Proteome_Profiling_mRNA

```
# INITIAL COMMANDS TO RESET THE SYSTEM
rm(list = ls())
if (is.integer(dev.list())){dev.off()}
```

```
## null device
##           1
```

```
cat("\014")
```

```
seedNo=14159
set.seed(seedNo)
```

R Markdown

```
#### required library and files

library(dplyr)
```

```
##
## 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(ggplot2)
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggplot2':
##
##     ggsave
```

```
library(tidyr)
library(stringr)
library(readr)

nameDictionary_RNA_Protein <-read_csv("~/Desktop/GitHub/proteome-profiling-ecoli/Files/nameDictionary_RNA_Protei
n.csv")
```

```
## Parsed with column specification:
## cols(
##   Protein_id = col_character(),
##   gene_name = col_character(),
##   mRNA_ID = col_character(),
##   `b#` = col_character()
## )
```

```
metaData_mrna <- read_csv("~/Desktop/GitHub/proteome-profiling-ecoli/umut/metaData_mrna_trT_set00_StcAllEx_SYAN_b
aseMgAllMg_baseNaAllNa_ExpAllPhase_noFilter_p1Sf_noNorm.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   sampleNum = col_integer(),
##   growthTime_hr = col_double(),
##   cellTotal = col_double(),
##   cellsPerTube = col_double(),
##   batchNumber = col_integer(),
##   Mg_mM = col_double(),
##   Na_mM = col_integer(),
##   doublingTimeMinutes = col_double(),
##   doublingTimeMinutes.95m = col_double(),
##   doublingTimeMinutes_95p = col_double(),
##   rSquared = col_double(),
##   sizeFactor = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
mRNA_untransformed  <- read_csv("~/Desktop/GitHub/proteome-profiling-ecoli/umut/resDf_mrna_trT_set00_StcAllEx_SYA
N_baseMgAllMg_baseNaAllNa_ExpAllPhase_noFilter_p1Sf_noNorm.csv")
```

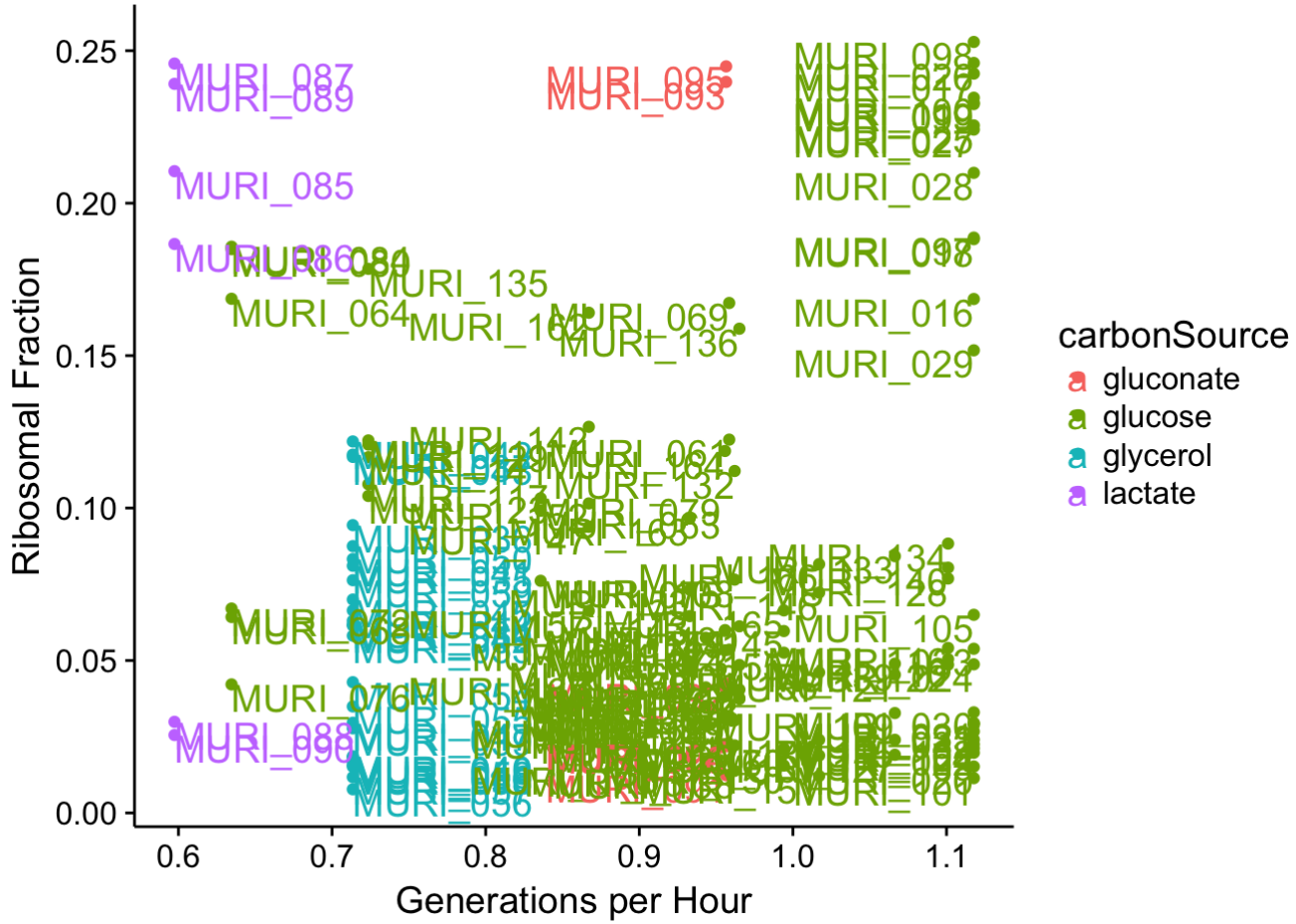
```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   X1 = col_character()
## )
## See spec(...) for full column specifications.
```

