Proteome Profiling

R Markdown

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

```r
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
```

```r
library(ggplot2)
library(cowplot)
```

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

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

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

resDf_protein_trT_set00_StcYtcNasAgrNgrMgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noMatchFilter_p1Sf_vst <- rea
d_csv("~/Desktop/Wilke Lab/ecoli_multiple_growth_conditions-master/a_results/resDf_protein_trT_set00_StcYtcNasAgr
NgrMgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noMatchFilter_p1Sf_vst.csv")
```

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

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   X1 = col_character()
## )
```

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

```
nameDictionary_RNA_Protein <- read_csv("~/Desktop/nameDictionary_RNA_Protein.csv")
```

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

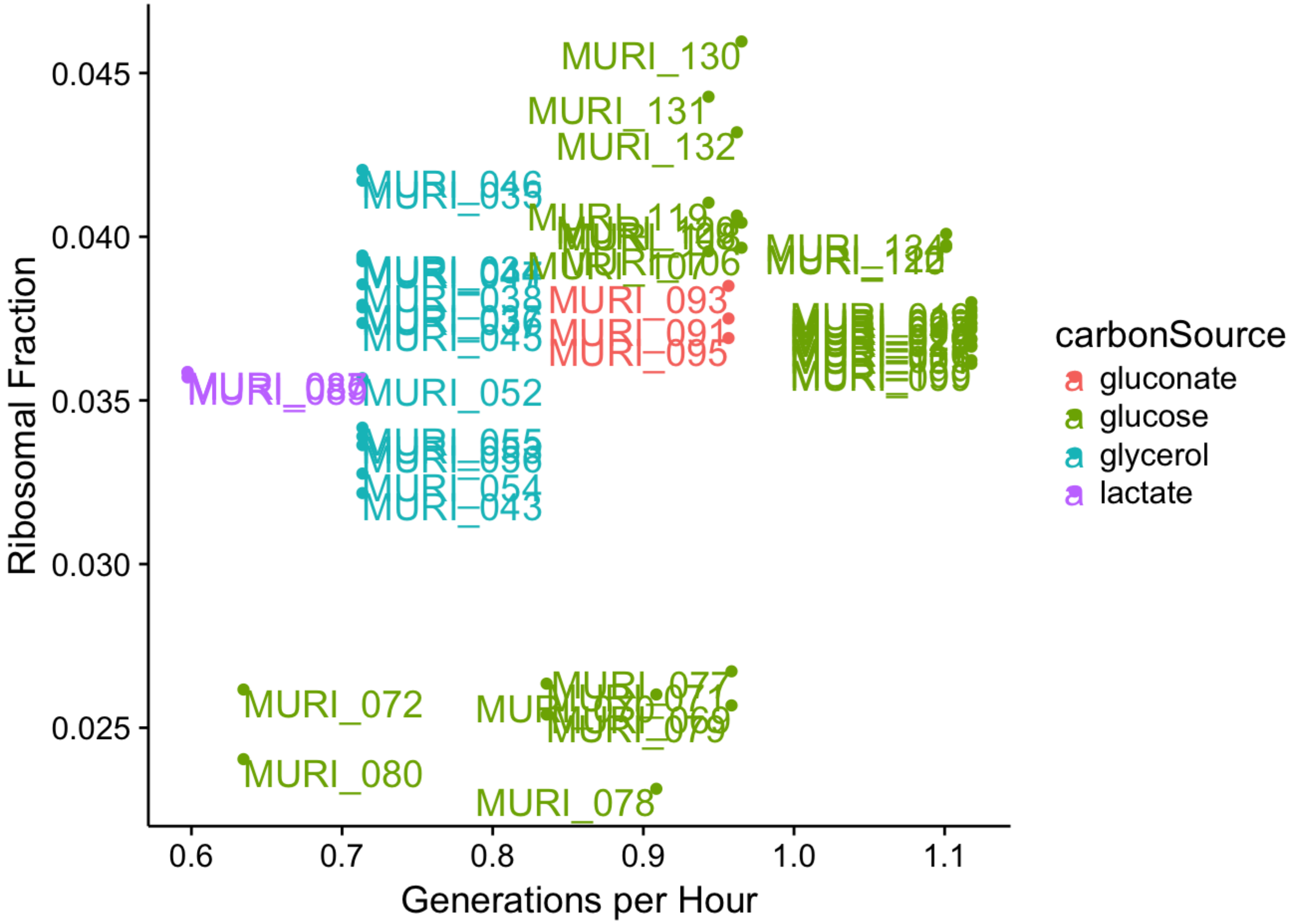
```
metaProtein <- read_csv("~/Desktop/Wilke Lab/ecoli_multiple_growth_conditions-master/a_results/metaProtein.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   growthTime_hr = col_double(),
##   cellTotal = col_double(),
##   cellsPerTube = col_double(),
##   Mg_mM = col_double(),
##   Na_mM = col_integer(),
##   doublingTimeMinutes = col_double(),
##   doublingTimeMinutes.95m = col_double(),
##   doublingTimeMinutes_95p = col_double(),
##   rSquared = col_double()
## )
## See spec(...) for full column specifications.
```

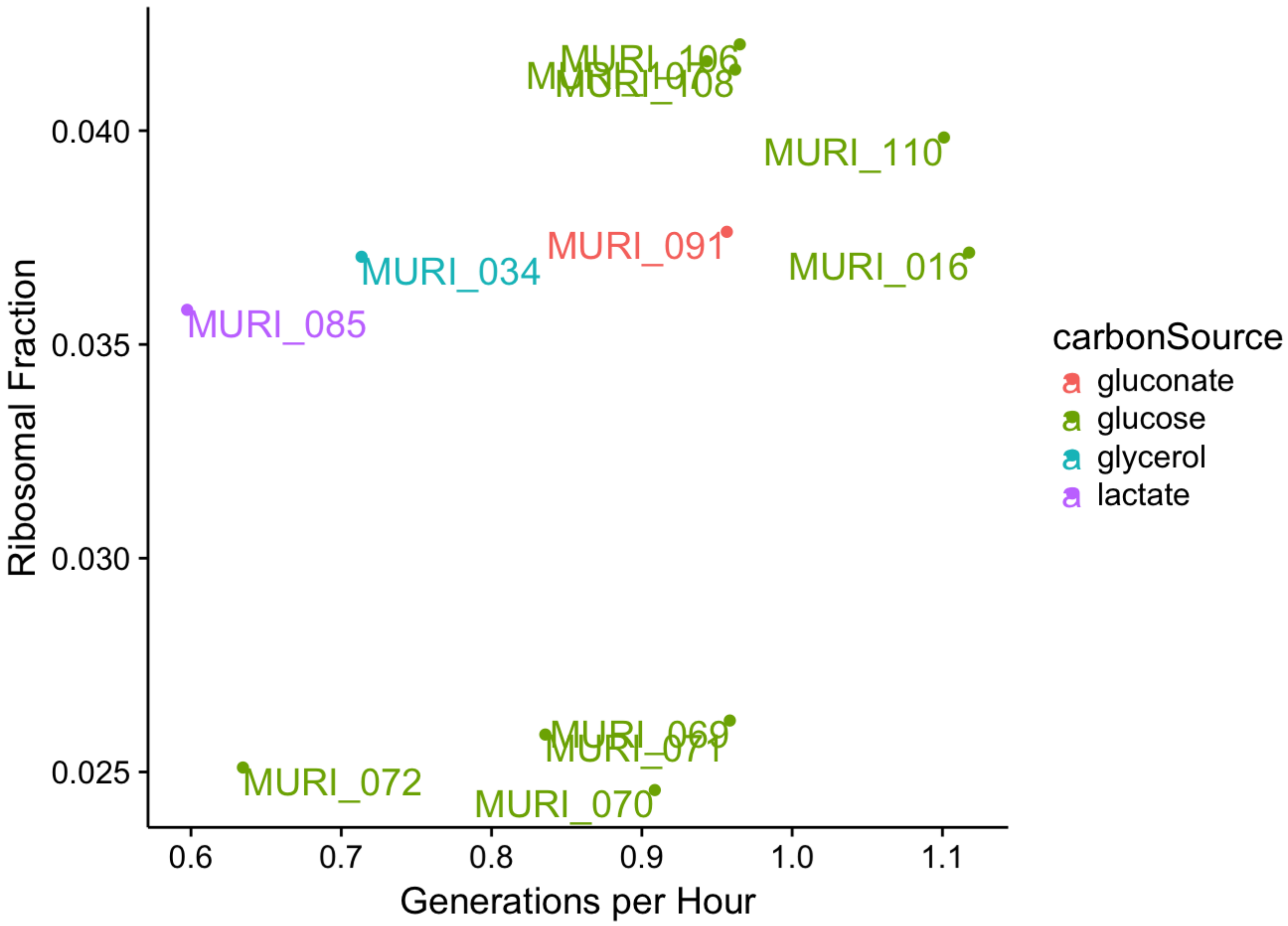
```
#change the name of columns on data frames to enable joining of tables
names(nameDictionary_RNA_Protein)[names(nameDictionary_RNA_Protein) == 'b#'] <- 'b'
names(nameDictionary_RNA_Protein)[names(nameDictionary_RNA_Protein) == 'Protein_id'] <- 'gene_id'
names(resDf_protein_trT_set00_StcYtcNasAgrNgrMgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noMatchFilter_p1Sf_vst)
[names(resDf_protein_trT_set00_StcYtcNasAgrNgrMgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noMatchFilter_p1Sf_vst)
 == 'X1'] <- 'gene_id'



#join normalized data to nameDictionary by gene_id
resDf_protein_trT_set00_StcYtcNasAgrNgrMgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noMatchFilter_p1Sf_vst %>% le
ft_join(nameDictionary_RNA_Protein, by = 'gene_id') %>% subset(select = -c(mRNA_ID, b)) %>% select(gene_id, gene_
name, everything())-> df

df %>% subset(select = (-c(gene_id))) -> df

#sum gene counts for each media, filter only ribosomal genes, sum ribosomal gene counts for each media, calculate
 ribosomal fraction
df %>% gather(media, count, MURI_016:MURI_140) %>% group_by(media) %>% mutate(total= sum(count)) %>% filter(str_d
etect(gene_name,regex('^rp'))) %>% group_by(media) %>% mutate(ribo = sum(count)) %>% mutate(fraction =
ribo/total)-> total_ribo

total_ribo %>% subset(select = -c(gene_name,count, total, ribo )) %>% distinct(fraction) -> fraction

#select growth condidtions and doubling times, and filter only exponential phase samples
metaProtein %>% select(growthPhase,carbonSource,
dataSet,experiment,growthTime_hr,doublingTimeMinutes,Mg_mM,Na_mM,Mg_mM_Levels,Na_mM_Levels) %>% filter(growthPhas
e == 'exponential') -> df2

#calculate generations per hour
df2 %>% mutate(generations_per_hour = ((growthTime_hr*60)/doublingTimeMinutes)/growthTime_hr) -> df2

MyData <- read_csv("~/Desktop/MyData.csv",
    col_types = cols(X1 = col_skip()))
```

