 4302 5077 5223 5272 5677 5873 6256   
## KK1912 KK2170 KK2027 KK1950 KK1932 KK2153 KK2091 KK1880 KK2111 KK1923 KK1949   
## 6826 6833 7467 7734 8140 8164 8227 8344 8380 9025 9115   
## KK1978 KK2137 KK1979 KK1916 KK1951 KK2132 KK2006 KK1909 KK2017 KK1919 KK1913   
## 9154 9238 9256 9349 9591 10240 10431 10591 10708 10816 10894   
## KK2135 KK2032 KK2086 KK2010 KK2018 KK1930 KK2084 KK2092 KK2155 KK1890 KK1980   
## 11006 11076 11218 11602 11731 11984 12342 12964 13067 13212 13293   
## KK2168 KK1946 KK2098 KK2133 KK2107 KK1818 KK1934 KK1917 KK2146 KK2110 KK1931   
## 13462 13660 13982 13990 14064 14176 14267 14300 14324 14455 14509   
## KK2022 KK1988 KK1927 KK2106 KK2158 KK1826 KK2126 KK1887 KK1888 KK1901 KK2105   
## 15062 15293 15383 15679 16022 16237 16258 16382 16717 16932 16962   
## KK2008 KK2116 KK1828 KK2016 KK2121 KK1829 KK1827 KK1987 KK1907 KK2076 KK2169   
## 17221 17277 17433 17823 17929 17988 18005 18062 18130 18365 18367   
## KK1881 KK2104 KK1945 KK1811 KK1830 KK1825 KK2082 KK2156 KK2095 KK1824 KK1914   
## 18417 18561 18601 18620 18663 18832 18889 18985 19118 19533 19658   
## KK1982 KK1993 KK1823 KK1874 KK1933 KK1822 KK1820 KK1814 KK2009 KK2167 KK2103   
## 19845 19857 20265 20327 20390 20637 21012 21083 21197 21288 21392   
## KK2031 KK2102 KK1843 KK1991 KK2154 KK2108 KK2150 KK2019 KK2088 KK1821 KK1813   
## 21419 21517 21552 21755 22124 22266 22498 22696 22945 23187 23226   
## KK1872 KK2024 KK1895 KK1810 KK1986 KK2166 KK1819 KK2113 KK1841 KK2077 KK1886   
## 23233 23352 23654 23854 23877 24000 24116 24154 24407 24479 24542   
## KK2128 KK1816 KK2163 KK2165 KK1809 KK1882 KK2090 KK1817 KK1812 KK2003 KK1990   
## 24565 24717 24748 24923 25075 25087 25092 25165 25473 25727 26069   
## KK2112 KK2157 KK1815 KK1836 KK2101 KK1877 KK1849 KK1915 KK1938 KK2130 KK2122   
## 26116 26398 26779 27130 27327 27373 27378 27477 27502 27732 27848   
## KK2148 KK2001 KK1854 KK1891 KK1995 KK1992 KK1878 KK1985 KK2085 KK1870 KK1876   
## 27958 28096 28106 28436 28646 28710 28726 28841 29151 29163 29433   
## KK1844 KK1989 KK1871 KK1892 KK1853 KK1889 KK2028 KK1852 KK1983 KK2160 KK1866   
## 29773 30019 30074 30166 30243 30669 30803 31024 31174 31197 31274   
## KK2054 KK2145 KK1972 KK2099 KK2087 KK1867 KK1869 KK1947 KK2109 KK2139 KK1837   
## 31275 31342 31425 31642 31676 31922 32109 32119 32222 32416 32708   
## KK2007 KK2048 KK1838 KK2052 KK1834 KK1899 KK1868 KK2161 KK2000 KK1835 KK1842   
## 32727 32771 32807 32862 32936 32966 32968 33163 33267 33275 33414   
## KK1875 KK1858 KK1846 KK1859 KK1897 KK2138 KK2093 KK2117 KK1883 KK1833 KK1865   
## 33462 33493 33750 33989 34095 34359 34824 34925 34951 35287 35362   
## KK1832 KK1973 KK1873 KK2023 KK1840 KK2152 KK1851 KK1996 KK2025 KK1862 KK1845   
## 35410 35603 35703 35733 35871 35895 36021 36661 36853 37325 37326   
## KK2071 KK1994 KK2083 KK2159 KK1857 KK1981 KK2172 KK1905 KK2011 KK1847 KK2002   
## 37682 37850 37939 38225 38349 38379 38623 38936 39073 39123 39142   
## KK2005 KK2100 KK1860 KK1997 KK2042 KK1864 KK2127 KK1884 KK1971 KK1898 KK1896   
## 39146 39189 39239 39503 39593 39739 39842 40203 40344 40825 41536   
## KK1921 KK2114 KK1850 KK2004 KK2034 KK1856 KK1863 KK1885 KK1893 KK2070 KK1906   
## 42438 42843 42946 43391 43394 45006 45612 45870 46261 46374 46493   
## KK2162 KK1969 KK1855 KK2029 KK1848 KK2013 KK2030 KK2033 KK1940 KK1977 KK1928   
## 46702 46745 47699 47852 48031 48236 48613 48640 48960 49722 49969   
## KK2129 KK1879 KK1831 KK2080 KK2089 KK2069 KK1861 KK1975 KK1904 KK1970 KK2060   
## 50409 50540 50610 50671 50991 51609 52380 52465 52648 52691 53049   
## KK2036 KK1902 KK1984 KK2014 KK2065 KK2134 KK1967 KK2015 KK2149 KK1965 KK1900   
## 53224 53932 54085 54458 55653 55719 55790 55965 56136 56768 56946   
## KK2059 KK1966 KK1943 KK2164 KK1939 KK1968 KK1976 KK2035 KK1894 KK1941 KK1961   
## 57459 57885 58743 58892 59357 59518 60788 61528 62676 62726 62745   
## KK2144 KK1908 KK2147 KK2123 KK2141 KK2094 KK2020 KK1903 KK1962 KK2131 KK2021   
## 62756 62949 63238 63841 64064 64443 65417 65888 66105 68067 68551   
## KK2120 KK1942 KK2058 KK2041 KK2151 KK2050 KK2125 KK2062 KK2097 KK2142 KK1998   
## 68635 69433 70437 70538 71860 72081 72093 73495 73520 75073 77466   
## KK1963 KK2056 KK2047 KK2096 KK2039 KK2140 KK2143 KK1964 KK2055 KK2124 KK2051   
## 78150 78716 79545 80237 81701 83329 83812 83869 84110 85135 85450   
## KK2040 KK2064 KK2044 KK2066 KK2049 KK1974 KK2057 KK2119 KK2068 KK2063 KK2043   
## 86063 86981 88322 89037 89043 91191 92575 97714 98630 101224 102205   
## KK2046 KK2053 KK2118 KK2038 KK1839 KK2061 KK1999 KK2037 KK2045 KK2067   
## 108927 110329 112187 114369 114603 122037 126300 157636 188889 222954

#How much data did we filter out from the original phyloseq object?  
  
ntaxa(psexp)-ntaxa(ps3) #filtered 3637 total taxa (ASVs)

## [1] 3637

(1-ntaxa(ps3)/ntaxa(psexp))\*100 #12% of the taxa were removed from the original dataset.

## [1] 12.03468

reads\_per\_OTU <- taxa\_sums(psexp) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU)) #Total number of reads is 13 158 716.

## [1] 13158716

reads\_per\_OTU\_filtered <- taxa\_sums(ps3) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU\_filtered)) #Total number of reads is now 13 112 089.

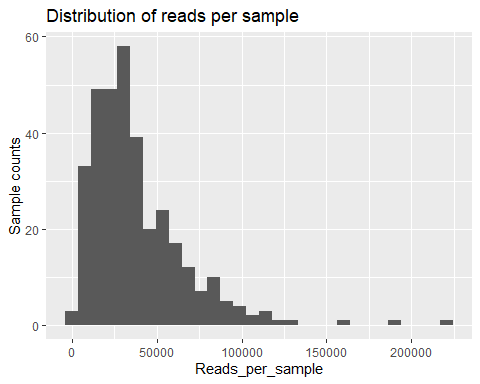
## [1] 13112089

(1-13112089/13158716)\*100 #0.35%. Only filtered out a very small percentage of the total data (reads).

## [1] 0.3543431

#Creating a distribution of reads per sample versus the sample counts.   
ps1\_df= data.table(as(sample\_data(ps3), "data.frame"), Reads\_per\_sample = sample\_sums(ps3), keep.rownames = TRUE)  
ps1\_df\_plot = ggplot(ps1\_df, aes(Reads\_per\_sample)) + geom\_histogram() + ggtitle("Distribution of reads per sample") + ylab("Sample counts")   
  
print(ps1\_df\_plot) #Looks right skewed. A couple of very large reads per sample. Probably the perlidae individuals because they are so big compared to some of the other samples like chironomids. Most are around 20,000 - 40,000 reads per sample though.

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



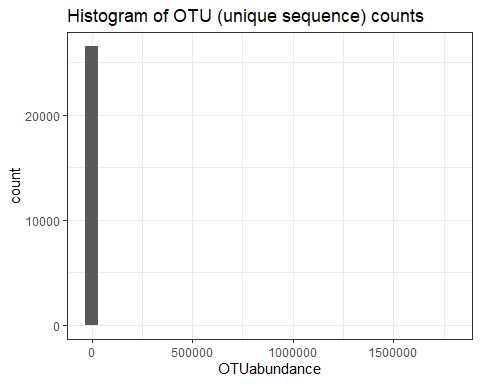
#Now we need to observe the counts of abundance of unique sequences across samples  
ps1.dt.taxa = data.table(tax\_table(ps3), OTUabundance = taxa\_sums(ps3), OTU = taxa\_names(ps3))

## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

## Warning in setDT(ans, key = key): Some columns are a multi-column type (such as  
## a matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

#makes a data table of the sum of abundance of each taxa vs the total number of taxa (ASVs)  
ps1.dt.tax.plot<- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram() + ggtitle("Histogram of OTU (unique sequence) counts") + theme\_bw() #Plots the number of times an ASV is repeated from the reads (abundance) on the x axis and the number of taxa on the y axis  
print(ps1.dt.tax.plot)

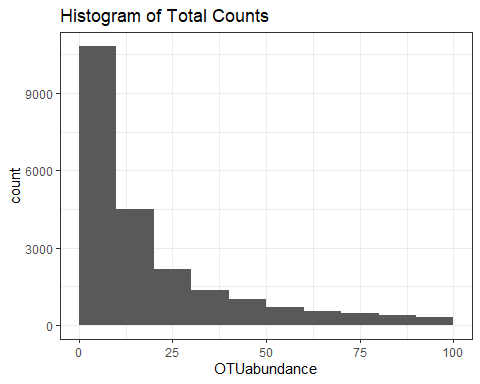
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



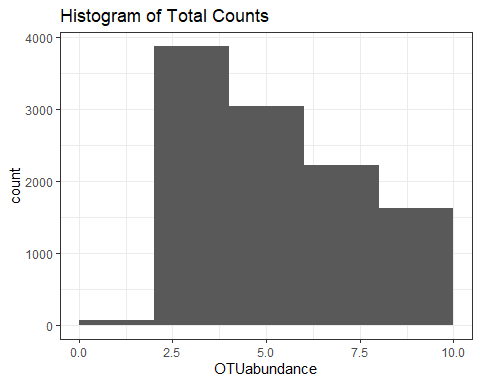
#You can"t see much on the plots panel in R  
  
write.table(ps1.dt.taxa, "ps1\_dt\_tax\_plot.txt", sep = "\t") #makes a table that can open in excel  
ggsave("Histogram of OTU counts.pdf", height = 6, width = 7) #pdf of the plot

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

#This plot shows that there is a very small proportion of ASVs with a really high abundance and most have relatively low abundance. Highly right skewed. We want to see how many have very low abundance because this may indicate that further processing/filtering should be done.   
  
#Zoom in on the lower abundance counts to get a better idea of its distribution  
plot.zoom1 <- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram(breaks=seq(0, 100, by =10)) + ggtitle("Histogram of Total Counts") + theme\_bw()  
print(plot.zoom1)



ggsave("Histogram of OTU counts 1 to 100.pdf", height = 6, width = 7)  
#A very high proportion of ASVs have an abundance of less than 10. This may indicate that they should be filtered out.   
  
plot.zoom2 <- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram(breaks=seq(0, 10, by =2)) + ggtitle("Histogram of Total Counts") + theme\_bw()  
print(plot.zoom2)



ggsave("Histogram of OTU counts 1 to 10.pdf", height = 6, width = 7)  
#Still a high proportion of ASVs with an abundance less than 10. How does this compare to the whole dataset? Would it remove too much of the dataset if we filtered these out? Need to figure that out and then decide if they should be removed.  
  
#Finding total number of reads vs # of reads with low abundance counts  
reads\_per\_OTU <- taxa\_sums(ps3)  
print(sum(reads\_per\_OTU)) #Total number of reads is 13 112 089..

## [1] 13112089

#How many taxa (ASVs) contain less than 5 reads (an abundance of 5 or less)?  
print(length(reads\_per\_OTU[reads\_per\_OTU <=5])) #5571. This seems quite high, but maybe worth removing.

## [1] 5571

#How many reads do these 5571 ASVs contain in total?  
print(sum(reads\_per\_OTU[reads\_per\_OTU <=5])) #21659

## [1] 21659

#What percentage of the total reads is that?  
print((sum(reads\_per\_OTU[reads\_per\_OTU <=5])/sum(reads\_per\_OTU))\*100) #0.17 %. A very small proportion of the entire dataset.

## [1] 0.1651834

#How many ASVs out of the total number of ASVs contains less than or equal to 5 reads?  
ntaxa(ps3) #26,594

## [1] 26584

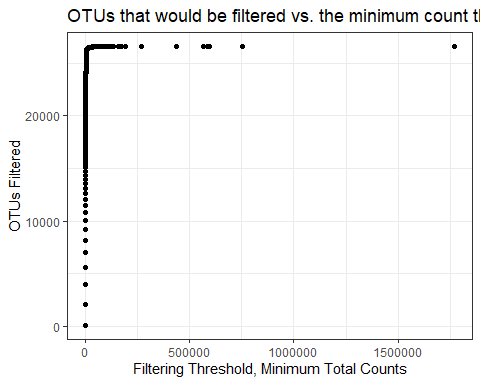
print((5571/26594)\*100) #20.9%

## [1] 20.94833

#20.9% of the ASVs. This sounds like a large number, but they do only contain 0.17% of the total data beacuse many taxa had very low abundance. This means that ASVs with a lower confidence (ie repeated 5 times or less are only a very small part of this dataset). This would affect subsequent diversity analyses though if we filtered them out.   
  
#From this we can conclude that a very small part of the data can have a significant effect on downstream analyses, if methods that use ASVs instead of the information that is contained in the sequences is implemented, such as Unifrac and phylogenetic diversity. Should I use these measures instead? Or use both?  
  
  
#Plotting the number of ASVs that would be filtered out given a minimum read threshold (min abundance of ASVs)  
ps1.dt.taxa = data.table(tax\_table(ps3), OTUabundance = taxa\_sums(ps3), OTU = taxa\_names(ps3))

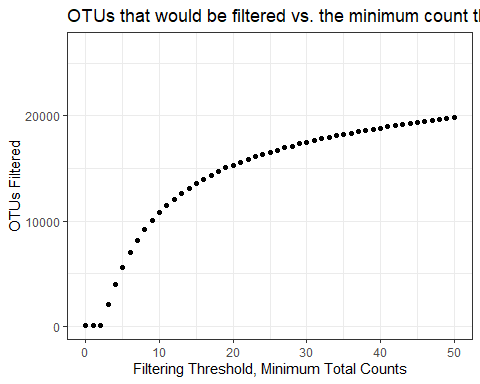
## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.  
  
## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

ps1.cumsum = ps1.dt.taxa[, .N, by = OTUabundance]  
setkey(ps1.cumsum, OTUabundance)  
ps1.cumsum[, CumSum := cumsum(N)]  
# Define the plot  
ps1.cumsum.plot = ggplot(ps1.cumsum, aes(OTUabundance, CumSum)) +   
 geom\_point() +  
 xlab("Filtering Threshold, Minimum Total Counts") +  
 ylab("OTUs Filtered") +  
 ggtitle("OTUs that would be filtered vs. the minimum count threshold") + theme\_bw()  
  
print(ps1.cumsum.plot) #Difficult to see the smaller values so we need to zoom in



ps1.cumsum.plot.zoom <- ps1.cumsum.plot + xlim(0, 50) + theme\_bw()   
  
print(ps1.cumsum.plot.zoom)

## Warning: Removed 1620 rows containing missing values (`geom\_point()`).



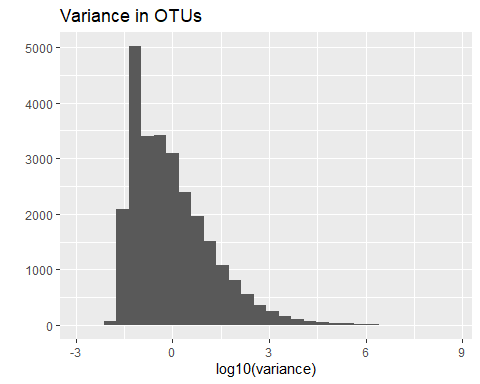
#Could maybe remove taxa with an abundance of less than 5 across all samples however if I go higher than that it filters out a ton of taxa (about 1/3 of all of them)  
  
#Check the variance to see any patterns in the data  
Variance.plot <- qplot(log10(apply(otu\_table(ps3), 1, var)), xlab = "log10(variance)", main = "Variance in OTUs")

## Warning: `qplot()` was deprecated in ggplot2 3.4.0.

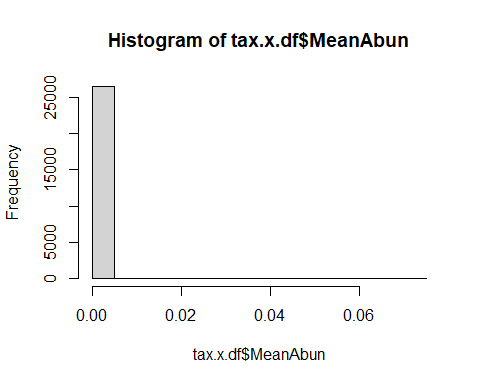
print(Variance.plot) #Variance is still right skewed after log transforming

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 66 rows containing non-finite values (`stat\_bin()`).

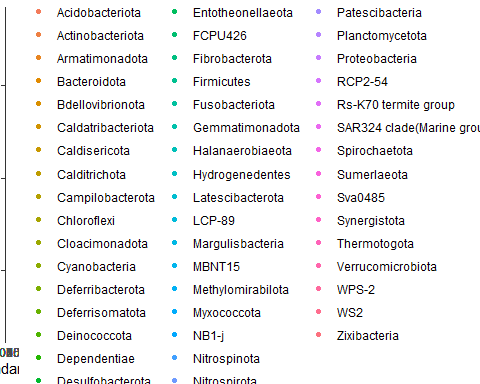


#Can also check the coefficient of variation for spurious observations. \*How do you tell which ones are spurious observations?\*  
ps1.rel <- microbiome::transform(ps3, "compositional") # transform to relative abundance  
p <- plot\_taxa\_cv(ps1.rel, plot.type="scatter")



print(p) #The low abundance ASVs have a very high coefficient of variation which may affect the differential abundance testing.

## Warning: Removed 66 rows containing missing values (`geom\_point()`).



#I will filter out taxa with an abundance of 5 or less.   
  
sample\_ps <- prune\_taxa(taxa\_sums(ps3) > 5, ps3)  
sample\_ps<- prune\_taxa(taxa\_sums(sample\_ps) > 0, sample\_ps)  
sample\_ps #21 013 taxa instead of 26584. Removed 5571 ASVs.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 21013 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 21013 taxa by 7 taxonomic ranks ]:  
## taxa are rows

any(taxa\_sums(sample\_ps)<=5) #FALSE. Removed them adequately.

## [1] FALSE

#How much data did we filter out from the original phyloseq object?  
  
ntaxa(psexp)-ntaxa(sample\_ps) #filtered 9208 total taxa (ASVs)

## [1] 9208

(1-ntaxa(sample\_ps)/ntaxa(psexp))\*100 #30.5% of the taxa were removed from the original dataset.

## [1] 30.46888

reads\_per\_OTU <- taxa\_sums(psexp) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU)) #Total number of reads is 13 158 716.

## [1] 13158716

reads\_per\_OTU\_filtered <- taxa\_sums(sample\_ps) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU\_filtered)) #Total number of reads is now 13 090 430.

## [1] 13090430

(1-13090430/13158716)\*100 #0.52%. Only filtered out a small percentage of the total data (reads).

## [1] 0.5189412

sample\_ps #21,013 taxa and 340 samples. Still waiting for 9 samples to come.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 21013 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 21013 taxa by 7 taxonomic ranks ]:  
## taxa are rows

sort(sample\_sums(sample\_ps)) #minimum is 2180, maximum is 222,954

## KK1924 KK1911 KK1944 KK1936 KK2079 KK1937 KK2136 KK2171 KK1920 KK1935 KK1948   
## 2180 2785 3648 4015 4302 5057 5223 5268 5666 5767 6243   
## KK1912 KK2170 KK2027 KK1950 KK1932 KK2153 KK2091 KK1880 KK2111 KK1923 KK1949   
## 6687 6807 7467 7725 8135 8164 8189 8210 8358 8902 9060   
## KK1978 KK2137 KK1916 KK1979 KK1951 KK2132 KK2006 KK1909 KK2017 KK1919 KK1913   
## 9151 9233 9248 9252 9500 10240 10427 10554 10691 10697 10768   
## KK2135 KK2032 KK2086 KK2010 KK2018 KK1930 KK2084 KK2092 KK1890 KK2155 KK1980   
## 11006 11061 11215 11584 11723 11915 12325 12941 13009 13067 13281   
## KK2168 KK1946 KK2098 KK2133 KK2107 KK1934 KK1818 KK1917 KK2146 KK2110 KK1931   
## 13308 13593 13895 13985 14052 14161 14172 14266 14320 14447 14495   
## KK2022 KK1988 KK1927 KK2106 KK1887 KK2158 KK1826 KK2126 KK1888 KK1901 KK2105   
## 15034 15259 15289 15602 16012 16019 16226 16239 16561 16849 16920   
## KK2008 KK2116 KK1828 KK2016 KK2121 KK1829 KK1827 KK1987 KK1907 KK1881 KK2169   
## 17214 17231 17416 17795 17887 17954 18004 18046 18094 18227 18299   
## KK2076 KK2104 KK1945 KK1811 KK1830 KK1825 KK2082 KK2156 KK2095 KK1824 KK1914   
## 18365 18442 18476 18610 18622 18812 18844 18980 19099 19516 19648   
## KK1982 KK1993 KK1874 KK1823 KK1933 KK1822 KK1820 KK1814 KK2009 KK2167 KK2103   
## 19841 19857 19896 20261 20307 20624 20997 21068 21193 21229 21300   
## KK2031 KK2102 KK1843 KK1991 KK2154 KK2108 KK2150 KK2019 KK2088 KK1872 KK1821   
## 21359 21505 21515 21743 22080 22216 22444 22687 22887 23106 23147   
## KK1813 KK2024 KK1895 KK1810 KK1986 KK2166 KK1886 KK1819 KK2113 KK1841 KK2077   
## 23182 23323 23594 23838 23847 23967 24086 24099 24133 24396 24466   
## KK2128 KK1816 KK2163 KK2165 KK1882 KK1809 KK2090 KK1817 KK1812 KK2003 KK1990   
## 24557 24687 24741 24911 24943 25036 25079 25146 25448 25691 26062   
## KK2112 KK2157 KK1815 KK1836 KK1877 KK2101 KK1915 KK1849 KK1938 KK2148 KK2130   
## 26067 26385 26757 27116 27284 27308 27348 27359 27374 27602 27688   
## KK2122 KK1854 KK2001 KK1891 KK1878 KK1995 KK1992 KK1985 KK1870 KK2085 KK1876   
## 27816 28063 28082 28184 28499 28643 28706 28820 28834 29143 29191   
## KK1844 KK1871 KK1989 KK1853 KK1892 KK1889 KK2028 KK1852 KK1983 KK2160 KK1866   
## 29755 29830 30005 30151 30165 30645 30787 30976 31161 31197 31231   
## KK2054 KK2145 KK1972 KK2099 KK2087 KK1867 KK1869 KK2109 KK1947 KK2139 KK1837   
## 31256 31308 31334 31616 31641 31887 31894 32029 32060 32404 32664   
## KK2007 KK2048 KK1838 KK2052 KK1899 KK1868 KK1834 KK2161 KK1875 KK1835 KK2000   
## 32702 32740 32785 32843 32883 32896 32924 33151 33201 33241 33255   
## KK1858 KK1842 KK1846 KK1859 KK1897 KK2138 KK1883 KK2093 KK2117 KK1833 KK1865   
## 33336 33391 33702 33910 34000 34359 34485 34711 34831 35221 35265   
## KK1873 KK1832 KK1973 KK2023 KK1840 KK2152 KK1851 KK1996 KK2025 KK1845 KK1862   
## 35367 35405 35570 35711 35835 35850 35994 36661 36662 37300 37311   
## KK2071 KK1994 KK2083 KK1857 KK2159 KK1981 KK2172 KK1905 KK2011 KK1847 KK2002   
## 37595 37850 37887 38196 38210 38371 38623 38773 39063 39068 39129   
## KK2005 KK2100 KK1860 KK1997 KK2042 KK1864 KK1884 KK2127 KK1971 KK1898 KK1896   
## 39143 39178 39221 39485 39582 39603 39799 39835 40246 40733 41346   
## KK1921 KK2114 KK1850 KK2004 KK2034 KK1856 KK1863 KK1885 KK1893 KK1906 KK2070   
## 42294 42805 42900 43371 43392 44983 45556 45656 46145 46325 46354   
## KK1969 KK2162 KK1855 KK2029 KK1848 KK2013 KK2030 KK1940 KK2033 KK1977 KK1928   
## 46556 46660 47539 47839 47956 48232 48496 48587 48640 49722 49911   
## KK2129 KK1879 KK1831 KK2080 KK2089 KK2069 KK1975 KK1970 KK1861 KK1904 KK2060   
## 50367 50490 50587 50667 50945 51606 52296 52321 52355 52431 53043   
## KK2036 KK1902 KK1984 KK2014 KK2065 KK1967 KK2134 KK2015 KK2149 KK1965 KK1900   
## 53224 53757 54082 54407 55568 55656 55713 55963 56129 56359 56666   
## KK2059 KK1966 KK1943 KK2164 KK1939 KK1968 KK1976 KK2035 KK1894 KK1908 KK1941   
## 57454 57753 58284 58889 59134 59273 60775 61523 62228 62442 62452   
## KK1961 KK2144 KK2147 KK2123 KK2141 KK2094 KK2020 KK1903 KK1962 KK2131 KK2021   
## 62602 62756 63238 63841 64060 64261 65325 65790 65954 68067 68547   
## KK2120 KK1942 KK2058 KK2041 KK2151 KK2125 KK2050 KK2062 KK2097 KK2142 KK1998   
## 68618 68941 70426 70535 71860 72080 72081 73495 73519 75073 77410   
## KK1963 KK2056 KK2047 KK2096 KK2039 KK2140 KK2143 KK1964 KK2055 KK2124 KK2051   
## 78135 78712 79545 80221 81701 83329 83809 83841 84110 85127 85447   
## KK2040 KK2064 KK2044 KK2066 KK2049 KK1974 KK2057 KK2119 KK2068 KK2063 KK2043   
## 86063 86979 88318 89026 89040 91161 92572 97706 98620 101211 102205   
## KK2046 KK2053 KK2118 KK2038 KK1839 KK2061 KK1999 KK2037 KK2045 KK2067   
## 108927 110329 112148 114359 114525 122037 126300 157625 188887 222954

mean(sample\_sums(sample\_ps)) #mean is 38,501.

## [1] 38501.26

#Still need to rarefy because there is a wide variation in the number of reads between samples.

#Subsetting experimental samples only from the original phyloseq object  
psbl<-subset\_samples(ps, Sample\_Type != "Sample") #Removing ASVs where the abundance is 0 (after removing the blank samples)  
psbl<-prune\_taxa(taxa\_sums(psbl) > 0, psbl)  
psbl #1294 taxa and 13 samples

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 1294 taxa and 13 samples ]:  
## sample\_data() Sample Data: [ 13 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 1294 taxa by 7 taxonomic ranks ]:  
## taxa are rows

#Removing low confidence data (where phylum could not be assigned)  
table(tax\_table(psbl)[,"Phylum"],exclude=NULL) #No taxa assigned as a phyla of NA

##   
## Abditibacteriota Acidobacteriota Actinobacteriota Armatimonadota   
## 2 32 194 1   
## Bacteroidota Bdellovibrionota Chloroflexi Cyanobacteria   
## 195 4 22 87   
## Deferribacterota Deinococcota Desulfobacterota Entotheonellaeota   
## 2 1 6 1   
## Firmicutes Gemmatimonadota Myxococcota Nitrospirota   
## 358 12 14 1   
## Patescibacteria Planctomycetota Proteobacteria RCP2-54   
## 2 26 324 2   
## Spirochaetota Synergistota Thermotogota Verrucomicrobiota   
## 1 3 1 2   
## WPS-2   
## 1

#Seems like some of these phyla with a prevalence of 1 also only showed up once in the experimental samples. These ones may need to be filtered out of the experimental samples if it originated just from the sheet/mesh/water blanks.  
  
#Getting rid of singletons and doubletons  
any(taxa\_sums(psbl)>=2) #TRUE. Means there are ASVs with a sum of sequences less than 2 across all samples.

## [1] TRUE

psbl0 <- prune\_taxa(taxa\_sums(psbl) > 2, psbl) #Getting rid of singletons and doubletons. (Taxa that only showed up in one or two sequences across all samples)  
any(taxa\_sums(psbl0) <2) #FALSE. Means they were removed.

## [1] FALSE

ntaxa(psbl)- ntaxa(psbl0) #Got rid of 32 ASVs.

## [1] 32

#How many bacterial taxa do we have/did we remove  
ntaxa(psbl) #1294 before any filtering was done.

## [1] 1294

ntaxa(psbl0) #1262 after sum of reads less than 2 were removed (32 removed).