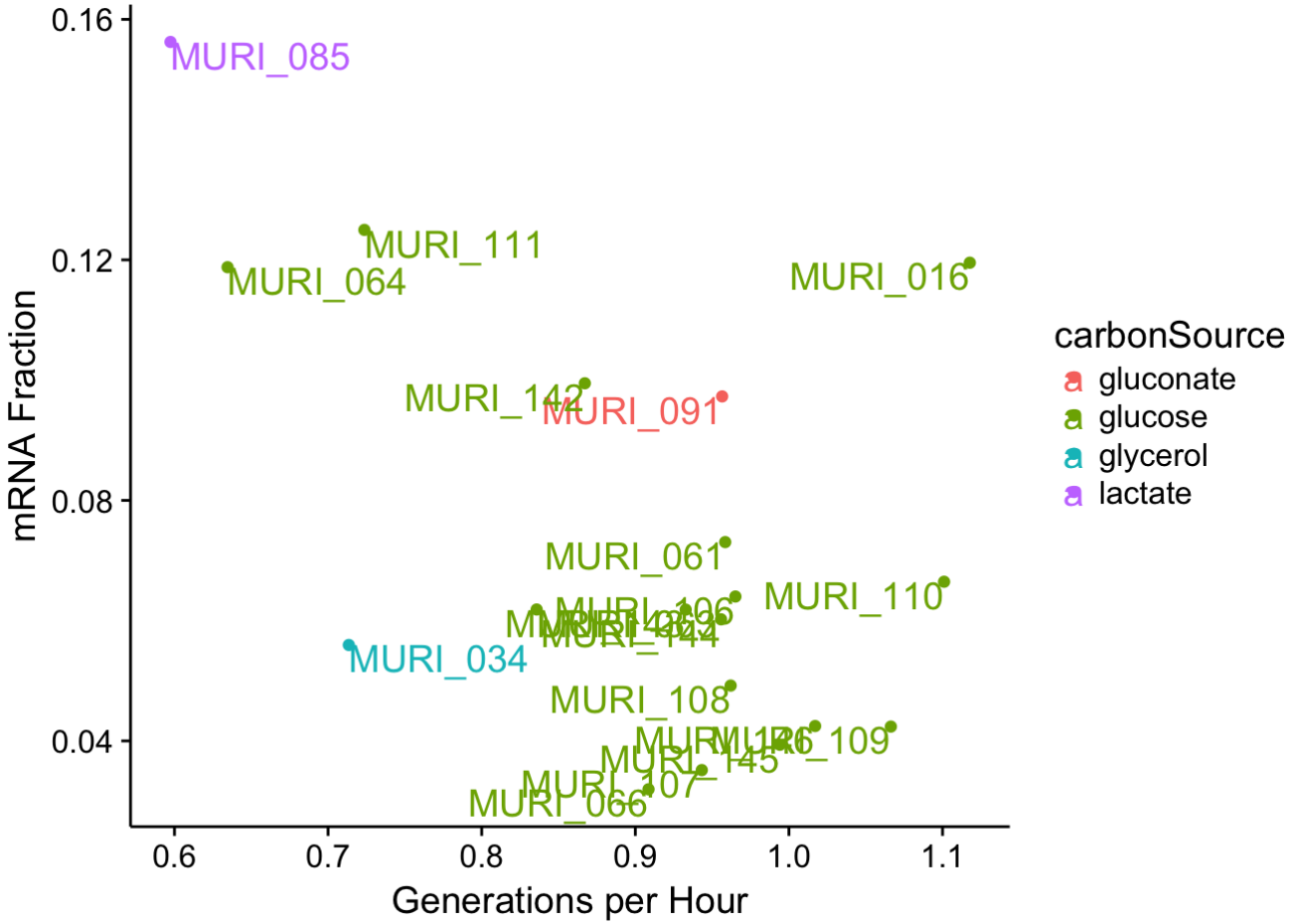
The following code is used to calculate and plot the ribosomal mRNA fraction of samples during exponential phase using untransformed data

```
############
#changing column names
names(nameDictionary_RNA_Protein)[names(nameDictionary_RNA_Protein) == 'b#'] <- 'b'

names(mRNA_untransformed)[names(mRNA_untransformed) == 'X1'] <- 'mRNA_ID'
############


############
#join untransformed data to nameDictionary by mRNA_ID
mRNA_untransformed %>% left_join(nameDictionary_RNA_Protein, by = 'mRNA_ID') %>% subset(select = -c(Protein_id,
b)) %>% select(mRNA_ID, gene_name, everything())-> UTmNames

UTmNames %>% subset(select = (-c(mRNA_ID))) -> UTmNames
############


############
#sum gene counts for each media, filter only ribosomal genes, sum ribosomal gene counts for each media, calculate
 ribosomal fraction
UTmNames %>% gather(media, count, MURI_016:MURI_171) %>% group_by(media) %>% mutate(total= sum(count)) %>%
filter(str_detect(gene_name,regex('^rp'))) %>% group_by(media) %>% mutate(mRNA = sum(count)) %>% mutate(fraction
= mRNA/total)-> UTtotalmRNA

UTtotalmRNA %>% subset(select = -c(gene_name,count, total, mRNA )) %>% distinct(fraction) -> UTmfraction
############


############
names(metaData_mrna)[names(metaData_mrna) == 'dataSet'] <- 'media'

#join Mydata (includes growth conditions and generation times) and fraction by sample
UTmfraction %>% left_join(metaData_mrna, by = 'media') %>%
  select(-fraction,fraction)-> UTmready
############


############
#calculate generations per hour
UTmready %>% mutate(generations_per_hour = ((growthTime_hr*60)/doublingTimeMinutes)/growthTime_hr) -> UTmready
```