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

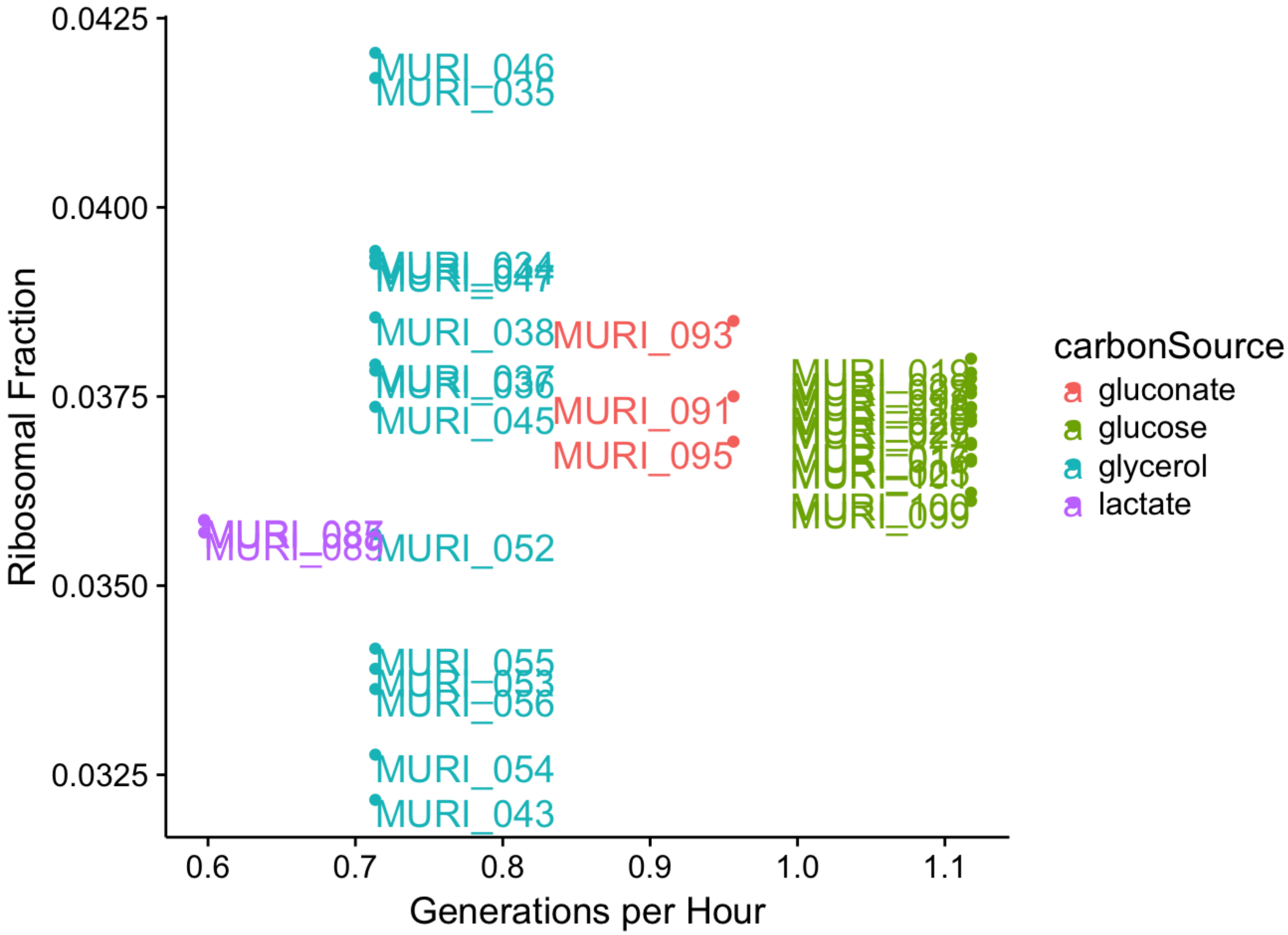
```
#join Mydata (includes growth conditions and generation times) and fraction by sample
fraction %>% left_join(MyData, by = 'media') %>% na.omit() %>% select(-fraction,fraction)-> graphready

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

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

#plot of fractions (non-averaged)
graphready %>% 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)
```



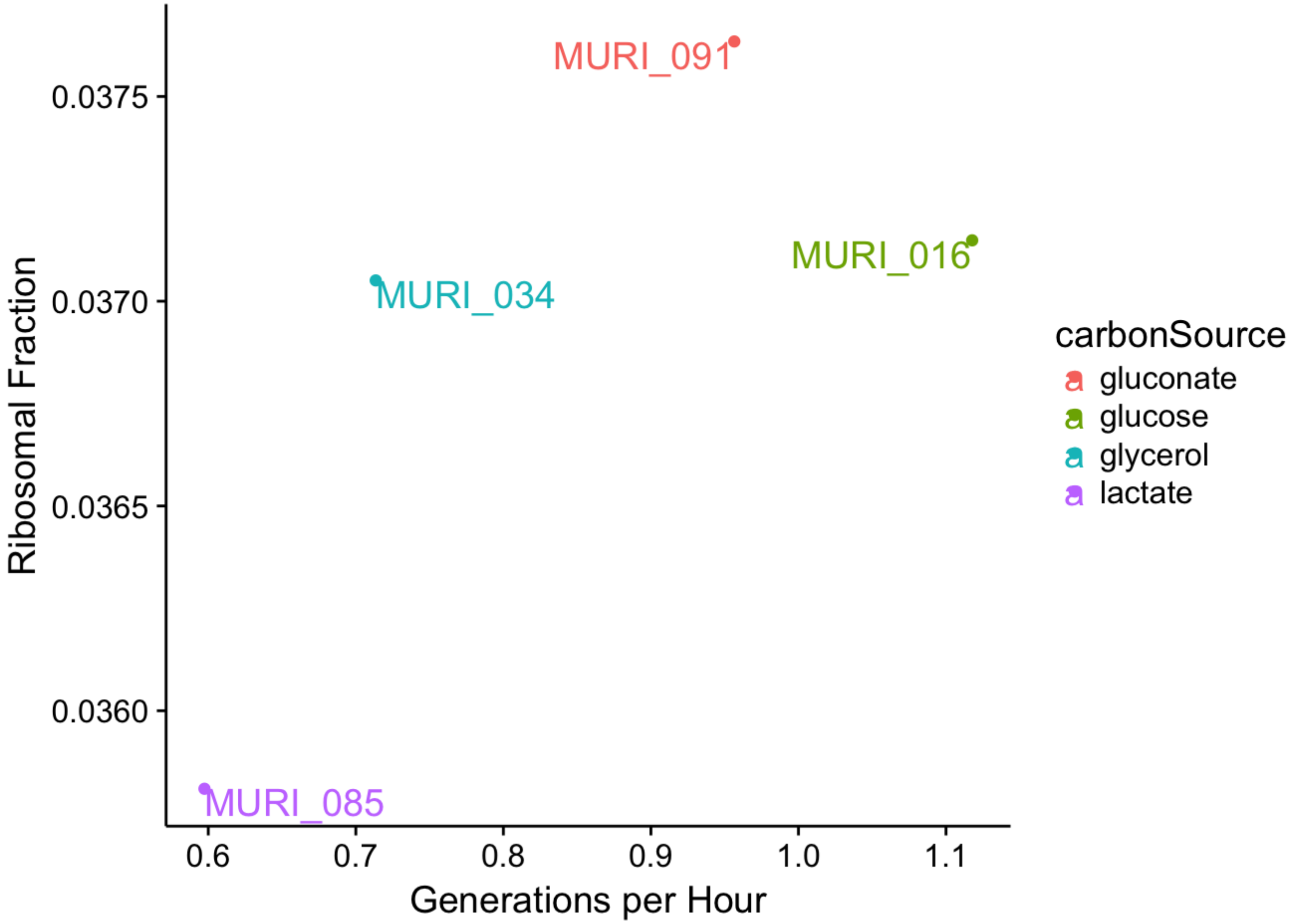
```
#plot of unique fraction averages
final %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per Hour')
+ ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```



