

# LOH\_H2O2\_20160121\_summary

*h qin*

*January 21, 2016*

20160121, start to port old R codes into a markdown file.

2013 Dec 9, using results from `_3b.2013Dec9.summarizeFitting.H2O2LOH.R`

2013Nov25, use summarized results from `_3.*R`, which contains standard deviations.

2008 CLS data uses M14, 2013H2O2 uses SGU57, so only 10 strains overlaps

Use 2013 fitting results and gnls results: 2013Nov14, use correlated regression; Reverse L0 axis in plot, 2012 July 30

2012Feb25,  $T_g$ .vs. $T_c \sim \ln(R_0) + G$

Partial correlations are all negative, what does this mean? Of course, they should be negative correlations.

$T_g/T_c$  is a measure of ability to maintain recombination rate during aging

Load 2013 H2O2-LOH results

```
list.files(pattern="csv", path='output')
```

```
## [1] "LOHH202_all_20131127.csv"          "LOHH202_averaged20131210_v1.csv"
```

```
tb2 = read.csv("output/LOHH202_averaged20131210_v1.csv")
```

Load previous LOH-CLS results, Qin Plos One 2008

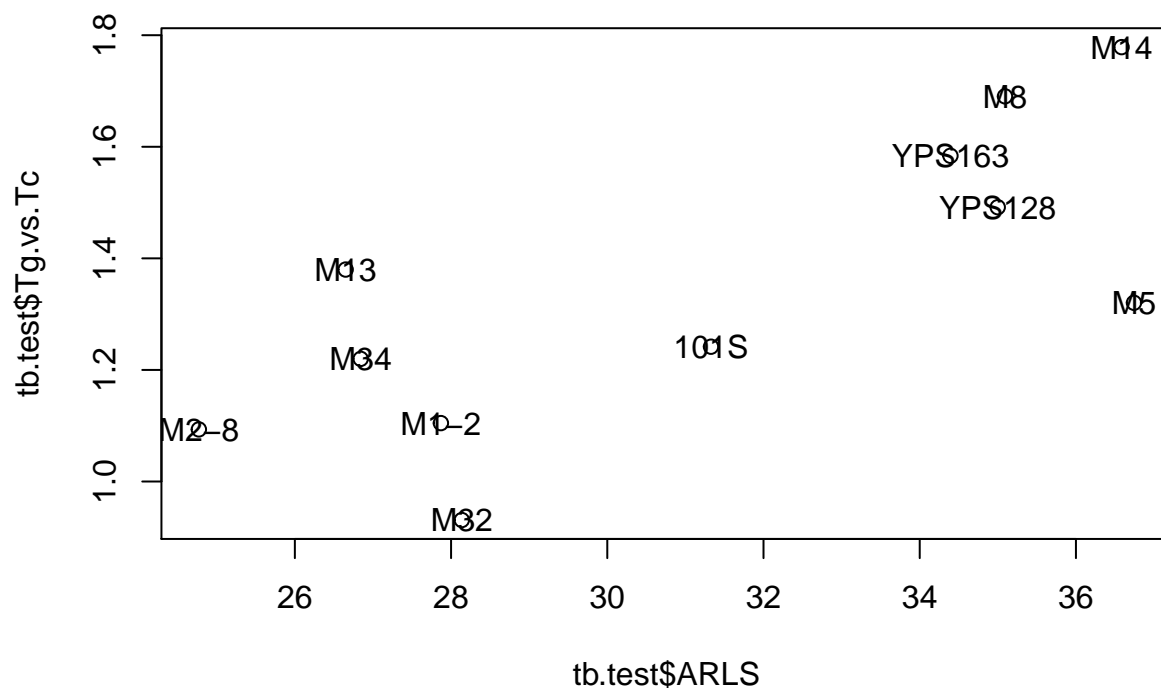
```
#tb = read.table("summary.new.by.strain.csv", header=T, sep="\t");
tb = read.table("021307.summary.by.strain.csv", header=T, sep="\t");
tb.old = tb;
labels = names( tb.old );
tb = tb.old[, c("strain","ARLS","R0","G","CLS","Tc", "Tg","Tmax","Tbmax", "Td", "Tdmax","TLmax","Lmax",
"b.max", "b.min", "strains", "L0.all", "L0.small" , "Pbt0","Pb0.5t0", "Pbt0.b") ];
tb$CLS.vs.Tc = tb$CLS / tb$Tc;
tb$Tg.vs.Tc = tb$Tg / tb$Tc;
tb$strain = as.character(tb$strain)
tb.old = tb;
tb = tb.old[1:13,] #remove rad52DD
tb.test = tb[1:11,]
```

```
summary(lm(Tg.vs.Tc ~ ARLS, data=tb.test)) #re-run the old results, just to double-check
```

```
##
## Call:
## lm(formula = Tg.vs.Tc ~ ARLS, data = tb.test)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28096 -0.10360  0.03093  0.13209  0.23406
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.03702   0.41283  -0.090  0.93052
## ARLS         0.04439   0.01310   3.388  0.00803 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1857 on 9 degrees of freedom
## Multiple R-squared:  0.5605, Adjusted R-squared:  0.5117
## F-statistic: 11.48 on 1 and 9 DF, p-value: 0.008026
```

```
plot( tb.test$Tg.vs.Tc ~ tb.test$ARLS )
text(tb.test$ARLS, tb.test$Tg.vs.Tc, tb.test$strains)
```



```
summary(lm( 1/Tg.vs.Tc ~ ARLS, data=tb.test)) #good, negative, p=0.012
```

```
##
## Call:
## lm(formula = 1/Tg.vs.Tc ~ ARLS, data = tb.test)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15828 -0.06543 -0.01532  0.04604  0.22840
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.54881   0.25270   6.129 0.000173 ***
## ARLS        -0.02499   0.00802  -3.115 0.012408 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.1137 on 9 degrees of freedom
## Multiple R-squared: 0.5189, Adjusted R-squared: 0.4654
## F-statistic: 9.706 on 1 and 9 DF, p-value: 0.01241
```

