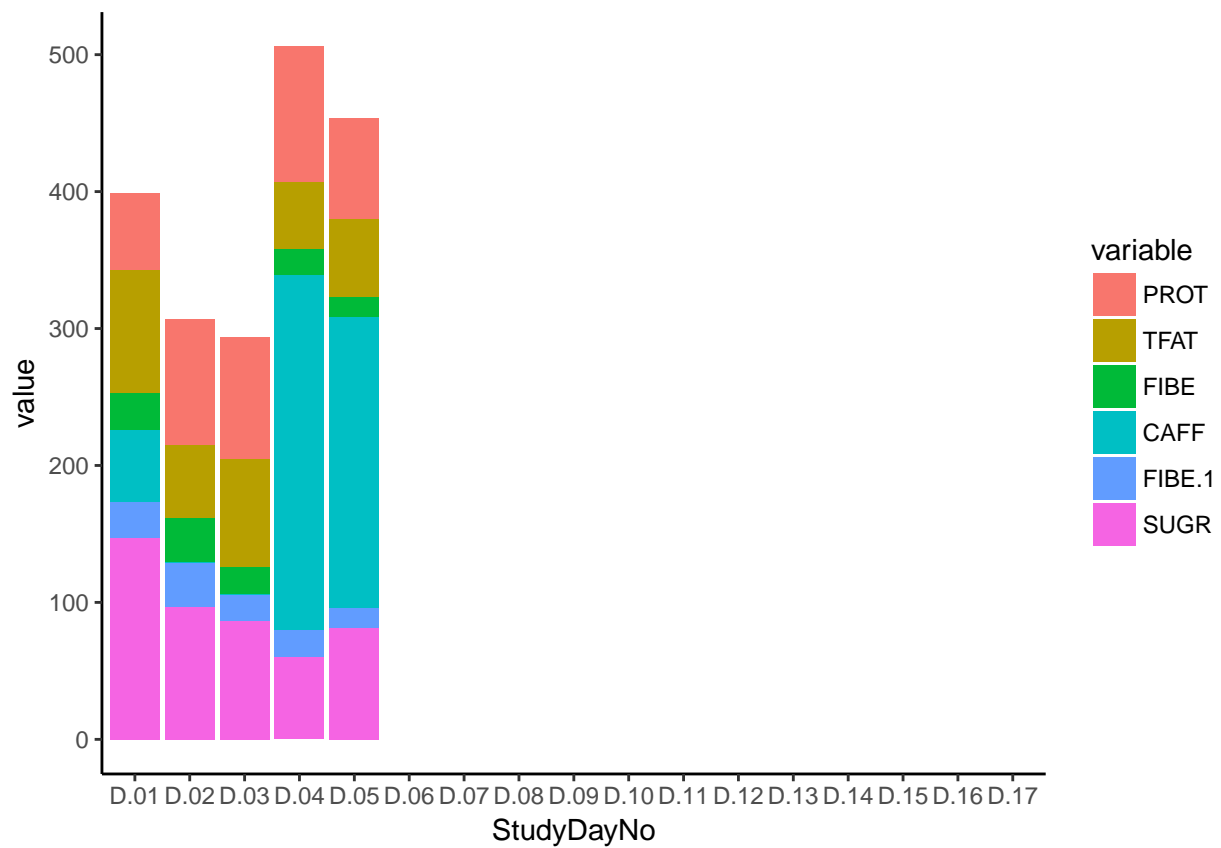


Subject_21

Type	Your Average	Total Average
CALORIES	NA	2080.11
PROTEIN	NA	88.57
TOTAL FAT	NA	89.97
CARBS	NA	225.55
FIBER	NA	21.96