## [1] 1262

# Compute prevalence of each feature, store as data.frame  
prevdf=apply(X=otu\_table(psbl0),MARGIN=ifelse(taxa\_are\_rows(psbl0),yes=1,no=2),FUN = function(x){sum(x > 0)})  
# Add taxonomy and total read counts to this data.frame  
prevdf= data.frame(Prevalence= prevdf,TotalAbundance=taxa\_sums(psbl0),tax\_table(psbl0))  
  
plyr::ddply(prevdf,"Phylum",function(df1){cbind(mean(df1$Prevalence),sum(df1$Prevalence))})

## Phylum 1 2  
## 1 Abditibacteriota 1.000000 2  
## 2 Acidobacteriota 1.322581 41  
## 3 Actinobacteriota 1.177083 226  
## 4 Bacteroidota 1.423280 269  
## 5 Bdellovibrionota 1.000000 4  
## 6 Chloroflexi 1.000000 21  
## 7 Cyanobacteria 1.388235 118  
## 8 Deferribacterota 1.000000 2  
## 9 Deinococcota 1.000000 1  
## 10 Desulfobacterota 1.000000 5  
## 11 Entotheonellaeota 1.000000 1  
## 12 Firmicutes 1.881159 649  
## 13 Gemmatimonadota 1.000000 12  
## 14 Myxococcota 1.142857 16  
## 15 Nitrospirota 1.000000 1  
## 16 Patescibacteria 1.000000 2  
## 17 Planctomycetota 1.038462 27  
## 18 Proteobacteria 1.476489 471  
## 19 RCP2-54 1.000000 2  
## 20 Spirochaetota 1.000000 1  
## 21 Synergistota 1.000000 3  
## 22 Thermotogota 1.000000 1  
## 23 Verrucomicrobiota 1.000000 2  
## 24 WPS-2 1.000000 1

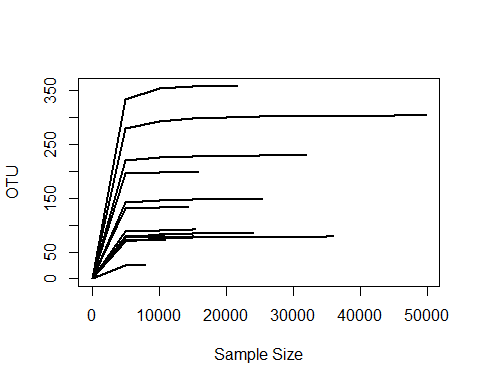
#The low prevalence phyla showed up in the experimental samples as well. Could be due to sequencing errors, contamination during extraction, or real observations in the blanks. Not sure.   
  
plyr::ddply(prevdf, "Phylum", function(df1){  
 data.frame(mean\_prevalence=mean(df1$Prevalence),total\_abundance=sum(df1$TotalAbundance,na.rm = T),stringsAsFactors = F)  
 })

## Phylum mean\_prevalence total\_abundance  
## 1 Abditibacteriota 1.000000 21  
## 2 Acidobacteriota 1.322581 2659  
## 3 Actinobacteriota 1.177083 5189  
## 4 Bacteroidota 1.423280 110637  
## 5 Bdellovibrionota 1.000000 140  
## 6 Chloroflexi 1.000000 420  
## 7 Cyanobacteria 1.388235 22369  
## 8 Deferribacterota 1.000000 36  
## 9 Deinococcota 1.000000 8  
## 10 Desulfobacterota 1.000000 57  
## 11 Entotheonellaeota 1.000000 8  
## 12 Firmicutes 1.881159 81285  
## 13 Gemmatimonadota 1.000000 195  
## 14 Myxococcota 1.142857 390  
## 15 Nitrospirota 1.000000 10  
## 16 Patescibacteria 1.000000 85  
## 17 Planctomycetota 1.038462 384  
## 18 Proteobacteria 1.476489 55078  
## 19 RCP2-54 1.000000 27  
## 20 Spirochaetota 1.000000 10  
## 21 Synergistota 1.000000 108  
## 22 Thermotogota 1.000000 12  
## 23 Verrucomicrobiota 1.000000 27  
## 24 WPS-2 1.000000 90

#Creating a rarefaction curve  
tab <- otu\_table(psbl0)  
class(tab) <- "matrix" # as.matrix() will do nothing

## Warning in class(tab) <- "matrix": Setting class(x) to "matrix" sets attribute  
## to NULL; result will no longer be an S4 object

## you get a warning here, but this is what we need to have  
tab <- t(tab) # transpose observations to rows  
library(vegan)  
rare <- rarecurve(tab, step=5000, lwd=2, ylab="OTU", label=F)



#The rarefaction curves are not crazy different so we can rarefy to the minimum sequencing depth of ~7000  
  
sort(sample\_sums(psbl0)) #minimum is 7949, maximum is 49,799

## KK1954 KK1956 KK1958 KK1960 KK1959 KK1955 KK2073 KK2074 KK1952 KK1957 KK2072   
## 7949 10832 10910 14418 15123 15372 15915 21605 24004 25366 31996   
## KK1953 KK2075   
## 35956 49799

mean(sample\_sums(psbl0)) #mean is 21,480

## [1] 21480.38

#How much data did we filter out from the original phyloseq object?  
  
ntaxa(psbl)-ntaxa(psbl0) #filtered 32 total taxa (ASVs)

## [1] 32

(1-ntaxa(psbl0)/ntaxa(psbl))\*100 #2.47% of the taxa were removed from the original dataset.

## [1] 2.472952

reads\_per\_OTU <- taxa\_sums(psbl) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU)) #Total number of reads is 279,297.

## [1] 279297

reads\_per\_OTU\_filtered <- taxa\_sums(psbl0) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU\_filtered)) #Total number of reads is now 279,245.

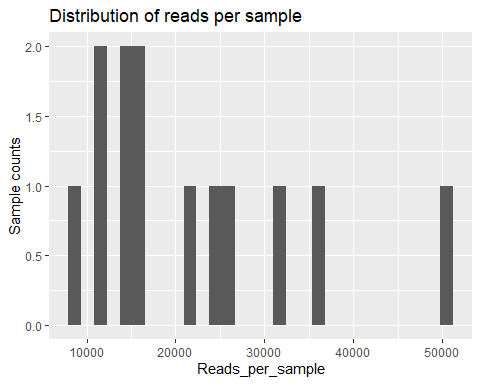
## [1] 279245

(1-279245/279297)\*100 #0.019%. Only filtered out a very small percentage of the total data (reads).

## [1] 0.01861817

#Creating a distribution of reads per sample versus the sample counts.   
ps1\_df= data.table(as(sample\_data(psbl0), "data.frame"), Reads\_per\_sample = sample\_sums(psbl0), keep.rownames = TRUE)  
ps1\_df\_plot = ggplot(ps1\_df, aes(Reads\_per\_sample)) + geom\_histogram() + ggtitle("Distribution of reads per sample") + ylab("Sample counts")   
  
print(ps1\_df\_plot) #All over the place. Not really any sequencing depth that is consistent across all samples

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



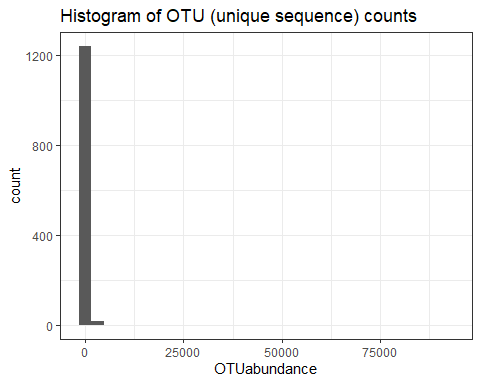
#Now we need to observe the counts of abundance of unique sequences across samples  
ps1.dt.taxa = data.table(tax\_table(psbl0), OTUabundance = taxa\_sums(psbl0), OTU = taxa\_names(psbl0))

## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

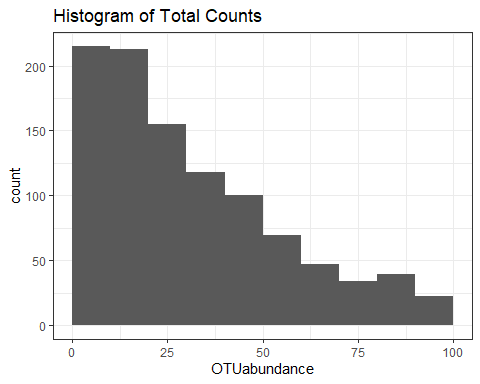
## Warning in setDT(ans, key = key): Some columns are a multi-column type (such as  
## a matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

#makes a data table of the sum of abundance of each taxa vs the total number of taxa (ASVs)  
ps1.dt.tax.plot<- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram() + ggtitle("Histogram of OTU (unique sequence) counts") + theme\_bw() #Plots the number of times an ASV is repeated from the reads (abundance) on the x axis and the number of taxa on the y axis  
print(ps1.dt.tax.plot)

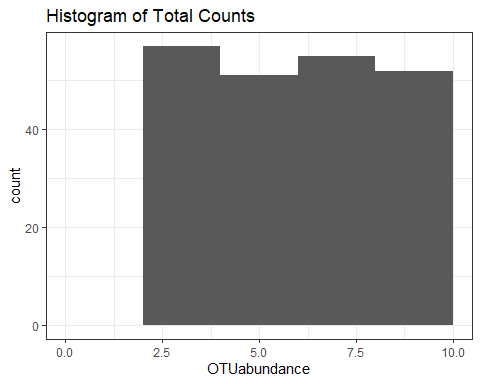
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#You can"t see much on the plots panel in R but not a very high abundance in most samples. One sample has a really high abundance.  
  
#This plot shows that there is a very small proportion of ASVs with a really high abundance and most have relatively low abundance. Highly right skewed. We want to see how many have very low abundance because this may indicate that further processing/filtering should be done.   
  
#Zoom in on the lower abundance counts to get a better idea of its distribution  
plot.zoom1 <- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram(breaks=seq(0, 100, by =10)) + ggtitle("Histogram of Total Counts") + theme\_bw()  
print(plot.zoom1)



#A small proportion of ASVs have an abundance of less than 10. These may be able to be filtered out.   
  
plot.zoom2 <- ggplot(ps1.dt.taxa, aes(OTUabundance)) + geom\_histogram(breaks=seq(0, 10, by =2)) + ggtitle("Histogram of Total Counts") + theme\_bw()  
print(plot.zoom2)



#Only about 200 taxa have a prevalence below 10.   
  
#Finding total number of reads vs # of reads with low abundance counts  
reads\_per\_OTU <- taxa\_sums(psbl0)  
print(sum(reads\_per\_OTU)) #Total number of reads is 279,245.

## [1] 279245

#How many taxa (ASVs) contain less than 10 reads (an abundance of 10 or less)?  
print(length(reads\_per\_OTU[reads\_per\_OTU <=10])) #215.

## [1] 215

#How many reads do these 215 ASVs contain in total?  
print(sum(reads\_per\_OTU[reads\_per\_OTU <=10])) #1385

## [1] 1385

#What percentage of the total reads is that?  
print((sum(reads\_per\_OTU[reads\_per\_OTU <=10])/sum(reads\_per\_OTU))\*100) #0.50 %. A very small proportion of the entire dataset.

## [1] 0.4959802

#How many ASVs out of the total number of ASVs contains less than or equal to 10 reads?  
ntaxa(psbl0) #1262

## [1] 1262

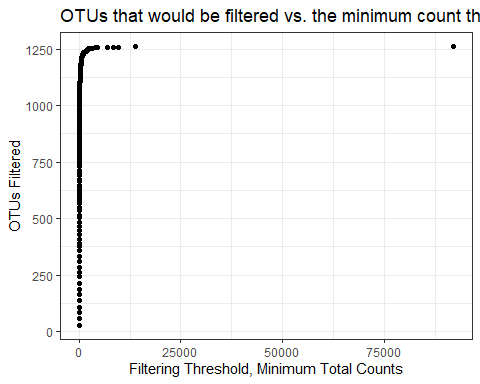
print((215/1262)\*100) #17.0%

## [1] 17.03645

#Plotting the number of ASVs that would be filtered out given a minimum read threshold (min abundance of ASVs)  
ps1.dt.taxa = data.table(tax\_table(psbl0), OTUabundance = taxa\_sums(psbl0), OTU = taxa\_names(psbl0))

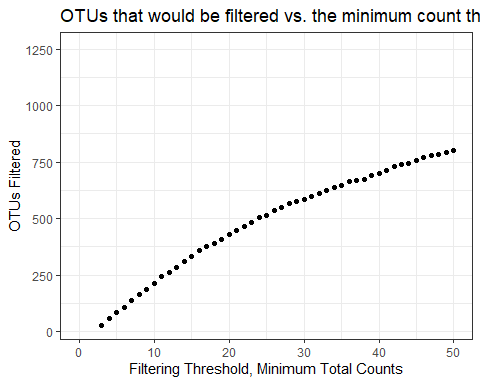
## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.  
  
## Warning in setDT(value): Some columns are a multi-column type (such as a  
## matrix column): [1, 2, 3, 4, 5, 6, 7]. setDT will retain these columns as-is  
## but subsequent operations like grouping and joining may fail. Please consider  
## as.data.table() instead which will create a new column for each embedded column.

ps1.cumsum = ps1.dt.taxa[, .N, by = OTUabundance]  
setkey(ps1.cumsum, OTUabundance)  
ps1.cumsum[, CumSum := cumsum(N)]  
# Define the plot  
ps1.cumsum.plot = ggplot(ps1.cumsum, aes(OTUabundance, CumSum)) +   
 geom\_point() +  
 xlab("Filtering Threshold, Minimum Total Counts") +  
 ylab("OTUs Filtered") +  
 ggtitle("OTUs that would be filtered vs. the minimum count threshold") + theme\_bw()  
  
print(ps1.cumsum.plot) #Difficult to see the smaller values so we need to zoom in



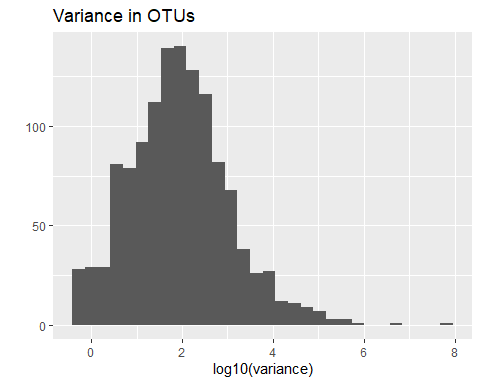
ps1.cumsum.plot.zoom <- ps1.cumsum.plot + xlim(0, 50) + theme\_bw()   
  
print(ps1.cumsum.plot.zoom)

## Warning: Removed 243 rows containing missing values (`geom\_point()`).

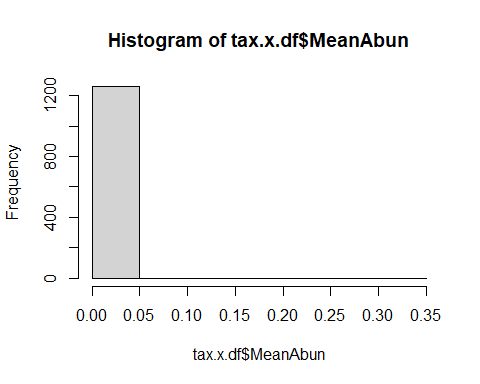


#Could maybe remove taxa with an abundance of less than 10 across all samples  
  
#Check the variance to see any patterns in the data  
Variance.plot <- qplot(log10(apply(otu\_table(psbl0), 1, var)), xlab = "log10(variance)", main = "Variance in OTUs")  
print(Variance.plot) #Variance is still right skewed after log transforming

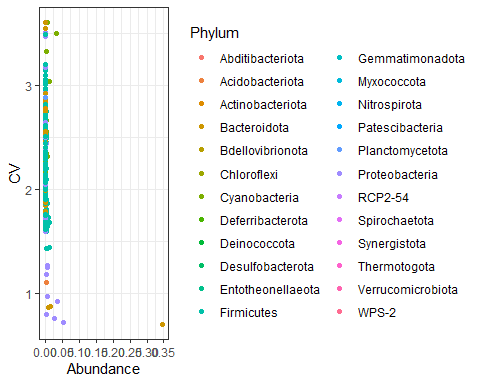
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#Can also check the coefficient of variation for spurious observations. \*How do you tell which ones are spurious observations?\*  
ps1.rel <- microbiome::transform(psbl0, "compositional") # transform to relative abundance  
p <- plot\_taxa\_cv(ps1.rel, plot.type="scatter")



print(p) #The low abundance ASVs have a very high coefficient of variation which may affect the differential abundance testing.



#I will filter out taxa with an abundance of 10 or less.   
  
  
blank\_ps <- prune\_taxa(taxa\_sums(psbl0) > 10, psbl0)  
blank\_ps<- prune\_taxa(taxa\_sums(blank\_ps) > 0, blank\_ps)  
blank\_ps #1047 taxa instead of 1294. Removed 247 ASVs in total.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 1047 taxa and 13 samples ]:  
## sample\_data() Sample Data: [ 13 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 1047 taxa by 7 taxonomic ranks ]:  
## taxa are rows

any(taxa\_sums(blank\_ps)<=10) #FALSE. Removed them adequately.

## [1] FALSE

sort(sample\_sums(blank\_ps)) #Lowest was 7949, highest was 49655. Seems pretty high for a blank sample but not too variable.

## KK1954 KK1956 KK1958 KK1960 KK1959 KK1955 KK2073 KK2074 KK1952 KK1957 KK2072   
## 7946 10776 10878 14348 15108 15316 15773 21051 23890 25284 31936   
## KK1953 KK2075   
## 35899 49655

mean(sample\_sums(blank\_ps)) #mean is 21,373

## [1] 21373.85

#How much data did we filter out from the original phyloseq object?  
  
ntaxa(psbl)-ntaxa(blank\_ps) #filtered 247 total taxa (ASVs)

## [1] 247

(1-ntaxa(blank\_ps)/ntaxa(psbl))\*100 #19.1% of the taxa were removed from the original dataset.

## [1] 19.0881

reads\_per\_OTU <- taxa\_sums(psbl) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU)) #Total number of reads is 279297.

## [1] 279297

reads\_per\_OTU\_filtered <- taxa\_sums(blank\_ps) #Original ps with no filtering done.  
print(sum(reads\_per\_OTU\_filtered)) #Total number of reads is now 277860.

## [1] 277860

(1-277860/279297)\*100 #0.51%. Only filtered out a small percentage of the total data (reads).

## [1] 0.5145061

##Prevalence threshold pruning ##  
# Define prevalence of each taxa (in how many samples did each taxa appear at least once)  
prev0 = apply(X = otu\_table(sample\_ps),  
 MARGIN = ifelse(taxa\_are\_rows(sample\_ps), yes = 1, no = 2),  
 FUN = function(x){sum(x > 0)})  
prevdf = data.frame(Prevalence = prev0,  
 TotalAbundance = taxa\_sums(sample\_ps),  
 tax\_table(sample\_ps))  
keepPhyla = table(prevdf$Phylum)[(table(prevdf$Phylum) > 2)] #Keeping phyla with a prevalence greater than 2  
prevdf1 = subset(prevdf, Phylum %in% names(keepPhyla))  
# Define prevalence threshold as 5% of total samples  
prevalenceThreshold = 0.02 \* nsamples(sample\_ps)  
prevalenceThreshold #We are keeping phyla that occur at least 6.88 samples (2% of samples) with a prevalence of 2 or greater

## [1] 6.8

psprune = prune\_taxa((prev0 > prevalenceThreshold), sample\_ps)  
psprune #2849 taxa instead of 27,069. Cut out a very large portion of the data. Too high of a threshold.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 2810 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 2810 taxa by 7 taxonomic ranks ]:  
## taxa are rows

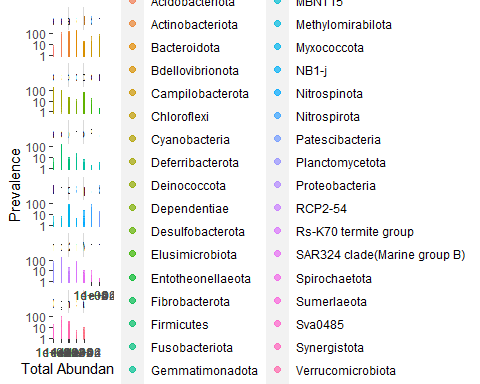
reads\_pruned <- taxa\_sums(psprune)  
print(sum(reads\_pruned)) #11,306,751 total reads left after pruning

## [1] 11191488

(1-(sum(reads\_pruned))/(sum(reads\_per\_OTU)))\*100 #14.55% of the total read data was removed by pruning. A decently large portion of the data was cut out from additional pruning

## [1] -3907.02

ggplot(prevdf1, aes(TotalAbundance, Prevalence, color = Phylum)) +  
 geom\_hline(yintercept = prevalenceThreshold, alpha = 0.5, linetype = 2) +  
 geom\_point(size = 2, alpha = 0.7) +  
 scale\_y\_log10() + scale\_x\_log10() +  
 xlab("Total Abundance") +  
 facet\_wrap(~Phylum)



#Prevalence threshold looks too high, cutting out a lot of the variation in microbial communities. I will try a lower threshold value.   
  
  
#Lowering the prevalence threshold to 0.2%  
  
# Define prevalence of each taxa  
# (in how many samples did each taxa appear at least once)  
prev1 = apply(X = otu\_table(sample\_ps),  
 MARGIN = ifelse(taxa\_are\_rows(sample\_ps), yes = 1, no = 2),  
 FUN = function(x){sum(x > 0)})  
prevdf2 = data.frame(Prevalence = prev1,  
 TotalAbundance = taxa\_sums(sample\_ps),  
 tax\_table(sample\_ps))  
keepPhyla1 = table(prevdf2$Phylum)[(table(prevdf2$Phylum) > 2)]  
prevdf3 = subset(prevdf2, Phylum %in% names(keepPhyla))  
# Define prevalence threshold as 5% of total samples  
prevalenceThreshold1 = 0.01 \* nsamples(sample\_ps)  
prevalenceThreshold1 #3.44.

## [1] 3.4

psprune1 = prune\_taxa((prev1 > prevalenceThreshold1), sample\_ps)  
psprune1 #4614 taxa instead of 27,069. Got rid of 22455 ASVs. Still a very large percentage of taxa removed.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 4556 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 4556 taxa by 7 taxonomic ranks ]:  
## taxa are rows

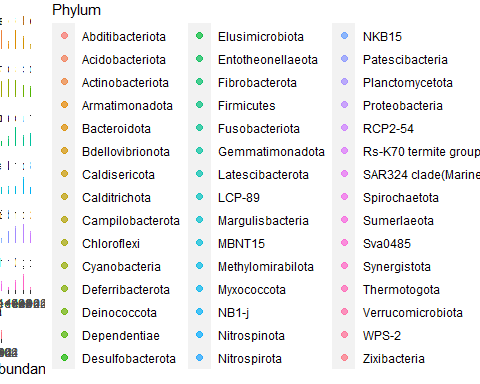
reads\_pruned1 <- taxa\_sums(psprune1)  
print(sum(reads\_pruned1)) #13 231 428 total reads left after pruning

## [1] 11956159

(1-(sum(reads\_pruned1))/(sum(reads\_per\_OTU)))\*100 #0% of data was removed. Why did it remove 297 ASVs but none of the reads??

## [1] -4180.805

ggplot(prevdf2, aes(TotalAbundance, Prevalence, color = Phylum)) +  
 geom\_hline(yintercept = prevalenceThreshold1, alpha = 0.5, linetype = 2) +  
 geom\_point(size = 2, alpha = 0.7) +  
 scale\_y\_log10() + scale\_x\_log10() +  
 xlab("Total Abundance") +  
 facet\_wrap(~Phylum)



#This threshold of filtering looks a lot better and I can be more confidence that we are not removing true taxa from samples. Looks like the ones that are removed are just artifacts of sampling or sequencing errors.

#Experimental Samples  
psexp<-subset\_samples(ps, Sample\_Type == "Sample") #Removing ASVs where the abundance is 0 (after removing the blank samples)  
psexp<-prune\_taxa(taxa\_sums(psexp) > 0, psexp)  
psexp #30,221 taxa and 350 samples

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 30221 taxa and 350 samples ]:  
## sample\_data() Sample Data: [ 350 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 30221 taxa by 7 taxonomic ranks ]:  
## taxa are rows

#Removing Eukaryotes  
ps0 <- subset\_taxa(psexp, Kingdom != "Eukaryota")  
ntaxa(psexp)-ntaxa(ps0) #Got rid of 39 ASVs

## [1] 39

#Removing low confidence data (where phylum could not be assigned)  
ps1<-subset\_taxa(ps0, !is.na(Phylum) & !Phylum %in% c("","uncharacterized")) #Removes the phyla characterized as NA  
ntaxa(ps0) - ntaxa(ps1) #Got rid of 393 ASVs

## [1] 393

#Getting rid of singletons and doubletons  
any(taxa\_sums(ps1)>=2) #TRUE. Means there are ASVs with a sum of sequences less than 2 across all samples.

## [1] TRUE

ps2 <- prune\_taxa(taxa\_sums(ps1) > 2, ps1) #Getting rid of singletons and doubletons. (Taxa that only showed up in one or two sequences across all samples)  
any(taxa\_sums(ps2) <2) #FALSE. Means they were removed.

## [1] FALSE

ntaxa(ps1)- ntaxa(ps2) #Got rid of 3205 ASVs.

## [1] 3205

ps3 <- subset\_samples(ps2, sample\_sums(ps2) > 2000)   
  
sample\_ps <- prune\_taxa(taxa\_sums(ps3) > 5, ps3)  
sample\_ps<- prune\_taxa(taxa\_sums(sample\_ps) > 0, sample\_ps)  
sample\_ps #21 013 taxa instead of 26584. Removed 5571 ASVs.

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 21013 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 21013 taxa by 7 taxonomic ranks ]:  
## taxa are rows

#Blank samples  
  
psbl<-subset\_samples(ps, Sample\_Type != "Sample") #Removing ASVs where the abundance is 0 (after removing the blank samples)  
psbl<-prune\_taxa(taxa\_sums(psbl) > 0, psbl)  
psbl

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 1294 taxa and 13 samples ]:  
## sample\_data() Sample Data: [ 13 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 1294 taxa by 7 taxonomic ranks ]:  
## taxa are rows

any(taxa\_sums(psbl)>=2) #TRUE. Means there are ASVs with a sum of sequences less than 2 across all samples.

## [1] TRUE

psbl0 <- prune\_taxa(taxa\_sums(psbl) > 2, psbl) #Getting rid of singletons and doubletons. (Taxa that only showed up in one or two sequences across all samples)  
any(taxa\_sums(psbl0) <2) #FALSE. Means they were removed.

## [1] FALSE

ntaxa(psbl)- ntaxa(psbl0) #Got rid of 32 ASVs.

## [1] 32

#How many bacterial taxa do we have/did we remove  
ntaxa(psbl) #1294 before any filtering was done.

## [1] 1294

ntaxa(psbl0) #1262 after sum of reads less than 2 were removed (32 removed).

## [1] 1262

blank\_ps <- prune\_taxa(taxa\_sums(psbl0) > 10, psbl0)  
blank\_ps<- prune\_taxa(taxa\_sums(blank\_ps) > 0, blank\_ps)  
blank\_ps #1047 taxa instead of 1294. Removed 247 ASVs in total. 13 samples

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 1047 taxa and 13 samples ]:  
## sample\_data() Sample Data: [ 13 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 1047 taxa by 7 taxonomic ranks ]:  
## taxa are rows

##Data diagnostics

In this section I filter out different taxonomic organizations of bacterial taxa to see the abundance at different levels of organization. Ex. how many phyla per site, what are the top phyla, families, genera and their percentage of the total.

#Experimental Samples  
ntaxa(sample\_ps) #21 239

## [1] 21013

sort(table(tax\_table(sample\_ps)[,2]))

##   
## Caldisericota Calditrichota   
## 1 1   
## NKB15 Thermotogota   
## 1 1   
## Armatimonadota Entotheonellaeota   
## 2 3   
## LCP-89 Margulisbacteria   
## 4 4   
## Sumerlaeota Zixibacteria   
## 4 4   
## NB1-j Nitrospinota   
## 5 5   
## MBNT15 Methylomirabilota   
## 8 8   
## RCP2-54 WPS-2   
## 8 9   
## Spirochaetota SAR324 clade(Marine group B)   
## 10 13   
## Sva0485 Synergistota   
## 13 14   
## Fibrobacterota Rs-K70 termite group   
## 16 16   
## Latescibacterota Deferribacterota   
## 20 21   
## Campilobacterota Fusobacteriota   
## 22 22   
## Elusimicrobiota Deinococcota   
## 30 36   
## Dependentiae Abditibacteriota   
## 40 42   
## Nitrospirota Patescibacteria   
## 46 118   
## Gemmatimonadota Bdellovibrionota   
## 126 158   
## Verrucomicrobiota Myxococcota   
## 209 312   
## Chloroflexi Desulfobacterota   
## 334 407   
## Planctomycetota Acidobacteriota   
## 512 622   
## Cyanobacteria Actinobacteriota   
## 1098 1716   
## Firmicutes Bacteroidota   
## 2133 5387   
## Proteobacteria   
## 7452

#Top 3 abundant phyla are Proteobacteria (35.1%), Bacteroidota (25.4%), Firmicutes (10.4%)  
  
#Sort by proteobacteria  
proteo <- subset\_taxa(sample\_ps, Phylum=="Proteobacteria")  
ntaxa(proteo) #7452

## [1] 7452

sort(table(tax\_table(proteo)[,3])) #class. Only 2 classes. Alphaproteobacteria (47.5%) and Gammaproteobacteria (53.5%)

##   
## Alphaproteobacteria Gammaproteobacteria   
## 3538 3842

sort(table(tax\_table(proteo)[,5])) #family Top 3: Comamonadaceae (18.3%), Sphingomonadaceae (7.6%), Rhodobacteraceae (6.4%)

##   
## Acidithiobacillaceae Amb-16S-1323   
## 1 1   
## Aquaspirillaceae bac2nit3   
## 1 1   
## Clade III Inquilinaceae   
## 1 1   
## Leeiaceae Methylococcaceae   
## 1 1   
## Parvularculaceae Shewanellaceae   
## 1 1   
## Spongiibacteraceae Stappiaceae   
## 1 1   
## Taonella Vibrionaceae   
## 1 1   
## Zavarziniaceae Azospirillales Incertae Sedis   
## 1 2   
## Budviciaceae Chitinimonadaceae   
## 2 2   
## Pseudohongiellaceae Salinisphaeraceae   
## 2 2   
## Thiotrichaceae Alteromonadaceae   
## 2 3   
## Labraceae Methylopilaceae   
## 3 3   
## Pectobacteriaceae Sulfuricellaceae   
## 3 3   
## Kaistiaceae Pasteurellaceae   
## 4 4   
## Rhodomicrobiaceae A21b   
## 4 5   
## Magnetospirillaceae Micropepsaceae   
## 5 5   
## Piscirickettsiaceae Woeseiaceae   
## 5 5   
## Acidiferrobacteraceae Candidatus Jidaibacter   
## 6 6   
## Elsteraceae Rhodospirillaceae   
## 6 6   
## Caedibacteraceae KF-JG30-B3   
## 7 7   
## Wohlfahrtiimonadaceae AB1   
## 8 9   
## Chromobacteriaceae Dongiaceae   
## 9 9   
## Micavibrionaceae Pleomorphomonadaceae   
## 9 9   
## C2U Candidatus Hepatincola   
## 10 10   
## Chromatiaceae Coxiellaceae   
## 10 10   
## Halomonadaceae SM2D12   
## 10 10   
## Gallionellaceae Morganellaceae   
## 11 11   
## Aeromonadaceae D05-2   
## 12 14   
## Methylomonadaceae Enterobacteriaceae   
## 14 15   
## B1-7BS Halieaceae   
## 16 16   
## Competibacteraceae Methyloligellaceae   
## 17 18   
## Erwiniaceae Hydrogenophilaceae   
## 19 19   
## Arenicellaceae Azospirillaceae   
## 20 21   
## Geminicoccaceae Hyphomonadaceae   
## 21 21   
## Chitinibacteraceae T34   
## 22 22   
## Defluviicoccaceae A0839   
## 24 25   
## Paracaedibacteraceae Reyranellaceae   
## 26 30   
## Neisseriaceae Holosporaceae   
## 32 35   
## Moraxellaceae Yersiniaceae   
## 36 36   
## Anaplasmataceae Alcaligenaceae   
## 39 41   
## Solimonadaceae Pseudomonadaceae   
## 42 52   
## SC-I-84 TRA3-20   
## 56 59   
## Unknown Family Burkholderiaceae   
## 60 61   
## Diplorickettsiaceae Steroidobacteraceae   
## 68 68   
## Sutterellaceae Legionellaceae   
## 72 78   
## Rhodanobacteraceae Devosiaceae   
## 79 86   
## Methylophilaceae Rickettsiaceae   
## 106 110   
## Xanthobacteraceae Caulobacteraceae   
## 120 124   
## Rhodocyclaceae Beijerinckiaceae   
## 135 173   
## Nitrosomonadaceae Acetobacteraceae   
## 177 187   
## Xanthomonadaceae Rhizobiaceae   
## 189 199   
## Hyphomicrobiaceae Rhizobiales Incertae Sedis   
## 202 212   
## Oxalobacteraceae Mitochondria   
## 460 463   
## Rhodobacteraceae Sphingomonadaceae   
## 480 560   
## Comamonadaceae   
## 1366

sort(table(tax\_table(proteo)[,6])) #genera Top 3: Rhodoferax (3.7%), Massilia (3.3%), Hyphomicrobium (1.7%)