```
#get rid of NaCl and MgSO4 stress
final_alt <- final[!grepl("stress", final$experiment),]
#get rid of NaCl and MgSO4 stress (non-unique fraction_avg)
final_alt2 <- graphready[!grepl("stress", graphready$experiment),]

final_alt2 %>% 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)
```

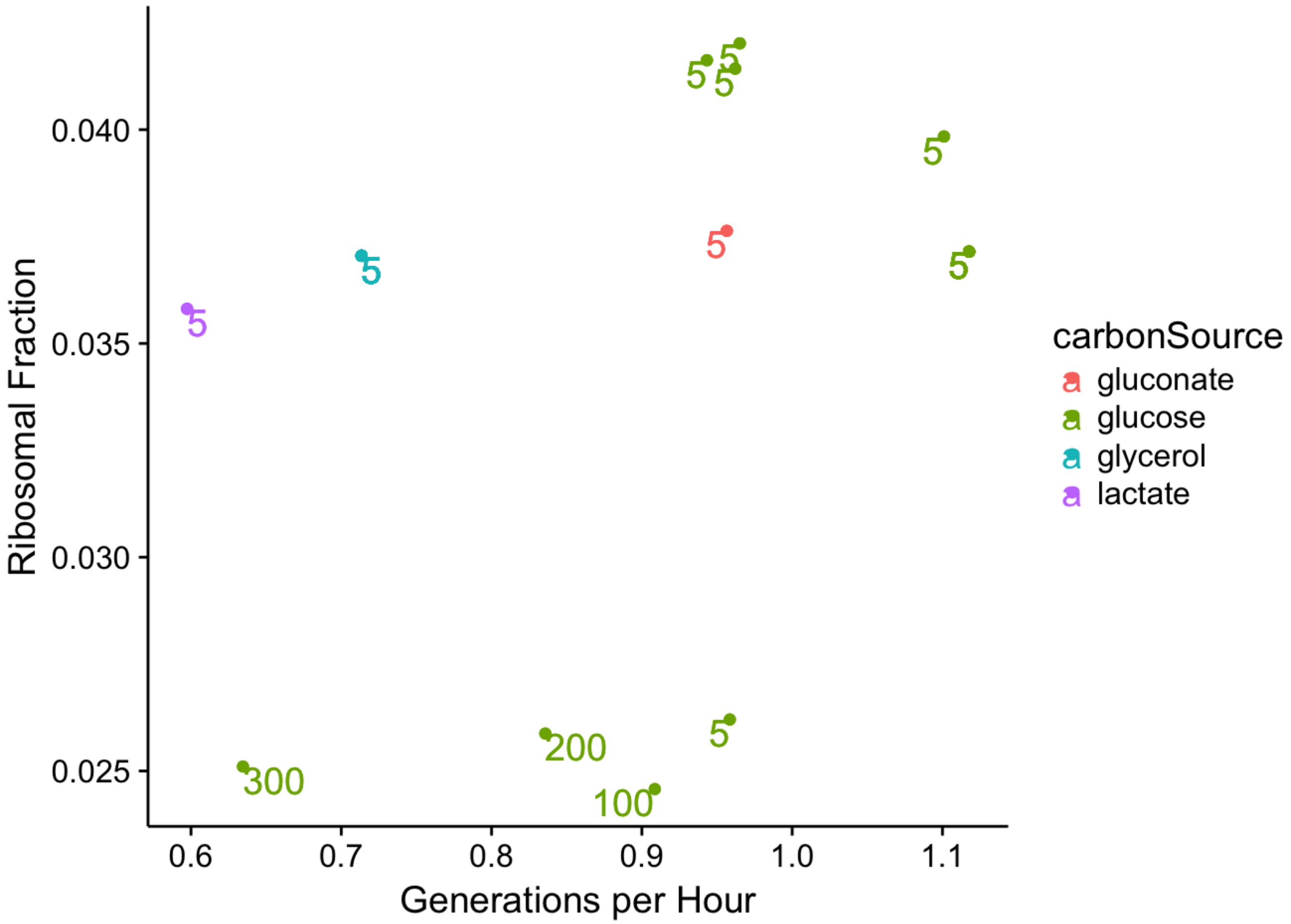


```
final_alt %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per Hou
r') + ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```