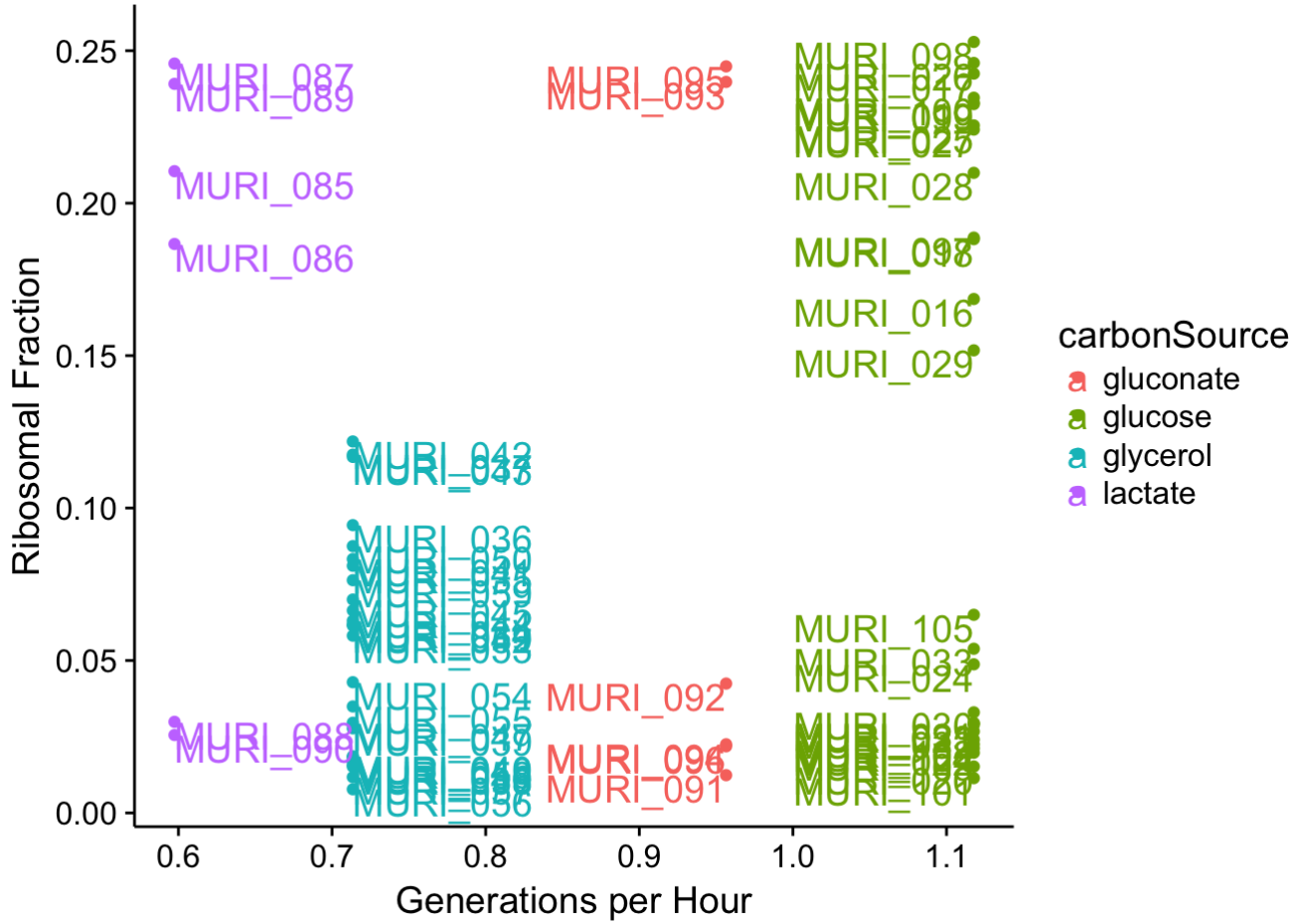
```
##########


###########
#average the ribosomal fractions of samples that have the same doubling time
UTmready %>% group_by(doublingTimeMinutes) %>%
  mutate(fraction_avg = (sum(fraction))/n()) -> UTmready
##########


###########
#unique fraction averages