Load exg06 data

```
r nat = read.table("062705.rls.cls.tab", sep="\t", header=T, colClasses=c("character",
rep(NA,4)) );
```

Merge tb and tb2, and do some regression.

```
tb2$strain = tb2$strains
#tb3a = merge(tb, tb2, by='strain', all=T)
tb3 = merge(tb, tb2, by='strain')
tb3 = tb3[ , -(grep('strains', names(tb3))) ]

#remove BY4743, will increae p-value
#tb3 = tb3[ tb3$strain != "BY4743", ]
summary(lm(tb3$Cv.vs.Cb ~ tb3$ARLS)) #p 0.078
```

```
##
## Call:
## lm(formula = tb3$Cv.vs.Cb ~ tb3$ARLS)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.35944 -0.58570 -0.03378  0.17894  1.83951
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.07933    1.78522   2.845  0.0174 *
## tb3$ARLS     -0.11426    0.05833  -1.959  0.0786 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8739 on 10 degrees of freedom
## Multiple R-squared: 0.2773, Adjusted R-squared: 0.205
## F-statistic: 3.837 on 1 and 10 DF, p-value: 0.07859
```

```
summary(lm(tb3$Cb.vs.Cv ~ tb3$ARLS)) #p 0.039
```

```
##
## Call:
## lm(formula = tb3$Cb.vs.Cv ~ tb3$ARLS)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.8629 -0.2748 -0.1479  0.3616  0.8908
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.60720    1.07869  -1.490  0.1671
```

```
## tb3$ARLS      0.08356    0.03524    2.371    0.0392 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.528 on 10 degrees of freedom
## Multiple R-squared:  0.3599, Adjusted R-squared:  0.2958
## F-statistic: 5.622 on 1 and 10 DF,  p-value: 0.03921
```

```
#summary(lm( tb3$ARLS ~ tb3$CvManu * tb3$CbManu ))
```