##   
## [Agitococcus] lubricus group   
## 1   
## Actimicrobium   
## 1   
## Agaricicola   
## 1   
## Albibacter   
## 1   
## Amphiplicatus   
## 1   
## Arsenophonus   
## 1   
## Aurantimonas   
## 1   
## Azohydromonas   
## 1   
## BD1-7 clade   
## 1   
## Beijerinckia   
## 1   
## Budvicia   
## 1   
## Candidatus Contendobacter   
## 1   
## Candidatus Endonucleariobacter   
## 1   
## Candidatus Finniella   
## 1   
## Candidatus Hamiltonella   
## 1   
## Candidatus Hemipteriphilus   
## 1   
## Candidatus Regiella   
## 1   
## Cavicella   
## 1   
## Chitinimonas   
## 1   
## Chitinivorax   
## 1   
## Commensalibacter   
## 1   
## Crenobacter   
## 1   
## Dechlorosoma   
## 1   
## Denitratisoma   
## 1   
## Diplorickettsia   
## 1   
## DSSF69   
## 1   
## Endobacter   
## 1   
## Enhydrobacter   
## 1   
## Escherichia-Shigella   
## 1   
## Formivibrio   
## 1   
## Gallionella   
## 1   
## Glaciimonas   
## 1   
## Haematospirillum   
## 1   
## Halioglobus   
## 1   
## Halotalea   
## 1   
## Hansschlegelia   
## 1   
## Herbaspirillum   
## 1   
## Hydrocarboniphaga   
## 1   
## Hyphomonas   
## 1   
## Inquilinus   
## 1   
## KCM-B-112   
## 1   
## Koukoulia   
## 1   
## Kozakia   
## 1   
## Lacibacterium   
## 1   
## Leeia   
## 1   
## Luteibacter   
## 1   
## Methylovulum   
## 1   
## MM2   
## 1   
## MN 122.2a   
## 1   
## Moraxella   
## 1   
## Morganella   
## 1   
## Neorhizobium   
## 1   
## Neorickettsia   
## 1   
## Nitrobacter   
## 1   
## Niveibacterium   
## 1   
## Niveispirillum   
## 1   
## Oligella   
## 1   
## Palleronia-Pseudomaribius   
## 1   
## Paludibacterium   
## 1   
## Parasutterella   
## 1   
## Perlucidibaca   
## 1   
## Pleomorphomonas   
## 1   
## Pragia   
## 1   
## Prosthecomicrobium   
## 1   
## Proteus   
## 1   
## Providencia   
## 1   
## Pseudoxanthobacter   
## 1   
## Psychrobacter   
## 1   
## Puniceibacterium   
## 1   
## Rhodobaculum   
## 1   
## Rivicola   
## 1   
## Roseovarius   
## 1   
## Rubrimonas   
## 1   
## s3t2d-1089   
## 1   
## Salinarimonas   
## 1   
## Salmonella   
## 1   
## Seohaeicola   
## 1   
## Shewanella   
## 1   
## Sphingosinicella   
## 1   
## Starkeya   
## 1   
## Sulfuricella   
## 1   
## Sulfurisoma   
## 1   
## Tepidimonas   
## 1   
## Tianweitania   
## 1   
## Tibeticola   
## 1   
## Tolumonas   
## 1   
## Variibacter   
## 1   
## Vibrio   
## 1   
## Vitreoscilla   
## 1   
## Wohlfahrtiimonas   
## 1   
## Xanthomonas   
## 1   
## Yokenella   
## 1   
## Alkanibacter   
## 2   
## Ancylobacter   
## 2   
## Arsenicitalea   
## 2   
## Azoarcus   
## 2   
## Bartonella   
## 2   
## BIyi10   
## 2   
## Bordetella   
## 2   
## C39   
## 2   
## Chelativorans   
## 2   
## Collimonas   
## 2   
## Craurococcus-Caldovatus   
## 2   
## Croceicoccus   
## 2   
## Cupriavidus   
## 2   
## Dyella   
## 2   
## Ellin6055   
## 2   
## Elstera   
## 2   
## Enterobacter   
## 2   
## Ferritrophicum   
## 2   
## FFCH5858   
## 2   
## Giesbergeria   
## 2   
## Gulbenkiania   
## 2   
## Ignatzschineria   
## 2   
## Malikia   
## 2   
## Methylobacillus   
## 2   
## Methyloglobulus   
## 2   
## Methylovirgula   
## 2   
## Neo-b11   
## 2   
## Oceanicella   
## 2   
## Oxalicibacterium   
## 2   
## Pelagibacterium   
## 2   
## Pigmentiphaga   
## 2   
## Piscinibacter   
## 2   
## Polycyclovorans   
## 2   
## Pseudochrobactrum   
## 2   
## Raoultella   
## 2   
## Roseobacter clade CHAB-I-5 lineage   
## 2   
## Salinisphaera   
## 2   
## Simplicispira   
## 2   
## Solimonas   
## 2   
## Stella   
## 2   
## SWB02   
## 2   
## Tahibacter   
## 2   
## Telmatospirillum   
## 2   
## Thauera   
## 2   
## Thiothrix   
## 2   
## Tropicimonas   
## 2   
## Wenxinia   
## 2   
## Achromobacter   
## 3   
## Azospira   
## 3   
## Azovibrio   
## 3   
## Candidatus Captivus   
## 3   
## Candidatus Neoehrlichia   
## 3   
## Chitinibacter   
## 3   
## GOUTA6   
## 3   
## Herminiimonas   
## 3   
## Jannaschia   
## 3   
## Klebsiella   
## 3   
## Labrys   
## 3   
## Magnetospirillum   
## 3   
## MM1   
## 3   
## oc32   
## 3   
## Phyllobacterium   
## 3   
## Pseudaminobacter   
## 3   
## Pseudoduganella   
## 3   
## Pseudofulvimonas   
## 3   
## Ralstonia   
## 3   
## Rheinheimera   
## 3   
## Rhodovarius   
## 3   
## Rhodovulum   
## 3   
## Robbsia   
## 3   
## Rubribacterium   
## 3   
## Sideroxydans   
## 3   
## Silicimonas   
## 3   
## Sodalis   
## 3   
## Sphingoaurantiacus   
## 3   
## Zoogloea   
## 3   
## Aliihoeflea   
## 4   
## Candidatus Megaira   
## 4   
## Citrobacter   
## 4   
## DSSD61   
## 4   
## Geminicoccus   
## 4   
## Haemophilus   
## 4   
## Hirschia   
## 4   
## Kaistia   
## 4   
## Pseudolabrys   
## 4   
## Pseudorhodoferax   
## 4   
## Rhodomicrobium   
## 4   
## Sandaracinobacter   
## 4   
## Sandarakinorhabdus   
## 4   
## Uliginosibacterium   
## 4   
## Yoonia-Loktanella   
## 4   
## [Aquaspirillum] arcticum group   
## 5   
## Belnapia   
## 5   
## Blastomonas   
## 5   
## Buchnera   
## 5   
## Burkholderia-Caballeronia-Paraburkholderia   
## 5   
## Caedibacter   
## 5   
## Candidatus Endoecteinascidia   
## 5   
## Delftia   
## 5   
## Dokdonella   
## 5   
## Ensifer   
## 5   
## Erwinia   
## 5   
## Erythrobacter   
## 5   
## Falsirhodobacter   
## 5   
## FukuN57   
## 5   
## Gemmobacter   
## 5   
## Oxalobacter   
## 5   
## Rhodopila   
## 5   
## Rickettsiella   
## 5   
## Shinella   
## 5   
## Tardiphaga   
## 5   
## Verticiella   
## 5   
## Vogesella   
## 5   
## Woeseia   
## 5   
## Asticcacaulis   
## 6   
## Candidatus Methylopumilus   
## 6   
## Candidatus Nitrotoga   
## 6   
## Ferribacterium   
## 6   
## GKS98 freshwater group   
## 6   
## Methylocella   
## 6   
## Neomegalonema   
## 6   
## Nitratireductor   
## 6   
## Sulfurifustis   
## 6   
## Sutterella   
## 6   
## Yersinia   
## 6   
## Afipia   
## 7   
## Azospirillum   
## 7   
## Candidatus Trichorickettsia   
## 7   
## Chiayiivirga   
## 7   
## Chthonobacter   
## 7   
## Comamonas   
## 7   
## Inhella   
## 7   
## Iodobacter   
## 7   
## Methylocystis   
## 7   
## Methylorosula   
## 7   
## Neisseria   
## 7   
## Roseococcus   
## 7   
## Candidatus Ovatusbacter   
## 8   
## Candidatus Paracaedibacter   
## 8   
## Curvibacter   
## 8   
## Janthinobacterium   
## 8   
## Methylophilus   
## 8   
## Psychroglaciecola   
## 8   
## Silanimonas   
## 8   
## Steroidobacter   
## 8   
## 1174-901-12   
## 9   
## Dongia   
## 9   
## Flavimaricola   
## 9   
## Halomonas   
## 9   
## IS-44   
## 9   
## Panacagrimonas   
## 9   
## Rubritepida   
## 9   
## 966-1   
## 10   
## Bradyrhizobium   
## 10   
## Candidatus Berkiella   
## 10   
## Coxiella   
## 10   
## Defluviimonas   
## 10   
## Luteimonas   
## 10   
## Parablastomonas   
## 10   
## Porphyrobacter   
## 10   
## Pseudorhodoplanes   
## 10   
## Aeromonas   
## 11   
## Alcaligenes   
## 11   
## Crenothrix   
## 11   
## Deefgea   
## 11   
## Nevskia   
## 11   
## Pantoea   
## 11   
## Parvibium   
## 11   
## Propionivibrio   
## 11   
## Stenotrophomonas   
## 11   
## Acinetobacter   
## 12   
## Nordella   
## 12   
## Phenylobacterium   
## 12   
## Polymorphobacter   
## 12   
## Polynucleobacter   
## 12   
## Rhodovastum   
## 12   
## alphaI cluster   
## 13   
## Bauldia   
## 13   
## Candidatus Symbiobacter   
## 13   
## Qipengyuania   
## 13   
## Rhodoplanes   
## 13   
## Skermanella   
## 13   
## UKL13-1   
## 13   
## Bosea   
## 14   
## Lautropia   
## 14   
## Limnobacter   
## 14   
## Methylibium   
## 14   
## MND1   
## 14   
## Pseudoxanthomonas   
## 14   
## Rhodopseudomonas   
## 14   
## Aminobacter   
## 15   
## Aquicella   
## 15   
## Candidatus Alysiosphaera   
## 15   
## OM60(NOR5) clade   
## 15   
## PMMR1   
## 15   
## Thiobacillus   
## 15   
## Candidatus Competibacter   
## 16   
## Cereibacter   
## 16   
## mle1-7   
## 16   
## Sphingopyxis   
## 16   
## Acidibacter   
## 17   
## Ahniella   
## 17   
## Alkanindiges   
## 18   
## Rubrivivax   
## 18   
## Xylophilus   
## 18   
## Acidiphilium   
## 19   
## Pelomonas   
## 19   
## Sphaerotilus   
## 20   
## Duganella   
## 21   
## Sulfuritalea   
## 21   
## Aureimonas   
## 22   
## Reyranella   
## 22   
## Thermomonas   
## 22   
## Caenimonas   
## 24   
## Defluviicoccus   
## 24   
## Nitrosomonas   
## 24   
## Undibacterium   
## 24   
## Mesorhizobium   
## 25   
## Serratia   
## 25   
## Candidatus Accumulibacter   
## 26   
## Ramlibacter   
## 28   
## Sphingobium   
## 29   
## Microvirga   
## 30   
## Aquabacterium   
## 31   
## Acidovorax   
## 32   
## Caulobacter   
## 33   
## Wolbachia   
## 33   
## Paracoccus   
## 34   
## Ideonella   
## 35   
## Lysobacter   
## 36   
## Methylobacterium-Methylorubrum   
## 36   
## Variovorax   
## 36   
## Dechloromonas   
## 37   
## Paucibacter   
## 39   
## Amaricoccus   
## 41   
## Brevundimonas   
## 45   
## CM1G08   
## 47   
## Tabrizicola   
## 47   
## Altererythrobacter   
## 49   
## Noviherbaspirillum   
## 49   
## Rhizorhapis   
## 49   
## AAP99   
## 50   
## Pseudomonas   
## 52   
## Rubellimicrobium   
## 52   
## Pedomicrobium   
## 54   
## Polaromonas   
## 56   
## Allorhizobium-Neorhizobium-Pararhizobium-Rhizobium   
## 63   
## Rhodobacter   
## 69   
## Rickettsia   
## 69   
## Phreatobacter   
## 72   
## Sphingorhabdus   
## 72   
## Devosia   
## 73   
## Hydrogenophaga   
## 77   
## Methylotenera   
## 77   
## Legionella   
## 78   
## Roseomonas   
## 78   
## Arenimonas   
## 82   
## Ellin6067   
## 89   
## Pseudorhodobacter   
## 97   
## Novosphingobium   
## 98   
## Leptothrix   
## 102   
## Rhizobacter   
## 111   
## Limnohabitans   
## 120   
## Sphingomonas   
## 121   
## Hyphomicrobium   
## 130   
## Massilia   
## 239   
## Rhodoferax   
## 277

#Sort by Bacteroidota  
bac <- subset\_taxa(sample\_ps, Phylum=="Bacteroidota")  
ntaxa(bac) #5424

## [1] 5387

sort(table(tax\_table(bac)[,3])) #class. Top class: Bacteroidia (97.7%)

##   
## OC31 Rhodothermia Ignavibacteria Kryptonia SJA-28   
## 1 7 18 20 31   
## Kapabacteria Bacteroidia   
## 35 5266

sort(table(tax\_table(bac)[,5])) #family Top 3: Saprospiraceae (13.9%), Chitinophagaceae (13.7%), Flavobacteriaceae (13.2%)

##   
## Balneolaceae Blattabacteriaceae FFCH9454   
## 1 1 1   
## Ika33 KD1-131 LD-RB-34   
## 1 1 1   
## MSB-3C8 Raineyaceae Rs-E47 termite group   
## 1 1 2   
## SR-FBR-L83 ST-12K33 Z4MB62   
## 2 2 2   
## Cryomorphaceae MWH-CFBk5 Ignavibacteriaceae   
## 5 5 6   
## Porphyromonadaceae Rhodothermaceae Tannerellaceae   
## 6 6 6   
## Barnesiellaceae Marinifilaceae Marinilabiliaceae   
## 7 7 8   
## Muribaculaceae PHOS-HE36 Prevotellaceae   
## 8 9 9   
## SB-5 Amoebophilaceae Bacteroidetes BD2-2   
## 10 16 16   
## 37-13 LiUU-11-161 BSV26   
## 17 18 19   
## KD3-93 Bacteroidetes vadinHA17 Williamwhitmaniaceae   
## 24 29 37   
## Cyclobacteriaceae Bacteroidaceae Lentimicrobiaceae   
## 38 39 39   
## Paludibacteraceae Cytophagaceae Dysgonomonadaceae   
## 40 42 45   
## AKYH767 NS9 marine group Rikenellaceae   
## 56 65 100   
## Weeksellaceae Prolixibacteraceae NS11-12 marine group   
## 106 107 123   
## Crocinitomicaceae Microscillaceae env.OPS 17   
## 135 153 237   
## Sphingobacteriaceae Spirosomaceae Hymenobacteraceae   
## 243 444 464   
## Flavobacteriaceae Chitinophagaceae Saprospiraceae   
## 715 736 752

sort(table(tax\_table(bac)[,6])) #genera Top 3: Flavobacterium (12.5%), Hymenobacter (7.1%), Ferruginibacter (3.4%)

##   
## Aequorivita Alloprevotella   
## 1 1   
## Anaerocella Apibacter   
## 1 1   
## Arachidicoccus BCf9-17 termite group   
## 1 1   
## Bergeyella Blvii28 wastewater-sludge group   
## 1 1   
## Candidatus Cardinium Candidatus Paenicardinium   
## 1 1   
## Capnocytophaga Cecembia   
## 1 1   
## CK06-06-Mud-MAS4B-21 Coprobacter   
## 1 1   
## Elizabethkingia F0058   
## 1 1   
## Filimonas Huanghella   
## 1 1   
## Lutibacter Myroides   
## 1 1   
## Niastella Odoribacter   
## 1 1   
## Olivibacter Persicitalea   
## 1 1   
## Proteiniphilum Pseudoflavitalea   
## 1 1   
## Raineya Rubidimonas   
## 1 1   
## Spongiimonas Thermoflexibacter   
## 1 1   
## Williamwhitmania Fluviimonas   
## 1 2   
## Indibacter Marinoscillum   
## 2 2   
## OLB8 Portibacter   
## 2 2   
## Rhabdobacter Saprospira   
## 2 2   
## Siphonobacter UTBCD1   
## 2 2   
## Acetobacteroides Actibacter   
## 3 3   
## Barnesiella Lewinella   
## 3 3   
## Rikenellaceae RC9 gut group Taibaiella   
## 3 3   
## Candidatus Aquirestis Flavitalea   
## 4 4   
## Gillisia Parasegetibacter   
## 4 4   
## Puia Rubrivirga   
## 4 4   
## Rurimicrobium Butyricimonas   
## 4 5   
## Cloacibacterium Flavihumibacter   
## 5 5   
## Larkinella M2PB4-61 termite group   
## 5 5   
## Parasediminibacterium Aureispira   
## 5 6   
## Cnuella Flaviaesturariibacter   
## 6 6   
## Flexibacter Ignavibacterium   
## 6 6   
## Ohtaekwangia Parabacteroides   
## 6 6   
## Porphyromonas [Cytophaga] xylanolytica group   
## 6 7   
## Roseimarinus Sporocytophaga   
## 7 7   
## Flectobacillus Lentimicrobium   
## 8 8   
## Prevotella Salinimicrobium   
## 8 8   
## Algoriphagus Cesiribacter   
## 9 9   
## Lacibacter Sphingobacterium   
## 9 9   
## Haliscomenobacter Rhodocytophaga   
## 10 10   
## Rufibacter Fibrella   
## 10 11   
## Nibribacter Candidatus Amoebophilus   
## 11 12   
## Pseudarcicella Solitalea   
## 13 13   
## Antarcticibacterium Chryseolinea   
## 14 15   
## Crocinitomix Arcticibacter   
## 15 17   
## Vibrionimonas Pontibacter   
## 17 18   
## Runella Cytophaga   
## 21 23   
## Phaeodactylibacter BSV13   
## 25 26   
## Lacihabitans Rs-D38 termite group   
## 26 28   
## Flavisolibacter OLB12   
## 31 32   
## Dinghuibacter Paludibacter   
## 33 33   
## Sediminibacterium Segetibacter   
## 34 35   
## Rudanella Aurantisolimonas   
## 36 38   
## Bacteroides Dysgonomonas   
## 39 39   
## Arcicella Mucilaginibacter   
## 40 40   
## Mucinivorans WCHB1-32   
## 40 44   
## Adhaeribacter Alistipes   
## 49 50   
## Edaphobaculum Spirosoma   
## 58 58   
## Dyadobacter Chryseobacterium   
## 59 70   
## Emticicia Terrimonas   
## 76 82   
## Fluviicola Pedobacter   
## 120 150   
## Ferruginibacter Hymenobacter   
## 184 374   
## Flavobacterium   
## 677

#Sort by Firmicutes  
firm <- subset\_taxa(sample\_ps, Phylum=="Firmicutes")  
ntaxa(firm) #2211

## [1] 2133

sort(table(tax\_table(firm)[,3])) #class. Top 2: Clostridia (56.8%), Bacilli (38.8%)

##   
## Desulfotomaculia Incertae Sedis Dethiobacteria   
## 1 1 2   
## Moorellia Thermoanaerobacteria Syntrophomonadia   
## 2 2 3   
## Limnochordia Symbiobacteriia Desulfitobacteriia   
## 5 5 29   
## Negativicutes Bacilli Clostridia   
## 43 845 1193

sort(table(tax\_table(firm)[,5])) #family Top 3: Lachnospiraceae (21.0%), Ruminococcaceae (9.9%), Bacillaceae (7.0%)

##   
## Alkalibacteraceae Alkaliphilus   
## 1 1   
## Caldalkalibacillaceae Caldicoprobacteraceae   
## 1 1   
## Candidatus Dichloromethanomonas Catellicoccaceae   
## 1 1   
## Desulfallas-Sporotomaculum Ethanoligenenaceae   
## 1 1   
## Ezakiella Fastidiosipila   
## 1 1   
## Fenollaria Gallicola   
## 1 1   
## Gottschalkia Gracilibacter   
## 1 1   
## Proteiniboraceae Saccharofermentans   
## 1 1   
## Sedimentibacteraceae Tepidimicrobium   
## 1 1   
## Acidaminobacteraceae Defluviitaleaceae   
## 2 2   
## Desulfitibacteraceae Dethiobacteraceae   
## 2 2   
## Fusibacteraceae Garciellaceae   
## 2 2   
## Helcococcus Hydrogenoanaerobacterium   
## 2 2   
## Peptoniphilus Selenomonadaceae   
## 2 2   
## SRB2 UCG-012   
## 2 2   
## Anaerobacterium Aneurinibacillaceae   
## 3 3   
## Finegoldia Gemellaceae   
## 3 3   
## HN-HF0106 Oxobacteraceae   
## 3 3   
## Syntrophomonadaceae type III   
## 3 4   
## Acidaminococcaceae Ercella   
## 5 5   
## Symbiobacteraceae Butyricicoccaceae   
## 5 6   
## Eubacteriaceae Peptococcaceae   
## 6 6   
## Leuconostocaceae Lutispora   
## 7 7   
## Thermincolaceae Tissierella   
## 7 7   
## Veillonellaceae Brevibacillaceae   
## 7 8   
## Aerococcaceae Caloramatoraceae   
## 9 9   
## Alicyclobacillaceae Erysipelatoclostridiaceae   
## 11 11   
## [Eubacterium] coprostanoligenes group UCG-010   
## 12 16   
## Vagococcaceae Anaerococcus   
## 16 17   
## Enterococcaceae Monoglobaceae   
## 17 18   
## Ruminiclostridium Exiguobacteraceae   
## 20 21   
## Lactobacillaceae Sporomusaceae   
## 21 23   
## Anaerovoracaceae Carnobacteriaceae   
## 24 25   
## Peptostreptococcaceae Desulfitobacteriaceae   
## 26 27   
## Spiroplasmataceae Streptococcaceae   
## 30 34   
## Thermoactinomycetaceae Mycoplasmataceae   
## 39 46   
## Christensenellaceae Erysipelotrichaceae   
## 55 58   
## Oscillospiraceae Staphylococcaceae   
## 61 77   
## Paenibacillaceae Planococcaceae   
## 103 105   
## Clostridiaceae Bacillaceae   
## 106 153   
## Ruminococcaceae Lachnospiraceae   
## 211 437

sort(table(tax\_table(firm)[,6])) #genera Top 3: Tyzzerella (5.7%), Bacillus (5.6%), Paenibacillus (3.7%)

##   
## [Ruminococcus] gnavus group Acetoanaerobium   
## 1 1   
## Acetonema Aerosphaera   
## 1 1   
## Alicyclobacillus Alkalibacter   
## 1 1   
## Alkalibacterium Anaerosinus   
## 1 1   
## Anaerospora Angelakisella   
## 1 1   
## CAG-352 Caldalkalibacillus   
## 1 1   
## Caldicoprobacter Candidatus Lumbricincola   
## 1 1   
## Candidatus Stoquefichus Caryophanon   
## 1 1   
## Catabacter Catellicoccus   
## 1 1   
## Catenibacterium Centipeda   
## 1 1   
## CHKCI001 Christensenella   
## 1 1   
## Chungangia Clostridium sensu stricto 10   
## 1 1   
## Clostridium sensu stricto 14 Clostridium sensu stricto 15   
## 1 1   
## Clostridium sensu stricto 7 Dehalobacterium   
## 1 1   
## Dethiobacter Dialister   
## 1 1   
## Dolosigranulum DTU089   
## 1 1   
## Effusibacillus Epulopiscium   
## 1 1   
## Eubacterium Faecalibaculum   
## 1 1   
## Family XIII UCG-001 Flavonifractor   
## 1 1   
## Hazenella Irregularibacter   
## 1 1   
## Jeotgalicoccus Kroppenstedtia   
## 1 1   
## Lachnospiraceae NK4B4 group Lachnospiraceae UCG-003   
## 1 1   
## Marinilactibacillus Megamonas   
## 1 1   
## Negativicoccus Novibacillus   
## 1 1   
## Oceanobacillus Oscillospira   
## 1 1   
## Paraclostridium Pasteuria   
## 1 1   
## Pediococcus Peptococcus   
## 1 1   
## Phocea Proteiniborus   
## 1 1   
## Pseudoflavonifractor Rhabdanaerobium   
## 1 1   
## Rummeliibacillus Salinicoccus   
## 1 1   
## Sarcina Sedimentibacter   
## 1 1   
## Shuttleworthia Sporacetigenium   
## 1 1   
## UCG-009 V9D2013 group   
## 1 1   
## Weissella XBB1006   
## 1 1   
## [Clostridium] innocuum group Acidaminobacter   
## 2 2   
## Ammoniphilus Anaerobium   
## 2 2   
## CAG-56 Caproiciproducens   
## 2 2   
## Catenibacillus Clostridium   
## 2 2   
## Clostridium sensu stricto 2 Defluviitaleaceae UCG-011   
## 2 2   
## Desulfitibacter Erysipelotrichaceae UCG-003   
## 2 2   
## Faecalitalea Fictibacillus   
## 2 2   
## Fusibacter Geothermomicrobium   
## 2 2   
## Globicatella Granulicatella   
## 2 2   
## Intestinibacter Jeotgalibaca   
## 2 2   
## Lachnospiraceae UCG-004 Lachnospiraceae UCG-010   
## 2 2   
## Lachnotalea Negativibacillus   
## 2 2   
## Nosocomiicoccus Paeniclostridium   
## 2 2   
## Phascolarctobacterium Polycladomyces   
## 2 2   
## Risungbinella Sellimonas   
## 2 2   
## Shimazuella Sporomusa   
## 2 2   
## UCG-003 [Eubacterium] ruminantium group   
## 2 3   
## [Eubacterium] ventriosum group Aerococcus   
## 3 3   
## Anaerosporobacter Aneurinibacillus   
## 3 3   
## Atopostipes Clostridium sensu stricto 5   
## 3 3   
## Clostridium sensu stricto 8 Colidextribacter   
## 3 3   
## Domibacillus Facklamia   
## 3 3   
## Gemella Laceyella   
## 3 3   
## Lachnospiraceae UCG-001 Oxobacter   
## 3 3   
## Solibacillus Sporobacter   
## 3 3   
## Syntrophomonas Trichococcus   
## 3 3   
## Ureibacillus Youngiibacter   
## 3 3   
## [Eubacterium] eligens group [Eubacterium] siraeum group   
## 4 4   
## [Eubacterium] xylanophilum group [Ruminococcus] gauvreauii group   
## 4 4   
## Breznakia GCA-900066575   
## 4 4   
## Lachnospiraceae ND3007 group Macrococcus   
## 4 4   
## NK4A214 group Oscillibacter   
## 4 4   
## Paludicola Saccharibacillus   
## 4 4   
## Symbiobacterium Terrisporobacter   
## 4 4   
## Thermobacillus Acetobacterium   
## 4 5   
## Anaerocolumna Butyricicoccus   
## 5 5   
## Clostridium sensu stricto 3 Herbinix   
## 5 5   
## Intestinimonas Lachnospira   
## 5 5   
## Lactococcus Marvinbryantia   
## 5 5   
## Natranaerovirga Thermoactinomyces   
## 5 5   
## Veillonella [Eubacterium] brachy group   
## 5 6   
## Clostridium sensu stricto 12 Dorea   
## 6 6   
## Fonticella Fusicatenibacter   
## 6 6   
## Lachnospiraceae FCS020 group Leuconostoc   
## 6 6   
## Mobilitalea Papillibacter   
## 6 6   
## Anaerostipes Erysipelatoclostridium   
## 7 7   
## Family XIII AD3011 group Geobacillus   
## 7 7   
## Proteiniclasticum Thermincola   
## 7 7   
## Thermoflavimicrobium Brevibacillus   
## 7 8   
## Paenisporosarcina Planifilum   
## 8 8   
## Subdoligranulum Tumebacillus   
## 8 8   
## Turicibacter Anoxybacillus   
## 8 9   
## Cohnella Psychrobacillus   
## 9 9   
## Anaerovorax Cellulosilyticum   
## 10 10   
## Clostridium sensu stricto 9 Incertae Sedis   
## 10 10   
## [Eubacterium] hallii group Carnobacterium   
## 11 11   
## Clostridium sensu stricto 13 Erysipelothrix   
## 11 11   
## UCG-002 Agathobacter   
## 11 12   
## Coprococcus Lachnospiraceae NK4A136 group   
## 12 13   
## Romboutsia Pelosinus   
## 13 15   
## Planomicrobium Ruminococcus   
## 15 15   
## UCG-005 Lysinibacillus   
## 15 16   
## Vagococcus Enterococcus   
## 16 17   
## Mycoplasma [Ruminococcus] torques group   
## 17 18   
## Monoglobus Lactobacillus   
## 18 20   
## Planococcus Roseburia   
## 20 20   
## Exiguobacterium Sporosarcina   
## 21 22   
## Candidatus Soleaferrea Lachnoclostridium   
## 25 26   
## ZOR0006 Desulfosporosinus   
## 26 27   
## Candidatus Bacilloplasma Blautia   
## 28 29   
## Streptococcus Spiroplasma   
## 29 30   
## Christensenellaceae R-7 group Clostridium sensu stricto 1   
## 35 45   
## Staphylococcus Faecalibacterium   
## 69 73   
## Paenibacillus Bacillus   
## 81 124   
## Tyzzerella   
## 126

##Get total numbers of ASVs at each level  
get\_taxa\_unique(sample\_ps, "Phylum") #45 phyla total