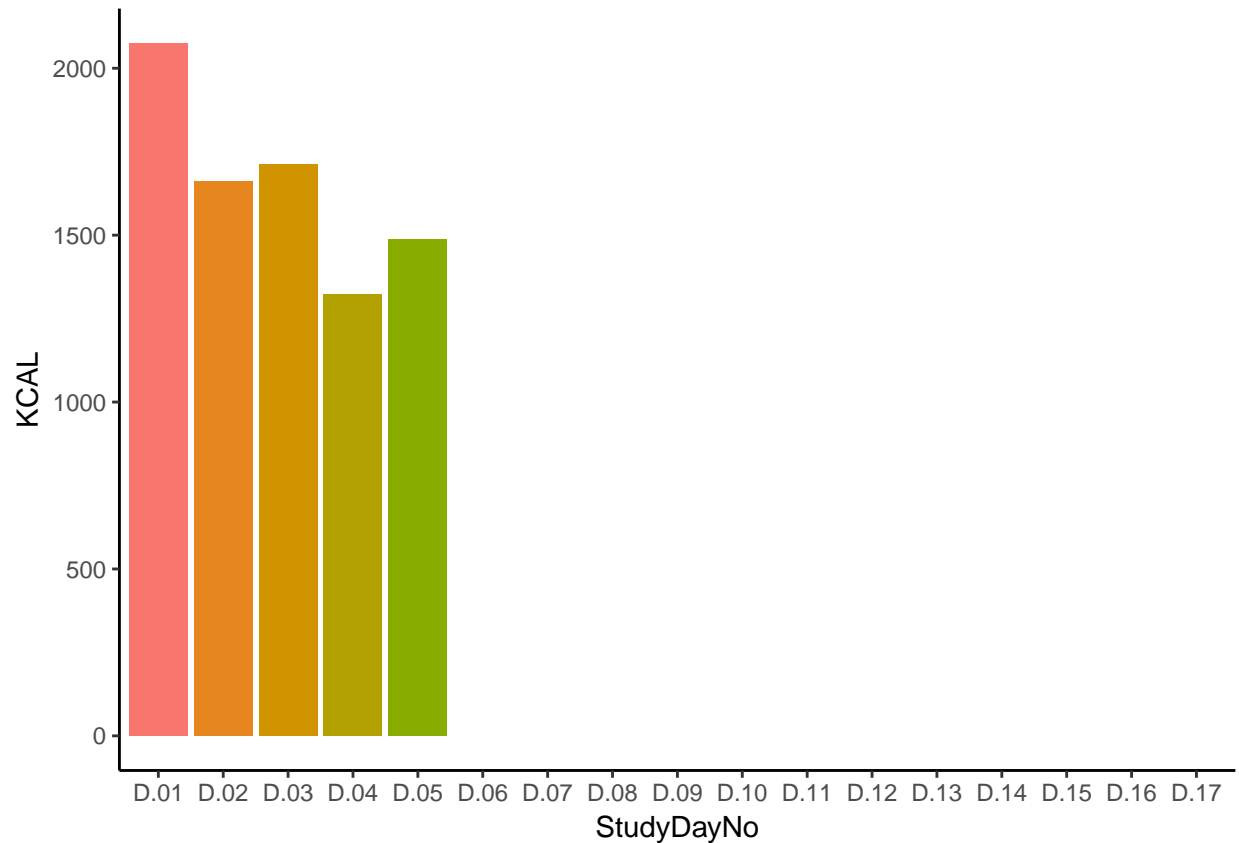
MicroNutrients

Warning: Removed 72 rows containing missing values (position_stack).



Daily Calorie Intake

Warning: Removed 12 rows containing missing values (geom_bar).