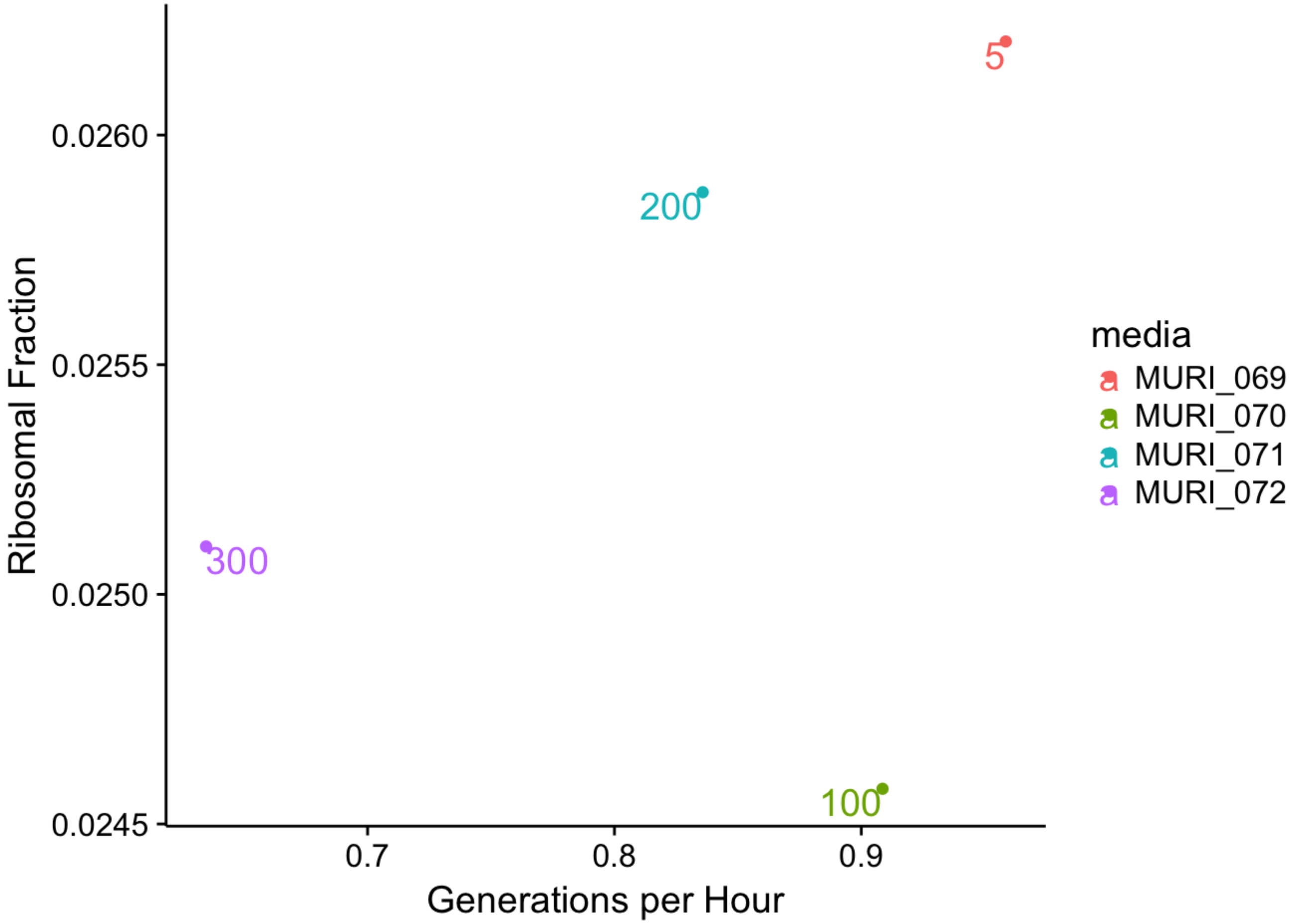


Na concentrations and NaCl stress

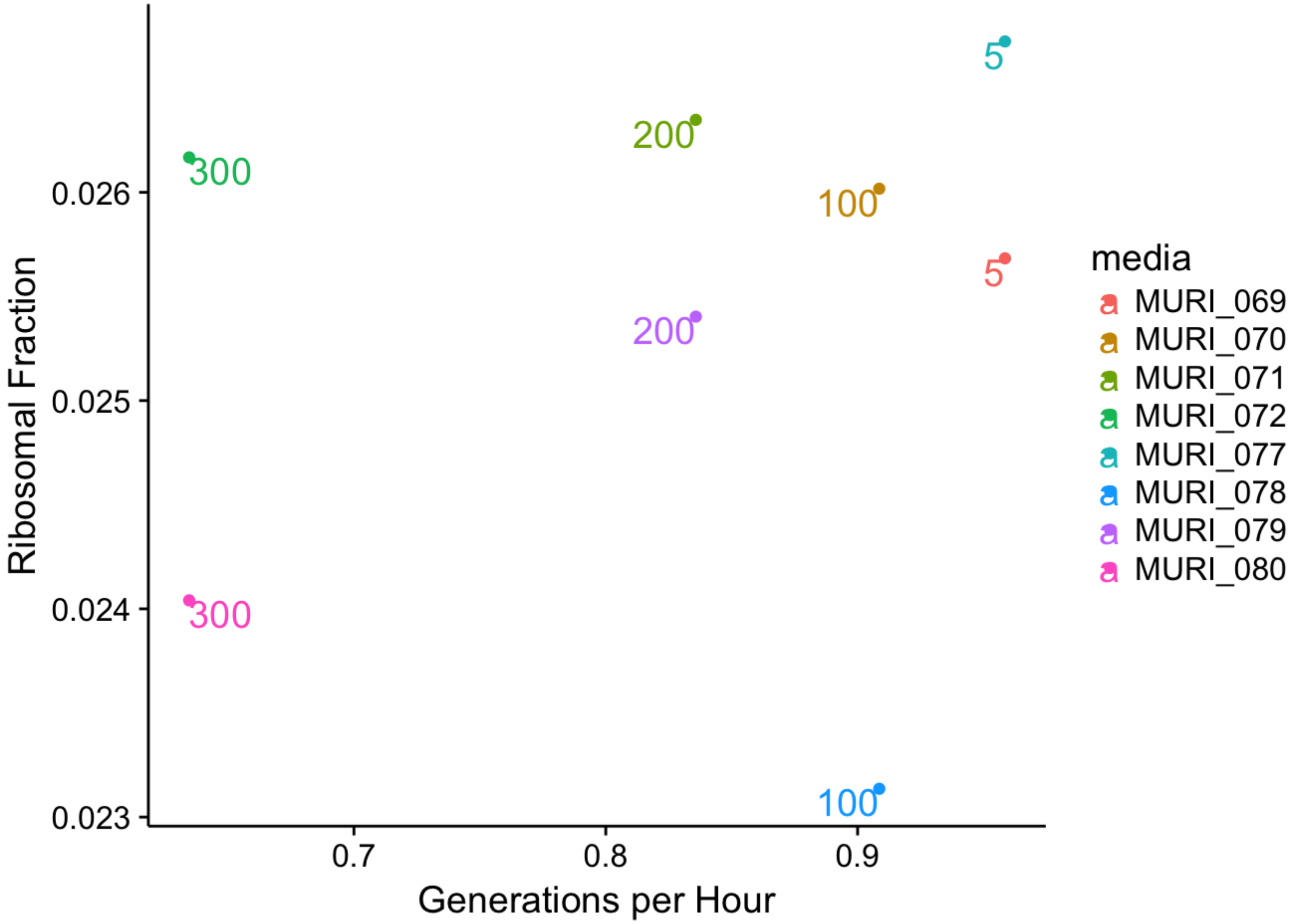
```
#keep only ond samples on NaCL stress with unique Na concentrations (5,100,200,300)
graphready %>% filter(experiment == 'NaCl_stress') %>% ungroup() %>% distinct(Na_mM, carbonSource,.keep_all = TRU
E) -> Na_final

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



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

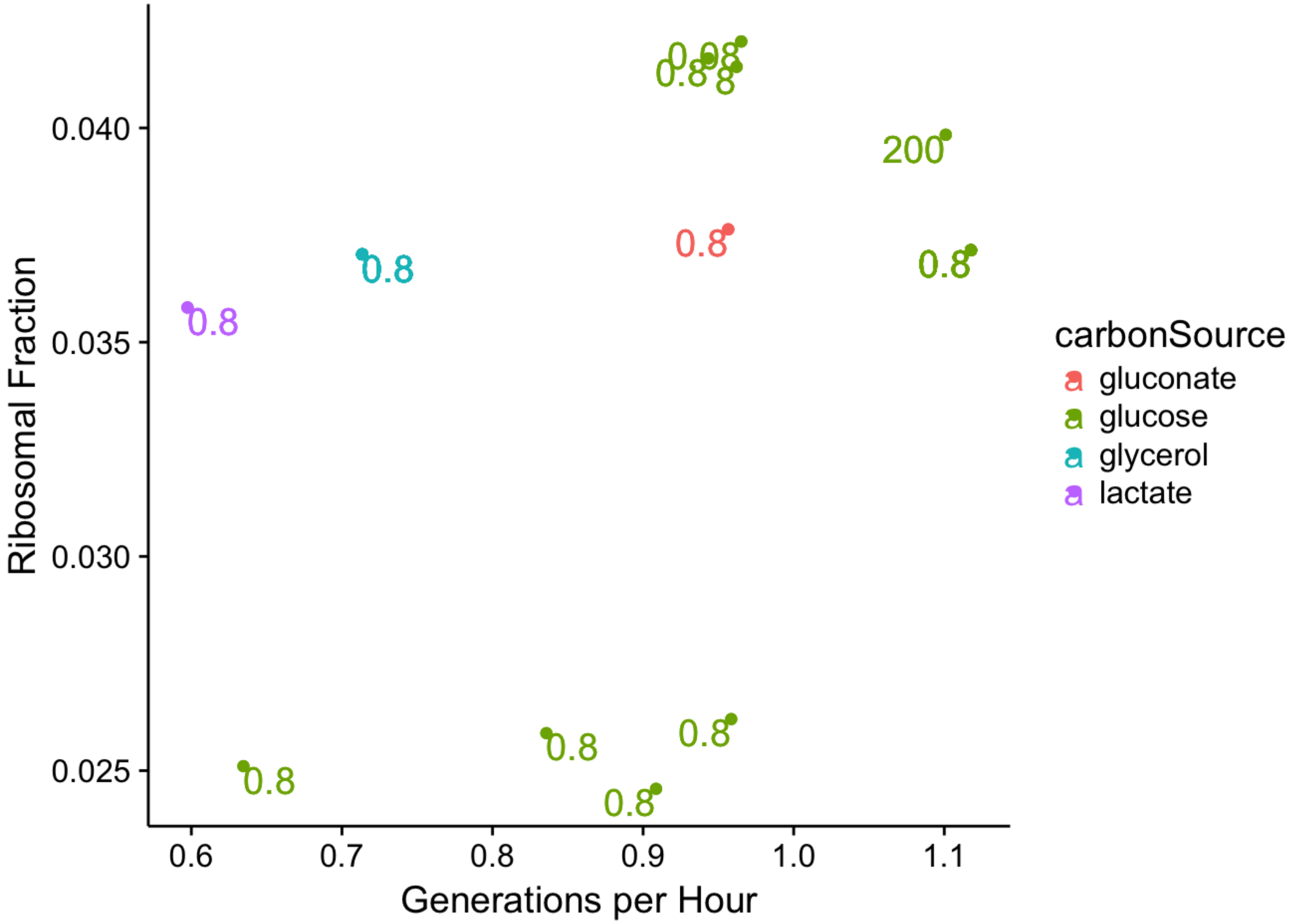


```
#non-distinct ribosomal fraction NaCl stress
graphready %>% filter(experiment == 'NaCl_stress') %>% ungroup() -> a