## [1] "Proteobacteria" "Bacteroidota"   
## [3] "Cyanobacteria" "Firmicutes"   
## [5] "Actinobacteriota" "Deferribacterota"   
## [7] "Myxococcota" "Campilobacterota"   
## [9] "Synergistota" "Acidobacteriota"   
## [11] "Rs-K70 termite group" "Nitrospirota"   
## [13] "Planctomycetota" "Chloroflexi"   
## [15] "Desulfobacterota" "RCP2-54"   
## [17] "Bdellovibrionota" "Spirochaetota"   
## [19] "Verrucomicrobiota" "Patescibacteria"   
## [21] "Nitrospinota" "Dependentiae"   
## [23] "Fusobacteriota" "Elusimicrobiota"   
## [25] "Gemmatimonadota" "Deinococcota"   
## [27] "Latescibacterota" "Abditibacteriota"   
## [29] "Zixibacteria" "LCP-89"   
## [31] "Fibrobacterota" "Sva0485"   
## [33] "SAR324 clade(Marine group B)" "MBNT15"   
## [35] "NB1-j" "WPS-2"   
## [37] "NKB15" "Entotheonellaeota"   
## [39] "Margulisbacteria" "Methylomirabilota"   
## [41] "Armatimonadota" "Thermotogota"   
## [43] "Sumerlaeota" "Calditrichota"   
## [45] "Caldisericota"

get\_taxa\_unique(sample\_ps, "Class") #117 Classes total

## [1] "Alphaproteobacteria" "Bacteroidia"   
## [3] "Cyanobacteriia" "Bacilli"   
## [5] "Gammaproteobacteria" "Actinobacteria"   
## [7] "Clostridia" "Deferribacteres"   
## [9] "Myxococcia" "Campylobacteria"   
## [11] "Synergistia" "Acidobacteriae"   
## [13] "Acidimicrobiia" NA   
## [15] "Nitrospiria" "Thermoleophilia"   
## [17] "vadinHA49" "Kapabacteria"   
## [19] "Anaerolineae" "Negativicutes"   
## [21] "Blastocatellia" "Desulfovibrionia"   
## [23] "Coriobacteriia" "Polyangia"   
## [25] "KD4-96" "Desulfobacteria"   
## [27] "Desulfuromonadia" "Desulfobulbia"   
## [29] "Planctomycetes" "Oligoflexia"   
## [31] "SJA-28" "Vicinamibacteria"   
## [33] "Syntrophobacteria" "Spirochaetia"   
## [35] "Ignavibacteria" "Desulfitobacteriia"   
## [37] "Verrucomicrobiae" "Chloroflexia"   
## [39] "P9X2b3D02" "Desulfobaccia"   
## [41] "Saccharimonadia" "Phycisphaerae"   
## [43] "Babeliae" "Limnochordia"   
## [45] "Parcubacteria" "Sericytochromatia"   
## [47] "Gracilibacteria" "Vampirivibrionia"   
## [49] "Fusobacteriia" "Endomicrobia"   
## [51] "Gemmatimonadetes" "Desulfarculia"   
## [53] "Deinococci" "Leptospirillia"   
## [55] "Elusimicrobia" "Lineage IIb"   
## [57] "MB-A2-108" "Thermoanaerobaculia"   
## [59] "Subgroup 11" "Longimicrobia"   
## [61] "Abditibacteria" "Moorellia"   
## [63] "Holophagae" "Thermoanaerobacteria"   
## [65] "Syntrophia" "bacteriap25"   
## [67] "S0134 terrestrial group" "Kryptonia"   
## [69] "Berkelbacteria" "Fibrobacteria"   
## [71] "Dethiobacteria" "Subgroup 22"   
## [73] "Rhodothermia" "Symbiobacteriia"   
## [75] "Omnitrophia" "Bdellovibrionia"   
## [77] "CPR2" "Kiritimatiellae"   
## [79] "Thermodesulfovibrionia" "Subgroup 25"   
## [81] "Gitt-GS-136" "Syntrophorhabdia"   
## [83] "Syntrophomonadia" "Ktedonobacteria"   
## [85] "Lentisphaeria" "Aminicenantia"   
## [87] "Incertae Sedis" "Lineage IIa"   
## [89] "ABY1" "Entotheonellia"   
## [91] "RBG-16-55-12" "Microgenomatia"   
## [93] "Methylomirabilia" "TK10"   
## [95] "Fimbriimonadia" "Lineage IIc"   
## [97] "Subgroup 5" "BD2-11 terrestrial group"  
## [99] "Thermotogae" "Desulfomonilia"   
## [101] "Pla3 lineage" "Sumerlaeia"   
## [103] "Calditrichia" "Dehalococcoidia"   
## [105] "OM190" "Latescibacteria"   
## [107] "WWE3" "4-29-1"   
## [109] "Caldisericia" "Desulfotomaculia"   
## [111] "JG30-KF-CM66" "OC31"   
## [113] "OLB14" "Leptospirae"   
## [115] "Chitinivibrionia" "AT-s3-28"   
## [117] "SHA-26"

get\_taxa\_unique(sample\_ps, "Order") #264 orders total

## [1] "Rickettsiales"   
## [2] "Chitinophagales"   
## [3] "Chloroplast"   
## [4] "Holosporales"   
## [5] "Mycoplasmatales"   
## [6] "Burkholderiales"   
## [7] "Entomoplasmatales"   
## [8] "Caulobacterales"   
## [9] "Lactobacillales"   
## [10] "Staphylococcales"   
## [11] "Cytophagales"   
## [12] "Enterobacterales"   
## [13] "Salinisphaerales"   
## [14] "Sphingomonadales"   
## [15] "Rhizobiales"   
## [16] "Flavobacteriales"   
## [17] "Bacteroidales"   
## [18] "Corynebacteriales"   
## [19] "Rhodobacterales"   
## [20] "RsaHf231"   
## [21] "Cardiobacteriales"   
## [22] "Erysipelotrichales"   
## [23] "Clostridiales"   
## [24] "Oscillospirales"   
## [25] "Lachnospirales"   
## [26] "Deferribacterales"   
## [27] "Myxococcales"   
## [28] "Peptostreptococcales-Tissierellales"  
## [29] NA   
## [30] "Aeromonadales"   
## [31] "Christensenellales"   
## [32] "Campylobacterales"   
## [33] "Hungateiclostridiaceae"   
## [34] "Synergistales"   
## [35] "Subgroup 2"   
## [36] "Pseudomonadales"   
## [37] "Azospirillales"   
## [38] "Exiguobacterales"   
## [39] "Microtrichales"   
## [40] "Micrococcales"   
## [41] "Gammaproteobacteria Incertae Sedis"   
## [42] "Steroidobacterales"   
## [43] "Xanthomonadales"   
## [44] "Sphingobacteriales"   
## [45] "Nitrospirales"   
## [46] "Solirubrobacterales"   
## [47] "Cyanobacteriales"   
## [48] "Leptolyngbyales"   
## [49] "Bacillales"   
## [50] "Oceanospirillales"   
## [51] "Rhodospirillales"   
## [52] "Bryobacterales"   
## [53] "Kapabacteriales"   
## [54] "RBG-13-54-9"   
## [55] "Veillonellales-Selenomonadales"   
## [56] "Acidaminococcales"   
## [57] "Acetobacterales"   
## [58] "Oxyphotobacteria Incertae Sedis"   
## [59] "Blastocatellales"   
## [60] "Monoglobales"   
## [61] "Desulfovibrionales"   
## [62] "PeM15"   
## [63] "Coriobacteriales"   
## [64] "Haliangiales"   
## [65] "Propionibacteriales"   
## [66] "Arenicellales"   
## [67] "Polyangiales"   
## [68] "Run-SP154"   
## [69] "Frankiales"   
## [70] "Pasteurellales"   
## [71] "Pseudanabaenales"   
## [72] "Clostridia vadinBB60 group"   
## [73] "Alteromonadales"   
## [74] "Desulfobacterales"   
## [75] "SJA-15"   
## [76] "Geobacterales"   
## [77] "Micavibrionales"   
## [78] "Paludibaculum"   
## [79] "Anaerolineales"   
## [80] "Synechococcales"   
## [81] "Desulfobulbales"   
## [82] "Reyranellales"   
## [83] "Planctomycetales"   
## [84] "Gaiellales"   
## [85] "OPB41"   
## [86] "Clostridia UCG-014"   
## [87] "Streptomycetales"   
## [88] "Pirellulales"   
## [89] "Legionellales"   
## [90] "Oligoflexales"   
## [91] "Acidobacteriales"   
## [92] "Thermoactinomycetales"   
## [93] "Defluviicoccales"   
## [94] "Subgroup 17"   
## [95] "Caedibacterales"   
## [96] "Syntrophobacterales"   
## [97] "Spirochaetales"   
## [98] "Ignavibacteriales"   
## [99] "24-Nov"   
## [100] "Desulfitobacteriales"   
## [101] "Verrucomicrobiales"   
## [102] "Vicinamibacterales"   
## [103] "Alphaproteobacteria Incertae Sedis"   
## [104] "Gemmatales"   
## [105] "Chloroflexales"   
## [106] "Kineosporiales"   
## [107] "Desulfobaccales"   
## [108] "Saccharimonadales"   
## [109] "Chromatiales"   
## [110] "CCM19a"   
## [111] "PLTA13"   
## [112] "Phycisphaerales"   
## [113] "Paracaedibacterales"   
## [114] "Ardenticatenales"   
## [115] "Paenibacillales"   
## [116] "Babeliales"   
## [117] "RF39"   
## [118] "Diplorickettsiales"   
## [119] "Izemoplasmatales"   
## [120] "Ga0077536"   
## [121] "Hydrogenispora"   
## [122] "Phormidesmiales"   
## [123] "Candidatus Moranbacteria"   
## [124] "Solibacterales"   
## [125] "Competibacterales"   
## [126] "Brevibacillales"   
## [127] "Cellvibrionales"   
## [128] "Micromonosporales"   
## [129] "Alicyclobacillales"   
## [130] "JGI 0000069-P22"   
## [131] "Obscuribacterales"   
## [132] "Tepidisphaerales"   
## [133] "Fusobacteriales"   
## [134] "Tistrellales"   
## [135] "Endomicrobiales"   
## [136] "Gemmatimonadales"   
## [137] "Desulfarculales"   
## [138] "Caldilineales"   
## [139] "Nannocystales"   
## [140] "Piscirickettsiales"   
## [141] "Silvanigrellales"   
## [142] "Deinococcales"   
## [143] "Leptospirillales"   
## [144] "Blfdi19"   
## [145] "Elusimicrobiales"   
## [146] "Peptococcales"   
## [147] "0319-6G20"   
## [148] "Opitutales"   
## [149] "Thermoanaerobaculales"   
## [150] "Gastranaerophilales"   
## [151] "MA-28-I98C"   
## [152] "Chthoniobacterales"   
## [153] "Desulfuromonadaceae"   
## [154] "Longimicrobiales"   
## [155] "Pseudonocardiales"   
## [156] "Bradymonadales"   
## [157] "Abditibacteriales"   
## [158] "Isosphaerales"   
## [159] "Desulfitibacterales"   
## [160] "Thermincolales"   
## [161] "Subgroup 7"   
## [162] "Thermoanaerobacterales"   
## [163] "Thiotrichales"   
## [164] "Syntrophales"   
## [165] "Pyrinomonadales"   
## [166] "CCD24"   
## [167] "Dongiales"   
## [168] "Methylococcales"   
## [169] "Kryptoniales"   
## [170] "Thermomicrobiales"   
## [171] "SBR1031"   
## [172] "R7C24"   
## [173] "0319-7L14"   
## [174] "mle1-27"   
## [175] "EC3"   
## [176] "Fibrobacterales"   
## [177] "Zavarziniales"   
## [178] "20-Jan"   
## [179] "PB19"   
## [180] "Pedosphaerales"   
## [181] "Dethiobacterales"   
## [182] "Actinomarinales"   
## [183] "SepB-3"   
## [184] "Aneurinibacillales"   
## [185] "Candidatus Kaiserbacteria"   
## [186] "Elsterales"   
## [187] "Rhodothermales"   
## [188] "Symbiobacteriales"   
## [189] "PAUC26f"   
## [190] "KF-JG30-C25"   
## [191] "Eubacteriales"   
## [192] "IMCC26256"   
## [193] "Candidatus Nomurabacteria"   
## [194] "Omnitrophales"   
## [195] "Bacteriovoracales"   
## [196] "WCHB1-41"   
## [197] "Acidiferrobacterales"   
## [198] "SM1A07"   
## [199] "Bdellovibrionales"   
## [200] "Micropepsales"   
## [201] "AMGG11"   
## [202] "SAR11 clade"   
## [203] "Caenarcaniphilales"   
## [204] "GOUTB8"   
## [205] "Syntrophorhabdales"   
## [206] "Syntrophomonadales"   
## [207] "Kiritimatiellales"   
## [208] "Bacteroidetes VC2.1 Bac22"   
## [209] "C0119"   
## [210] "Euzebyales"   
## [211] "Coxiellales"   
## [212] "1013-28-CG33"   
## [213] "Victivallales"   
## [214] "Aminicenantales"   
## [215] "DTU014"   
## [216] "Candidatus Campbellbacteria"   
## [217] "Caldicoprobacterales"   
## [218] "DS-100"   
## [219] "CCM11a"   
## [220] "Candidatus Komeilibacteria"   
## [221] "JG36-GS-52"   
## [222] "HglApr721"   
## [223] "Entotheonellales"   
## [224] "Gloeobacterales"   
## [225] "Lineage IV"   
## [226] "Candidatus Pacebacteria"   
## [227] "Elev-16S-1166"   
## [228] "Methylomirabilales"   
## [229] "Streptosporangiales"   
## [230] "Fimbriimonadales"   
## [231] "Candidatus Azambacteria"   
## [232] "MidBa8"   
## [233] "Gracilibacteraceae"   
## [234] "Thermotogales"   
## [235] "Desulfomonilales"   
## [236] "Geothermobacteraceae"   
## [237] "Ferrovibrionales"   
## [238] "Sumerlaeales"   
## [239] "Vibrionales"   
## [240] "Actinomycetales"   
## [241] "Calditrichales"   
## [242] "FFCH16263"   
## [243] "LD1-PB3"   
## [244] "S085"   
## [245] "Latescibacterales"   
## [246] "Thermales"   
## [247] "KI89A clade"   
## [248] "Acidithiobacillales"   
## [249] "Lentisphaerales"   
## [250] "Vampirovibrionales"   
## [251] "Subgroup 9"   
## [252] "Rokubacteriales"   
## [253] "Absconditabacteriales (SR1)"   
## [254] "Desulfatiglandales"   
## [255] "Caldalkalibacillales"   
## [256] "EV818SWSAP88"   
## [257] "Caldisericales"   
## [258] "Desulfotomaculales"   
## [259] "Candidatus Vogelbacteria"   
## [260] "Kallotenuales"   
## [261] "Balneolales"   
## [262] "Holophagales"   
## [263] "Leptospirales"   
## [264] "Bifidobacteriales"

get\_taxa\_unique(sample\_ps, "Family") #415 families total

## [1] "Rickettsiaceae"   
## [2] "Chitinophagaceae"   
## [3] NA   
## [4] "Holosporaceae"   
## [5] "Anaplasmataceae"   
## [6] "Mycoplasmataceae"   
## [7] "Comamonadaceae"   
## [8] "Spiroplasmataceae"   
## [9] "Caulobacteraceae"   
## [10] "Carnobacteriaceae"   
## [11] "Staphylococcaceae"   
## [12] "Spirosomaceae"   
## [13] "Yersiniaceae"   
## [14] "Solimonadaceae"   
## [15] "Sphingomonadaceae"   
## [16] "Rhizobiales Incertae Sedis"   
## [17] "Burkholderiaceae"   
## [18] "Flavobacteriaceae"   
## [19] "Saprospiraceae"   
## [20] "Rikenellaceae"   
## [21] "Alcaligenaceae"   
## [22] "Oxalobacteraceae"   
## [23] "Nocardiaceae"   
## [24] "Rhodobacteraceae"   
## [25] "Erysipelotrichaceae"   
## [26] "Dysgonomonadaceae"   
## [27] "Clostridiaceae"   
## [28] "Ruminococcaceae"   
## [29] "Candidatus Hepatincola"   
## [30] "Weeksellaceae"   
## [31] "Lachnospiraceae"   
## [32] "Deferribacteraceae"   
## [33] "Myxococcaceae"   
## [34] "Peptostreptococcaceae"   
## [35] "Aeromonadaceae"   
## [36] "Rhodocyclaceae"   
## [37] "Sutterellaceae"   
## [38] "Williamwhitmaniaceae"   
## [39] "Christensenellaceae"   
## [40] "Sulfurospirillaceae"   
## [41] "Crocinitomicaceae"   
## [42] "[Eubacterium] coprostanoligenes group"  
## [43] "Methylophilaceae"   
## [44] "Rhizobiaceae"   
## [45] "Streptococcaceae"   
## [46] "Xanthobacteraceae"   
## [47] "Saccharofermentans"   
## [48] "Synergistaceae"   
## [49] "Pseudomonadaceae"   
## [50] "Anaerococcus"   
## [51] "Azospirillaceae"   
## [52] "Exiguobacteraceae"   
## [53] "Ilumatobacteraceae"   
## [54] "Oscillospiraceae"   
## [55] "Beijerinckiaceae"   
## [56] "Bacteroidaceae"   
## [57] "Unknown Family"   
## [58] "Steroidobacteraceae"   
## [59] "Erysipelatoclostridiaceae"   
## [60] "Xanthomonadaceae"   
## [61] "Corynebacteriaceae"   
## [62] "env.OPS 17"   
## [63] "Sphingobacteriaceae"   
## [64] "NS11-12 marine group"   
## [65] "Nitrospiraceae"   
## [66] "Moraxellaceae"   
## [67] "TRA3-20"   
## [68] "Micrococcaceae"   
## [69] "Solirubrobacteraceae"   
## [70] "Phormidiaceae"   
## [71] "Rs-E47 termite group"   
## [72] "Enterococcaceae"   
## [73] "Leptolyngbyaceae"   
## [74] "Microbacteriaceae"   
## [75] "Mitochondria"   
## [76] "Vagococcaceae"   
## [77] "Hyphomicrobiaceae"   
## [78] "Bacillaceae"   
## [79] "Halomonadaceae"   
## [80] "Chitinibacteraceae"   
## [81] "Magnetospirillaceae"   
## [82] "Bryobacteraceae"   
## [83] "Hymenobacteraceae"   
## [84] "Nitrosomonadaceae"   
## [85] "C2U"   
## [86] "Morganellaceae"   
## [87] "Acidaminococcaceae"   
## [88] "Prolixibacteraceae"   
## [89] "Tannerellaceae"   
## [90] "Acetobacteraceae"   
## [91] "Blastocatellaceae"   
## [92] "Anaerovoracaceae"   
## [93] "Barnesiellaceae"   
## [94] "Methyloligellaceae"   
## [95] "Monoglobaceae"   
## [96] "Devosiaceae"   
## [97] "Erwiniaceae"   
## [98] "UCG-010"   
## [99] "Mycobacteriaceae"   
## [100] "Desulfovibrionaceae"   
## [101] "Chroococcidiopsaceae"   
## [102] "Coriobacteriales Incertae Sedis"   
## [103] "Haliangiaceae"   
## [104] "Paludibacteraceae"   
## [105] "Propionibacteriaceae"   
## [106] "Arenicellaceae"   
## [107] "37-13"   
## [108] "T34"   
## [109] "Lactobacillaceae"   
## [110] "Nocardioidaceae"   
## [111] "Wohlfahrtiimonadaceae"   
## [112] "Neisseriaceae"   
## [113] "Peptoniphilus"   
## [114] "LiUU-11-161"   
## [115] "Amoebophilaceae"   
## [116] "Polyangiaceae"   
## [117] "Sporichthyaceae"   
## [118] "Enterobacteriaceae"   
## [119] "Butyricicoccaceae"   
## [120] "NS9 marine group"   
## [121] "SC-I-84"   
## [122] "Planococcaceae"   
## [123] "Pasteurellaceae"   
## [124] "Pseudanabaenaceae"   
## [125] "67-14"   
## [126] "Intrasporangiaceae"   
## [127] "Leuconostocaceae"   
## [128] "Alteromonadaceae"   
## [129] "Desulfosarcinaceae"   
## [130] "Arcobacteraceae"   
## [131] "Sandaracinaceae"   
## [132] "Hyphomonadaceae"   
## [133] "Chromobacteriaceae"   
## [134] "Microtrichaceae"   
## [135] "Geobacteraceae"   
## [136] "Micavibrionaceae"   
## [137] "B1-7BS"   
## [138] "Anaerolineaceae"   
## [139] "Cyanobiaceae"   
## [140] "D05-2"   
## [141] "Desulfocapsaceae"   
## [142] "Reyranellaceae"   
## [143] "Woeseiaceae"   
## [144] "Microscillaceae"   
## [145] "Gallionellaceae"   
## [146] "Rubinisphaeraceae"   
## [147] "Lentimicrobiaceae"   
## [148] "Aerococcaceae"   
## [149] "Cytophagaceae"   
## [150] "Phaselicystidaceae"   
## [151] "Cyclobacteriaceae"   
## [152] "Finegoldia"   
## [153] "Promicromonosporaceae"   
## [154] "Streptomycetaceae"   
## [155] "Pirellulaceae"   
## [156] "Cellulomonadaceae"   
## [157] "Legionellaceae"   
## [158] "Hydrogenophilaceae"   
## [159] "KD3-93"   
## [160] "Caloramatoraceae"   
## [161] "Oligoflexus"   
## [162] "Geodermatophilaceae"   
## [163] "Gaiellaceae"   
## [164] "Acidobacteriaceae (Subgroup 1)"   
## [165] "Thermoactinomycetaceae"   
## [166] "Defluviicoccaceae"   
## [167] "Caedibacteraceae"   
## [168] "Spirochaetaceae"   
## [169] "Xenococcaceae"   
## [170] "PHOS-HE36"   
## [171] "AKYH767"   
## [172] "Coleofasciculaceae"   
## [173] "Chamaesiphonaceae"   
## [174] "Desulfitobacteriaceae"   
## [175] "Verrucomicrobiaceae"   
## [176] "Gemellaceae"   
## [177] "Dermacoccaceae"   
## [178] "Gemmataceae"   
## [179] "Rhodanobacteraceae"   
## [180] "Herpetosiphonaceae"   
## [181] "Kineosporiaceae"   
## [182] "A0839"   
## [183] "Helcococcus"   
## [184] "Cyanobacteriaceae"   
## [185] "SM2D12"   
## [186] "Desulfobaccaceae"   
## [187] "Chromatiaceae"   
## [188] "BIrii41"   
## [189] "Nostocaceae"   
## [190] "Phycisphaeraceae"   
## [191] "Paracaedibacteraceae"   
## [192] "Ardenticatenaceae"   
## [193] "Paenibacillaceae"   
## [194] "Fenollaria"   
## [195] "Diplorickettsiaceae"   
## [196] "Rhodomicrobiaceae"   
## [197] "Bacteroidetes vadinHA17"   
## [198] "Nodosilineaceae"   
## [199] "Nakamurellaceae"   
## [200] "Solibacteraceae"   
## [201] "Competibacteraceae"   
## [202] "Cryomorphaceae"   
## [203] "Brevibacillaceae"   
## [204] "Dermabacteraceae"   
## [205] "Halieaceae"   
## [206] "Micromonosporaceae"   
## [207] "Alicyclobacillaceae"   
## [208] "Marinilabiliaceae"   
## [209] "Obscuribacteraceae"   
## [210] "WD2101 soil group"   
## [211] "Sanguibacteraceae"   
## [212] "Rubritaleaceae"   
## [213] "Fusobacteriaceae"   
## [214] "Geminicoccaceae"   
## [215] "KF-JG30-B3"   
## [216] "AB1"   
## [217] "Endomicrobiaceae"   
## [218] "Salinisphaeraceae"   
## [219] "Rhodospirillaceae"   
## [220] "Leptotrichiaceae"   
## [221] "Gemmatimonadaceae"   
## [222] "Desulfobulbaceae"   
## [223] "Caldilineaceae"   
## [224] "Nannocystaceae"   
## [225] "Piscirickettsiaceae"   
## [226] "Bacteroidetes BD2-2"   
## [227] "Silvanigrellaceae"   
## [228] "Deinococcaceae"   
## [229] "Leptospirillaceae"   
## [230] "Elusimicrobiaceae"   
## [231] "Peptococcaceae"   
## [232] "Opitutaceae"   
## [233] "Ignavibacteriaceae"   
## [234] "Bogoriellaceae"   
## [235] "Thermoanaerobaculaceae"   
## [236] "Chthoniobacteraceae"   
## [237] "Desulfuromonas"   
## [238] "Longimicrobiaceae"   
## [239] "Pseudonocardiaceae"   
## [240] "Vicinamibacteraceae"   
## [241] "Abditibacteriaceae"   
## [242] "Isosphaeraceae"   
## [243] "Chloroflexaceae"   
## [244] "Desulfitibacteraceae"   
## [245] "Stappiaceae"   
## [246] "Thermincolaceae"   
## [247] "Iamiaceae"   
## [248] "SRB2"   
## [249] "Thiotrichaceae"   
## [250] "Dietziaceae"   
## [251] "Pyrinomonadaceae"   
## [252] "Dongiaceae"   
## [253] "Ruminiclostridium"   
## [254] "Syntrophobacteraceae"   
## [255] "LD-RB-34"   
## [256] "Anaeromyxobacteraceae"   
## [257] "Sporomusaceae"   
## [258] "Methylomonadaceae"   
## [259] "bac2nit3"   
## [260] "BSV26"   
## [261] "Defluviitaleaceae"   
## [262] "JG30-KF-CM45"   
## [263] "A4b"   
## [264] "Proteiniboraceae"   
## [265] "Atopobiaceae"   
## [266] "Fibrobacteraceae"   
## [267] "Ezakiella"   
## [268] "Vermiphilaceae"   
## [269] "Zavarziniaceae"   
## [270] "Koribacteraceae"   
## [271] "Brevibacteriaceae"   
## [272] "Budviciaceae"   
## [273] "Phormidesmiaceae"   
## [274] "Pedosphaeraceae"   
## [275] "Selenomonadaceae"   
## [276] "Dethiobacteraceae"   
## [277] "Trueperaceae"   
## [278] "Aneurinibacillaceae"   
## [279] "Pleomorphomonadaceae"   
## [280] "Elsteraceae"   
## [281] "Rhodothermaceae"   
## [282] "Symbiobacteraceae"   
## [283] "Tepidisphaeraceae"   
## [284] "MWH-CFBk5"   
## [285] "Inquilinaceae"   
## [286] "SB-5"   
## [287] "Alkalibacteraceae"   
## [288] "Ethanoligenenaceae"   
## [289] "Eubacteriaceae"   
## [290] "Omnitrophaceae"   
## [291] "Roseiflexaceae"   
## [292] "Bacteriovoracaceae"   
## [293] "Azospirillales Incertae Sedis"   
## [294] "Candidatus Jidaibacter"   
## [295] "Coriobacteriaceae"   
## [296] "Schlesneriaceae"   
## [297] "Frankiaceae"   
## [298] "Pectobacteriaceae"   
## [299] "Ercella"   
## [300] "Marinifilaceae"   
## [301] "Acidiferrobacteraceae"   
## [302] "Chitinimonadaceae"   
## [303] "Porphyromonadaceae"   
## [304] "Methylopilaceae"   
## [305] "Bdellovibrionaceae"   
## [306] "Spongiibacteraceae"   
## [307] "Micropepsaceae"   
## [308] "Gallicola"   
## [309] "Prevotellaceae"   
## [310] "Eggerthellaceae"   
## [311] "Syntrophaceae"   
## [312] "Clade III"   
## [313] "Labraceae"   
## [314] "Puniceicoccaceae"   
## [315] "Oxobacteraceae"   
## [316] "Syntrophorhabdaceae"   
## [317] "Syntrophomonadaceae"   
## [318] "Methylococcaceae"   
## [319] "Tissierella"   
## [320] "B122"   
## [321] "Sulfurimonadaceae"   
## [322] "Kiritimatiellaceae"   
## [323] "Acidothermaceae"   
## [324] "Akkermansiaceae"   
## [325] "Veillonellaceae"   
## [326] "Aquaspirillaceae"   
## [327] "Muribaculaceae"   
## [328] "Euzebyaceae"   
## [329] "Coxiellaceae"   
## [330] "Kaistiaceae"   
## [331] "Microcystaceae"   
## [332] "Ika33"   
## [333] "vadinBE97"   
## [334] "ST-12K33"   
## [335] "A21b"   
## [336] "Saccharimonadaceae"   
## [337] "Caldicoprobacteraceae"   
## [338] "HN-HF0106"   
## [339] "Tsukamurellaceae"   
## [340] "Anaerobacterium"   
## [341] "Demequinaceae"   
## [342] "Entotheonellaceae"   
## [343] "Gloeobacteraceae"   
## [344] "Smithellaceae"   
## [345] "Dermatophilaceae"   
## [346] "Methylomirabilaceae"   
## [347] "Acidaminobacteraceae"   
## [348] "Streptosporangiaceae"   
## [349] "Fimbriimonadaceae"   
## [350] "Alkaliphilus"   
## [351] "FFCH9454"   
## [352] "type III"   
## [353] "Geitlerinemaceae"   
## [354] "Lutispora"   
## [355] "Fervidobacteriaceae"   
## [356] "Tepidimicrobium"   
## [357] "Shewanellaceae"   
## [358] "Garciellaceae"   
## [359] "Synechococcales Incertae Sedis"   
## [360] "Desulfomonilaceae"   
## [361] "Geothermobacter"   
## [362] "Taonella"   
## [363] "Sumerlaeaceae"   
## [364] "AKYG1722"   
## [365] "Vibrionaceae"   
## [366] "Actinomycetaceae"   
## [367] "Calditrichaceae"   
## [368] "27F-1492R"   
## [369] "Z4MB62"   
## [370] "Latescibacteraceae"   
## [371] "LWQ8"   
## [372] "Thermaceae"   
## [373] "Babeliaceae"   
## [374] "Nocardiopsaceae"   
## [375] "Amb-16S-1323"   
## [376] "Acidithiobacillaceae"   
## [377] "Lentisphaeraceae"   
## [378] "Desulfatiglandaceae"   
## [379] "SR-FBR-L83"   
## [380] "Caldalkalibacillaceae"   
## [381] "Thermomonosporaceae"   
## [382] "Gloeocapsaceae"   
## [383] "WCHB1-02"   
## [384] "Sulfuricellaceae"   
## [385] "Desulfallas-Sporotomaculum"   
## [386] "Fusibacteraceae"   
## [387] "Hydrogenoanaerobacterium"   
## [388] "Gracilibacter"   
## [389] "Fastidiosipila"   
## [390] "Pseudohongiellaceae"   
## [391] "Catellicoccaceae"   
## [392] "AKIW781"   
## [393] "Gottschalkia"   
## [394] "KD1-131"   
## [395] "WX65"   
## [396] "Balneolaceae"   
## [397] "Raineyaceae"   
## [398] "UBA12409"   
## [399] "UCG-012"   
## [400] "MSB-3C8"   
## [401] "Cryptosporangiaceae"   
## [402] "Helicobacteraceae"   
## [403] "Desulfococcaceae"   
## [404] "Holophagaceae"   
## [405] "Leeiaceae"   
## [406] "Candidatus Dichloromethanomonas"   
## [407] "Sedimentibacteraceae"   
## [408] "Desulforegulaceae"   
## [409] "Leptospiraceae"   
## [410] "Parvularculaceae"   
## [411] "BBMC-4"   
## [412] "Blattabacteriaceae"   
## [413] "Xiphinematobacteraceae"   
## [414] "Bifidobacteriaceae"