Microbiome Daily Relative Abundance

```
# make ggplot bar chart of top 10 most abundant species per day
#melt subtaxasp to get our dataframe in the long format for future usage
meltdf<- melt(subtaxasp)
```

```
## Warning in melt.data.table(subtaxasp): To be consistent with reshape2's
## melt, id.vars and measure.vars are internally guessed when both are 'NULL'.
## All non-numeric/integer/logical type columns are considered id.vars, which
## in this case are columns [rn]. Consider providing at least one of 'id' or
## 'measure' vars in future.
```

```
#merge to get access to Day var
mergedf<- merge(x=meltdf, y=map, by.x = "variable", by.y= "X.SampleID", all.x=TRUE)

#convert our dataframe species (rn) column to a character
mergedf$rn <- as.character(mergedf$rn)

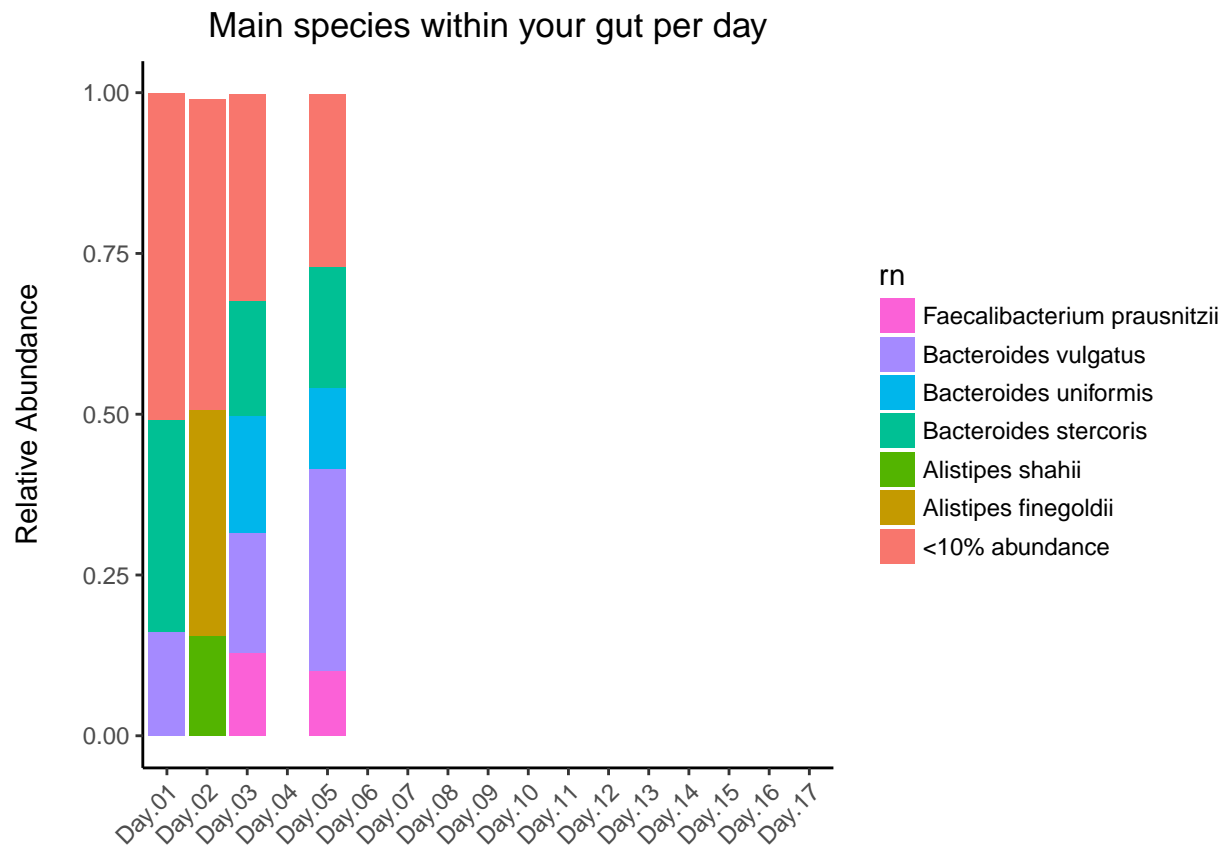
#series of gsub commands meant to neaten and clarify legend content
mergedf$rn <- gsub(".*s__", "", mergedf$rn)
mergedf$rn <- gsub("\\\\[", "",mergedf$rn)
mergedf$rn <- gsub("\\\\]", "",mergedf$rn)
mergedf$rn <- gsub("_", " ",mergedf$rn)
```

```

#create <10% abundance category
mergedf$rn[mergedf$value < 0.1] <- "<10% abundance"

ggplot(mergedf, aes(x = StudyDayNo, y = value, fill = rn)) +
  geom_bar(stat = "identity") +
  scale_x_discrete(drop = FALSE) +
  theme_classic() +
  theme(strip.text.y = element_text(angle = 0, size = 8, face = "italic"),
        axis.text.x = element_text(angle = 45, hjust = 1),
        axis.title.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        strip.background = element_rect(color = "grey")) +
  guides(fill = guide_legend(reverse = TRUE,
                              keywidth = 1,
                              keyheight = 1,
                              ncol = 1)) +
  ylab("Relative Abundance\n") +
  ggtitle("Main species within your gut per day")