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

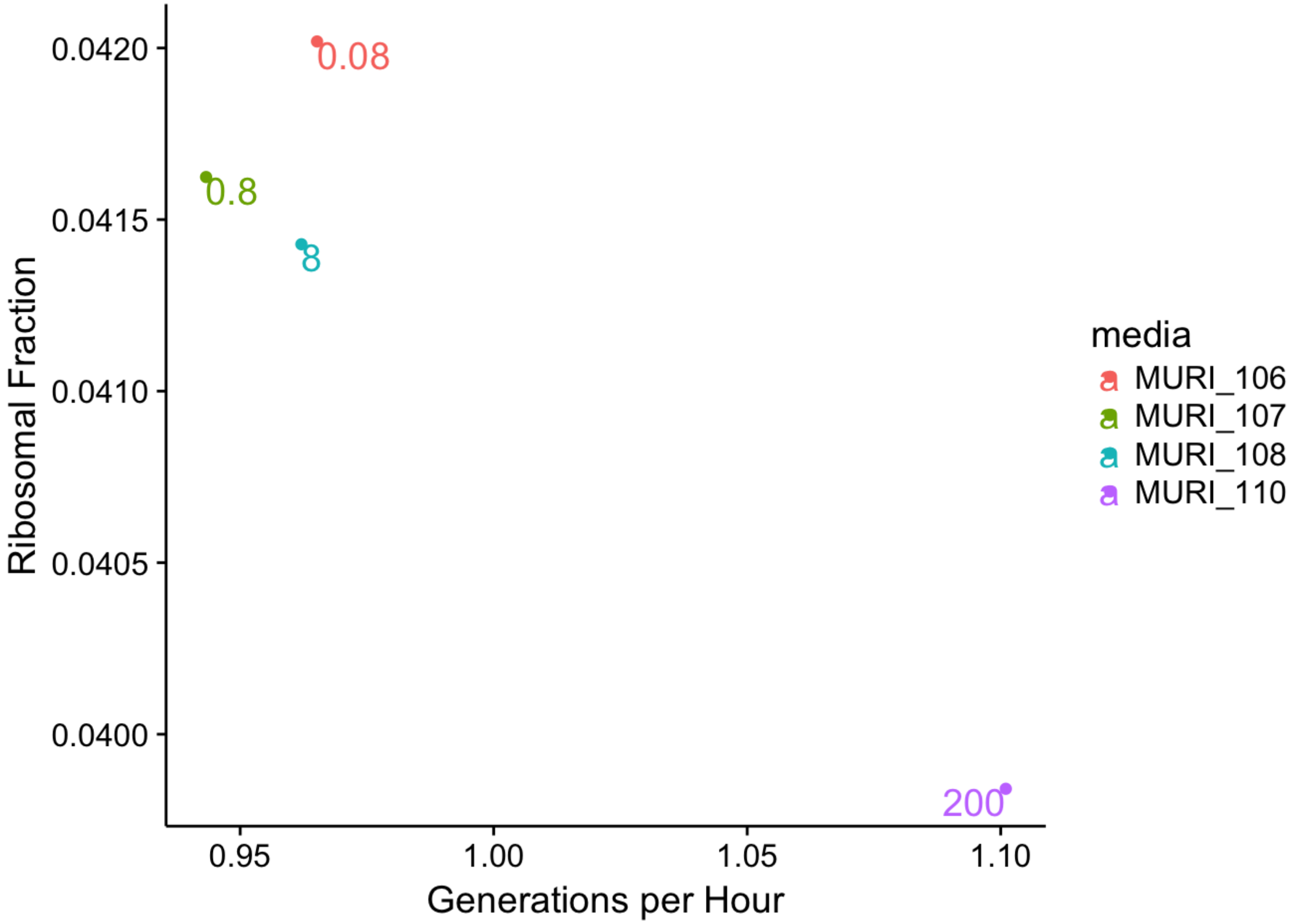


Mg concentrations and MgSO4 stress

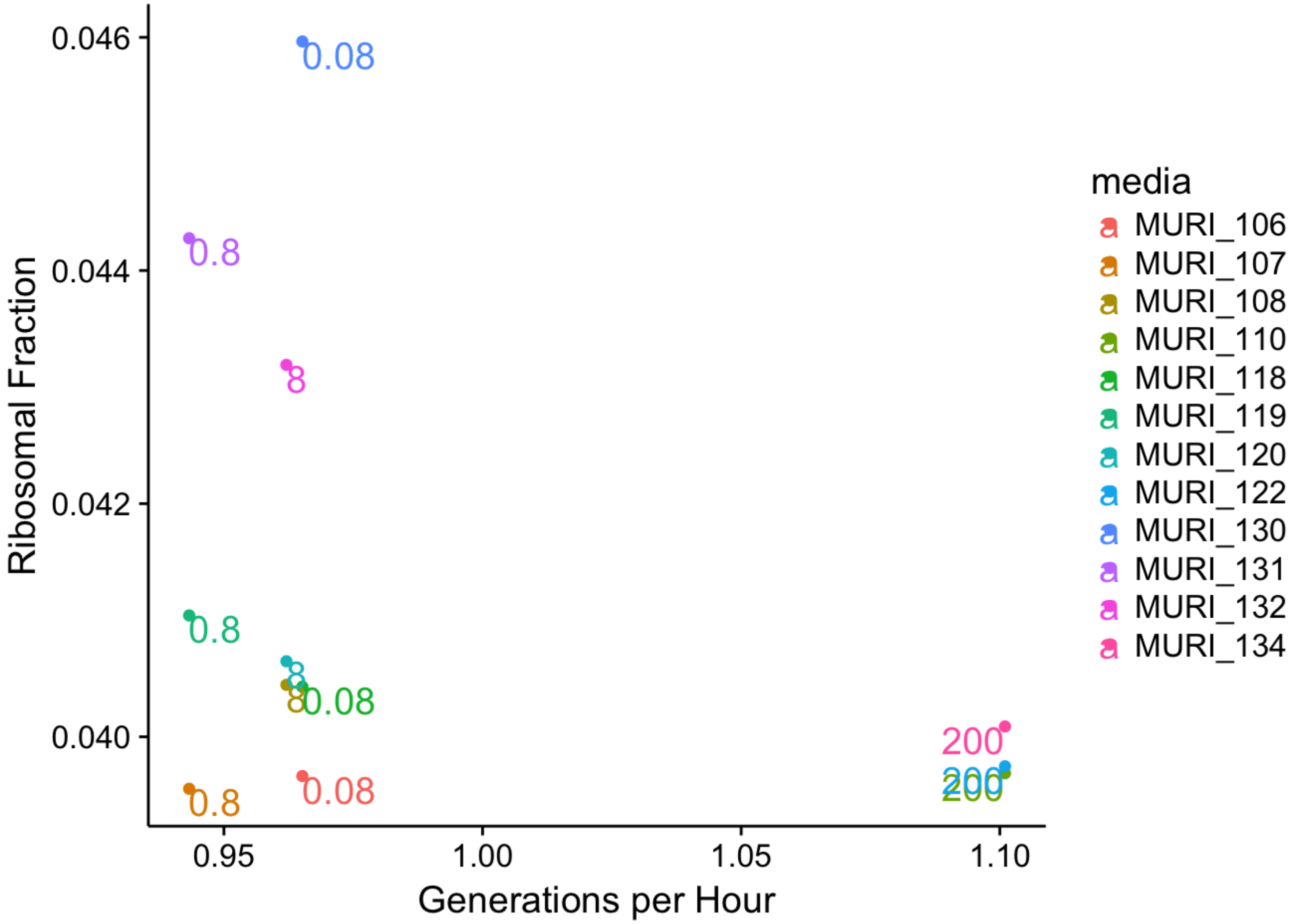
```
#keep only ond samples on MgSO4 stress with unique Mg concentrations (0.08,0.8,8,200)
graphready %>% filter(experiment == 'MgSO4_stress_high') %>% ungroup() %>% distinct(Mg_mM, carbonSource,.keep_all
 = TRUE) -> Mg_final

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



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



```
#non-distinct ribosomal fraction MgSO4 stress
graphready %>% filter(experiment == 'MgSO4_stress_high') %>% ungroup() -> b

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



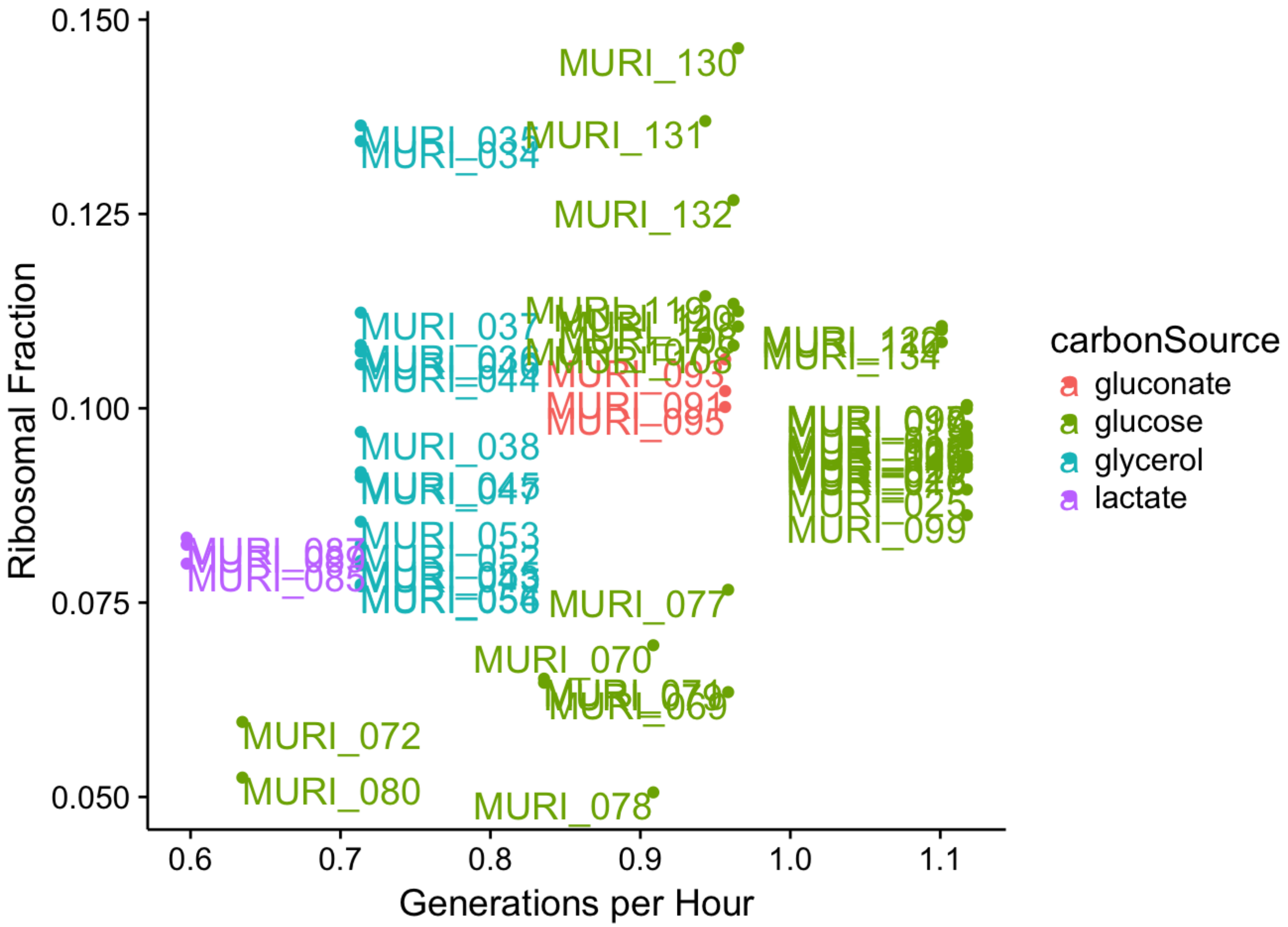
The following code is used to calculate and plot the ribosomal fraction of samples during exponential phase using untransformed data

```
Untransformed <- read_csv("~/Desktop/GitHub/proteome-profiling-ecoli/umut/resDf_protein_trT_set00_StcYtcNasAgrNgr
Mgh_SYAN_baseMgAllMg_baseNaAllNa_ExpAllPhase_noFilter_p1Sf_noNorm.csv")
```