UTmready %>% distinct(fraction_avg, .keep_all = TRUE) -> UTmfinal
##########


##########
#plot of fractions
UTmready %>% ggplot(aes(x=generations_per_hour, y=fraction, color=carbonSource)) + xlab('Generations per Hour') +
 ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```



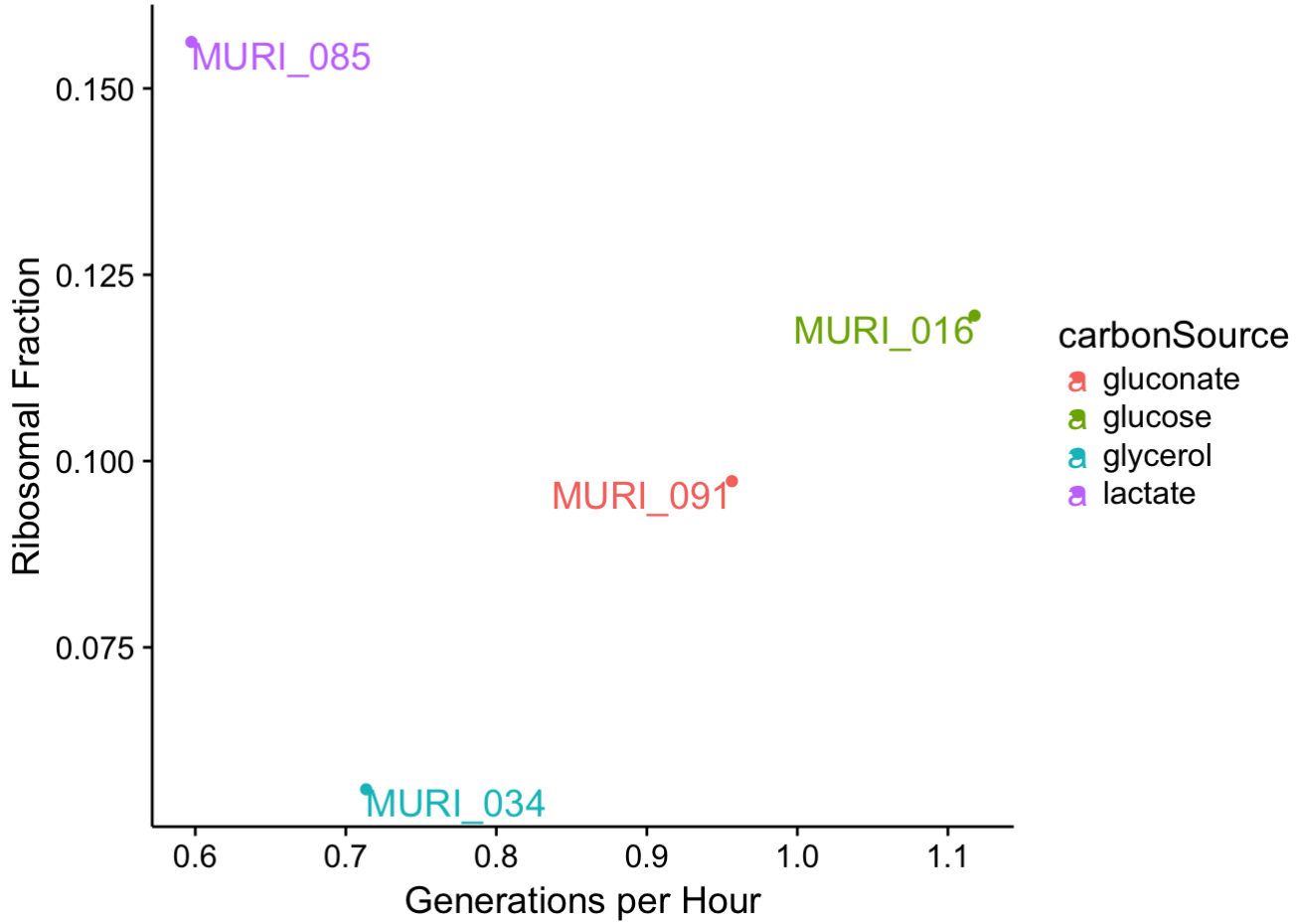