get\_taxa\_unique(sample\_ps, "Genus") #1058 genera total

## [1] "Rickettsia"   
## [2] "Vibrionimonas"   
## [3] NA   
## [4] "Wolbachia"   
## [5] "Candidatus Bacilloplasma"   
## [6] "Ideonella"   
## [7] "Spiroplasma"   
## [8] "Candidatus Neoehrlichia"   
## [9] "Rhodoferax"   
## [10] "Carnobacterium"   
## [11] "Staphylococcus"   
## [12] "Lacihabitans"   
## [13] "Nevskia"   
## [14] "Sphingorhabdus"   
## [15] "Flavobacterium"   
## [16] "Sediminibacterium"   
## [17] "Mucinivorans"   
## [18] "Alistipes"   
## [19] "Serratia"   
## [20] "Alcaligenes"   
## [21] "CM1G08"   
## [22] "Limnohabitans"   
## [23] "Rhodococcus"   
## [24] "Tabrizicola"   
## [25] "Hydrogenophaga"   
## [26] "Ferruginibacter"   
## [27] "Dysgonomonas"   
## [28] "Clostridium sensu stricto 13"   
## [29] "Ruminococcus"   
## [30] "Pseudorhodobacter"   
## [31] "Achromobacter"   
## [32] "Emticicia"   
## [33] "Sphaerotilus"   
## [34] "Apibacter"   
## [35] "Faecalibacterium"   
## [36] "Agathobacter"   
## [37] "Mucispirillum"   
## [38] "Melittangium"   
## [39] "Cereibacter"   
## [40] "ZOR0006"   
## [41] "Romboutsia"   
## [42] "Rhodobacter"   
## [43] "Runella"   
## [44] "Aeromonas"   
## [45] "Dechloromonas"   
## [46] "AAP99"   
## [47] "Rs-D38 termite group"   
## [48] "Massilia"   
## [49] "Christensenellaceae R-7 group"   
## [50] "Rhizorhapis"   
## [51] "Sulfurospirillum"   
## [52] "Pelomonas"   
## [53] "[Eubacterium] hallii group"   
## [54] "Blautia"   
## [55] "Fluviicola"   
## [56] "Candidatus Methylopumilus"   
## [57] "Candidatus Soleaferrea"   
## [58] "Streptococcus"   
## [59] "Mycoplasma"   
## [60] "Anaerocella"   
## [61] "Bradyrhizobium"   
## [62] "Chryseobacterium"   
## [63] "Clostridium sensu stricto 1"   
## [64] "Allorhizobium-Neorhizobium-Pararhizobium-Rhizobium"  
## [65] "Fusicatenibacter"   
## [66] "Propionivibrio"   
## [67] "Pseudomonas"   
## [68] "Skermanella"   
## [69] "M2PB4-61 termite group"   
## [70] "Subdoligranulum"   
## [71] "Exiguobacterium"   
## [72] "Ilumatobacter"   
## [73] "Flavimaricola"   
## [74] "UCG-002"   
## [75] "Rhizobacter"   
## [76] "Methylobacterium-Methylorubrum"   
## [77] "Bacteroides"   
## [78] "Brevundimonas"   
## [79] "Turicibacter"   
## [80] "Aureimonas"   
## [81] "Paracoccus"   
## [82] "Acidibacter"   
## [83] "Tyzzerella"   
## [84] "Sphingopyxis"   
## [85] "Erysipelotrichaceae UCG-003"   
## [86] "Arenimonas"   
## [87] "Corynebacterium"   
## [88] "Leptothrix"   
## [89] "Pedobacter"   
## [90] "Pseudarcicella"   
## [91] "Nitrospira"   
## [92] "Acinetobacter"   
## [93] "Sphingomonas"   
## [94] "Novosphingobium"   
## [95] "[Ruminococcus] torques group"   
## [96] "Variovorax"   
## [97] "Planktothrix NIVA-CYA 15"   
## [98] "Defluviimonas"   
## [99] "Enterococcus"   
## [100] "Phreatobacter"   
## [101] "Microterricola"   
## [102] "Vagococcus"   
## [103] "Arcicella"   
## [104] "Alkanindiges"   
## [105] "Hyphomicrobium"   
## [106] "Bacillus"   
## [107] "Renibacterium"   
## [108] "[Eubacterium] siraeum group"   
## [109] "Burkholderia-Caballeronia-Paraburkholderia"   
## [110] "Halomonas"   
## [111] "Parvibium"   
## [112] "Candidatus Symbiobacter"   
## [113] "Iodobacter"   
## [114] "Aquabacterium"   
## [115] "Anaerostipes"   
## [116] "Telmatospirillum"   
## [117] "Caulobacter"   
## [118] "Roseburia"   
## [119] "Intestinibacter"   
## [120] "Bryobacter"   
## [121] "Methylotenera"   
## [122] "Breznakia"   
## [123] "Hymenobacter"   
## [124] "Haliscomenobacter"   
## [125] "[Ruminococcus] gauvreauii group"   
## [126] "Polymorphobacter"   
## [127] "Noviherbaspirillum"   
## [128] "Sandaracinobacter"   
## [129] "[Eubacterium] ruminantium group"   
## [130] "Azovibrio"   
## [131] "Nitrosomonas"   
## [132] "Rubellimicrobium"   
## [133] "Acidovorax"   
## [134] "Qipengyuania"   
## [135] "Arsenophonus"   
## [136] "Gemmobacter"   
## [137] "Rubrivivax"   
## [138] "Polaromonas"   
## [139] "Dorea"   
## [140] "Terrimonas"   
## [141] "Nesterenkonia"   
## [142] "Parabacteroides"   
## [143] "Roseomonas"   
## [144] "Dinghuibacter"   
## [145] "Micrococcus"   
## [146] "Lachnospira"   
## [147] "Calothrix KVSF5"   
## [148] "Flectobacillus"   
## [149] "Blastocatella"   
## [150] "Fretibacterium"   
## [151] "Arthrobacter"   
## [152] "[Eubacterium] brachy group"   
## [153] "Barnesiella"   
## [154] "Curvibacter"   
## [155] "Rathayibacter"   
## [156] "Monoglobus"   
## [157] "Candidatus Aquirestis"   
## [158] "Devosia"   
## [159] "Clavibacter"   
## [160] "CL500-29 marine group"   
## [161] "Pantoea"   
## [162] "Sutterella"   
## [163] "Mycobacterium"   
## [164] "Desulfovibrio"   
## [165] "Aliterella"   
## [166] "Methylophilus"   
## [167] "Solitalea"   
## [168] "Haliangium"   
## [169] "Parasediminibacterium"   
## [170] "Erysipelatoclostridium"   
## [171] "Moraxella"   
## [172] "NK4A214 group"   
## [173] "[Eubacterium] ventriosum group"   
## [174] "Altererythrobacter"   
## [175] "Paeniclostridium"   
## [176] "DSSF69"   
## [177] "Coprococcus"   
## [178] "Crocinitomix"   
## [179] "Cutibacterium"   
## [180] "Lactobacillus"   
## [181] "alphaI cluster"   
## [182] "Nocardioides"   
## [183] "FukuN57"   
## [184] "Azospira"   
## [185] "Ellin6067"   
## [186] "Porphyrobacter"   
## [187] "Rurimicrobium"   
## [188] "Galbitalea"   
## [189] "Lachnoclostridium"   
## [190] "Candidatus Amoebophilus"   
## [191] "Pseudoduganella"   
## [192] "Pajaroellobacter"   
## [193] "Beijerinckia"   
## [194] "Aetherobacter"   
## [195] "Colidextribacter"   
## [196] "Spirosoma"   
## [197] "Geobacillus"   
## [198] "Marisediminicola"   
## [199] "Raoultella"   
## [200] "OLB8"   
## [201] "Pseudoflavitalea"   
## [202] "Curtobacterium"   
## [203] "UCG-009"   
## [204] "Kocuria"   
## [205] "Rufibacter"   
## [206] "Planococcus"   
## [207] "Microvirga"   
## [208] "Haemophilus"   
## [209] "Pseudanabaena PCC-6802"   
## [210] "Family XIII UCG-001"   
## [211] "Knoellia"   
## [212] "Yoonia-Loktanella"   
## [213] "Marmoricola"   
## [214] "[Eubacterium] xylanophilum group"   
## [215] "Aminobacter"   
## [216] "Leuconostoc"   
## [217] "UCG-005"   
## [218] "Methylorosula"   
## [219] "Rheinheimera"   
## [220] "Rothia"   
## [221] "Buchnera"   
## [222] "Sva0081 sediment group"   
## [223] "Family XIII AD3011 group"   
## [224] "Antarcticibacterium"   
## [225] "Pseudarcobacter"   
## [226] "Planomicrobium"   
## [227] "Lachnospiraceae NK4A136 group"   
## [228] "Polynucleobacter"   
## [229] "Candidatus Limnoluna"   
## [230] "[Eubacterium] eligens group"   
## [231] "Deefgea"   
## [232] "Sandaracinus"   
## [233] "UKL13-1"   
## [234] "Ramlibacter"   
## [235] "Sphingobacterium"   
## [236] "Neisseria"   
## [237] "Vogesella"   
## [238] "Parablastomonas"   
## [239] "Polyangium"   
## [240] "Geobacter"   
## [241] "Leifsonia"   
## [242] "Acidiphilium"   
## [243] "GKS98 freshwater group"   
## [244] "Cloacibacterium"   
## [245] "Paenisporosarcina"   
## [246] "Nosocomiicoccus"   
## [247] "Fluviimonas"   
## [248] "Cyanobium PCC-6307"   
## [249] "Butyricicoccus"   
## [250] "Reyranella"   
## [251] "CAG-56"   
## [252] "Stenotrophomonas"   
## [253] "Woeseia"   
## [254] "Lautropia"   
## [255] "Shinella"   
## [256] "Amaricoccus"   
## [257] "Candidatus Nitrotoga"   
## [258] "Oxalobacter"   
## [259] "Granulicatella"   
## [260] "Cryobacterium"   
## [261] "[Desulfobacterium] catecholicum group"   
## [262] "Ornithinimicrobium"   
## [263] "SH-PL14"   
## [264] "Aurantisolimonas"   
## [265] "Steroidobacter"   
## [266] "Erythrobacter"   
## [267] "Williamsia"   
## [268] "Paenarthrobacter"   
## [269] "Sporosarcina"   
## [270] "Synechococcus PCC-7502"   
## [271] "Desulfatirhabdium"   
## [272] "Aerococcus"   
## [273] "Tychonema CCAP 1459-11B"   
## [274] "Cytophaga"   
## [275] "Phaselicystis"   
## [276] "Bosea"   
## [277] "Lysobacter"   
## [278] "Rhodoluna"   
## [279] "Rudanella"   
## [280] "Promicromonospora"   
## [281] "Lachnospiraceae ND3007 group"   
## [282] "Panacagrimonas"   
## [283] "UCG-003"   
## [284] "Streptomyces"   
## [285] "Hirschia"   
## [286] "Candidatus Microthrix"   
## [287] "Paucibacter"   
## [288] "Edaphobaculum"   
## [289] "Pirellula"   
## [290] "Arcticibacter"   
## [291] "Flavisolibacter"   
## [292] "Cellulomonas"   
## [293] "Legionella"   
## [294] "Paludibacter"   
## [295] "Thiobacillus"   
## [296] "Tropicimonas"   
## [297] "Subtercola"   
## [298] "XBB1006"   
## [299] "Yersinia"   
## [300] "Pedomicrobium"   
## [301] "Tardiphaga"   
## [302] "Nitratireductor"   
## [303] "Sphingobium"   
## [304] "Fonticella"   
## [305] "mle1-7"   
## [306] "Limnobacter"   
## [307] "Tibeticola"   
## [308] "P3OB-42"   
## [309] "1174-901-12"   
## [310] "Undibacterium"   
## [311] "Mesorhizobium"   
## [312] "Adhaeribacter"   
## [313] "Parafrigoribacterium"   
## [314] "Blastococcus"   
## [315] "Gaiella"   
## [316] "Phascolarctobacterium"   
## [317] "Aridibacter"   
## [318] "Edaphobacter"   
## [319] "Pseudaminobacter"   
## [320] "Blastomonas"   
## [321] "Archangium"   
## [322] "Zoogloea"   
## [323] "OLB12"   
## [324] "Dolosigranulum"   
## [325] "Defluviicoccus"   
## [326] "Caedibacter"   
## [327] "Lachnospiraceae UCG-001"   
## [328] "Methylocella"   
## [329] "Pseudanabaena PCC-7429"   
## [330] "Spirochaeta 2"   
## [331] "Roseococcus"   
## [332] "Mucilaginibacter"   
## [333] "Pleurocapsa PCC-7319"   
## [334] "Pseudorhodoplanes"   
## [335] "Glutamicibacter"   
## [336] "Nitrobacter"   
## [337] "Pseudoxanthomonas"   
## [338] "Facklamia"   
## [339] "Symplocastrum CPER-KK1"   
## [340] "Macrococcus"   
## [341] "Chamaesiphon PCC-6605"   
## [342] "Erysipelothrix"   
## [343] "Pseudochrobactrum"   
## [344] "Jannaschia"   
## [345] "Anaerovorax"   
## [346] "Puia"   
## [347] "Terrisporobacter"   
## [348] "Desulfosporosinus"   
## [349] "Anaerobium"   
## [350] "Sandarakinorhabdus"   
## [351] "Phyllobacterium"   
## [352] "Thermomonas"   
## [353] "Starkeya"   
## [354] "Duganella"   
## [355] "Neomegalonema"   
## [356] "Gemella"   
## [357] "Parasegetibacter"   
## [358] "Dermacoccus"   
## [359] "Phenylobacterium"   
## [360] "Rhodocytophaga"   
## [361] "Frigoribacterium"   
## [362] "Stenotrophobacter"   
## [363] "Lysinibacillus"   
## [364] "Ahniella"   
## [365] "hgcI clade"   
## [366] "Phaeodactylibacter"   
## [367] "Herpetosiphon"   
## [368] "Robbsia"   
## [369] "Klenkia"   
## [370] "Frondihabitans"   
## [371] "Saprospira"   
## [372] "Pontibacter"   
## [373] "Caenimonas"   
## [374] "Candidatus Planktophila"   
## [375] "Algoriphagus"   
## [376] "Desulfoprunum"   
## [377] "Ornithinibacter"   
## [378] "Candidatus Planktoluna"   
## [379] "Ferribacterium"   
## [380] "Lacibacter"   
## [381] "Pseudarthrobacter"   
## [382] "Geminocystis PCC-6308"   
## [383] "Desulfobacca"   
## [384] "Kineococcus"   
## [385] "Lactococcus"   
## [386] "Bergeyella"   
## [387] "Candidatus Tammella"   
## [388] "Klebsiella"   
## [389] "Lachnotalea"   
## [390] "Trichococcus"   
## [391] "Afipia"   
## [392] "CL500-3"   
## [393] "Gillisia"   
## [394] "Candidatus Finniella"   
## [395] "Paenibacillus"   
## [396] "Dyadobacter"   
## [397] "IMCC26207"   
## [398] "Neorhizobium"   
## [399] "Leptolinea"   
## [400] "Lawsonella"   
## [401] "Rhodopseudomonas"   
## [402] "Xylophilus"   
## [403] "Siphonobacter"   
## [404] "Citrobacter"   
## [405] "Lachnospiraceae UCG-010"   
## [406] "Incertae Sedis"   
## [407] "Chamaesiphon PCC-7430"   
## [408] "Methylibium"   
## [409] "Rhodomicrobium"   
## [410] "Rhodopila"   
## [411] "Magnetospirillum"   
## [412] "F0058"   
## [413] "Janthinobacterium"   
## [414] "Nodosilinea PCC-7104"   
## [415] "Nakamurella"   
## [416] "Candidatus Solibacter"   
## [417] "Koukoulia"   
## [418] "BSV13"   
## [419] "Rhodovastum"   
## [420] "Candidatus Competibacter"   
## [421] "Actibacter"   
## [422] "Sporichthya"   
## [423] "Psychrobacillus"   
## [424] "Acetobacteroides"   
## [425] "Anoxybacillus"   
## [426] "Fibrella"   
## [427] "Brevibacillus"   
## [428] "Brachybacterium"   
## [429] "OM60(NOR5) clade"   
## [430] "Actinoplanes"   
## [431] "Tumebacillus"   
## [432] "Candidatus Captivus"   
## [433] "MND1"   
## [434] "Verticiella"   
## [435] "Pseudoglutamicibacter"   
## [436] "[Cytophaga] xylanolytica group"   
## [437] "Flaviaesturariibacter"   
## [438] "Candidatus Accumulibacter"   
## [439] "Niveibacterium"   
## [440] "Erwinia"   
## [441] "Modestobacter"   
## [442] "Sanguibacter"   
## [443] "Patulibacter"   
## [444] "Luteolibacter"   
## [445] "Geothermomicrobium"   
## [446] "Cetobacterium"   
## [447] "Candidatus Alysiosphaera"   
## [448] "Aurantimicrobium"   
## [449] "Terriglobus"   
## [450] "Leptolyngbya PCC-6306"   
## [451] "Nordella"   
## [452] "Rhodoplanes"   
## [453] "Cnuella"   
## [454] "Scytonema PCC-7110"   
## [455] "966-1"   
## [456] "Psychrobacter"   
## [457] "Endomicrobium"   
## [458] "Salinisphaera"   
## [459] "Candidatus Megaira"   
## [460] "Globicatella"   
## [461] "Herminiimonas"   
## [462] "Aeromicrobium"   
## [463] "Leptotrichia"   
## [464] "Oscillospira"   
## [465] "Lachnospiraceae UCG-004"   
## [466] "Atopostipes"   
## [467] "Oscillatoria SAG 1459-8"   
## [468] "Angustibacter"   
## [469] "JGI 0001001-H03"   
## [470] "oc32"   
## [471] "Pseudolabrys"   
## [472] "Gemmatimonas"   
## [473] "Clostridium sensu stricto 2"   
## [474] "Papillibacter"   
## [475] "Glaciimonas"   
## [476] "Desulfobulbus"   
## [477] "Asticcacaulis"   
## [478] "Nannocystis"   
## [479] "Aureispira"   
## [480] "Clostridium sensu stricto 5"   
## [481] "Candidatus Endoecteinascidia"   
## [482] "Cesiribacter"   
## [483] "Deinococcus"   
## [484] "Leptospirillum"   
## [485] "Inhella"   
## [486] "Solibacillus"   
## [487] "Elusimicrobium"   
## [488] "Salinimicrobium"   
## [489] "Larkinella"   
## [490] "Lentimicrobium"   
## [491] "Citricoccus"   
## [492] "Rubritepida"   
## [493] "Citrifermentans"   
## [494] "Rubripirellula"   
## [495] "Hypnocyclicus"   
## [496] "Lacunisphaera"   
## [497] "WCHB1-32"   
## [498] "Methylocystis"   
## [499] "Ignavibacterium"   
## [500] "Segetibacter"   
## [501] "IS-44"   
## [502] "Comamonas"   
## [503] "PMMR1"   
## [504] "Saccharibacillus"   
## [505] "Herbiconiux"   
## [506] "Georgenia"   
## [507] "Quadrisphaera"   
## [508] "Negativibacillus"   
## [509] "Solimonas"   
## [510] "Portibacter"   
## [511] "Filimonas"   
## [512] "Leptolyngbya SAG 2411"   
## [513] "Acetoanaerobium"   
## [514] "Neo-b11"   
## [515] "Cystobacter"   
## [516] "Leptolyngbya FYG"   
## [517] "Delftia"   
## [518] "Subgroup 10"   
## [519] "Candidatus Trichorickettsia"   
## [520] "Catabacter"   
## [521] "Chthoniobacter"   
## [522] "Proteiniclasticum"   
## [523] "Longimicrobium"   
## [524] "Saccharopolyspora"   
## [525] "Luteimonas"   
## [526] "Lachnospiraceae UCG-003"   
## [527] "Intestinimonas"   
## [528] "BCf9-17 termite group"   
## [529] "Abditibacterium"   
## [530] "Zavarzinella"   
## [531] "Candidatus Chloroploca"   
## [532] "Chloronema"   
## [533] "Desulfitibacter"   
## [534] "Agaricicola"   
## [535] "Alkalibacterium"   
## [536] "Thermincola"   
## [537] "Falsirhodobacter"   
## [538] "Conexibacter"   
## [539] "Elizabethkingia"   
## [540] "Iamia"   
## [541] "Rickettsiella"   
## [542] "Calothrix PCC-6303"   
## [543] "Mastigocladopsis PCC-10914"   
## [544] "Thiothrix"   
## [545] "Dietzia"   
## [546] "Candidatus Stoquefichus"   
## [547] "Paeniglutamicibacter"   
## [548] "Candidatus Aquiluna"   
## [549] "RB41"   
## [550] "Clostridium sensu stricto 12"   
## [551] "Dongia"   
## [552] "Syntrophobacter"   
## [553] "Candidatus Obscuribacter"   
## [554] "Anaeromyxobacter"   
## [555] "Jeotgalibaca"   
## [556] "Ellin6055"   
## [557] "Pseudorhodoferax"   
## [558] "Solirubrobacter"   
## [559] "Rhodovulum"   
## [560] "Crenothrix"   
## [561] "Gemmata"   
## [562] "Clostridium sensu stricto 9"   
## [563] "Fimbriiglobus"   
## [564] "Chryseoglobus"   
## [565] "Puniceibacterium"   
## [566] "Christensenella"   
## [567] "Rhodopirellula"   
## [568] "Tianweitania"   
## [569] "Vitreoscilla"   
## [570] "MM1"   
## [571] "Defluviitaleaceae UCG-011"   
## [572] "Oscillibacter"   
## [573] "Sulfuritalea"   
## [574] "Aliihoeflea"   
## [575] "Psychroglaciecola"   
## [576] "Ralstonia"   
## [577] "Ancylobacter"   
## [578] "Cohnella"   
## [579] "Rummeliibacillus"   
## [580] "Aquicella"   
## [581] "Leptolyngbya ANT.L52.2"   
## [582] "Proteiniborus"   
## [583] "Taibaiella"   
## [584] "Amnibacterium"   
## [585] "Silanimonas"   
## [586] "Chryseolinea"   
## [587] "Luteibacter"   
## [588] "Salinispira"   
## [589] "Kineosporia"   
## [590] "Agromyces"   
## [591] "Chelativorans"   
## [592] "Youngiibacter"   
## [593] "Peptococcus"   
## [594] "Silicimonas"   
## [595] "Atopobium"   
## [596] "Giesbergeria"   
## [597] "Candidatus Berkiella"   
## [598] "Domibacillus"   
## [599] "Candidatus Koribacter"   
## [600] "Thermoactinomyces"   
## [601] "Laceyella"   
## [602] "Pelagibacterium"   
## [603] "Chiayiivirga"   
## [604] "Brevibacterium"   
## [605] "Pseudonocardia"   
## [606] "MWH-Ta3"   
## [607] "Malikia"   
## [608] "Pelosinus"   
## [609] "Nibribacter"   
## [610] "Wenxinia"   
## [611] "Candidatus Paenicardinium"   
## [612] "Budvicia"   
## [613] "Candidatus Hamiltonella"   
## [614] "Phormidesmis ANT.LACV5.1"   
## [615] "Sarcina"   
## [616] "Candidatus Regiella"   
## [617] "DEV114"   
## [618] "Piscinibacter"   
## [619] "Thermoflavimicrobium"   
## [620] "Pseudofulvimonas"   
## [621] "Pragia"   
## [622] "Megamonas"   
## [623] "Wohlfahrtiimonas"   
## [624] "Oryzihumus"   
## [625] "Pir4 lineage"   
## [626] "Truepera"   
## [627] "AKYG587"   
## [628] "Scytonema UTEX 2349"   
## [629] "Aneurinibacillus"   
## [630] "IMCC26134"   
## [631] "SWB02"   
## [632] "Oceanobacillus"   
## [633] "Ensifer"   
## [634] "Microcoleus PCC-7113"   
## [635] "Longispora"   
## [636] "Sphingosinicella"   
## [637] "DSSD61"   
## [638] "Paraclostridium"   
## [639] "Tundrisphaera"   
## [640] "Pleomorphomonas"   
## [641] "Rubrivirga"   
## [642] "Litorilinea"   
## [643] "Natranaerovirga"   
## [644] "Kitasatospora"   
## [645] "Symbiobacterium"   
## [646] "Tepidisphaera"   
## [647] "Escherichia-Shigella"   
## [648] "Longivirga"   
## [649] "[Ruminococcus] gnavus group"   
## [650] "Lapillicoccus"   
## [651] "Inquilinus"   
## [652] "Clostridium sensu stricto 8"   
## [653] "Parviterribacter"   
## [654] "Alloactinosynnema"   
## [655] "Cellulosilyticum"   
## [656] "Marvinbryantia"   
## [657] "Geminicoccus"   
## [658] "Candidatus Paracaedibacter"   
## [659] "Tuwongella"   
## [660] "Enteractinococcus"   
## [661] "Neorickettsia"   
## [662] "Sporacetigenium"   
## [663] "Haematospirillum"   
## [664] "Dyella"   
## [665] "Candidatus Contendobacter"   
## [666] "Intrasporangium"   
## [667] "Salinarimonas"   
## [668] "Opitutus"   
## [669] "Centipeda"   
## [670] "Bryocella"   
## [671] "Shimazuella"   
## [672] "Huanghella"   
## [673] "Alkalibacter"   
## [674] "Lewinella"   
## [675] "Caproiciproducens"   
## [676] "Arsenicitalea"   
## [677] "Variibacter"   
## [678] "Flavihumibacter"   
## [679] "Lysinimonas"   
## [680] "Croceicoccus"   
## [681] "Roseimarinus"   
## [682] "Acetobacterium"   
## [683] "Novibacillus"   
## [684] "Shuttleworthia"   
## [685] "Candidatus Omnitrophus"   
## [686] "GWD2-49-16"   
## [687] "Flexibacter"   
## [688] "Potamolinea 1PC"   
## [689] "Gulbenkiania"   
## [690] "Saccharothrix"   
## [691] "[Clostridium] innocuum group"   
## [692] "Xylanimonas"   
## [693] "Neomicrococcus"   
## [694] "Geotalea"   
## [695] "Actinotalea"   
## [696] "Peredibacter"   
## [697] "Stella"   
## [698] "Prosthecomicrobium"   
## [699] "MN 122.2a"   
## [700] "Cavicella"   
## [701] "Myroides"   
## [702] "Collinsella"   
## [703] "Angelakisella"   
## [704] "Lachnospiraceae FCS020 group"   
## [705] "Schlesneria"   
## [706] "Perlucidibaca"   
## [707] "Marinoscillum"   
## [708] "Streptobacillus"   
## [709] "Jatrophihabitans"   
## [710] "Xanthomonas"   
## [711] "Niastella"   
## [712] "Phycicoccus"   
## [713] "Methylovirgula"   
## [714] "Sodalis"   
## [715] "Bauldia"   
## [716] "Rhodovarius"   
## [717] "Roseobacter clade CHAB-I-5 lineage"   
## [718] "C39"   
## [719] "Paludicola"   
## [720] "Dokdonella"   
## [721] "Odoribacter"   
## [722] "Butyricimonas"   
## [723] "FFCH5858"   
## [724] "Sulfurifustis"   
## [725] "Hazenella"   
## [726] "Silvanigrella"   
## [727] "Dermabacter"   
## [728] "E1B-B3-114"   
## [729] "Oligella"   
## [730] "Chthonobacter"   
## [731] "Candidatus Cardinium"   
## [732] "Chitinivorax"   
## [733] "Fictibacillus"   
## [734] "Sulfurisoma"   
## [735] "Porphyromonas"   
## [736] "Mobilitalea"   
## [737] "Candidatus Ovatusbacter"   
## [738] "possible genus 04"   
## [739] "Ureibacillus"   
## [740] "Epulopiscium"   
## [741] "Actinomycetospora"   
## [742] "Leptolyngbya EcFYyyy-00"   
## [743] "Diplorickettsia"   
## [744] "Hansschlegelia"   
## [745] "Anaerolinea"   
## [746] "Gordonia"   
## [747] "Leptolyngbya VRUC 135"   
## [748] "OM27 clade"   
## [749] "BD1-7 clade"   
## [750] "Flavitalea"   
## [751] "Micromonospora"   
## [752] "UTBCD1"   
## [753] "Ignatzschineria"   
## [754] "Weissella"   
## [755] "Roseisolibacter"   
## [756] "Prevotella"   
## [757] "Anaerosporobacter"   
## [758] "Thermobacillus"   
## [759] "Olivibacter"   
## [760] "Clostridium sensu stricto 14"   
## [761] "Bdellovibrio"   
## [762] "Syntrophus"   
## [763] "Pasteuria"   
## [764] "Catenibacterium"   
## [765] "Jeotgalicoccus"   
## [766] "s3t2d-1089"   
## [767] "Sideroxydans"   
## [768] "Brevifollis"   
## [769] "Frankia"   
## [770] "Chroococcopsis"   
## [771] "Labrys"   
## [772] "KD3-10"   
## [773] "Kroppenstedtia"   
## [774] "Lentimonas"   
## [775] "GCA-900066575"   
## [776] "Oxobacter"   
## [777] "[Agitococcus] lubricus group"   
## [778] "Collimonas"   
## [779] "Tahibacter"   
## [780] "Nostoc PCC-73102"   
## [781] "YC-ZSS-LKJ147"   
## [782] "Bartonella"   
## [783] "Blastopirellula"   
## [784] "Hyphomonas"   
## [785] "Denitratisoma"   
## [786] "Syntrophorhabdus"   
## [787] "Syntrophomonas"   
## [788] "Azospirillum"   
## [789] "Tolypothrix"   
## [790] "Halotalea"   
## [791] "Halioglobus"   
## [792] "Bacteriovorax"   
## [793] "Sporobacter"   
## [794] "Belnapia"   
## [795] "CENA359"   
## [796] "Endobacter"   
## [797] "Lutibacter"   
## [798] "Desulfovirga"   
## [799] "Ohtaekwangia"   
## [800] "Planifilum"   
## [801] "Pseudoflavonifractor"   
## [802] "Cupriavidus"   
## [803] "Clostridium sensu stricto 10"   
## [804] "Sulfuricurvum"   
## [805] "MSBL3"   
## [806] "Crenobacter"   
## [807] "Candidatus Anammoximicrobium"   
## [808] "Acidothermus"   
## [809] "Lacibacterium"   
## [810] "Alloprevotella"   
## [811] "Clostridium sensu stricto 7"   
## [812] "Luteitalea"   
## [813] "Akkermansia"   
## [814] "SM1A02"   
## [815] "Polycladomyces"   
## [816] "Candidatus Udaeobacter"   
## [817] "Veillonella"   
## [818] "Rivicola"   
## [819] "Catenibacillus"   
## [820] "Rubribacterium"   
## [821] "Euzebya"   
## [822] "Dialister"   
## [823] "Coxiella"   
## [824] "Phormidesmis ANT.L52.6"   
## [825] "Kaistia"   
## [826] "Prosthecobacter"   
## [827] "Sporocytophaga"   
## [828] "Synechocystis PCC-6803"   
## [829] "Bordetella"   
## [830] "Aurantimonas"   
## [831] "DTU089"   
## [832] "Methylobacillus"   
## [833] "Candidatus Hemipteriphilus"   
## [834] "Microcystis PCC-7914"   
## [835] "Herbinix"   
## [836] "Candidatus Saccharimonas"   
## [837] "Methyloglobulus"   
## [838] "Isosphaera"   
## [839] "Chitinimonas"   
## [840] "Thauera"   
## [841] "Craurococcus-Caldovatus"   
## [842] "Palleronia-Pseudomaribius"   
## [843] "Conyzicola"   
## [844] "Clostridium sensu stricto 15"   
## [845] "Clostridium"   
## [846] "Alkanibacter"   
## [847] "CENA518"   
## [848] "JCM 18997"   
## [849] "Phocea"   
## [850] "Caldicoprobacter"   
## [851] "Sellimonas"   
## [852] "Chroococcidiopsis PCC-6712"   
## [853] "Tsukamurella"   
## [854] "Persicitalea"   
## [855] "Providencia"   
## [856] "LB3-76"   
## [857] "Amycolatopsis"   
## [858] "Phormidium SAG 37.90"   
## [859] "Demequina"   
## [860] "Termite planctomycete cluster"   
## [861] "[Aquaspirillum] arcticum group"   
## [862] "Proteus"   
## [863] "Desulfocapsa"   
## [864] "Crinalium SAG 22.89"   
## [865] "Ammoniphilus"   
## [866] "Gloeobacter PCC-7421"   
## [867] "Uliginosibacterium"   
## [868] "Simplicispira"   
## [869] "Anaerosinus"   
## [870] "Mycetocola"   
## [871] "Geodermatophilus"   
## [872] "Aequorivita"   
## [873] "Coriobacteriaceae UCG-002"   
## [874] "Telmatocola"   
## [875] "Anaerocolumna"   
## [876] "Desulforhopalus"   
## [877] "Smithella"   
## [878] "Friedmanniella"   
## [879] "Sporomusa"   
## [880] "Candidatus Endonucleariobacter"   
## [881] "Herbaspirillum"   
## [882] "Candidatus Methylomirabilis"   
## [883] "Rhabdobacter"   
## [884] "Cyanothece PCC-7424"   
## [885] "Polycyclovorans"   
## [886] "Sphingoaurantiacus"   
## [887] "Acidaminobacter"   
## [888] "Enhygromyxa"   
## [889] "Kytococcus"   
## [890] "Salinicoccus"   
## [891] "Lachnospiraceae NK4B4 group"   
## [892] "Nonomuraea"   
## [893] "Turicella"   
## [894] "Phormidium CYN64"   
## [895] "Azoarcus"   
## [896] "Kozakia"   
## [897] "Leptolyngbya Es-Yyy1000"   
## [898] "Pseudactinotalea"   
## [899] "UTCFX1"   
## [900] "Geitlerinema PCC-7105"   
## [901] "Anaerospora"   
## [902] "Paludibacterium"   
## [903] "Dechlorosoma"   
## [904] "Fervidobacterium"   
## [905] "Shewanella"   
## [906] "Arachidicoccus"   
## [907] "Rhabdanaerobium"   
## [908] "Faecalitalea"   
## [909] "Schizothrix LEGE 07164"   
## [910] "Desulfomonile"   
## [911] "Synechocystis CCALA 700"   
## [912] "Sumerlaea"   
## [913] "Rikenellaceae RC9 gut group"   
## [914] "Treponema"   
## [915] "Vibrio"   
## [916] "Oxalicibacterium"   
## [917] "Irregularibacter"   
## [918] "Kribbella"   
## [919] "Elstera"   
## [920] "JdFR-76"   
## [921] "Indibacter"   
## [922] "GOUTA6"   
## [923] "Oceanicella"   
## [924] "Yokenella"   
## [925] "Parasutterella"   
## [926] "Marinilactibacillus"   
## [927] "Flavonifractor"   
## [928] "Corallococcus"   
## [929] "Smaragdicoccus"   
## [930] "Clostridium sensu stricto 3"   
## [931] "Formivibrio"   
## [932] "Alterococcus"   
## [933] "Commensalibacter"   
## [934] "Thermus"   
## [935] "Seohaeicola"   
## [936] "Nocardiopsis"   
## [937] "Mobilicoccus"   
## [938] "Raoultibacter"   
## [939] "KCM-B-112"   
## [940] "Faecalibaculum"   
## [941] "Lentisphaera"   
## [942] "Rubritalea"   
## [943] "Pelolinea"   
## [944] "Desulfatiglans"   
## [945] "Caldalkalibacillus"   
## [946] "Negativicoccus"   
## [947] "Merismopedia AICB1015"   
## [948] "Risungbinella"   
## [949] "Actinomadura"   
## [950] "Pseudoxanthobacter"   
## [951] "Oscillochloris"   
## [952] "Antricoccus"   
## [953] "MM2"   
## [954] "RBG-16-58-14"   
## [955] "Cecembia"   
## [956] "Capnocytophaga"   
## [957] "Fusobacterium"   
## [958] "Thermobispora"   
## [959] "Pediococcus"   
## [960] "CAG-352"   
## [961] "Enterobacter"   
## [962] "Phytoactinopolyspora"   
## [963] "OLB17"   
## [964] "Candidatus Electronema"   
## [965] "Albibacter"   
## [966] "Lechevalieria"   
## [967] "Pigmentiphaga"   
## [968] "Eubacterium"   
## [969] "Thermoflexibacter"   
## [970] "Chitinibacter"   
## [971] "Diplosphaera"   
## [972] "Rubrimonas"   
## [973] "Gleocapsa"   
## [974] "Actinomyces"   
## [975] "Spirochaeta"   
## [976] "Enhydrobacter"   
## [977] "Sulfuricella"   
## [978] "Leucobacter"   
## [979] "SH3-11"   
## [980] "Fusibacter"   
## [981] "Caryophanon"   
## [982] "Planctomicrobium"   
## [983] "Rubidimonas"   
## [984] "Effusibacillus"   
## [985] "FFCH7168"   
## [986] "Terrabacter"   
## [987] "Proteiniphilum"   
## [988] "BIyi10"   
## [989] "Catellicoccus"   
## [990] "Azohydromonas"   
## [991] "Asanoa"   
## [992] "Yaniella"   
## [993] "CHKCI001"   
## [994] "Adlercreutzia"   
## [995] "Ornithinicoccus"   
## [996] "Roseovarius"   
## [997] "Cephaloticoccus"   
## [998] "Snowella 0TU37S04"   
## [999] "Planctopirus"   
## [1000] "Dehalobacterium"   
## [1001] "CK06-06-Mud-MAS4B-21"   
## [1002] "Raineya"   
## [1003] "Aerosphaera"   
## [1004] "Aquipuribacter"   
## [1005] "Tepidimonas"   
## [1006] "Alicyclobacillus"   
## [1007] "Methylovulum"   
## [1008] "Salmonella"   
## [1009] "OLB13"   
## [1010] "Blvii28 wastewater-sludge group"   
## [1011] "Coprobacter"   
## [1012] "Actimicrobium"   
## [1013] "Chungangia"   
## [1014] "Saccharomonospora"   
## [1015] "Rhodobaculum"   
## [1016] "Hydrocarboniphaga"   
## [1017] "Cryptosporangium"   
## [1018] "Dethiobacter"   
## [1019] "Candidatus Lumbricincola"   
## [1020] "Candidatus Nostocoida"   
## [1021] "Helicobacter"   
## [1022] "Subgroup 23"   
## [1023] "Spongiimonas"   
## [1024] "Desulfococcus"   
## [1025] "Agrococcus"   
## [1026] "Williamwhitmania"   
## [1027] "Leptolyngbya ANT.L67.1"   
## [1028] "Tolumonas"   
## [1029] "Leeia"   
## [1030] "Gallionella"   
## [1031] "Ferritrophicum"   
## [1032] "Geothrix"   
## [1033] "V9D2013 group"   
## [1034] "Sedimentibacter"   
## [1035] "Austwickia"   
## [1036] "Desulforegula"   
## [1037] "Leptospira"   
## [1038] "Longilinea"   
## [1039] "Desulfopila"   
## [1040] "Desulforhabdus"   
## [1041] "Sh765B-TzT-35"   
## [1042] "Niveispirillum"   
## [1043] "GWE2-31-10"   
## [1044] "Amphiplicatus"   
## [1045] "Acetonema"   
## [1046] "Candidatus Xiphinematobacter"   
## [1047] "Bifidobacterium"   
## [1048] "Morganella"