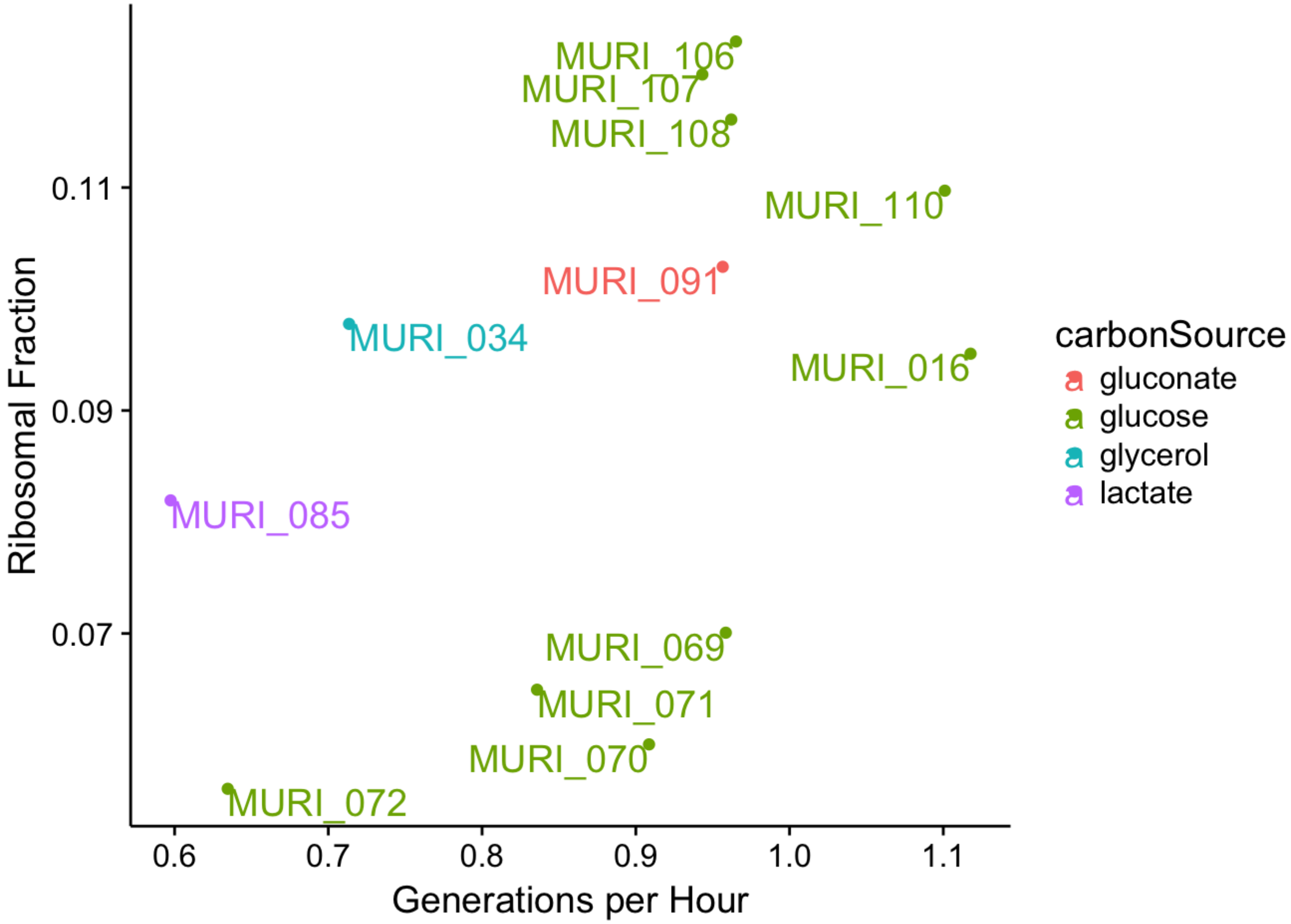
```
## Warning: Missing column names filled in: 'X1' [1]
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   X1 = col_character()
## )
```

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

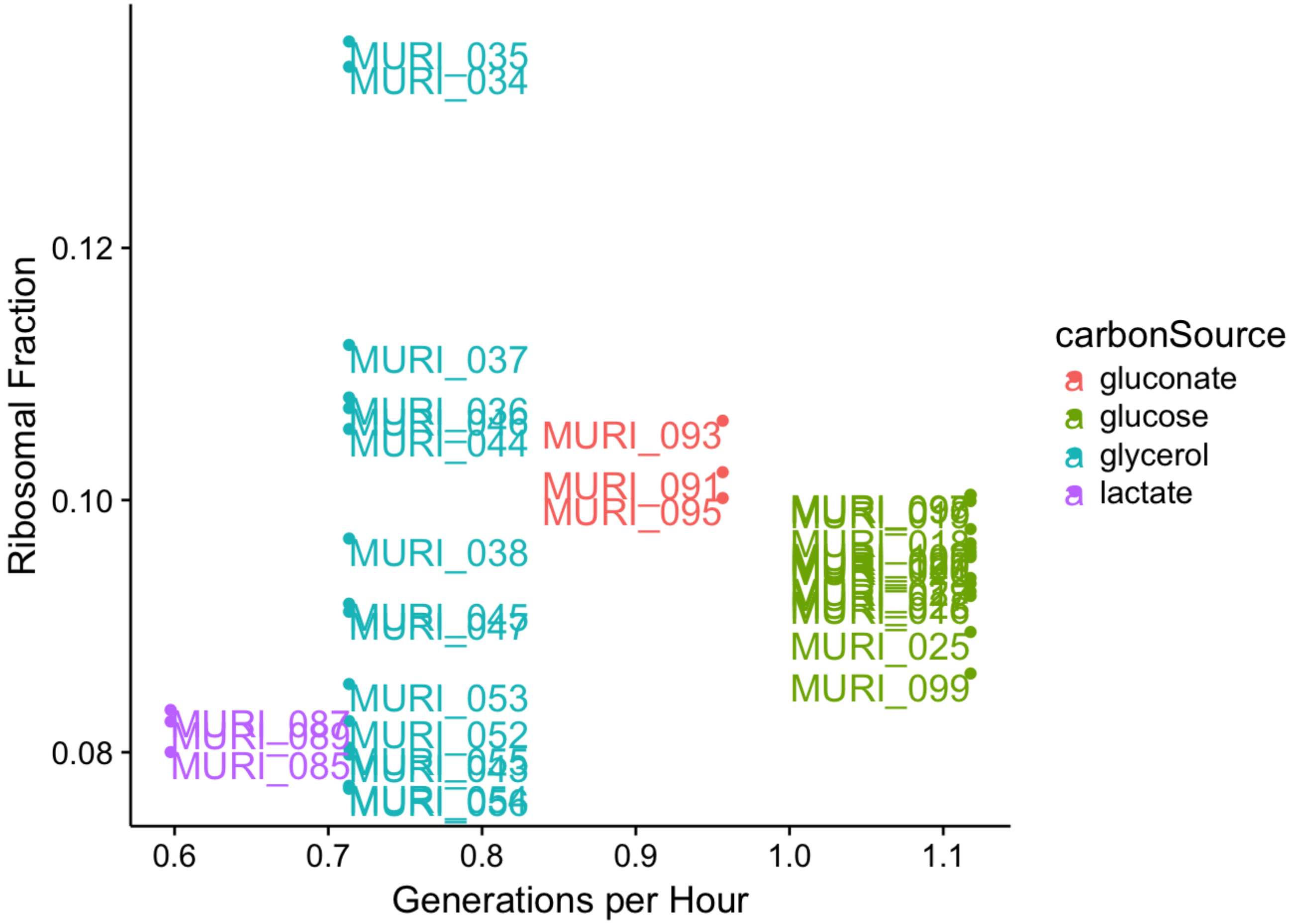
```
names(Untransformed)[names(Untransformed) == 'X1'] <- 'gene_id'

#join untransformed data to nameDictionary by gene_id
Untransformed %>% left_join(nameDictionary_RNA_Protein, by = 'gene_id') %>% subset(select = -c(mRNA_ID, b)) %>% s
elect(gene_name, everything())-> UTnames

UTnames %>% subset(select = (-c(gene_id))) -> UTnames

#sum gene counts for each media, filter only ribosomal genes, sum ribosomal gene counts for each media, calculate
 ribosomal fraction
UTnames %>% gather(media, count, MURI_016:MURI_140) %>% group_by(media) %>% mutate(total= sum(count)) %>%
filter(str_detect(gene_name,regex('^rp'))) %>% group_by(media) %>% mutate(ribo = sum(count)) %>% mutate(fraction
= ribo/total)-> UTtotalribo

UTtotalribo %>% subset(select = -c(gene_name,count, total, ribo )) %>% distinct(fraction) -> UTfraction

#join Mydata (includes growth conditions and generation times) and fraction by sample
UTfraction %>% left_join(MyData, by = 'media') %>% na.omit() %>% select(-fraction,fraction)-> UTready

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

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

#plot of fractions
UTready %>% 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)
```