```
############


############
#plot of fraction averages
UTmfinal %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per Hou
r') + ylab('mRNA Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```




```
############


############
#get rid of NaCl and MgSO4 stress
UTmfinal_alt <- UTmfinal[!grepl("stress", UTmfinal$experiment),]
#get rid of NaCl and MgSO4 stress (non-unique fraction_avg)
UTmfinal_alt2 <- UTmready[!grepl("stress", UTmready$experiment),]
############

UTmfinal_alt2 %>% ggplot(aes(x=generations_per_hour, y=fraction, color=carbonSource)) + xlab('Generations per Hou
r') + ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```

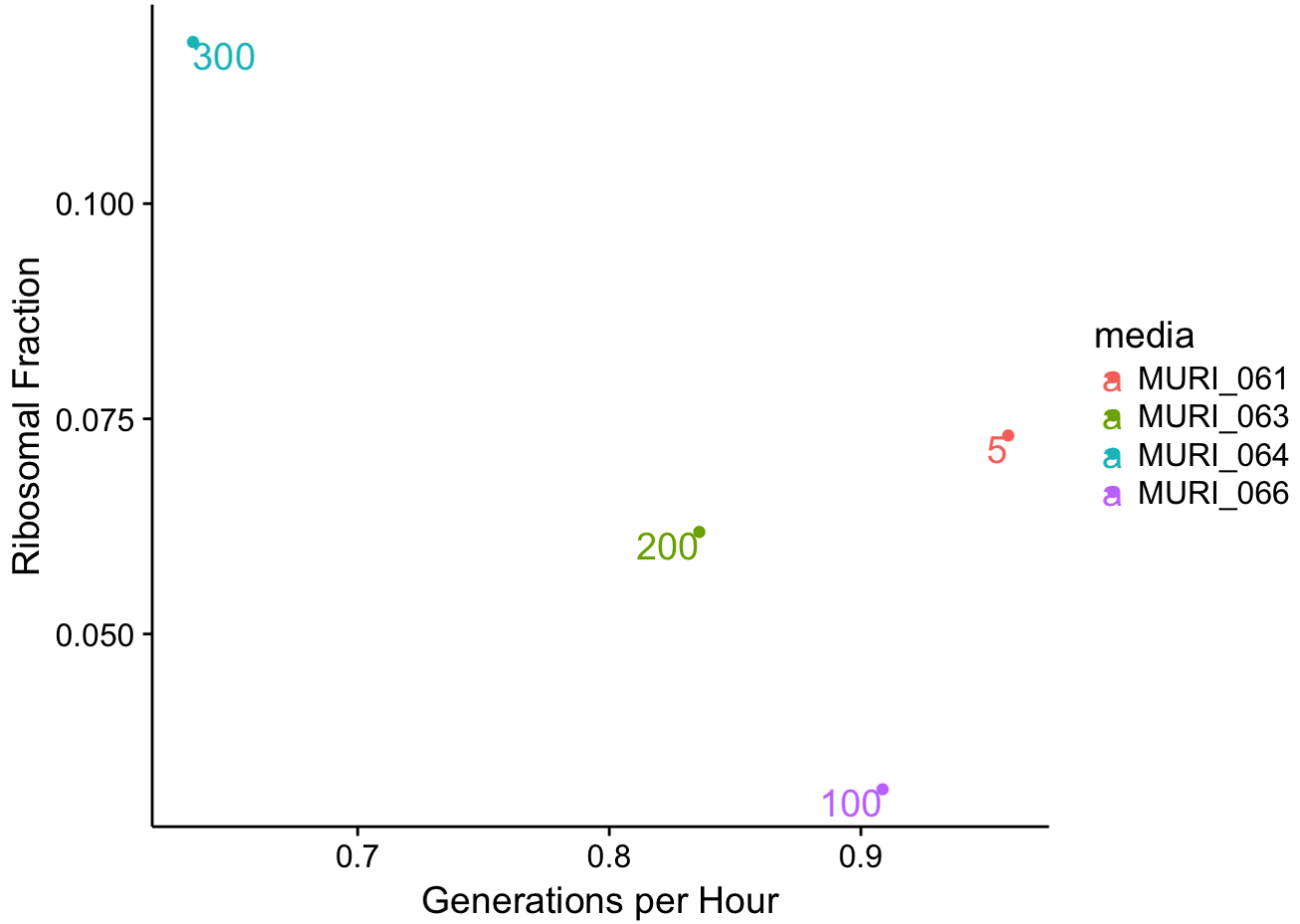



