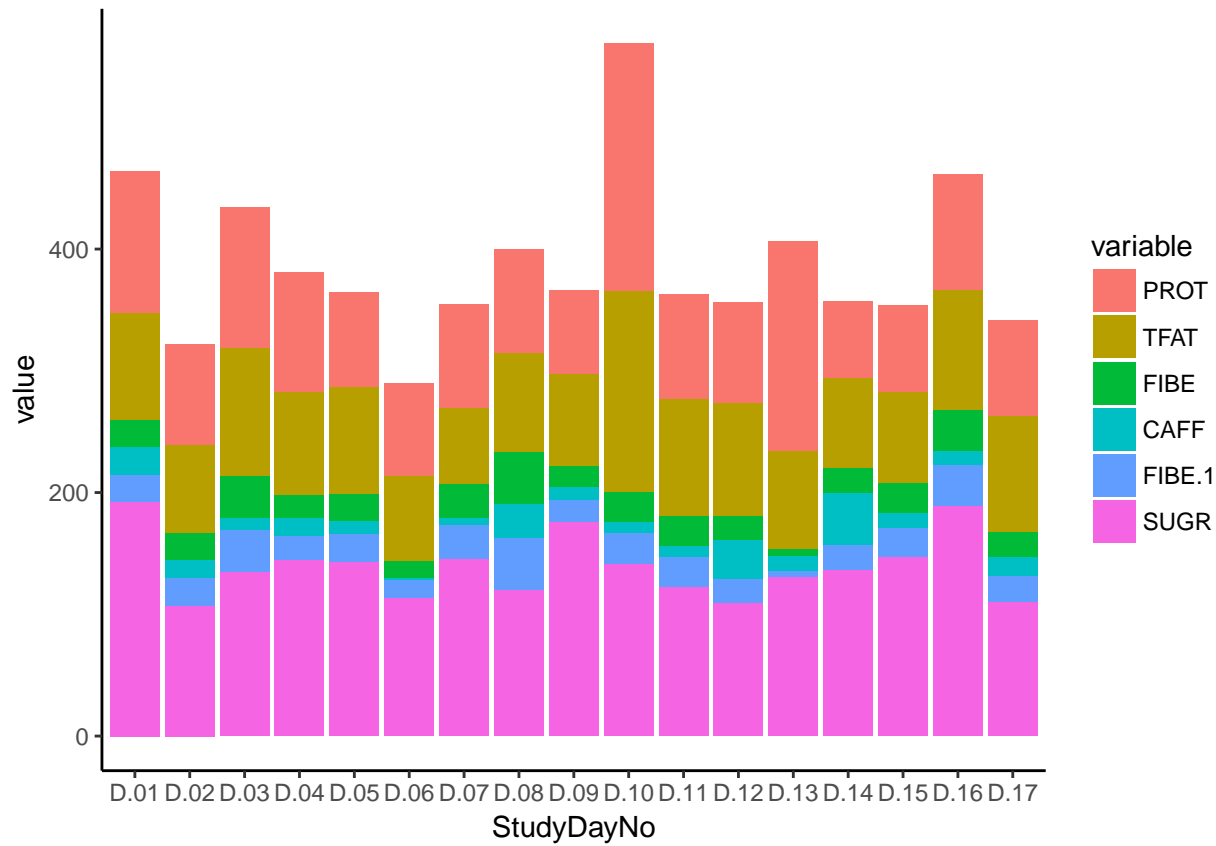


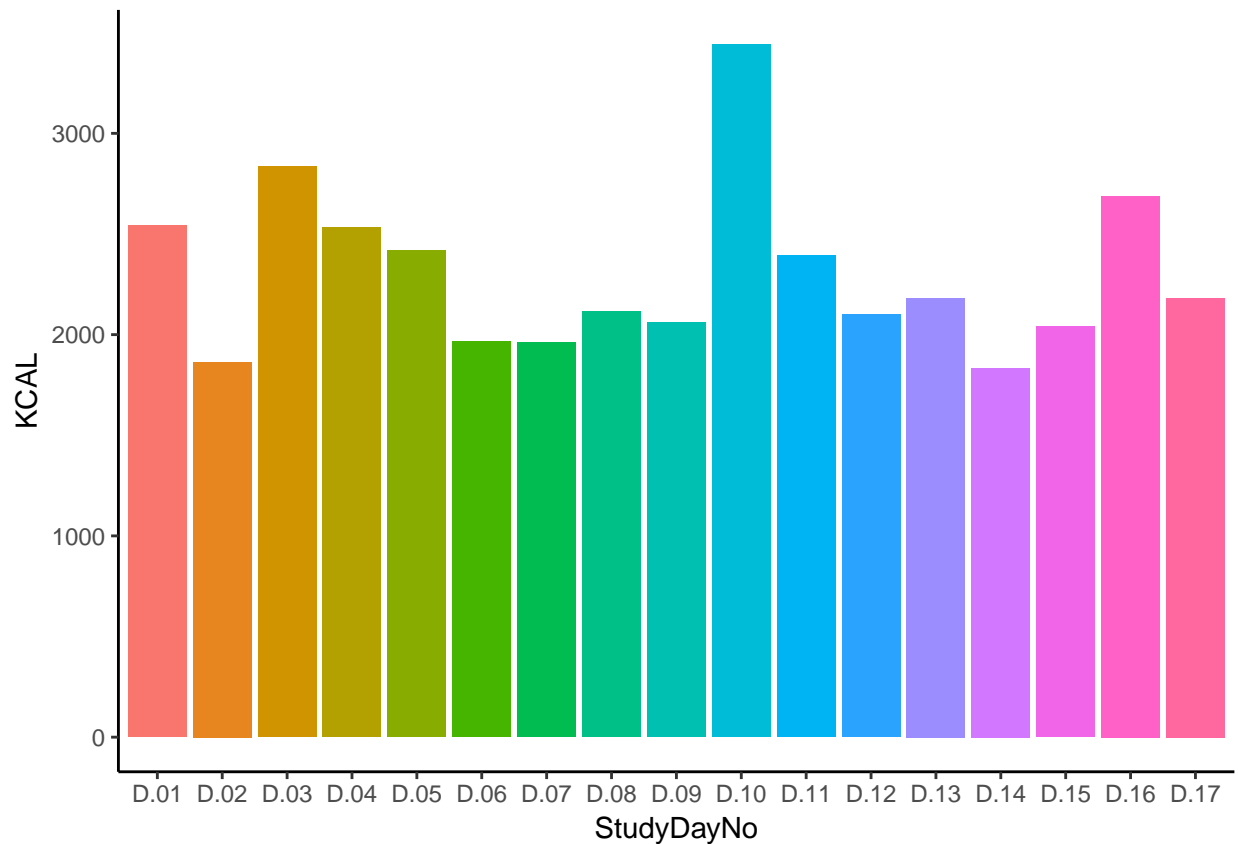
Subject_29

Type	Your Average	Total Average
CALORIES	2303.48	2080.11
PROTEIN	97.37	88.57
TOTAL FAT	88.39	89.97
CARBS	286.88	225.55
FIBER	23.44	21.96

MicroNutrients



Daily Calorie Intake



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)
```

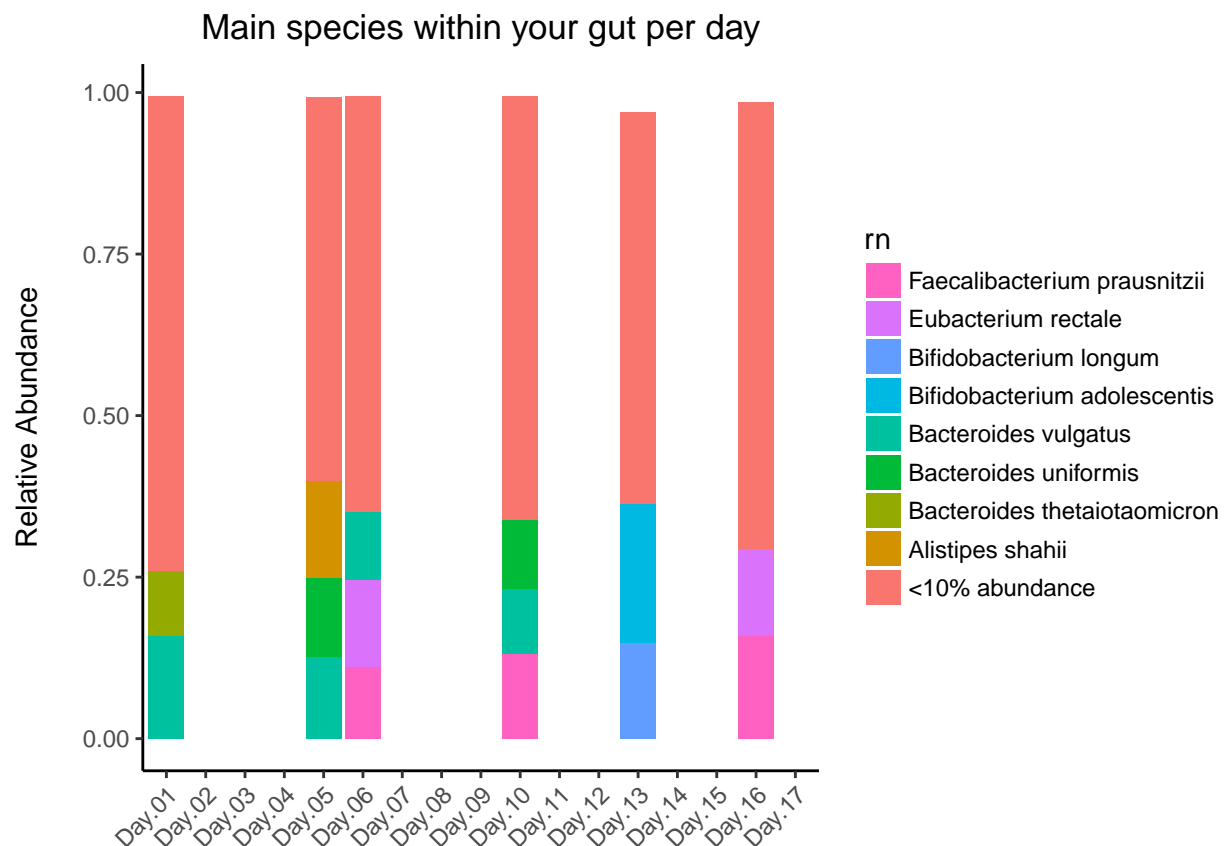
```
## Warning in melt.data.table(subtaxasp): To be consistent with reshape2's  
## melt, id.vars and measure.vars are internally guessed when both are 'NULL'.  
## All non-numeric/integer/logical type columns are considered id.vars, which  
## in this case are columns [rn]. Consider providing at least 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, : Chernobyl! trL>n 6  
  
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =  
## parametric, : Chernobyl! trL>n 6  
  
## Warning in stats::qt(level/2 + 0.5, pred$df): NaNs produced
```