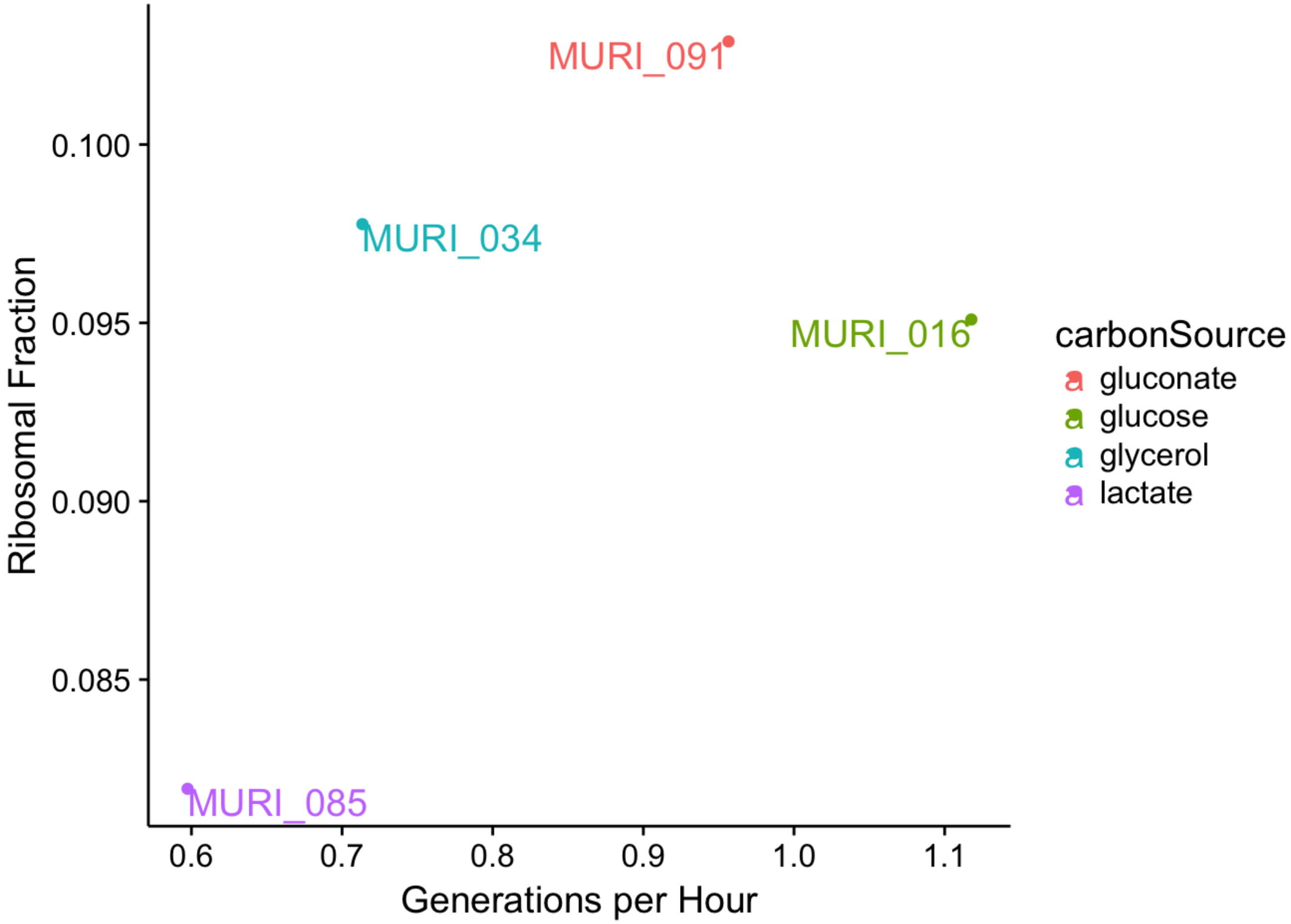
```
#plot of fraction averages
UTfinal %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per
Hour') + ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size=
5)
```



```
#get rid of NaCl and MgSO4 stress
UTfinal_alt <- UTfinal[!grepl("stress", UTfinal$experiment),]
#get rid of NaCl and MgSO4 stress (non-unique fraction_avg)
UTfinal_alt2 <- UTready[!grepl("stress", UTready$experiment),]

UTfinal_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)
```

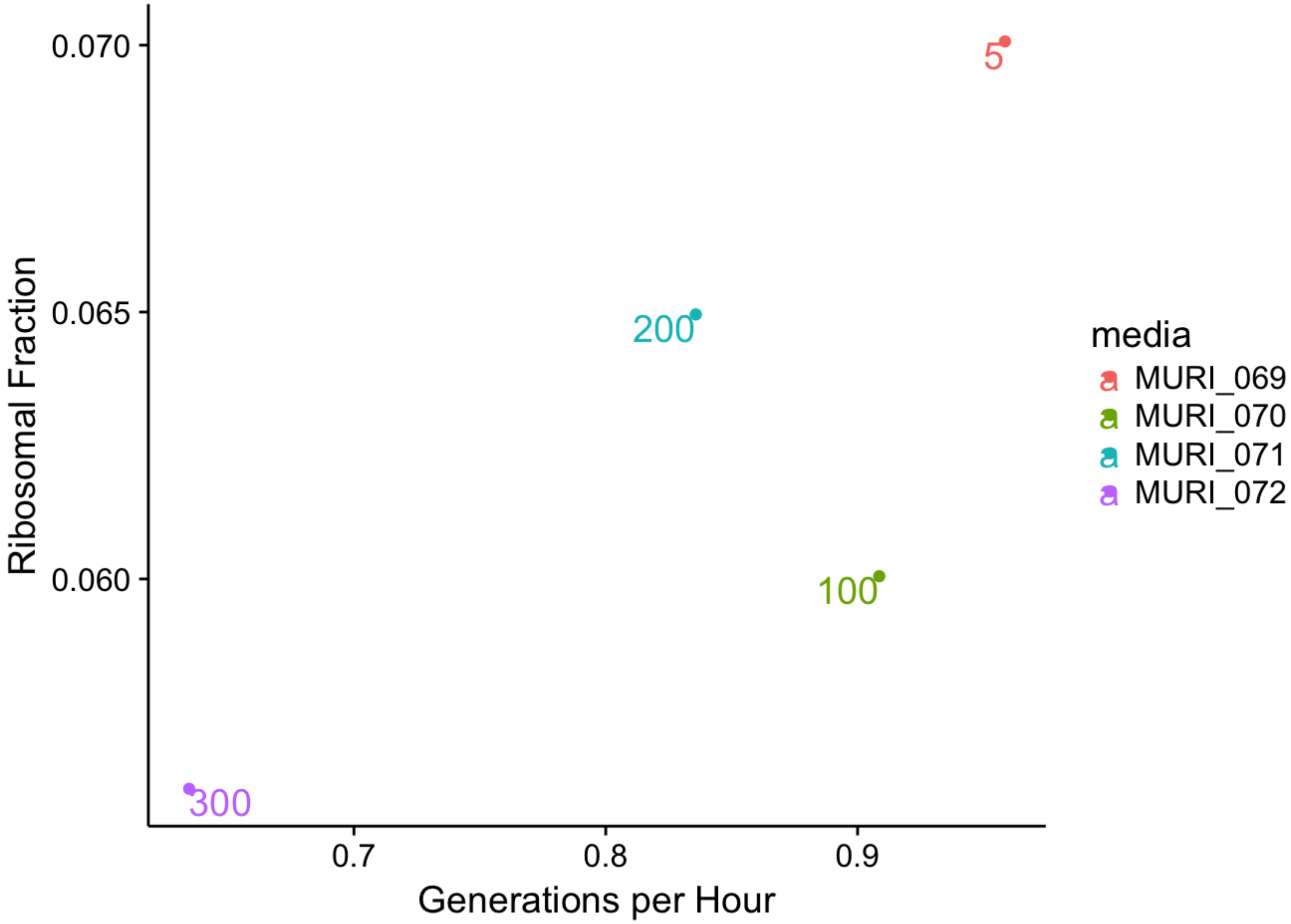


```
UTfinal_alt %>% ggplot(aes(x=generations_per_hour, y=fraction_avg, color=carbonSource)) + xlab('Generations per H
our') + ylab('Ribosomal Fraction') + geom_point() + geom_text(aes(label=media),hjust='inward', vjust=1 , size= 5)
```