```
UTmfinal_alt %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per
 Hour') + ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size=
5)
```

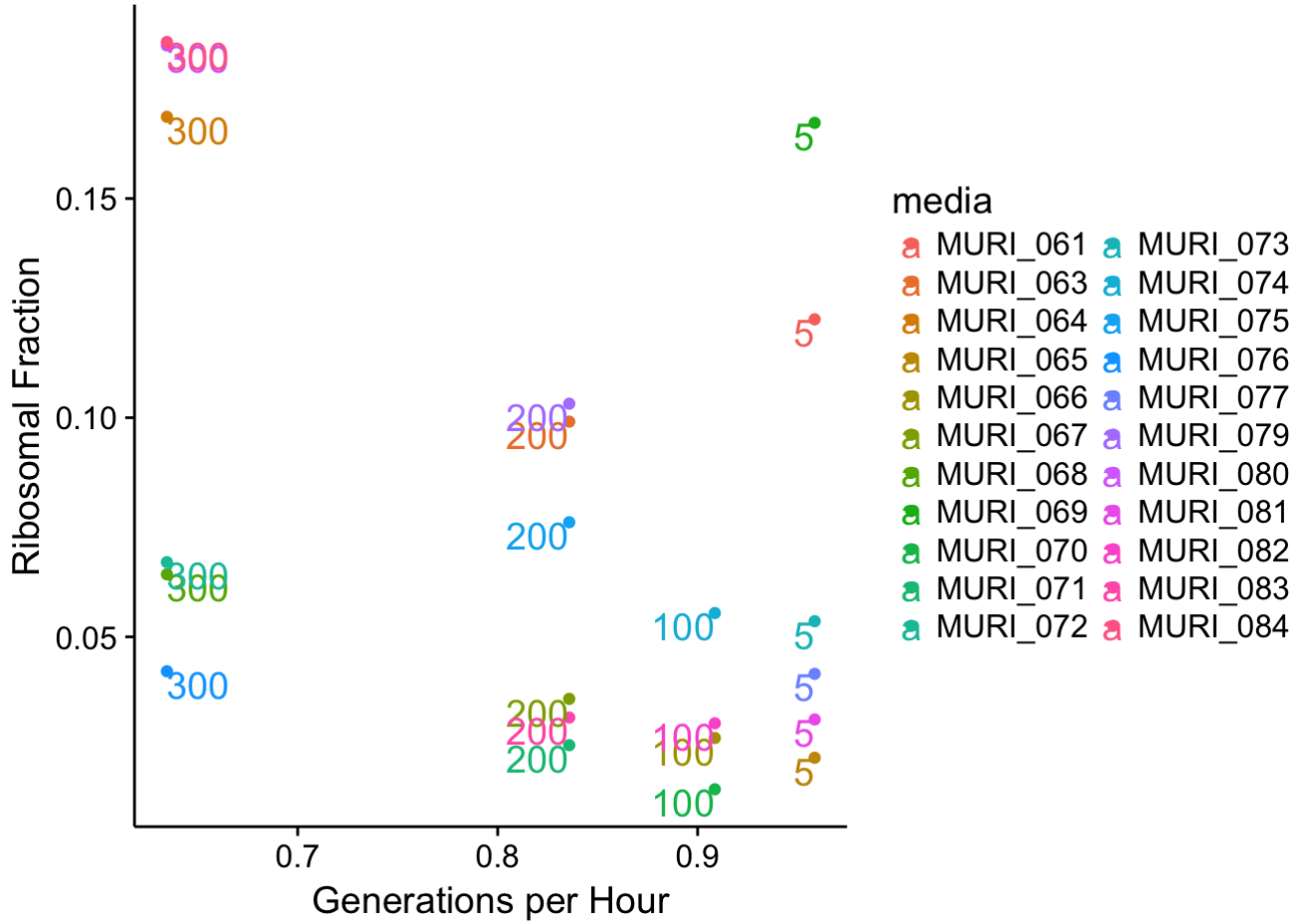


Na concentrations and NaCl stress with untransformed data

```
#keep only ond samples on NaCL stress with unique Na concentrations (5,100,200,300)
UTmready %>% filter(experiment == 'NaCl_stress') %>% ungroup() %>% distinct(Na_mM, carbonSource,.keep_all = TRUE)
 -> UTmNa