#Phyla per insect (sample) taxa  
  
  
#Total phyla per site  
sample.df <- psmelt(sample\_ps) #make phyloseq object a dataframe

## Warning in psmelt(sample\_ps): The sample variables:   
## Family  
## have been renamed to:   
## sample\_Family  
## to avoid conflicts with taxonomic rank names.

sample.df.agg <- aggregate(Abundance ~ Phylum + Site + sample\_Family + Taxa, data = sample.df, FUN = mean) #aggregate to Phylum level, take the mean  
sample.df.agg$Site <- factor(sample.df.agg$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats"))  
  
(with(sample.df.agg[sample.df.agg$Phylum == "Proteobacteria", ], table(Abundance, Site)))

## Site  
## Abundance Cochrane Sunalta Cushing Bridge Graves Bridge  
## 0.387094068706388 0 1 0 0  
## 0.472145540986121 0 0 0 0  
## 0.518736580783682 1 0 0 0  
## 0.602908615136876 0 0 0 1  
## 0.642797235641439 0 0 0 1  
## 0.751936201211564 0 0 1 0  
## 0.819226382179281 0 0 1 0  
## 0.958702361782072 1 0 0 0  
## 1.10540794417606 0 0 0 0  
## 1.25135869565217 1 0 0 0  
## 1.26516371443908 0 0 1 0  
## 1.31226516371444 0 1 0 0  
## 1.35882984433709 0 0 0 1  
## 1.38626604206314 1 0 0 0  
## 1.41255703166935 0 0 0 0  
## 1.46625067096082 0 0 0 0  
## 1.61291754407696 0 0 0 0  
## 1.67931169742764 0 0 1 0  
## 1.74688003220612 0 0 0 0  
## 1.98030730005368 0 0 0 0  
## 1.990371712292 1 0 0 0  
## 2.04153247450349 0 1 0 0  
## 2.11825684380032 0 0 0 1  
## 2.12199558656886 0 0 0 0  
## 2.36334205582394 0 0 0 0  
## 2.44014190821256 1 0 0 0  
## 2.63665235283593 0 0 0 0  
## 2.67741098586509 0 1 0 0  
## 2.89704777241009 0 0 1 0  
## 3.38998758722491 0 0 0 0  
## 7.40098295759528 0 0 1 0  
## 8.03532608695652 0 0 1 0  
## 8.87830448201825 0 0 0 0  
## 9.05704844337091 1 0 0 0  
## 10.7011540526033 0 1 0 0  
## 11.9870169082126 0 0 0 1  
## Site  
## Abundance Policeman Flats  
## 0.387094068706388 0  
## 0.472145540986121 1  
## 0.518736580783682 0  
## 0.602908615136876 0  
## 0.642797235641439 0  
## 0.751936201211564 0  
## 0.819226382179281 0  
## 0.958702361782072 0  
## 1.10540794417606 1  
## 1.25135869565217 0  
## 1.26516371443908 0  
## 1.31226516371444 0  
## 1.35882984433709 0  
## 1.38626604206314 0  
## 1.41255703166935 0  
## 1.46625067096082 1  
## 1.61291754407696 1  
## 1.67931169742764 0  
## 1.74688003220612 0  
## 1.98030730005368 0  
## 1.990371712292 0  
## 2.04153247450349 0  
## 2.11825684380032 0  
## 2.12199558656886 1  
## 2.36334205582394 1  
## 2.44014190821256 0  
## 2.63665235283593 0  
## 2.67741098586509 0  
## 2.89704777241009 0  
## 3.38998758722491 0  
## 7.40098295759528 0  
## 8.03532608695652 0  
## 8.87830448201825 1  
## 9.05704844337091 0  
## 10.7011540526033 0  
## 11.9870169082126 0

(with(sample.df.agg[sample.df.agg$Phylum == "Bacteroidota", ], table(Abundance, Site)))

## Site  
## Abundance Cochrane Sunalta Cushing Bridge Graves Bridge  
## 0.0134351215890106 0 1 0 0  
## 0.0586597364024503 0 0 1 0  
## 0.126331066371905 1 0 0 0  
## 0.142048317377814 0 0 1 0  
## 0.150315574531279 0 0 0 0  
## 0.203568776684611 1 0 0 0  
## 0.262808613328383 0 0 0 1  
## 0.31698997586783 0 0 1 0  
## 0.328940040839057 0 0 0 0  
## 0.367133840727678 0 0 1 0  
## 0.467607202524596 1 0 0 0  
## 0.476845968214077 0 0 0 0  
## 0.5129478373863 0 0 0 0  
## 0.535339706701318 0 1 0 0  
## 0.635534620382402 0 0 0 1  
## 0.639618526081307 1 0 0 0  
## 0.653285687766846 0 0 0 1  
## 0.705564321514758 0 1 0 0  
## 0.739233339521069 0 1 0 0  
## 0.888759977724151 0 0 0 0  
## 0.99102004826434 1 0 0 0  
## 0.994443413155127 0 1 0 0  
## 1.14296525824278 0 0 1 0  
## 1.23412845739744 0 0 0 0  
## 1.44714126601077 1 0 0 0  
## 1.56693273931069 0 0 0 1  
## 2.30488908483386 0 0 0 0  
## 2.31267867087433 0 0 1 0  
## 2.47120846482272 0 0 0 0  
## 2.90667347317616 0 0 0 0  
## 2.95275663634676 0 0 0 1  
## 2.97986923251449 0 0 0 0  
## 2.98231854464451 0 0 0 0  
## 3.55029303349333 0 0 1 0  
## 3.55093280118805 1 0 0 0  
## 3.57766382030815 0 0 0 0  
## Site  
## Abundance Policeman Flats  
## 0.0134351215890106 0  
## 0.0586597364024503 0  
## 0.126331066371905 0  
## 0.142048317377814 0  
## 0.150315574531279 1  
## 0.203568776684611 0  
## 0.262808613328383 0  
## 0.31698997586783 0  
## 0.328940040839057 1  
## 0.367133840727678 0  
## 0.467607202524596 0  
## 0.476845968214077 1  
## 0.5129478373863 1  
## 0.535339706701318 0  
## 0.635534620382402 0  
## 0.639618526081307 0  
## 0.653285687766846 0  
## 0.705564321514758 0  
## 0.739233339521069 0  
## 0.888759977724151 1  
## 0.99102004826434 0  
## 0.994443413155127 0  
## 1.14296525824278 0  
## 1.23412845739744 1  
## 1.44714126601077 0  
## 1.56693273931069 0  
## 2.30488908483386 0  
## 2.31267867087433 0  
## 2.47120846482272 0  
## 2.90667347317616 0  
## 2.95275663634676 0  
## 2.97986923251449 1  
## 2.98231854464451 0  
## 3.55029303349333 0  
## 3.55093280118805 0  
## 3.57766382030815 0

(with(sample.df.agg[sample.df.agg$Phylum == "Firmicutes", ], table(Abundance, Site)))

## Site  
## Abundance Cochrane Sunalta Cushing Bridge Graves Bridge  
## 0.0255508673230192 1 0 0 0  
## 0.149742147210502 0 0 1 0  
## 0.183368495077356 0 0 0 0  
## 0.340365682137834 0 0 0 0  
## 0.506797937177684 0 1 0 0  
## 0.584095171120488 0 0 1 0  
## 0.671940928270042 0 1 0 0  
## 0.697784810126582 0 0 0 1  
## 0.836497890295359 1 0 0 0  
## 0.908403656821378 0 0 0 1  
## 0.924485968789766 0 0 1 0  
## 0.98763478668542 0 0 0 0  
## 1.03000468823254 1 0 0 0  
## 1.17469075696924 0 0 0 0  
## 1.22925457102672 1 0 0 0  
## 1.24027191748711 0 0 0 0  
## 1.33212778782399 0 0 0 0  
## 1.33951327622214 1 0 0 0  
## 1.35402853124372 0 0 1 0  
## 1.62734802312861 0 0 0 1  
## 1.68055555555556 0 0 0 0  
## 1.70287024014169 0 0 0 0  
## 1.74882794186592 0 0 1 0  
## 1.78117674636662 0 0 0 1  
## 2.02232770745429 0 0 1 0  
## 2.08761134552274 0 0 0 0  
## 2.52523831848726 0 1 0 0  
## 2.68483356774496 0 0 0 0  
## 2.69227613689639 0 0 0 0  
## 3.0785819899744 0 0 1 0  
## 3.75357477730895 0 1 0 0  
## 3.8301101734646 0 0 0 0  
## 3.94924988279419 0 0 0 1  
## 4.89006094702297 0 1 0 0  
## 6.73013361462729 1 0 0 0  
## 9.31715893108298 1 0 0 0  
## Site  
## Abundance Policeman Flats  
## 0.0255508673230192 0  
## 0.149742147210502 0  
## 0.183368495077356 1  
## 0.340365682137834 0  
## 0.506797937177684 0  
## 0.584095171120488 0  
## 0.671940928270042 0  
## 0.697784810126582 0  
## 0.836497890295359 0  
## 0.908403656821378 0  
## 0.924485968789766 0  
## 0.98763478668542 0  
## 1.03000468823254 0  
## 1.17469075696924 1  
## 1.22925457102672 0  
## 1.24027191748711 0  
## 1.33212778782399 1  
## 1.33951327622214 0  
## 1.35402853124372 0  
## 1.62734802312861 0  
## 1.68055555555556 0  
## 1.70287024014169 1  
## 1.74882794186592 0  
## 1.78117674636662 0  
## 2.02232770745429 0  
## 2.08761134552274 1  
## 2.52523831848726 0  
## 2.68483356774496 1  
## 2.69227613689639 0  
## 3.0785819899744 0  
## 3.75357477730895 0  
## 3.8301101734646 1  
## 3.94924988279419 0  
## 4.89006094702297 0  
## 6.73013361462729 0  
## 9.31715893108298 0

##Rarefying the data

#How to rarefy to an even sequencing depth (usually the minimum read so that no sequences are removed)  
(ps\_rare <- phyloseq::rarefy\_even\_depth(sample\_ps, rngseed = 123, replace = FALSE))

## `set.seed(123)` was used to initialize repeatable random subsampling.

## Please record this for your records so others can reproduce.

## Try `set.seed(123); .Random.seed` for the full vector

## ...

## 5365OTUs were removed because they are no longer   
## present in any sample after random subsampling

## ...

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 15648 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 15648 taxa by 7 taxonomic ranks ]:  
## taxa are rows

#Rarefying to the minimum sample depth removes a lot of the ASVs because there was such high variation in number of reads. ASVs were removed because they are no longer present in any sample after random subsampling  
  
head(phyloseq::sample\_sums(ps\_rare)) #2259 bacterial taxa per sample. Cuts a lot of sequencing depth but some people argue it is better to retain for samples rather than a larger sequencing depth.

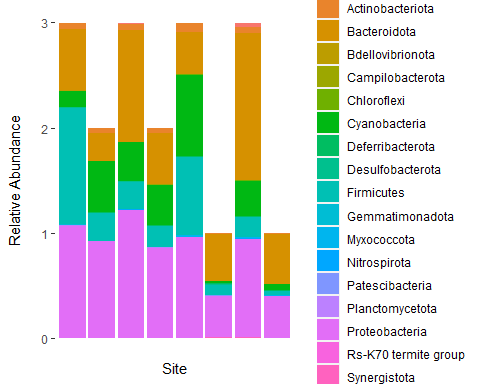
## KK1809 KK1810 KK1811 KK1812 KK1813 KK1814   
## 2180 2180 2180 2180 2180 2180

##Relative Abundance

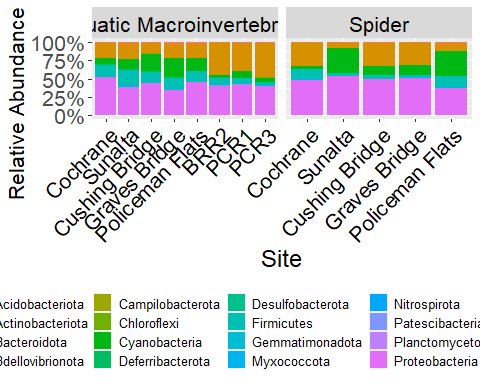
#Relative Abundance - insect/spiders, phylum level  
##Glomerate to Phylum  
  
#Without extra filtering beforehand  
ps\_phylum <- phyloseq::tax\_glom(ps, "Phylum")  
ps\_rel\_abund = phyloseq::transform\_sample\_counts(ps\_phylum, function(x){x / sum(x)})  
ps\_rel\_abund\_prune = prune\_taxa(taxa\_sums(ps\_rel\_abund) > 0.1, ps\_rel\_abund) #10% abundance of total  
ps\_melt<-psmelt(ps\_rel\_abund\_prune)

## Warning in psmelt(ps\_rel\_abund\_prune): The sample variables:   
## Family  
## have been renamed to:   
## sample\_Family  
## to avoid conflicts with taxonomic rank names.

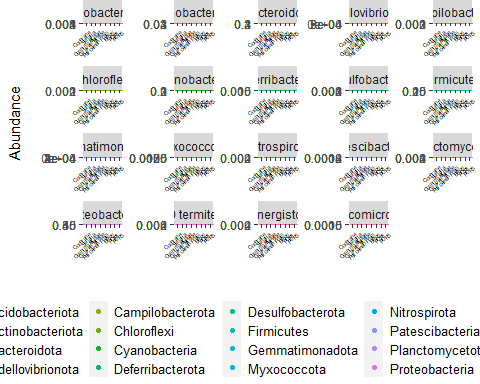
ps\_agg<-aggregate(Abundance~Phylum+Site+Taxa, data=ps\_melt, FUN= mean)  
  
ggplot(ps\_agg, aes(x=Site, y=Abundance, fill = "Phylum")) +  
 geom\_bar(aes(fill = Phylum), stat = "identity", position="stack")+  
 labs(x = "Site", y = "Relative Abundance\n") +  
 theme(panel.background = element\_blank(),  
 axis.text.x=element\_blank(),  
 axis.ticks.x=element\_blank())



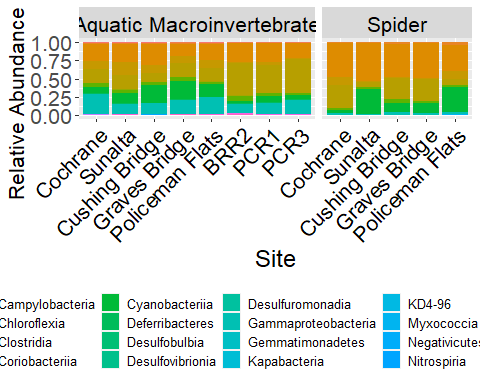
#Shows all samples at the phylum level.  
  
#With the pre-processing steps  
rel\_abun\_phylum <- aggregate\_taxa(sample\_ps, "Phylum")  
comp\_rel\_abun\_phylum <- transform\_sample\_counts(rel\_abun\_phylum, function(x) x/sum(x))  
#comp\_rel\_abun\_phylum <- microbiome::transform(rel\_abun\_phylum, "compositional") #This is the exact same as the code above. "compositional" means transform to relative abundance.  
rel\_abun\_all\_prune = prune\_taxa(taxa\_sums(comp\_rel\_abun\_phylum) > 0.05, comp\_rel\_abun\_phylum)   
  
rel\_abun\_prune\_sample <- psmelt(rel\_abun\_all\_prune)  
  
sample.df.agg <- aggregate(Abundance ~ Phylum + Site + Taxa, data = rel\_abun\_prune\_sample, FUN = mean) #aggregate to Phylum level, take the mean  
sample.df.agg$Site <- factor(sample.df.agg$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
sample.df.agg$Taxa <- factor(sample.df.agg$Taxa, levels=c("Insect", "Spider"), labels=c("Aquatic Macroinvertebrate", "Spider"))  
  
##Relative Abundance Plot - Phylum Level  
rel\_abun\_plot\_phylum <- ggplot(sample.df.agg, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >5%") +  
 facet\_wrap(.~Taxa, scale="free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))+  
 scale\_y\_continuous(label = scales::percent)  
rel\_abun\_plot\_phylum



#Plotting abundance of phyla composing at least 5% of the total abundance across Sites  
ggplot(data =sample.df.agg, aes(x = Site, y = Abundance)) +  
 geom\_boxplot(outlier.shape = NA) +  
 geom\_jitter(aes(color = Phylum), height = 0, width = .2) +  
 labs(x = "", y = "Abundance\n") +  
 facet\_wrap(~ Phylum, scales = "free")+  
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10), axis.text.x=element\_text(angle=45, hjust=1, size=4, color="black"))



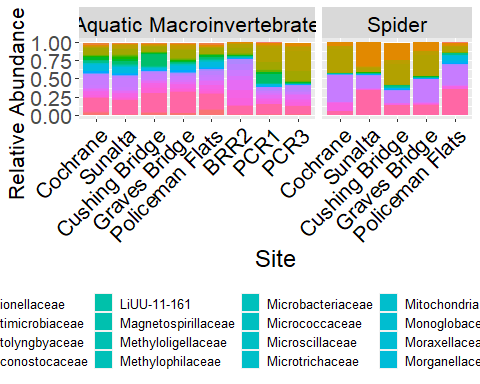
#Looks like some differences in certain phyla across sites  
  
#Relative abundance - Class Level  
##Glomerate to Class  
rel\_abun\_class <- aggregate\_taxa(sample\_ps, "Class")  
comp\_rel\_abun\_class <- microbiome::transform(rel\_abun\_class, "compositional")  
rel\_abun\_all\_prune\_class = prune\_taxa(taxa\_sums(comp\_rel\_abun\_class) > 0.05, comp\_rel\_abun\_class)   
  
rel\_abun\_prune\_sample\_class <- psmelt(rel\_abun\_all\_prune\_class)  
  
sample.df.agg.class <- aggregate(Abundance ~ Class + Site + Taxa, data = rel\_abun\_prune\_sample\_class, FUN = mean) #aggregate to Phylum level, take the mean  
sample.df.agg.class$Site <- factor(sample.df.agg.class$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
sample.df.agg.class$Taxa <- factor(sample.df.agg.class$Taxa, levels=c("Insect", "Spider"), labels=c("Aquatic Macroinvertebrate", "Spider"))  
  
rel\_abun\_plot\_class <- ggplot(sample.df.agg.class, aes(x=Site, y=Abundance, fill=Class)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >5%") +  
 facet\_grid(.~Taxa, scale="free\_x", space = "free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
rel\_abun\_plot\_class



#Relative abundance - Family Level  
rel\_abun\_fam <- aggregate\_taxa(sample\_ps, "Family")  
comp\_rel\_abun\_fam <- microbiome::transform(rel\_abun\_fam, "compositional")  
rel\_abun\_all\_prune\_fam = prune\_taxa(taxa\_sums(comp\_rel\_abun\_fam) > 0.05, comp\_rel\_abun\_fam)   
  
rel\_abun\_prune\_sample\_fam <- psmelt(rel\_abun\_all\_prune\_fam)

## Warning in psmelt(rel\_abun\_all\_prune\_fam): The sample variables:   
## Family  
## have been renamed to:   
## sample\_Family  
## to avoid conflicts with taxonomic rank names.

sample.df.agg\_fam <- aggregate(Abundance ~ Family + Site + Taxa, data = rel\_abun\_prune\_sample\_fam, FUN = mean) #aggregate to Phylum level, take the mean  
sample.df.agg\_fam$Site <- factor(sample.df.agg\_fam$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
sample.df.agg\_fam$Taxa <- factor(sample.df.agg\_fam$Taxa, levels=c("Insect", "Spider"), labels=c("Aquatic Macroinvertebrate", "Spider"))  
  
  
rel\_abun\_plot\_fam <- ggplot(sample.df.agg\_fam, aes(x=Site, y=Abundance, fill=Family)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >5%") +  
 facet\_grid(.~Taxa, scale="free\_x", space = "free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
rel\_abun\_plot\_fam



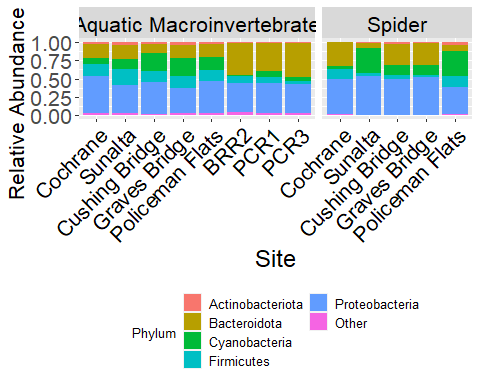
#These graphs don"t tell us much about the data but it looks like not much is changing in terms of composition of different taxonomic ranks across sites.  
  
#Top 5 phyla  
top\_phyla <- sample.df.agg %>%  
 group\_by(Site, Phylum) %>%  
 summarize(Mean = mean(Abundance)) %>%  
 arrange(-Mean)

## `summarise()` has grouped output by 'Site'. You can override using the  
## `.groups` argument.

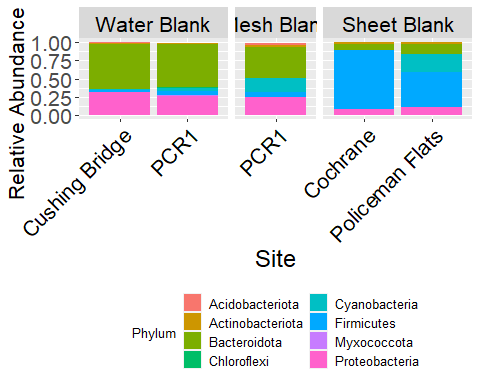
top\_phyla

## # A tibble: 152 x 3  
## # Groups: Site [8]  
## Site Phylum Mean  
## <fct> <chr> <dbl>  
## 1 Cochrane Proteobacteria 0.500  
## 2 PCR3 Bacteroidota 0.471  
## 3 Cushing Bridge Proteobacteria 0.464  
## 4 Sunalta Proteobacteria 0.460  
## 5 BRR2 Bacteroidota 0.445  
## 6 Graves Bridge Proteobacteria 0.427  
## 7 PCR1 Proteobacteria 0.412  
## 8 Policeman Flats Proteobacteria 0.409  
## 9 PCR3 Proteobacteria 0.395  
## 10 BRR2 Proteobacteria 0.388  
## # ... with 142 more rows

top5 <- top\_phyla$Phylum[1:35]  
df0 <- sample.df.agg %>%  
 mutate(Phylum = fct\_other(Phylum, top5))  
  
top5phylumplot <- ggplot(df0, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >5%") +  
 facet\_grid(.~Taxa, scale="free\_x", space = "free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
top5phylumplot



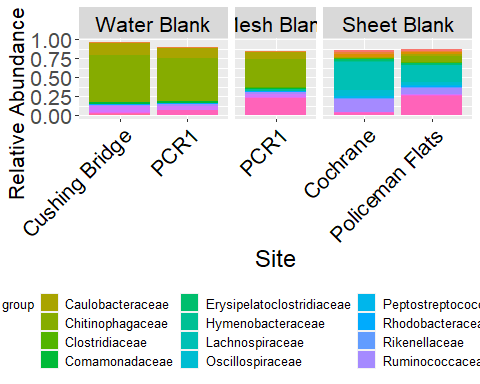
#Blank samples  
#Relative Abundance - phylum level  
bl\_rel\_abun\_phylum <- aggregate\_taxa(blank\_ps, "Phylum")  
bl\_comp\_rel\_abun\_phylum <- microbiome::transform(bl\_rel\_abun\_phylum, "compositional")  
bl\_rel\_abun\_all\_prune\_phylum = prune\_taxa(taxa\_sums(bl\_comp\_rel\_abun\_phylum) > 0.02, bl\_comp\_rel\_abun\_phylum)   
  
bl\_rel\_abun\_prune\_sample\_phylum <- psmelt(bl\_rel\_abun\_all\_prune\_phylum)  
  
bl.sample.df.agg.phylum <- aggregate(Abundance ~ Phylum + Site + Taxa, data = bl\_rel\_abun\_prune\_sample\_phylum, FUN = mean) #aggregate to Phylum level, take the mean  
bl.sample.df.agg.phylum$Site <- factor(bl.sample.df.agg.phylum$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
bl.sample.df.agg.phylum$Taxa <- factor(bl.sample.df.agg.phylum$Taxa, levels=c("Water Blank", "Mesh Blank", "Sheet Blank", "Negative Blank"))  
  
  
bl\_rel\_abun\_plot\_phylum <- ggplot(bl.sample.df.agg.phylum, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >2%") +  
 facet\_grid(.~Taxa, scale="free\_x", space = "free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
bl\_rel\_abun\_plot\_phylum



#Family Level  
bl\_rel\_abun\_fam <- aggregate\_taxa(blank\_ps, "Family")  
bl\_comp\_rel\_abun\_fam <- microbiome::transform(bl\_rel\_abun\_fam, "compositional")  
bl\_rel\_abun\_all\_prune\_fam = prune\_taxa(taxa\_sums(bl\_comp\_rel\_abun\_fam) > 0.05, bl\_comp\_rel\_abun\_fam)   
  
bl\_rel\_abun\_prune\_sample\_fam <- psmelt(bl\_rel\_abun\_all\_prune\_fam)

## Warning in psmelt(bl\_rel\_abun\_all\_prune\_fam): The sample variables:   
## Family  
## have been renamed to:   
## sample\_Family  
## to avoid conflicts with taxonomic rank names.

bl.sample.df.agg.fam <- aggregate(Abundance ~ Family + Site + Taxa, data = bl\_rel\_abun\_prune\_sample\_fam, FUN = mean) #aggregate to Phylum level, take the mean  
bl.sample.df.agg.fam$Site <- factor(bl.sample.df.agg.fam$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
bl.sample.df.agg.fam$Taxa <- factor(bl.sample.df.agg.fam$Taxa, levels=c("Water Blank", "Mesh Blank", "Sheet Blank", "Negative Blank"))  
  
  
bl\_rel\_abun\_plot\_fam <- ggplot(bl.sample.df.agg.fam, aes(x=Site, y=Abundance, fill=Family)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >5%") +  
 facet\_grid(.~Taxa, scale="free\_x", space = "free\_x") +   
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
bl\_rel\_abun\_plot\_fam



#Hydropsychids  
hydro\_ps<-subset\_samples(sample\_ps, Family == "Hydropsychidae")  
hydro\_ps<-prune\_taxa(taxa\_sums(hydro\_ps) > 0, hydro\_ps)  
hydro\_ps #92 samples, 8496 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 8496 taxa and 92 samples ]:  
## sample\_data() Sample Data: [ 92 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 8496 taxa by 7 taxonomic ranks ]:  
## taxa are rows

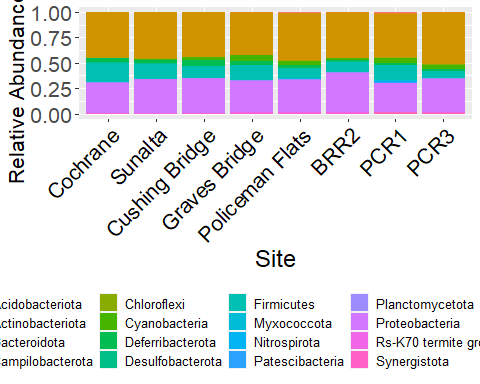
hydro\_ps1<-subset\_samples(hydro\_ps, Site %in% c("Cochrane", "Cushing Bridge", "Policeman Flats"))  
hydro\_ps1<-prune\_taxa(taxa\_sums(hydro\_ps1) > 0, hydro\_ps1)  
hydro\_ps1 #38 samples, 4758 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 4758 taxa and 38 samples ]:  
## sample\_data() Sample Data: [ 38 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 4758 taxa by 7 taxonomic ranks ]:  
## taxa are rows

hydro\_lar<-subset\_samples(hydro\_ps, Stage=="Larvae")  
hydro\_lar<-prune\_taxa(taxa\_sums(hydro\_lar)>0, hydro\_lar)  
hydro\_lar #60 samples, 4124 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 4124 taxa and 60 samples ]:  
## sample\_data() Sample Data: [ 60 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 4124 taxa by 7 taxonomic ranks ]:  
## taxa are rows

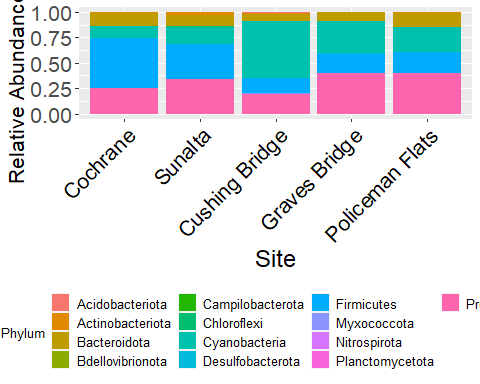
hydro\_rel\_abund\_phylum <- aggregate\_taxa(hydro\_lar, "Phylum")  
comp\_rel\_abund\_phylum\_hydro <- microbiome::transform(hydro\_rel\_abund\_phylum, "compositional")  
comp\_rel\_abund\_phylum\_hydro\_prune = prune\_taxa(taxa\_sums(comp\_rel\_abund\_phylum\_hydro) > 0.02, comp\_rel\_abund\_phylum\_hydro)   
  
rel\_abund\_hydro\_df <- psmelt(comp\_rel\_abund\_phylum\_hydro\_prune)  
  
rel\_abund\_hydro\_agg <- aggregate(Abundance ~ Phylum + Site + Taxa, data = rel\_abund\_hydro\_df, FUN = mean) #aggregate to Phylum level, take the mean  
rel\_abund\_hydro\_agg$Site <- factor(rel\_abund\_hydro\_agg$Site, levels=c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3"))  
  
  
rel\_abund\_hydro\_phylum\_plot <- ggplot(rel\_abund\_hydro\_agg, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >2%") +  
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
rel\_abund\_hydro\_phylum\_plot



#Not much change across any sites  
  
#Heptageniidae  
hep\_ps<-subset\_samples(sample\_ps, Family=="Heptageniidae ")  
hep\_lar<-subset\_samples(hep\_ps, Stage=="Larvae")  
hep\_lar<-prune\_taxa(taxa\_sums(hep\_lar)>0, hep\_lar)  
hep\_lar #40 samples, 6960 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 6960 taxa and 40 samples ]:  
## sample\_data() Sample Data: [ 40 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 6960 taxa by 7 taxonomic ranks ]:  
## taxa are rows

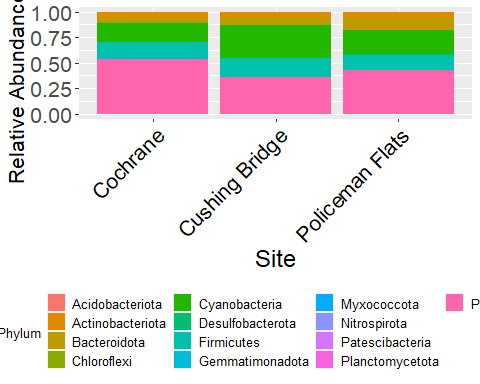
hep\_agg\_taxa<-aggregate\_taxa(hep\_lar, "Phylum")  
comp\_hep\_lar<-microbiome::transform(hep\_agg\_taxa, "compositional")  
hep\_prune<-prune\_taxa(taxa\_sums(comp\_hep\_lar)>0.02, comp\_hep\_lar)  
hep\_df<-psmelt(hep\_prune)  
  
hep\_agg<-aggregate(Abundance~Phylum+Site+Taxa, data=hep\_df, FUN=mean)  
  
rel\_abund\_hep\_lar\_plot <- ggplot(hep\_agg, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >2%") +  
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
rel\_abund\_hep\_lar\_plot



#A bit of change at Cushing Bridge  
  
#Chironomidae  
chiro\_ps<-subset\_samples(sample\_ps, Family == "Chironomidae")  
chiro\_ps<-subset\_samples(chiro\_ps, Site %in% c("Cochrane", "Cushing Bridge", "Policeman Flats"))   
chiro\_ps<-prune\_taxa(taxa\_sums(chiro\_ps) > 0, chiro\_ps)  
chiro\_ps #38 samples, 3759 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 3759 taxa and 38 samples ]:  
## sample\_data() Sample Data: [ 38 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 3759 taxa by 7 taxonomic ranks ]:  
## taxa are rows