```



```
## `geom_smooth()`` using method = 'loess'
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : span too small. fewer data values than degrees of freedom.
```

```

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : pseudoinverse used at 0.985

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : neighborhood radius 2.015

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
## parametric, : There are other near singularities as well. 4.0602

## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : span too small.
## fewer data values than degrees of freedom.

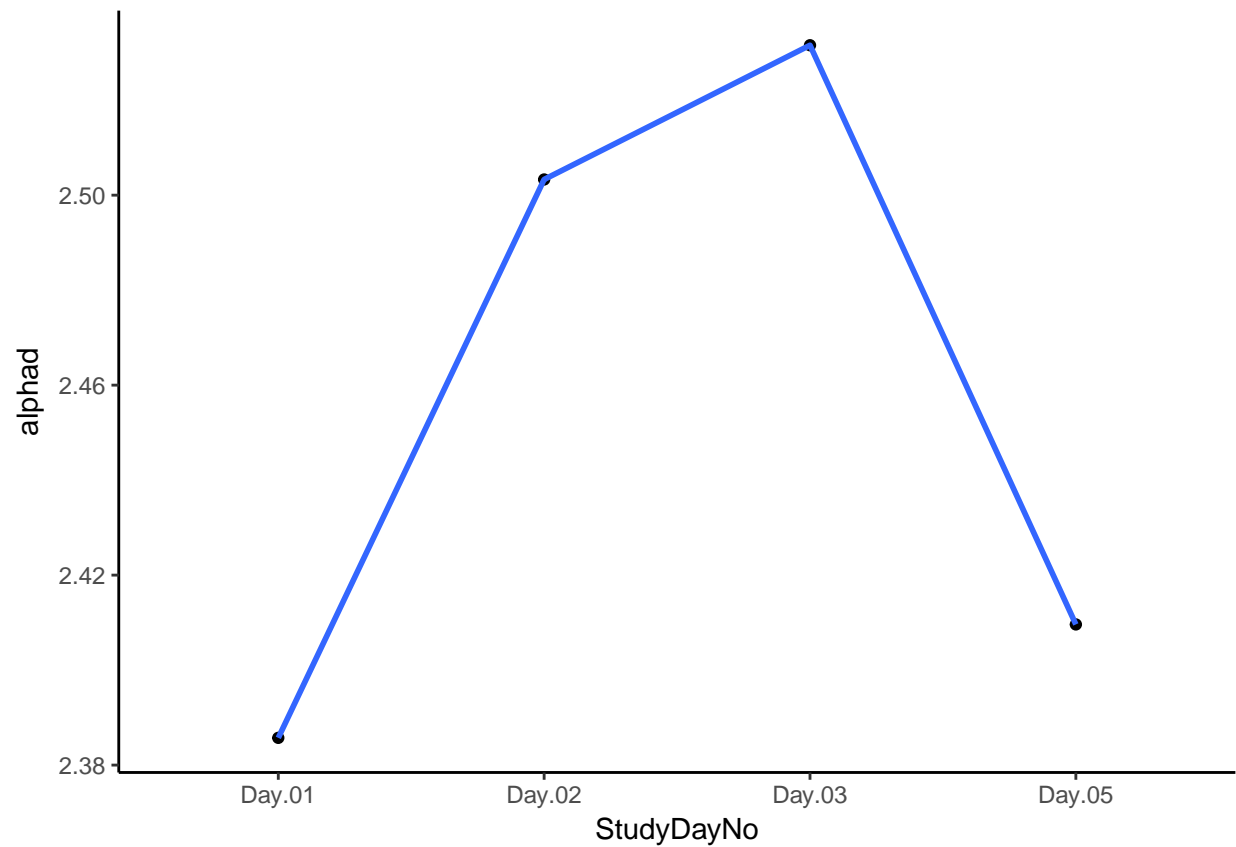
## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used
## at 0.985

## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius
## 2.015

## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal
## condition number 0

## Warning in predLoess(object$y, object$x, newx = if
## (is.null(newdata)) object$x else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other
## near singularities as well. 4.0602

```



```
#create PCOA plot of beta diversities of pertinent subjects  
#  
# ggplot(PCOA) +  
#   geom_point(aes(x=PC1,y=PC2, col=UserName, shape=UserName, size=4)) + #gives axes and item of focus  
#   labs(title="PCOA plot") +  
#   scale_shape_manual(values=shape_pal3) +  
#   theme_classic()
```