UTmNa %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=media)) + xlab('Generations per Hour') +
ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=Na_mM),hjust='inward', vjust=1 , size= 5)
```

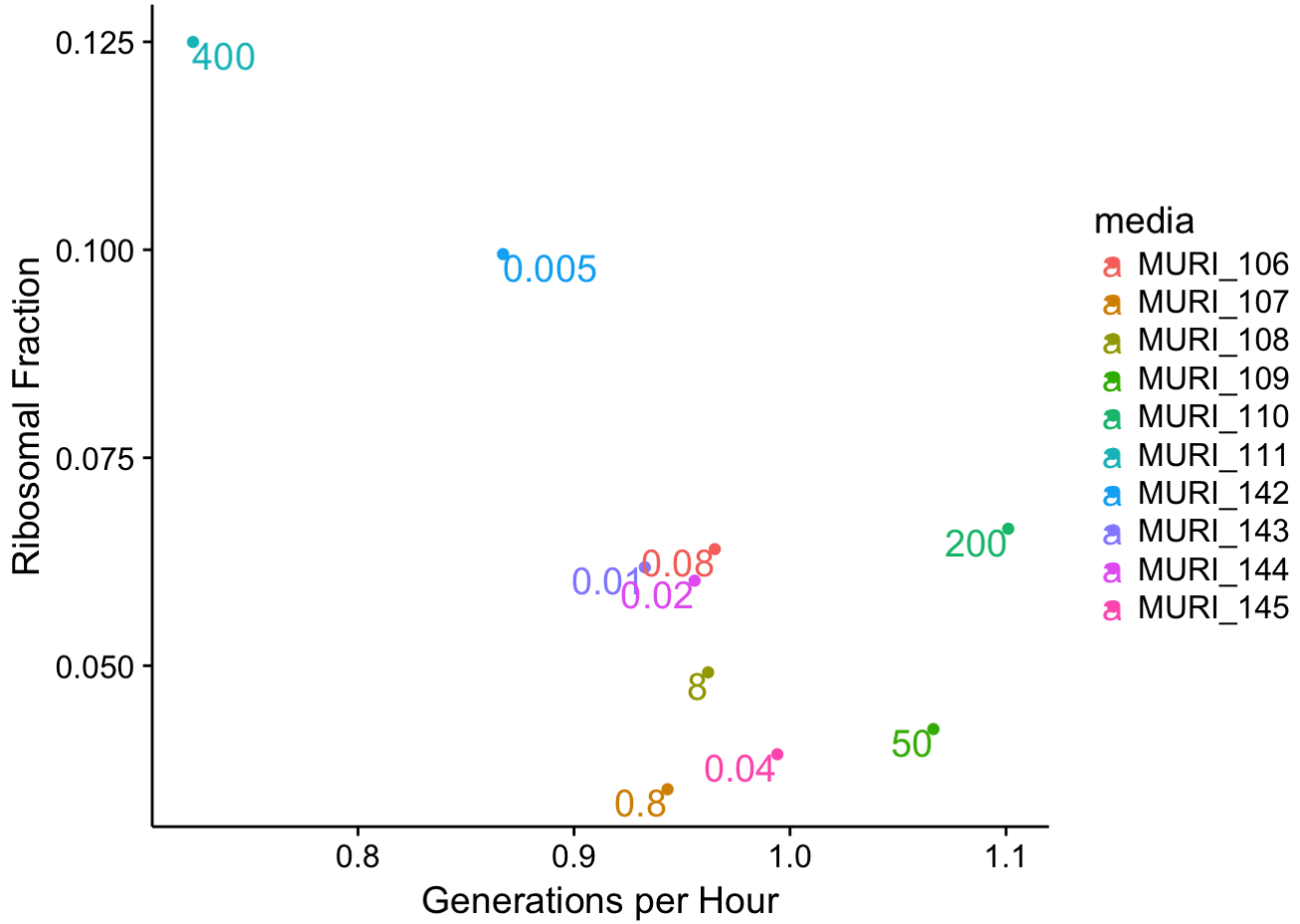



```
#non-distinct ribosomal fraction NaCl stress
UTmready %>% filter(experiment == 'NaCl_stress') %>% ungroup() -> UTma