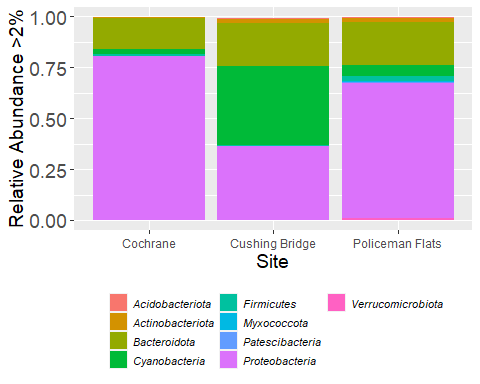
chiro\_rel\_abund\_phylum <- aggregate\_taxa(chiro\_ps, "Phylum")  
comp\_rel\_abund\_phylum\_chiro <- microbiome::transform(chiro\_rel\_abund\_phylum, "compositional")  
comp\_rel\_abund\_phylum\_chiro\_prune = prune\_taxa(taxa\_sums(comp\_rel\_abund\_phylum\_chiro) > 0.02, comp\_rel\_abund\_phylum\_chiro)   
  
rel\_abund\_chiro\_df <- psmelt(comp\_rel\_abund\_phylum\_chiro\_prune)  
  
rel\_abund\_chiro\_agg <- aggregate(Abundance ~ Phylum + Site + Taxa, data = rel\_abund\_chiro\_df, FUN = mean) #aggregate to Phylum level, take the mean  
  
rel\_abund\_chiro\_phylum\_plot <- ggplot(rel\_abund\_chiro\_agg, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >2%") +  
 theme(strip.text = element\_text(size = 16, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=16),  
 axis.text.y=element\_text(size=16),  
 axis.text.x=element\_text(angle=45, hjust=1, size=16, color="black"),  
 axis.title=element\_text(size=18))  
rel\_abund\_chiro\_phylum\_plot



#Not much change across sites here  
  
#Perlidae  
perl\_ps<-subset\_samples(sample\_ps, Family == "Perlidae")  
perl\_ps<-subset\_samples(perl\_ps, Site %in% c("Cochrane", "Cushing Bridge", "Policeman Flats"))   
perl\_ps<-prune\_taxa(taxa\_sums(perl\_ps) > 0, perl\_ps)  
perl\_ps #21 samples, 3114 taxa

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 3114 taxa and 21 samples ]:  
## sample\_data() Sample Data: [ 21 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 3114 taxa by 7 taxonomic ranks ]:  
## taxa are rows

perl\_rel\_abund\_phylum <- aggregate\_taxa(perl\_ps, "Phylum")  
comp\_rel\_abund\_phylum\_perl <- microbiome::transform(perl\_rel\_abund\_phylum, "compositional")  
comp\_rel\_abund\_phylum\_perl\_prune = prune\_taxa(taxa\_sums(comp\_rel\_abund\_phylum\_perl) > 0.02, comp\_rel\_abund\_phylum\_perl)   
  
rel\_abund\_perl\_df <- psmelt(comp\_rel\_abund\_phylum\_perl\_prune)  
  
rel\_abund\_perl\_agg <- aggregate(Abundance ~ Phylum + Site + Taxa, data = rel\_abund\_perl\_df, FUN = mean) #aggregate to Phylum level, take the mean  
  
rel\_abund\_perl\_phylum\_plot <- ggplot(rel\_abund\_perl\_agg, aes(x=Site, y=Abundance, fill=Phylum)) + geom\_bar(stat="identity") +   
 ylab("Relative Abundance >2%") +  
 theme(strip.text = element\_text(size = 14, color="black")) +   
 theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=8, face="italic"), legend.title=element\_text(size=8)) +  
 guides (fill=guide\_legend(nrow=4)) +  
 theme(axis.title.y=element\_text(size=14),axis.text.y=element\_text(size=14), axis.title=element\_text(size=14))+  
 scale\_fill\_discrete(name=NULL)  
rel\_abund\_perl\_phylum\_plot

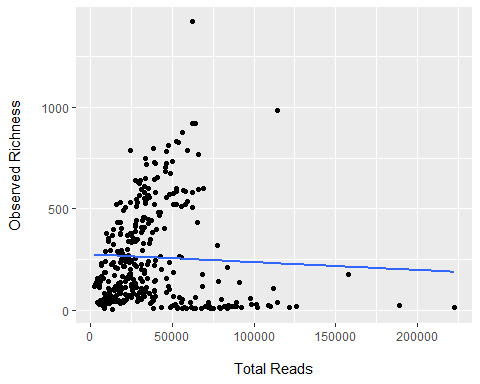


#Some change at Cushing Bridge

##Alpha Diversity

#Alpha Diversity   
#Check to see if total reads is correlated with observed richness. Should be since you likely have more taxa of bacteria with a higher total abundance.   
ggplot(data = data.frame("total\_reads" = phyloseq::sample\_sums(sample\_ps),  
 "observed" = phyloseq::estimate\_richness(sample\_ps, measures = "Observed")[, 1]),  
 aes(x = total\_reads, y = observed)) +  
 geom\_point() +  
 geom\_smooth(method="lm", se = FALSE) +  
 labs(x = "\nTotal Reads", y = "Observed Richness\n")

## `geom\_smooth()` using formula = 'y ~ x'



#Looks like there is not a positive correlation btw increasing reads/total abundance and observed richness. This might be because it only takes a certain amount of reads to capture all the unique bacteria so past this point there is no increase. Could also be that either contamination or sequencing errors are causing there to be really low amount of richness even though there are a large amouont of reads?  
  
#Rarifying  
(ps\_rare <- phyloseq::rarefy\_even\_depth(sample\_ps, rngseed = 123, replace = FALSE))

## `set.seed(123)` was used to initialize repeatable random subsampling.  
## Please record this for your records so others can reproduce.  
## Try `set.seed(123); .Random.seed` for the full vector  
## ...  
## 5365OTUs were removed because they are no longer   
## present in any sample after random subsampling  
##   
## ...

## phyloseq-class experiment-level object  
## otu\_table() OTU Table: [ 15648 taxa and 340 samples ]:  
## sample\_data() Sample Data: [ 340 samples by 8 sample variables ]:  
## tax\_table() Taxonomy Table: [ 15648 taxa by 7 taxonomic ranks ]:  
## taxa are rows

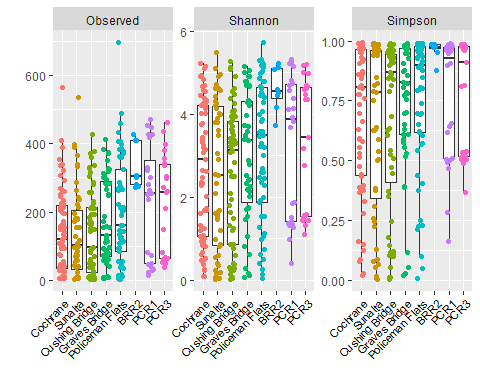
head(phyloseq::sample\_sums(ps\_rare)) #minimum sequencing depth, 2180.

## KK1809 KK1810 KK1811 KK1812 KK1813 KK1814   
## 2180 2180 2180 2180 2180 2180

#Generate a data frame with diversity measures  
adiv\_rare <- data.frame(  
 "Observed" = phyloseq::estimate\_richness(ps\_rare, measures = "Observed"),  
 "Shannon" = phyloseq::estimate\_richness(ps\_rare, measures = "Shannon"),  
 "Simpson" = phyloseq::estimate\_richness(ps\_rare, measures = "Simpson"),  
 "Site" = phyloseq::sample\_data(ps\_rare)$Site)  
 #"Family" = phyloseq::sample\_data(ps\_rare)$Family))  
head(adiv\_rare)

## Observed Shannon Simpson Site  
## KK1809 247 4.322890 0.9670057 Cochrane  
## KK1810 196 4.133454 0.9634972 Cochrane  
## KK1811 198 4.211592 0.9700694 Cochrane  
## KK1812 267 4.691059 0.9833331 Cochrane  
## KK1813 197 3.234588 0.8595055 Cochrane  
## KK1814 204 4.236695 0.9689370 Cochrane

#Plot adiv measures  
adiv\_rare %>%  
 gather(key = metric, value = value, c("Observed", "Shannon", "Simpson")) %>%  
 mutate(metric = factor(metric, levels = c("Observed", "Shannon", "Simpson"))) %>%  
 ggplot(aes(x = Site, y = value)) +  
 geom\_boxplot(outlier.color = NA) +  
 geom\_jitter(aes(color = Site), height = 0, width = .2) +  
 labs(x = "", y = "") +  
 facet\_wrap(~ metric, scales = "free") +  
 theme(legend.position="none", axis.text.x=element\_text(angle=45, hjust=1, size=8, color="black"))



#Summarize  
adiv\_rare %>%  
 group\_by(Site) %>%  
 dplyr::summarise(median\_observed = median(Observed),  
 median\_shannon = median(Shannon),  
 median\_pd = median(Simpson))

## # A tibble: 8 x 4  
## Site median\_observed median\_shannon median\_pd  
## <fct> <dbl> <dbl> <dbl>  
## 1 Cochrane 119 2.90 0.807  
## 2 Sunalta 102 2.52 0.785  
## 3 Cushing Bridge 97 3.13 0.868  
## 4 Graves Bridge 133 3.36 0.826  
## 5 Policeman Flats 160 3.49 0.899  
## 6 BRR2 304. 4.56 0.974  
## 7 PCR1 254. 3.87 0.928  
## 8 PCR3 257 3.45 0.911

#Statistical test. Use Kruskal test bc the data is not normal and we have more than 2 sites (non-parametric ANOVA)  
#Observed richness across sites  
kruskal.test(Observed ~ Site, data = adiv\_rare) #p<0.001. Significant differences across sites

##   
## Kruskal-Wallis rank sum test  
##   
## data: Observed by Site  
## Kruskal-Wallis chi-squared = 27.607, df = 7, p-value = 0.0002591

dunnTest(Observed~Site, data=adiv\_rare)

## Dunn (1964) Kruskal-Wallis multiple comparison  
## p-values adjusted with the Holm method.

## Comparison Z P.unadj P.adj  
## 1 BRR2 - Cochrane 3.60978995 0.0003064451 0.007967572  
## 2 BRR2 - Cushing Bridge 3.78087141 0.0001562804 0.004375851  
## 3 Cochrane - Cushing Bridge 0.39190257 0.6951302005 1.000000000  
## 4 BRR2 - Graves Bridge 2.94670980 0.0032117446 0.080293616  
## 5 Cochrane - Graves Bridge -1.17359329 0.2405579716 1.000000000  
## 6 Cushing Bridge - Graves Bridge -1.51536107 0.1296809892 1.000000000  
## 7 BRR2 - PCR1 2.06894962 0.0385508146 0.578262219  
## 8 Cochrane - PCR1 -2.12557406 0.0335387413 0.536619861  
## 9 Cushing Bridge - PCR1 -2.39471822 0.0166331426 0.349295995  
## 10 Graves Bridge - PCR1 -1.12544460 0.2604006800 1.000000000  
## 11 BRR2 - PCR3 1.96123209 0.0498519556 0.648075423  
## 12 Cochrane - PCR3 -2.25635728 0.0240482646 0.432868764  
## 13 Cushing Bridge - PCR3 -2.52058207 0.0117160914 0.269470102  
## 14 Graves Bridge - PCR3 -1.26534847 0.2057464374 1.000000000  
## 15 PCR1 - PCR3 -0.13583390 0.8919525884 1.000000000  
## 16 BRR2 - Policeman Flats 2.62143684 0.0087559984 0.210143962  
## 17 Cochrane - Policeman Flats -2.06010117 0.0393888697 0.551444176  
## 18 Cushing Bridge - Policeman Flats -2.41250722 0.0158432236 0.348550919  
## 19 Graves Bridge - Policeman Flats -0.72792458 0.4666597587 1.000000000  
## 20 PCR1 - Policeman Flats 0.58548252 0.5582233150 1.000000000  
## 21 PCR3 - Policeman Flats 0.73854247 0.4601848686 1.000000000  
## 22 BRR2 - Sunalta 3.69804970 0.0002172624 0.005866084  
## 23 Cochrane - Sunalta 0.33668778 0.7363522750 1.000000000  
## 24 Cushing Bridge - Sunalta -0.02446719 0.9804799583 0.980479958  
## 25 Graves Bridge - Sunalta 1.39302145 0.1636133033 1.000000000  
## 26 PCR1 - Sunalta 2.27082369 0.0231576533 0.439995413  
## 27 PCR3 - Sunalta 2.39459224 0.0166388579 0.332777158  
## 28 Policeman Flats - Sunalta 2.20848400 0.0272105521 0.462579386

#Significant differences in bacterial richness between:  
#BRR2 - Cochrane p=0.008  
#BRR2 - Cushing Bridge p=0.0044  
#BRR2 - Sunalta p=0.0059  
  
kruskal.test(Shannon ~ Site, data = adiv\_rare) #p=0.0031

##   
## Kruskal-Wallis rank sum test  
##   
## data: Shannon by Site  
## Kruskal-Wallis chi-squared = 21.492, df = 7, p-value = 0.003107

dunnTest(Shannon~Site, data=adiv\_rare)

## Dunn (1964) Kruskal-Wallis multiple comparison  
## p-values adjusted with the Holm method.

## Comparison Z P.unadj P.adj  
## 1 BRR2 - Cochrane 3.251446962 0.0011481920 0.02985299  
## 2 BRR2 - Cushing Bridge 3.512281098 0.0004442778 0.01243978  
## 3 Cochrane - Cushing Bridge 0.581274577 0.5610554123 1.00000000  
## 4 BRR2 - Graves Bridge 2.684578793 0.0072621263 0.18155316  
## 5 Cochrane - Graves Bridge -0.996007182 0.3192466554 1.00000000  
## 6 Cushing Bridge - Graves Bridge -1.512480560 0.1304116544 1.00000000  
## 7 BRR2 - PCR1 2.056945732 0.0396914462 0.83352037  
## 8 Cochrane - PCR1 -1.582705847 0.1134885249 1.00000000  
## 9 Cushing Bridge - PCR1 -1.994885566 0.0460553622 0.87505188  
## 10 Graves Bridge - PCR1 -0.745372464 0.4560465851 1.00000000  
## 11 BRR2 - PCR3 2.043611909 0.0409919018 0.81983804  
## 12 Cochrane - PCR3 -1.561779051 0.1183400485 1.00000000  
## 13 Cushing Bridge - PCR3 -1.967987023 0.0490695308 0.88325156  
## 14 Graves Bridge - PCR3 -0.738493498 0.4602146168 1.00000000  
## 15 PCR1 - PCR3 -0.003158097 0.9974802071 0.99748021  
## 16 BRR2 - Policeman Flats 2.353904409 0.0185773877 0.42727992  
## 17 Cochrane - Policeman Flats -1.871112667 0.0613294662 1.00000000  
## 18 Cushing Bridge - Policeman Flats -2.411328424 0.0158945284 0.38146868  
## 19 Graves Bridge - Policeman Flats -0.729694143 0.4655771621 1.00000000  
## 20 PCR1 - Policeman Flats 0.188324644 0.8506221679 1.00000000  
## 21 PCR3 - Policeman Flats 0.189219721 0.8499206116 1.00000000  
## 22 BRR2 - Sunalta 3.385242043 0.0007111552 0.01920119  
## 23 Cochrane - Sunalta 0.412407897 0.6800404756 1.00000000  
## 24 Cushing Bridge - Sunalta -0.122145025 0.9027841640 1.00000000  
## 25 Graves Bridge - Sunalta 1.299064434 0.1939218173 1.00000000  
## 26 PCR1 - Sunalta 1.813497605 0.0697551077 1.00000000  
## 27 PCR3 - Sunalta 1.791571616 0.0732016132 1.00000000  
## 28 Policeman Flats - Sunalta 2.110392253 0.0348245832 0.76614083

#Significant differences in bacterial alpha diversity between:  
#BRR2 - Cochrane p=0.03  
#Cochrane - Cushing Bridge p=0.012  
#BRR2 - Sunalta p=0.019  
  
kruskal.test(Simpson ~ Site, data = adiv\_rare) #p=0.014

##   
## Kruskal-Wallis rank sum test  
##   
## data: Simpson by Site  
## Kruskal-Wallis chi-squared = 17.612, df = 7, p-value = 0.01385

dunnTest(Simpson~Site, data=adiv\_rare)

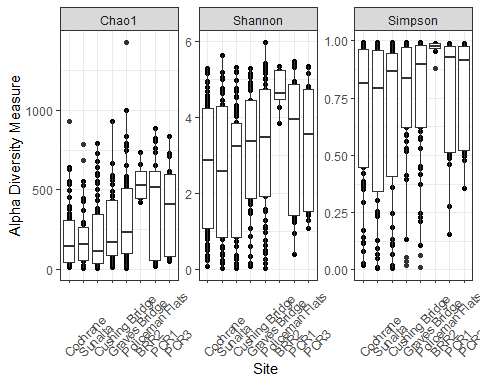
## Dunn (1964) Kruskal-Wallis multiple comparison  
## p-values adjusted with the Holm method.

## Comparison Z P.unadj P.adj  
## 1 BRR2 - Cochrane 3.12129572 0.0018005712 0.04681485  
## 2 BRR2 - Cushing Bridge 3.32162716 0.0008949419 0.02505837  
## 3 Cochrane - Cushing Bridge 0.45093964 0.6520330507 1.00000000  
## 4 BRR2 - Graves Bridge 2.60980082 0.0090594954 0.22648739  
## 5 Cochrane - Graves Bridge -0.89044312 0.3732279965 1.00000000  
## 6 Cushing Bridge - Graves Bridge -1.28959976 0.1971896594 1.00000000  
## 7 BRR2 - PCR1 2.13378268 0.0328605747 0.72293264  
## 8 Cochrane - PCR1 -1.24619136 0.2126941499 1.00000000  
## 9 Cushing Bridge - PCR1 -1.56579920 0.1173956348 1.00000000  
## 10 Graves Bridge - PCR1 -0.50634567 0.6126140061 1.00000000  
## 11 BRR2 - PCR3 2.13507671 0.0327547447 0.75335913  
## 12 Cochrane - PCR3 -1.20497660 0.2282123358 1.00000000  
## 13 Cushing Bridge - PCR3 -1.52015657 0.1284716298 1.00000000  
## 14 Graves Bridge - PCR3 -0.47857101 0.6322438483 1.00000000  
## 15 PCR1 - PCR3 0.01800115 0.9856379321 1.00000000  
## 16 BRR2 - Policeman Flats 2.32801296 0.0199114156 0.47787397  
## 17 Cochrane - Policeman Flats -1.65102396 0.0987336820 1.00000000  
## 18 Cushing Bridge - Policeman Flats -2.06716032 0.0387190498 0.81310005  
## 19 Graves Bridge - Policeman Flats -0.63250154 0.5270592039 1.00000000  
## 20 PCR1 - Policeman Flats 0.01773655 0.9858490244 1.00000000  
## 21 PCR3 - Policeman Flats -0.00400073 0.9968078877 0.99680789  
## 22 BRR2 - Sunalta 3.28216588 0.0010301299 0.02781351  
## 23 Cochrane - Sunalta 0.46108928 0.6447345518 1.00000000  
## 24 Cushing Bridge - Sunalta 0.04458329 0.9644394615 1.00000000  
## 25 Graves Bridge - Sunalta 1.24659818 0.2125448690 1.00000000  
## 26 PCR1 - Sunalta 1.53133167 0.1256874419 1.00000000  
## 27 PCR3 - Sunalta 1.48911770 0.1364563780 1.00000000  
## 28 Policeman Flats - Sunalta 1.95743640 0.0502961828 1.00000000

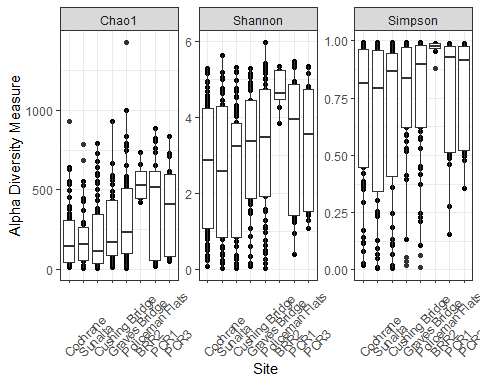
#Significant differences in bacterial alpha diversity between:  
#BRR2 - Sunalta p=0.028  
#BRR2 - Cushing Bridge p=0.025  
#BRR2 - Cochrane p=0.047  
  
  
#From Joey711 GitHub: You do not want to transform or filter your data before estimating richness, other than quality assurance filtering that would remove non-target sequences. Usually that sort of filtering would be done on the sequence data before it is in the form of a contingency table (table of counts). Basically, the very first table of counts that you have in your workflow is probably the one that you want to use for estimate\_richness and plot\_richness. There are several different alpha diversity estimators supported in the estimate\_richness function, and they all incorporate library size into their estimates in different ways... Except for the "Observed" option, which is simply showing you graphically the OTUs that were observed at least once in each sample. I think this is the one you were actually asking about.To be clear, the other methods require that you use raw counts. Do not use rarefied counts for the Chao-I estimate, or the Shannon index, for example. The "Observed" option is not a method at all, just showing you what is in your data as you provided. If you transform your data, especially if you rarefy, and then you want to estimate richness, the "Observed" result is now the only one still available to you at all. Rarefying reduces the precision with which you would estimate the diversity in the first place, and so this generally shouldn't be done. I suspect, however, that is the workflow you had in mind. All alpha-diversity indices/estimates are aware of differences in sample size (library size, number of reads in this case), because this has always been a problem when attempting to count things, even trees in a forest (see Sanders original paper describing rarefaction, a different technique than rarefying, if rarefying can be called a technique).  
  
#Now trying without rarefaction.   
##alpha diversity metrics  
  
adiv <- data.frame(  
 "Observed" = phyloseq::estimate\_richness(sample\_ps, measures = "Observed"),  
 "Shannon" = phyloseq::estimate\_richness(sample\_ps, measures = "Shannon"),  
 "Simpson" = phyloseq::estimate\_richness(sample\_ps, measures = "Simpson"),  
 "Site" = phyloseq::sample\_data(sample\_ps)$Site)  
 #"Family" = phyloseq::sample\_data(ps\_rare)$Family))  
head(adiv)

## Observed Shannon Simpson Site  
## KK1809 332 4.347870 0.9664416 Cochrane  
## KK1810 251 4.175786 0.9641535 Cochrane  
## KK1811 249 4.294790 0.9714985 Cochrane  
## KK1812 345 4.790204 0.9843738 Cochrane  
## KK1813 264 3.260907 0.8543057 Cochrane  
## KK1814 263 4.301367 0.9701458 Cochrane

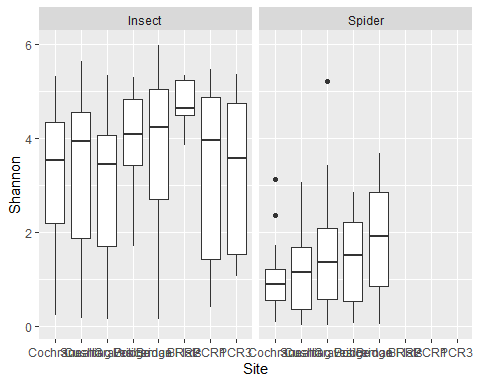
newSiteorder = c("Cochrane", "Sunalta", "Cushing Bridge", "Graves Bridge", "Policeman Flats", "BRR2", "PCR1", "PCR3")  
  
p <- plot\_richness(sample\_ps, x="Site", measures=c("Shannon", "Simpson", "Chao1"))  
p = p + theme\_bw() + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 45, hjust=0, vjust=0))  
p



p$sample\_ps$Site <- as.character(p$sample\_ps$Site)  
p$sample\_ps$Site <- factor(p$sample\_ps$Site, levels= newSiteorder)  
print(p)



er <- estimate\_richness(sample\_ps, measures=c("Shannon", "Simpson"))  
er <- data.frame(er)  
er<- rownames\_to\_column(er, var="sample\_id")  
metadata<-rownames\_to\_column(metadata, var="sample\_id")  
  
div.df<-left\_join(er,metadata, by = "sample\_id")  
  
div.df$Stream\_Type<-factor(div.df$Stream\_Type, levels = "Bow River", "ACWA Streams")  
  
ggplot(div.df, aes(x=Site, y=Shannon))+  
 facet\_grid(~Taxa)+  
 geom\_boxplot()



div.df$Site <-as.factor(div.df$Site)  
  
kruskal.test(Simpson~Site, data = div.df)

##   
## Kruskal-Wallis rank sum test  
##   
## data: Simpson by Site  
## Kruskal-Wallis chi-squared = 17.469, df = 7, p-value = 0.01461

dunnTest(Simpson~Site, data = div.df)

## Dunn (1964) Kruskal-Wallis multiple comparison  
## p-values adjusted with the Holm method.

## Comparison Z P.unadj P.adj  
## 1 BRR2 - Cochrane 3.08034755 0.002067592 0.05375738  
## 2 BRR2 - Cushing Bridge 3.26424275 0.001097571 0.03073198  
## 3 Cochrane - Cushing Bridge 0.41549239 0.677781487 1.00000000  
## 4 BRR2 - Graves Bridge 2.56148165 0.010422675 0.26056686  
## 5 Cochrane - Graves Bridge -0.90706645 0.364371660 1.00000000  
## 6 Cushing Bridge - Graves Bridge -1.27373445 0.202757542 1.00000000  
## 7 BRR2 - PCR1 2.10159423 0.035588836 0.78295439  
## 8 Cochrane - PCR1 -1.23704246 0.216071318 1.00000000  
## 9 Cushing Bridge - PCR1 -1.53079234 0.125820722 1.00000000  
## 10 Graves Bridge - PCR1 -0.48506370 0.627631164 1.00000000  
## 11 BRR2 - PCR3 2.11204213 0.034682836 0.79770523  
## 12 Cochrane - PCR3 -1.18071808 0.237714738 1.00000000  
## 13 Cushing Bridge - PCR3 -1.47055774 0.141410760 1.00000000  
## 14 Graves Bridge - PCR3 -0.44310350 0.657690877 1.00000000  
## 15 PCR1 - PCR3 0.03063354 0.975561790 1.00000000  
## 16 BRR2 - Policeman Flats 2.26541885 0.023486989 0.56368773  
## 17 Cochrane - Policeman Flats -1.69746947 0.089607938 1.00000000  
## 18 Cushing Bridge - Policeman Flats -2.07828724 0.037682911 0.79134113  
## 19 Graves Bridge - Policeman Flats -0.65855486 0.510181657 1.00000000  
## 20 PCR1 - Policeman Flats -0.02540796 0.979729558 0.97972956  
## 21 PCR3 - Policeman Flats -0.06155198 0.950919623 1.00000000  
## 22 BRR2 - Sunalta 3.25988105 0.001114589 0.03009392  
## 23 Cochrane - Sunalta 0.49679802 0.619331493 1.00000000  
## 24 Cushing Bridge - Sunalta 0.11210888 0.910737075 1.00000000  
## 25 Graves Bridge - Sunalta 1.29486713 0.195366068 1.00000000  
## 26 PCR1 - Sunalta 1.54973770 0.121204484 1.00000000  
## 27 PCR3 - Sunalta 1.49277910 0.135495020 1.00000000  
## 28 Policeman Flats - Sunalta 2.03479191 0.041871815 0.83743631

kruskal.test(Shannon~Site, data = div.df)

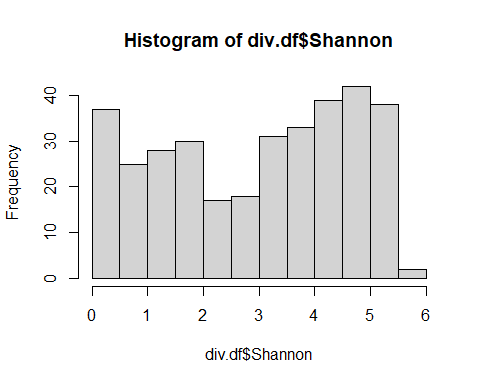
##   
## Kruskal-Wallis rank sum test  
##   
## data: Shannon by Site  
## Kruskal-Wallis chi-squared = 22.242, df = 7, p-value = 0.002308

dunnTest(Shannon~Site, data = div.df)

## Dunn (1964) Kruskal-Wallis multiple comparison  
## p-values adjusted with the Holm method.

## Comparison Z P.unadj P.adj  
## 1 BRR2 - Cochrane 3.32694990 0.0008780216 0.02282856  
## 2 BRR2 - Cushing Bridge 3.55358497 0.0003800184 0.01064052  
## 3 Cochrane - Cushing Bridge 0.50867706 0.6109786065 1.00000000  
## 4 BRR2 - Graves Bridge 2.73551554 0.0062282646 0.15570661  
## 5 Cochrane - Graves Bridge -1.04205668 0.2973854009 1.00000000  
## 6 Cushing Bridge - Graves Bridge -1.49183784 0.1357416622 1.00000000  
## 7 BRR2 - PCR1 2.09744088 0.0359545656 0.75504588  
## 8 Cochrane - PCR1 -1.63193436 0.1026933157 1.00000000  
## 9 Cushing Bridge - PCR1 -1.99059386 0.0465255558 0.88398556  
## 10 Graves Bridge - PCR1 -0.75712788 0.4489732521 1.00000000  
## 11 BRR2 - PCR3 2.07701879 0.0377998252 0.75599650  
## 12 Cochrane - PCR3 -1.62190632 0.1048234091 1.00000000  
## 13 Cushing Bridge - PCR3 -1.97533607 0.0482300230 0.86814041  
## 14 Graves Bridge - PCR3 -0.76116460 0.4465587608 1.00000000  
## 15 PCR1 - PCR3 -0.01282188 0.9897699041 0.98976990  
## 16 BRR2 - Policeman Flats 2.39914515 0.0164333992 0.37796818  
## 17 Cochrane - Policeman Flats -1.93457788 0.0530421162 0.90171598  
## 18 Cushing Bridge - Policeman Flats -2.40292797 0.0162643907 0.39034538  
## 19 Graves Bridge - Policeman Flats -0.74241913 0.4578334301 1.00000000  
## 20 PCR1 - Policeman Flats 0.19031460 0.8490626151 1.00000000  
## 21 PCR3 - Policeman Flats 0.20270366 0.8393666594 1.00000000  
## 22 BRR2 - Sunalta 3.46221313 0.0005357527 0.01446532  
## 23 Cochrane - Sunalta 0.41869046 0.6754423674 1.00000000  
## 24 Cushing Bridge - Sunalta -0.04984202 0.9602482848 1.00000000  
## 25 Graves Bridge - Sunalta 1.34733339 0.1778728837 1.00000000  
## 26 PCR1 - Sunalta 1.86494103 0.0621896703 0.99503472  
## 27 PCR3 - Sunalta 1.85337163 0.0638291110 0.95743667  
## 28 Policeman Flats - Sunalta 2.17441944 0.0296736497 0.65282029

hist(div.df$Shannon) #Does not look normal. Potential binomial distribution?



shapiro.test(er$Shannon) #Not normal

##   
## Shapiro-Wilk normality test  
##   
## data: er$Shannon  
## W = 0.93235, p-value = 2.633e-11

res.aov <- glm.nb(er$Shannon ~ Site + Family, data = as(sample\_data(sample\_ps),"data.frame"))