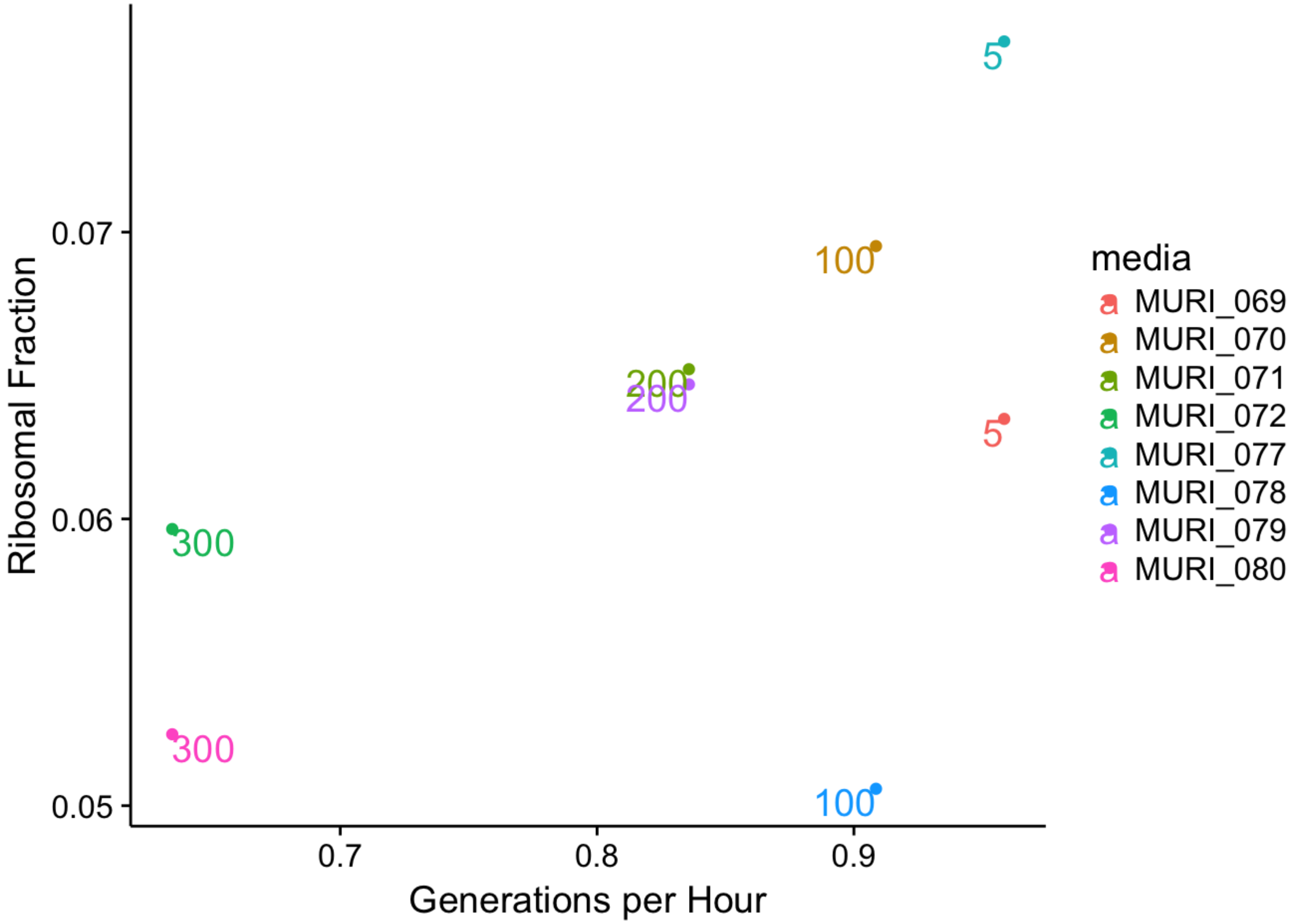


Na concentrations and NaCl stress with untransformed data

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

UTNa %>% 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)
```

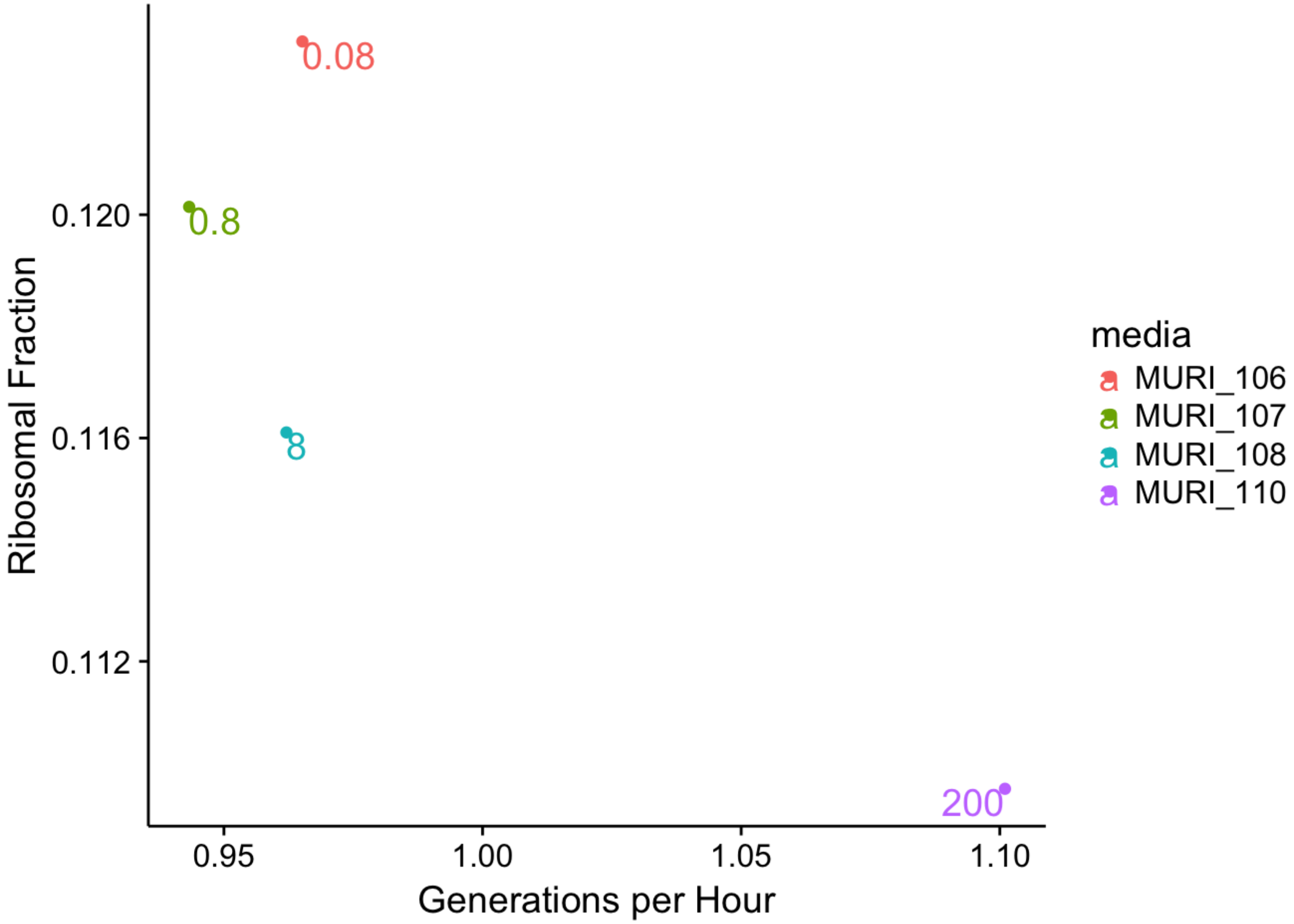


```
#non-distinct ribosomal fraction NaCl stress
UTready %>% filter(experiment == 'NaCl_stress') %>% ungroup() -> UTa

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

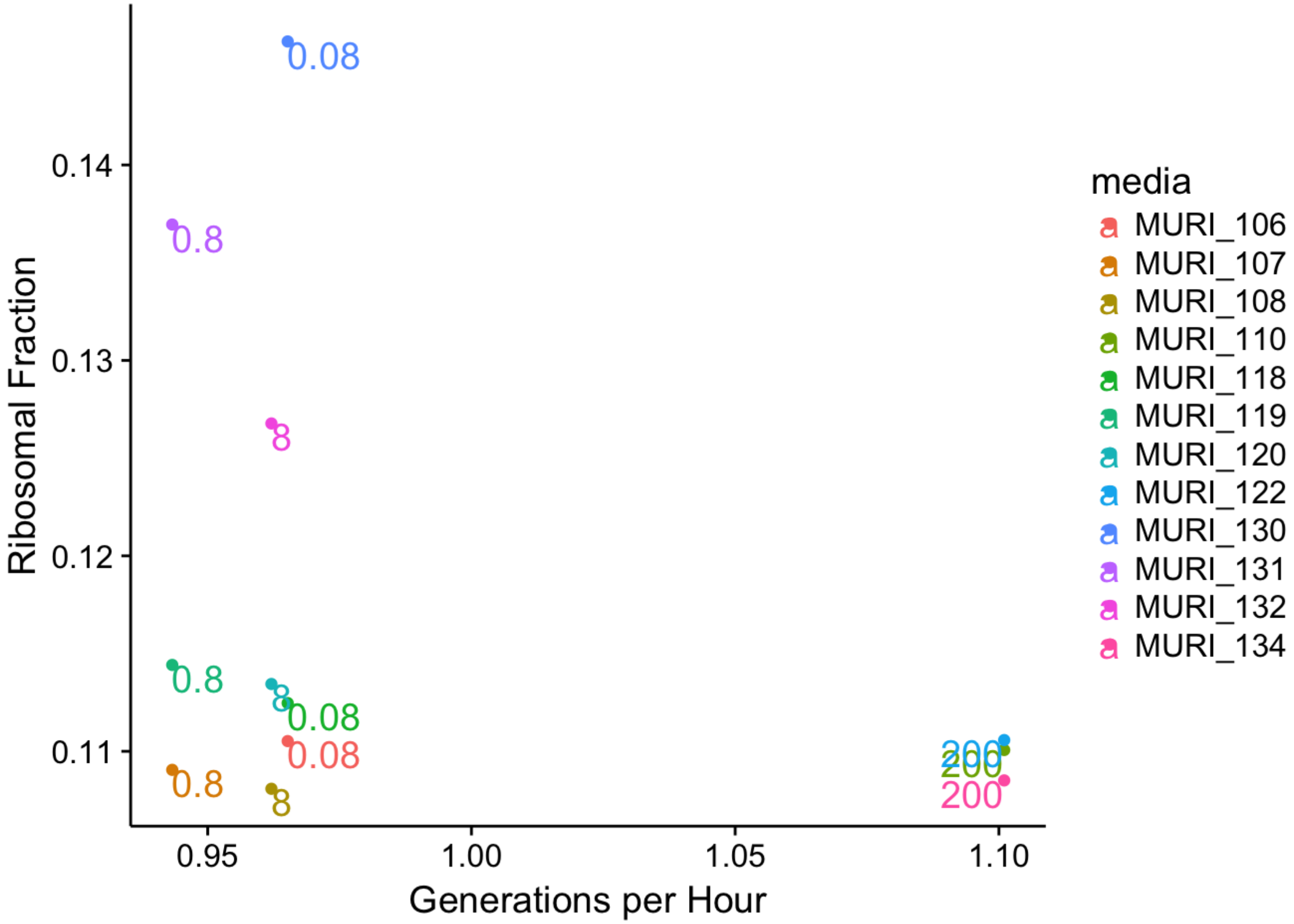


Mg concentrations and MgSO4 stress

```
#keep only ond samples on MgSO4 stress with unique Mg concentrations (0.08,0.8,8,200)
UTready %>% filter(experiment == 'MgSO4_stress_high') %>% ungroup() %>% distinct(Mg_mM, carbonSource,.keep_all =
TRUE) -> UTMg

UTMg %>% 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
UTready %>% filter(experiment == 'MgSO4_stress_high') %>% ungroup() -> UTb

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


0.08
0.14
0.8
0.13
8
0.12
0.8
8
0.08
0.11
0.08
0.8
8
200
200
200
Ribosomal Fraction
0.95
1.00
1.05
1.10
Generations per Hour
media
a MURI_106
a MURI_107
a MURI_108
a MURI_110
a MURI_118
a MURI_119
a MURI_120
a MURI_122
a MURI_130
a MURI_131
a MURI_132
a MURI_134