UTma %>% ggplot(aes(x=generations_per_hour, y=fraction, color=media)) + xlab('Generations per Hour') + ylab('Ribo
somal Fraction') + geom_point() + geom_text(aes(label=Na_mM),hjust='inward', vjust=1 , size= 5)
```

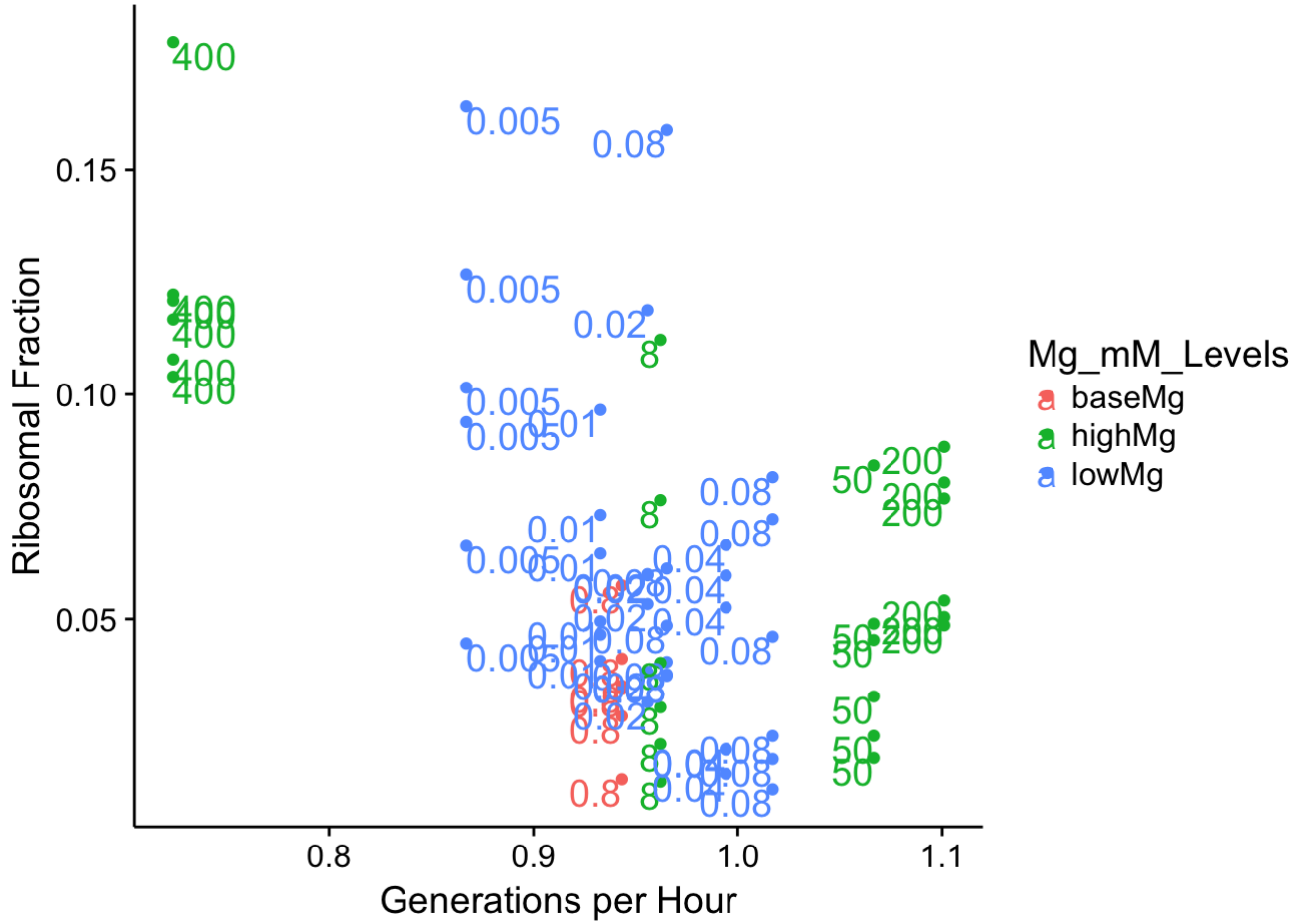


Mg concentrations and MgSO4 stress

```
#keep only ond samples on MgSO4 stress with unique Mg concentrations
UTmready %>% filter(experiment == 'MgSO4_stress_high'| experiment == 'MgSO4_stress_low') %>% ungroup() %>% distin
ct(Mg_mM, carbonSource,.keep_all = TRUE) -> UTmMg

UTmMg %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=media)) + xlab('Generations per Hour') +
ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=Mg_mM),hjust='inward', vjust=1 , size= 5)
```




```
#non-distinct ribosomal fraction MgSO4 stress
UTmready %>% filter(experiment == 'MgSO4_stress_high'| experiment == 'MgSO4_stress_low') %>% ungroup() -> UTmb

UTmb %>% ggplot(aes(x=generations_per_hour, y=fraction,color=Mg_mM_Levels )) + xlab('Generations per Hour') + yla
b('Ribosomal Fraction') + geom_point() + geom_text(aes(label=Mg_mM),hjust='inward', vjust=1 , size= 5)
```


Ribosomal Fraction
0.15
0.10
0.05
0.8
0.9
1.0
1.1
Generations per Hour
Mg_mM_Levels
baseMg
highMg
lowMg