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.347870

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.175786

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.294790

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.790204

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.260907

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.301367

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.055523

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.694098

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.112188

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.055423

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.677783

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.031617

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.552319

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.940339

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.991751

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.307291

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.353414

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.819245

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.078310

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.771887

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.057733

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.832445

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.916147

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.813321

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.172406

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.713983

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.296780

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.780884

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.381884

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.305827

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.385073

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.341083

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.427875

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.188341

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.333937

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.402465

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.262422

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.232898

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.346582

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.856936

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.572294

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.701824

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.562692

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.230086

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.340666

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.857165

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.942372

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.478125

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.327714

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.437709

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.473488

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.668397

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.773971

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.563082

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.154387

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.711471

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.811267

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.405731

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.695791

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.157537

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.321719

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.818501

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.044934

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.370804

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.739996

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.541891

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.458434

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.679170

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.704076

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.239880

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.524296

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.346992

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.164094

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.445141

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.549256

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.100816

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.001540

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.641555

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.645618

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.790139

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.386385

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.700913

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.293455

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.058574

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.513248

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.072610

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.014165

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.611150

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.876747

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.057516

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.457538

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.463280

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.164982

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.995493

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.989548

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.358467

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.030836

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.731547

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.130194

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.365195

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.472982

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.190993

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.271879

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.754611

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.842814

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.929834

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.555172

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.510258

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.482685

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.215964

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.621847

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.415781

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.988371

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.701938

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.357032

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.511951

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.563183

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.643912

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.614510

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.011626

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.776548

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.060206

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.541172

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.580933

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.378362

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.628104

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.227516

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.933254

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.920058

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.469333

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.364973

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.628766

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.651082

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.788528

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.568469

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.517078

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.077941

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.093024

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.871011

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.086901

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.418352

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.162732

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.072976

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.335105

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.127232

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.605881

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.873301

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.074503

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.381973

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.753521

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.174928

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.026710

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.198532

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.043577

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.796220

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.024134

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.158163

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.041081

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.557200

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.042323

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.603497

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.424663

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.293879

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.046892

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.038320

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.430316

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.512105

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.092215

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.471122

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.675676

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.934123

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.432356

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.316216

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.318178

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.458445

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.603131

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.012330

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.876164

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.737802

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.655042

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.175296

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.406642

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.983916

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.868587

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.250028

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.218796

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.157430

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.571827

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.090907

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.156547

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.836452

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.972577

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.499354

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.445222

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.674678

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.816901

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.684545

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.042319

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.734650

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.143571

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.929202

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.768656

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.493764

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.535350

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.319589

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.681398

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.124542

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.772925

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.247328

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.554035

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.409424

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.823882

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.997241

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.248998

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.519043

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.013895

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.548060

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.231410

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.525664

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.866606

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.265361

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.901075

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.942917

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.882741

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.904700

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.200598

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.781679

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.025856

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.318813

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.352387

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.565200

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.282904

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.772685

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.072735

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.519499

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.338602

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.408297

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.375220

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.245552

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.327996

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.473298

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.443873

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.885553

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.457754

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.542857

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.518470

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.670726

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.306281

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.426446

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.543992

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.783574

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.673459

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.223473

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.311501

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.720376

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.049422

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.748495

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.499306

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.308893

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.435315

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.917954

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.542073

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.159240

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.963205

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.232330

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.835983

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.243312

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.304534

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.295124

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.091747

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.703577

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.466735

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.366360

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.561659

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.578709

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.808852

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.295111

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.476563

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.322090

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.688016

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.093685

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.746981

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.512237

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.291918

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.703079

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.667734

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.027709

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.289839

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.488052

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.649628

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.373156

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.248457

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.163319

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.199397

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.596307

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.450547

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.872915

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.048699

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.162251

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.606302

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.822391

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.379673

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.390212

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.826093

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.591570

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.595032

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.258568

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.296289

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.257819

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.263908

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.263650

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.256682

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.232041

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.561100

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.271033

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.224309

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.263193

## Warning in dpois(y, mu, log = TRUE): non-integer x = 5.333183

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.610205

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.309386

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.838511

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.883595

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.160280

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.910101

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.472531

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.454625

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.760653

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.141014

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.988255

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.518827

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.571608

## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.258374

## Warning in dpois(y, mu, log = TRUE): non-integer x = 3.488999

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.777670

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.586211

## Warning in dpois(y, mu, log = TRUE): non-integer x = 2.321555

## Warning in dpois(y, mu, log = TRUE): non-integer x = 1.094486

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.541233

## Warning in dpois(y, mu, log = TRUE): non-integer x = 4.060042

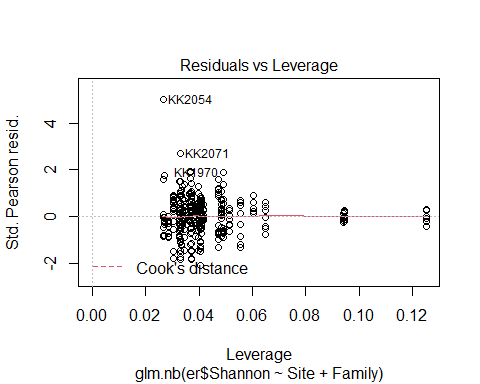
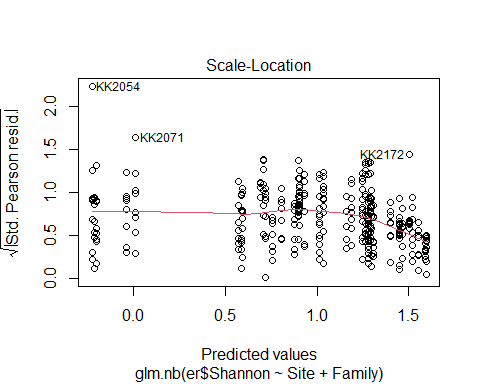
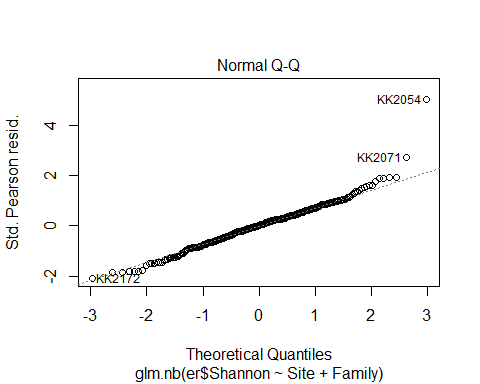
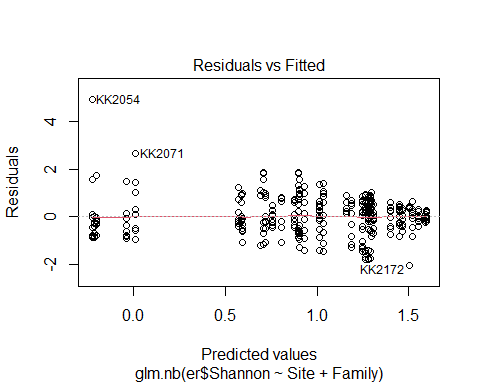
## Warning in dpois(y, mu, log = TRUE): non-integer x = 0.155507

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =  
## control$trace > : iteration limit reached  
  
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =  
## control$trace > : iteration limit reached

# Summary of the analysis  
summary(res.aov)

##   
## Call:  
## glm.nb(formula = er$Shannon ~ Site + Family, data = as(sample\_data(sample\_ps),   
## "data.frame"), init.theta = 56910.432, link = log)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.8662 -0.6412 0.0152 0.4354 3.2725   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.717456 0.127121 5.644 1.66e-08 \*\*\*  
## SiteSunalta -0.009158 0.120835 -0.076 0.93959   
## SiteCushing Bridge -0.024917 0.107940 -0.231 0.81744   
## SiteGraves Bridge 0.162030 0.115300 1.405 0.15993   
## SitePoliceman Flats 0.213045 0.103010 2.068 0.03862 \*   
## SiteBRR2 0.261591 0.186617 1.402 0.16099   
## SitePCR1 0.306409 0.165710 1.849 0.06445 .   
## SitePCR3 0.301466 0.165902 1.817 0.06920 .   
## FamilyAraneidae -0.123485 0.157703 -0.783 0.43361   
## FamilyBaetidae -0.121874 0.202514 -0.602 0.54730   
## FamilyChironomidae 0.318177 0.145451 2.188 0.02870 \*   
## FamilyHeptageniidae 0.589894 0.132763 4.443 8.86e-06 \*\*\*  
## FamilyHydropsychidae 0.572914 0.123259 4.648 3.35e-06 \*\*\*  
## FamilyPerlidae 0.471027 0.160840 2.929 0.00341 \*\*   
## FamilyTetragnathidae -0.921173 0.199564 -4.616 3.91e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for Negative Binomial(56910.43) family taken to be 1)  
##   
## Null deviance: 434.52 on 339 degrees of freedom  
## Residual deviance: 262.15 on 325 degrees of freedom  
## AIC: 1188.9  
##   
## Number of Fisher Scoring iterations: 1  
##   
##   
## Theta: 56910   
## Std. Err.: 334822   
## Warning while fitting theta: iteration limit reached   
##   
## 2 x log-likelihood: -1156.883

plot(res.aov) #Doesn"t fit assumptions of anova well. Let"s use kruskal test instead



summary(glht(res.aov, linfct=mcp(Site="Tukey")))

##   
## Simultaneous Tests for General Linear Hypotheses  
##   
## Multiple Comparisons of Means: Tukey Contrasts  
##   
##   
## Fit: glm.nb(formula = er$Shannon ~ Site + Family, data = as(sample\_data(sample\_ps),   
## "data.frame"), init.theta = 56910.432, link = log)  
##   
## Linear Hypotheses:  
## Estimate Std. Error z value Pr(>|z|)  
## Sunalta - Cochrane == 0 -0.009158 0.120835 -0.076 1.000  
## Cushing Bridge - Cochrane == 0 -0.024917 0.107940 -0.231 1.000  
## Graves Bridge - Cochrane == 0 0.162030 0.115300 1.405 0.845  
## Policeman Flats - Cochrane == 0 0.213045 0.103010 2.068 0.417  
## BRR2 - Cochrane == 0 0.261591 0.186617 1.402 0.846  
## PCR1 - Cochrane == 0 0.306409 0.165710 1.849 0.568  
## PCR3 - Cochrane == 0 0.301466 0.165902 1.817 0.590  
## Cushing Bridge - Sunalta == 0 -0.015758 0.123818 -0.127 1.000  
## Graves Bridge - Sunalta == 0 0.171188 0.123723 1.384 0.855  
## Policeman Flats - Sunalta == 0 0.222203 0.120972 1.837 0.577  
## BRR2 - Sunalta == 0 0.270749 0.191863 1.411 0.842  
## PCR1 - Sunalta == 0 0.315568 0.171596 1.839 0.575  
## PCR3 - Sunalta == 0 0.310624 0.171781 1.808 0.596  
## Graves Bridge - Cushing Bridge == 0 0.186947 0.118423 1.579 0.749  
## Policeman Flats - Cushing Bridge == 0 0.237961 0.105871 2.248 0.307  
## BRR2 - Cushing Bridge == 0 0.286508 0.189158 1.515 0.787  
## PCR1 - Cushing Bridge == 0 0.331326 0.168566 1.966 0.487  
## PCR3 - Cushing Bridge == 0 0.326382 0.168756 1.934 0.508  
## Policeman Flats - Graves Bridge == 0 0.051015 0.115444 0.442 1.000  
## BRR2 - Graves Bridge == 0 0.099561 0.188426 0.528 0.999  
## PCR1 - Graves Bridge == 0 0.144379 0.167744 0.861 0.988  
## PCR3 - Graves Bridge == 0 0.139435 0.167934 0.830 0.991  
## BRR2 - Policeman Flats == 0 0.048546 0.188353 0.258 1.000  
## PCR1 - Policeman Flats == 0 0.093365 0.167662 0.557 0.999  
## PCR3 - Policeman Flats == 0 0.088421 0.167852 0.527 0.999  
## PCR1 - BRR2 == 0 0.044818 0.213533 0.210 1.000  
## PCR3 - BRR2 == 0 0.039874 0.213682 0.187 1.000  
## PCR3 - PCR1 == 0 -0.004944 0.160689 -0.031 1.000  
## (Adjusted p values reported -- single-step method)

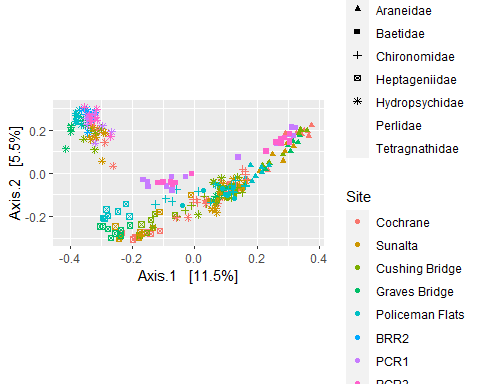
#chao.aov <- aov(log(er$Chao1) ~ Site + Family, data = as(sample\_data(sample\_ps),"data.frame"))  
#summary(chao.aov)  
#plot(chao.aov)  
#hist(log(er$Chao1)) #A bit better but not great  
#shapiro.test(log(er$Chao1)) #Still not normal   
  
#summary(glm(er$Shannon ~ Site + Family, family = "beta", data = as(sample\_data(sample\_ps), "data.frame")))  
  
kruskal.test(er$Shannon ~ Site, data = as(sample\_data(sample\_ps), "data.frame"))#Yes differences across sites

##   
## Kruskal-Wallis rank sum test  
##   
## data: er$Shannon by Site  
## Kruskal-Wallis chi-squared = 22.242, df = 7, p-value = 0.002308

##Beta Diversity

logt = transform\_sample\_counts(sample\_ps, function(x) log(1 + x) )  
out.mds.logt <- ordinate(logt, method = "MDS", distance = "bray")  
evals <- out.mds.logt$values$Eigenvalues  
plot\_ordination(logt, out.mds.logt, type = "samples",  
 color = "Site", shape = "Family") + labs(col = "Site") +  
 coord\_fixed(sqrt(evals[2] / evals[1]))

## Warning: The shape palette can deal with a maximum of 6 discrete values because  
## more than 6 becomes difficult to discriminate; you have 8. Consider  
## specifying shapes manually if you must have them.



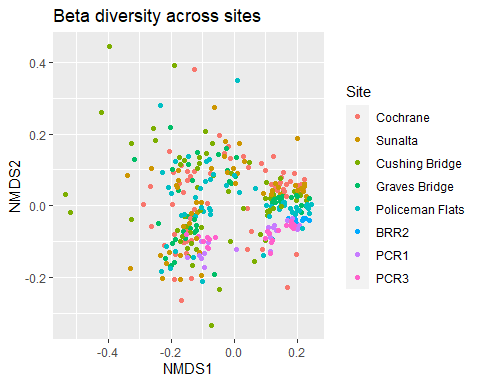
#d <- distance(sample\_ps, method="bray", type="samples")  
#clust <- hclust(d, method="average")  
#plot(clust)  
  
ord <- ordinate(sample\_ps, "NMDS", "bray")

## Square root transformation  
## Wisconsin double standardization  
## Run 0 stress 0.1885117   
## Run 1 stress 0.1900286   
## Run 2 stress 0.1867684   
## ... New best solution  
## ... Procrustes: rmse 0.01643195 max resid 0.1720931   
## Run 3 stress 0.1936741   
## Run 4 stress 0.1873941   
## Run 5 stress 0.1845646   
## ... New best solution  
## ... Procrustes: rmse 0.01703876 max resid 0.1428751   
## Run 6 stress 0.1850484   
## ... Procrustes: rmse 0.01853757 max resid 0.1412708   
## Run 7 stress 0.1853173   
## Run 8 stress 0.1864278   
## Run 9 stress 0.1978125   
## Run 10 stress 0.188469   
## Run 11 stress 0.1865944   
## Run 12 stress 0.184013   
## ... New best solution  
## ... Procrustes: rmse 0.01838383 max resid 0.1410178   
## Run 13 stress 0.1913175   
## Run 14 stress 0.1999938   
## Run 15 stress 0.1893979   
## Run 16 stress 0.187614   
## Run 17 stress 0.1992065   
## Run 18 stress 0.1831733   
## ... New best solution  
## ... Procrustes: rmse 0.009328274 max resid 0.1142053   
## Run 19 stress 0.1896455   
## Run 20 stress 0.1906081   
## \*\*\* Best solution was not repeated -- monoMDS stopping criteria:  
## 18: stress ratio > sratmax  
## 2: scale factor of the gradient < sfgrmin

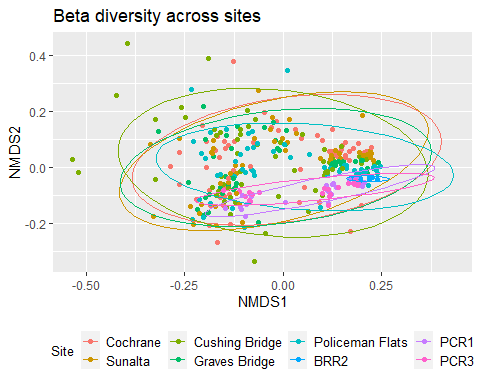
ord.plot = plot\_ordination(sample\_ps, ord, type="Family", color="Site", title="Beta diversity across sites")

## Warning in plot\_ordination(sample\_ps, ord, type = "Family", color = "Site", : type argument not supported. `type` set to 'samples'.  
## See `plot\_ordination('list')`

print(ord.plot)



ord.plot = ord.plot + theme(legend.position="bottom", legend.key.size = unit(0.5, "cm"),  
 legend.text=element\_text(size=10), legend.title=element\_text(size=10)) + stat\_ellipse()  
ord.plot



#Clustering based on Bray-Curtis dissimilarity with dendrogram, coloured by Site and insect Taxa  
ps\_rel\_otu <- data.frame(phyloseq::otu\_table(sample\_ps))  
ps\_rel\_otu <- t(ps\_rel\_otu)  
bc\_dist <- vegan::vegdist(ps\_rel\_otu, method = "bray")  
  
#Save as dendrogram  
ward <- as.dendrogram(hclust(bc\_dist, method = "ward.D2"))  
#Provide color codes  
meta <- data.frame(phyloseq::sample\_data(sample\_ps))  
colorCode <- c("Cochrane" = "red", "Sunalta" = "blue", "Cushing Bridge" = "orange", "Graves Bridge" = "green", "Policeman Flats" = "purple", "BRR2"="pink", "PCR1"="yellow", "PCR3"="brown")  
library(dendextend)

## Warning: package 'dendextend' was built under R version 4.1.3

## Registered S3 method overwritten by 'dendextend':  
## method from   
## rev.hclust vegan

##   
## ---------------------  
## Welcome to dendextend version 1.17.1  
## Type citation('dendextend') for how to cite the package.  
##   
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##   
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues  
## You may ask questions at stackoverflow, use the r and dendextend tags:   
## https://stackoverflow.com/questions/tagged/dendextend  
##   
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))  
## ---------------------

##   
## Attaching package: 'dendextend'

## The following object is masked from 'package:data.table':  
##   
## set

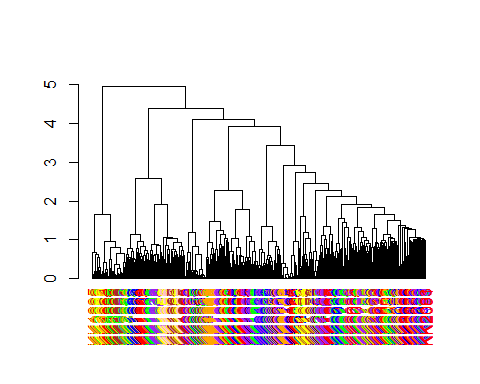
## The following object is masked from 'package:ggpubr':  
##   
## rotate

## The following objects are masked from 'package:ape':  
##   
## ladderize, rotate

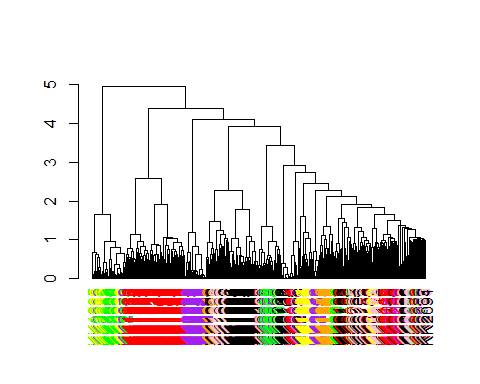
## The following object is masked from 'package:permute':  
##   
## shuffle

## The following object is masked from 'package:stats':  
##   
## cutree

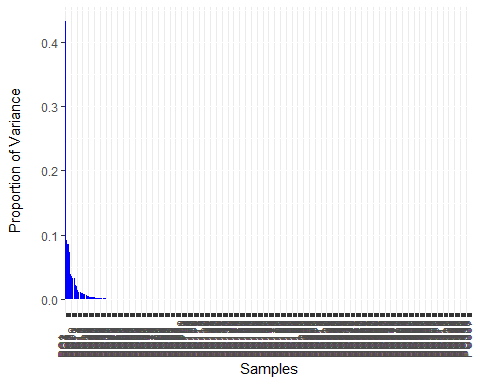
labels\_colors(ward) <- colorCode[meta$Site][order.dendrogram(ward)]  
#Plot  
plot(ward)



#Can't really read the sample names but the colours represent which site they came from. Seems like the colours are all mixing together for the most part, indicating that the sites don't cluster together in terms of their microbiome  
  
colorCode1<-c("Hydropsychidae" = "red", "Heptageniidae"="blue", "Perlidae"="orange", "Araneidae"="green", "Tetragnathidae"="purple", "Chironomidae"="pink", "Baetidae"="yellow")  
labels\_colors(ward) <- colorCode1[meta$Family][order.dendrogram(ward)]  
plot(ward)



#When colours by the family of insect sample it is more obvious that there are clusters. This makes sense because taxa that are more similar will have a more similar microbiome. Variation in clusters from the same family might be because of different genera within the family or individual variation.  
  
#PCA  
ord\_sample <- phyloseq::ordinate(sample\_ps, "RDA")  
#Plot scree plot  
phyloseq::plot\_scree(ord\_sample) +   
 geom\_bar(stat="identity", fill = "blue") +  
 labs(x = "Samples", y = "Proportion of Variance\n")+  
 theme(axis.text.x = element\_text(angle = 90, hjust=0, vjust=0))



head(ord\_sample$CA$eig)

## PC1 PC2 PC3 PC4 PC5 PC6   
## 520850418 109697798 103887949 87474916 46415327 43876117

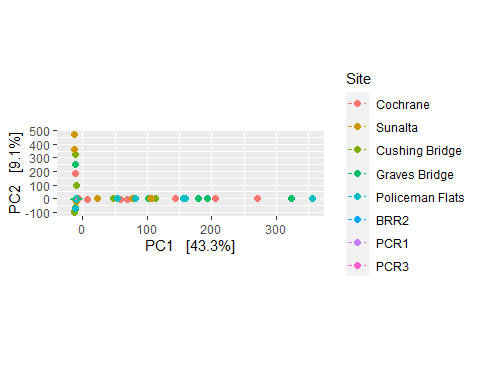
sapply(ord\_sample$CA$eig[1:5], function(x) x / sum(ord\_sample$CA$eig))

## PC1 PC2 PC3 PC4 PC5   
## 0.43263653 0.09111882 0.08629295 0.07265972 0.03855419

#PC1 represents 43% of variation, PC2 9.1%, PC3 8.6%, PC4 7.3%  
  
#Scale axes and plot ordination  
clr1 <- ord\_sample$CA$eig[1] / sum(ord\_sample$CA$eig)  
clr2 <- ord\_sample$CA$eig[2] / sum(ord\_sample$CA$eig)  
phyloseq::plot\_ordination(sample\_ps, ord\_sample, type="samples", color="Site") +   
 geom\_point(size = 2) +  
 coord\_fixed(clr2 / clr1) +  
 stat\_ellipse(aes(group = Site), linetype = 2)

## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure

## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure  
  
## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure  
  
## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure  
  
## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure



#Statistical testing  
  
#transform data  
clr\_sample <- microbiome::transform(sample\_ps, 'clr')  
clr\_bray <- phyloseq::distance(clr\_sample, method = "bray")

## Warning in vegdist(structure(c(-0.0826534128952847, -0.0660583198430373, :  
## results may be meaningless because data have negative entries in method "bray"

#ADONIS test  
#vegan::adonis(clr\_dist\_matrix ~ phyloseq::sample\_data(sample\_ps)$Site)  
sampledf <- data.frame(sample\_data(sample\_ps))  
adonis2(clr\_bray ~ Site, data = sampledf) #p=0.361 No difference in beta diversity across sites with all taxa included

## Permutation test for adonis under reduced model  
## Terms added sequentially (first to last)  
## Permutation: free  
## Number of permutations: 999  
##   
## adonis2(formula = clr\_bray ~ Site, data = sampledf)  
## Df SumOfSqs R2 F Pr(>F)  
## Site 7 5.1388e+31 0.86435 302.21 0.355  
## Residual 332 8.0647e+30 0.13565   
## Total 339 5.9453e+31 1.00000

sample\_nmds <- ordinate(  
 physeq = clr\_sample,   
 method = "NMDS",   
 distance = "bray")

## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

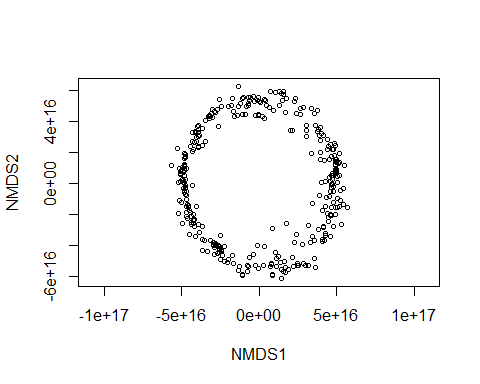
## Warning in distfun(comm, method = distance, ...): results may be meaningless because data have negative entries  
## in method "bray"

## Warning in metaMDSdist(comm, distance = distance, autotransform =  
## autotransform, : some dissimilarities exceed expected maximum 1

## Run 0 stress 0.2845715   
## Run 1 stress 0.3058429   
## Run 2 stress 0.3020355   
## Run 3 stress 0.3027884   
## Run 4 stress 0.294421   
## Run 5 stress 0.3002277   
## Run 6 stress 0.3055749   
## Run 7 stress 0.2963099   
## Run 8 stress 0.3001023   
## Run 9 stress 0.3041686   
## Run 10 stress 0.3064462   
## Run 11 stress 0.3011504   
## Run 12 stress 0.3006396   
## Run 13 stress 0.3017109   
## Run 14 stress 0.2913353   
## Run 15 stress 0.3075054   
## Run 16 stress 0.3041736   
## Run 17 stress 0.2913237   
## Run 18 stress 0.3014155   
## Run 19 stress 0.3051346   
## Run 20 stress 0.3009659   
## \*\*\* Best solution was not repeated -- monoMDS stopping criteria:  
## 13: no. of iterations >= maxit  
## 7: stress ratio > sratmax

plot(sample\_nmds)

## species scores not available



##Permanova and Pairwise Permanova  
#permanova  
set.seed(1)  
#dat\_bray <- phyloseq::distance(dat\_lessKK, method="bray")  
#sampledf <- data.frame(sample\_data(dat\_lessKK))  
#adonis(dat\_bray ~ Location, data=sampledf) #PERMANOVA Location  
#adonis(dat\_bray ~ Site, data=sampledf) #PERMANOVA Site  
   
#pairwise  
library(devtools)

## Warning: package 'devtools' was built under R version 4.1.3

## Loading required package: usethis

## Warning: package 'usethis' was built under R version 4.1.3

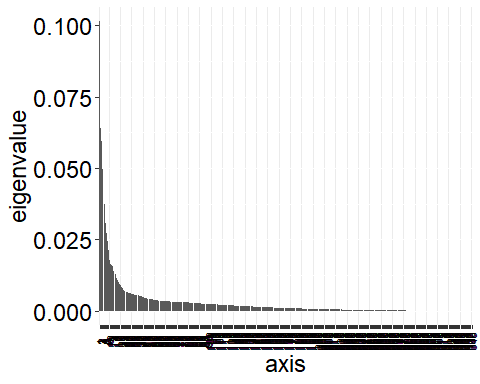
##   
## Attaching package: 'devtools'

## The following object is masked from 'package:permute':  
##   
## check

#install\_github("pmartinezarbizu/pairwiseAdonis/pairwiseAdonis", force=T)  
library(pairwiseAdonis)  
  
  
dat\_bray <- phyloseq::distance(sample\_ps, method="bray") #from above  
sampledf <- data.frame(sample\_data(sample\_ps)) #from above  
pairwise.adonis2(dat\_bray ~ Site, data=sampledf) #Pairwise Adonis by SITE with Padj

## $parent\_call  
## [1] "dat\_bray ~ Site , strata = Null , permutations 999"  
##   
## $`Cochrane\_vs\_Cushing Bridge`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.207 0.01999 2.6113 0.001 \*\*\*  
## Residual 128 59.178 0.98001   
## Total 129 60.385 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Cochrane\_vs\_Sunalta  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 0.656 0.01244 1.4105 0.06 .  
## Residual 112 52.052 0.98756   
## Total 113 52.707 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cochrane\_vs\_Graves Bridge`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 0.949 0.01814 2.069 0.006 \*\*  
## Residual 112 51.383 0.98186   
## Total 113 52.333 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cochrane\_vs\_Policeman Flats`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.477 0.02488 3.2155 0.001 \*\*\*  
## Residual 126 57.867 0.97512   
## Total 127 59.344 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Cochrane\_vs\_BRR2  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.337 0.06786 5.3143 0.001 \*\*\*  
## Residual 73 32.102 0.93214   
## Total 74 34.439 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Cochrane\_vs\_PCR1  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.979 0.04723 4.4118 0.001 \*\*\*  
## Residual 89 39.922 0.95277   
## Total 90 41.901 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Cochrane\_vs\_PCR3  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.055 0.05002 4.6335 0.001 \*\*\*  
## Residual 88 39.032 0.94998   
## Total 89 41.087 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_Sunalta`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.050 0.02065 2.277 0.003 \*\*  
## Residual 108 49.793 0.97935   
## Total 109 50.843 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_Graves Bridge`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.146 0.02279 2.5192 0.001 \*\*\*  
## Residual 108 49.125 0.97721   
## Total 109 50.271 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_Policeman Flats`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.545 0.02704 3.3902 0.001 \*\*\*  
## Residual 122 55.609 0.97296   
## Total 123 57.154 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_BRR2`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.397 0.07436 5.543 0.001 \*\*\*  
## Residual 69 29.843 0.92564   
## Total 70 32.241 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_PCR1`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.887 0.04771 4.2581 0.001 \*\*\*  
## Residual 85 37.664 0.95229   
## Total 86 39.550 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Cushing Bridge\_vs\_PCR3`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.997 0.05152 4.5626 0.001 \*\*\*  
## Residual 84 36.773 0.94848   
## Total 85 38.771 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Sunalta\_vs\_Graves Bridge`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 0.852 0.01989 1.8667 0.007 \*\*  
## Residual 92 41.999 0.98011   
## Total 93 42.851 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Sunalta\_vs\_Policeman Flats`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 0.974 0.01969 2.1293 0.001 \*\*\*  
## Residual 106 48.483 0.98031   
## Total 107 49.457 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Sunalta\_vs\_BRR2  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.2584 0.09042 5.2688 0.001 \*\*\*  
## Residual 53 22.7173 0.90958   
## Total 54 24.9757 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Sunalta\_vs\_PCR1  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.998 0.0614 4.5138 0.001 \*\*\*  
## Residual 69 30.538 0.9386   
## Total 70 32.535 1.0000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $Sunalta\_vs\_PCR3  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.222 0.06972 5.0962 0.001 \*\*\*  
## Residual 68 29.647 0.93028   
## Total 69 31.869 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Graves Bridge\_vs\_Policeman Flats`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 0.948 0.01944 2.1018 0.002 \*\*  
## Residual 106 47.814 0.98056   
## Total 107 48.763 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Graves Bridge\_vs\_BRR2`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.1245 0.08788 5.1067 0.001 \*\*\*  
## Residual 53 22.0490 0.91212   
## Total 54 24.1735 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Graves Bridge\_vs\_PCR1`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.6053 0.051 3.7084 0.001 \*\*\*  
## Residual 69 29.8692 0.949   
## Total 70 31.4745 1.000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Graves Bridge\_vs\_PCR3`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.5989 0.05229 3.7519 0.001 \*\*\*  
## Residual 68 28.9790 0.94771   
## Total 69 30.5779 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Policeman Flats\_vs\_BRR2`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.0315 0.06647 4.7704 0.001 \*\*\*  
## Residual 67 28.5327 0.93353   
## Total 68 30.5643 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Policeman Flats\_vs\_PCR1`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.992 0.05194 4.547 0.001 \*\*\*  
## Residual 83 36.353 0.94806   
## Total 84 38.345 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $`Policeman Flats\_vs\_PCR3`  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 2.256 0.0598 5.2155 0.001 \*\*\*  
## Residual 82 35.463 0.9402   
## Total 83 37.718 1.0000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $BRR2\_vs\_PCR1  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.4787 0.12255 4.1899 0.001 \*\*\*  
## Residual 30 10.5875 0.87745   
## Total 31 12.0662 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $BRR2\_vs\_PCR3  
## Df SumOfSqs R2 F Pr(>F)   
## Site 1 1.4806 0.13246 4.4279 0.002 \*\*  
## Residual 29 9.6972 0.86754   
## Total 30 11.1779 1.00000   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## $PCR1\_vs\_PCR3  
## Df SumOfSqs R2 F Pr(>F)  
## Site 1 0.4419 0.02461 1.1353 0.309  
## Residual 45 17.5175 0.97539   
## Total 46 17.9594 1.00000   
##   
## attr(,"class")  
## [1] "pwadstrata" "list"

#Scree plot  
bray\_dist = phyloseq::distance(sample\_ps, method = "bray")  
field\_ord <- ordinate(sample\_ps, method = "PCoA", distance = "bray\_dist")  
scree <- plot\_scree(field\_ord)  
scree + theme(axis.text.x=element\_text(size=8, color="black"), axis.text.y=element\_text(size=18, color="black"),   
 axis.text=element\_text(size=18, color="black"), axis.title=element\_text(size=18)) #axis 1, 2 account for most variation explained



##Differential Abundance