Regression analysis

```
pTb = 1: length(tb3[1,])
names(pTb) = names(tb3)
for( j in c(2:41) ) {
  #m = lm( tb3a[, j] ~ tb3a$ARLS, na.rm=T)
  m = lm( tb3[, j] ~ tb3$Cb.vs.Cv )
  #m = lm( tb3a[, j] ~ tb3a$CvMean, na.rm=T)
  #m = lm( tb3a[, j] ~ tb3a$CbMean, na.rm=T)
  sm = summary(m)
  pTb[j] = 1 - pf(sm$fsta[1], sm$fsta[2], sm$fsta[3])
}
```

```
## Warning in summary.lm(m): essentially perfect fit: summary may be
## unreliable
```

```
pTb[pTb<0.05]
```

```
          ARLS      L0.all      CLS.vs.Tc      bminH202      bmaxH202
0.0392109400 0.0078946548 0.0354310798 0.0139759187 0.0047376191
      Cv.vs.Cb Cv.vs.Cb.STD      Cb.vs.Cv
0.0001815387 0.0362846069 0.0000000000
```

```
pTb = 1: length(tb3[1,])
names(pTb) = names(tb3)
for( j in c(2:41) ) {
  #m = lm( tb3a[, j] ~ tb3a$ARLS, na.rm=T)
  m = lm( tb3[, j] ~ tb3$Cb0.5.vs.Cv)
  #m = lm( tb3a[, j] ~ tb3a$CvMean, na.rm=T)
  #m = lm( tb3a[, j] ~ tb3a$CbMean, na.rm=T)
  sm = summary(m)
  pTb[j] = 1 - pf(sm$fsta[1], sm$fsta[2], sm$fsta[3])
}
```

```
## Warning in summary.lm(m): essentially perfect fit: summary may be
## unreliable
```

```
pTb[pTb<0.1]
```

```
##          TLmax          b.min          bminH202          bmaxH202
## 6.321115e-02 3.604597e-02 2.648181e-04 1.371843e-03
## Cb0.5.vs.Cv Cb0.5.vs.Cv.STD Cb.05STD
## 0.000000e+00 1.551670e-07 5.132811e-03
```

```

pTb2 = 1: length(tb3[1,])
names(pTb2) = names(tb3)
for( j in c(2:38) ) {
  m = lm( tb3[, j] ~ ( tb3$Cb0.5Mean))
  #m = lm( tb3a[, j] ~ tb3a$CvMean, na.rm=T)
  #m = lm( tb3a[, j] ~ tb3a$CbMean, na.rm=T)
  sm = summary(m)
  pTb2[j] = 1 - pf(sm$fsta[1], sm$fsta[2], sm$fsta[3])
}

```

```

## Warning in summary.lm(m): essentially perfect fit: summary may be
## unreliable

```

```

pTb2[pTb2<0.05]

```

```

##          CvManu      CvManuSTD      CvMean      CvSTD      CbMean
## 0.0050616007 0.0005836548 0.0074151552 0.0007418509 0.0002412994
##          CbSTD      Cv.vs.Cb  Cv.vs.Cb.STD      Cb0.5Mean
## 0.0035395976 0.0057037439 0.0002793327 0.0000000000

```

```

pTb2 = 1: length(tb3[1,])
names(pTb2) = names(tb3)
for( j in c(2:38) ) {
  m = lm( tb3[, j] ~ ( tb3$CbManu))
  #m = lm( tb3a[, j] ~ tb3a$CvMean, na.rm=T)
  #m = lm( tb3a[, j] ~ tb3a$CbMean, na.rm=T)
  sm = summary(m)
  pTb2[j] = 1 - pf(sm$fsta[1], sm$fsta[2], sm$fsta[3])
}

```

```

## Warning in summary.lm(m): essentially perfect fit: summary may be
## unreliable

```

```

pTb2[pTb2<0.075]

```

```

##   L0.small  CLS.vs.Tc  Tg.vs.Tc      CvManu      CvMean      CvSTD
## 0.01231221 0.06537462 0.05274154 0.06441624 0.03964427 0.06795442
##          CbManu      CbMean
## 0.00000000 0.03873183

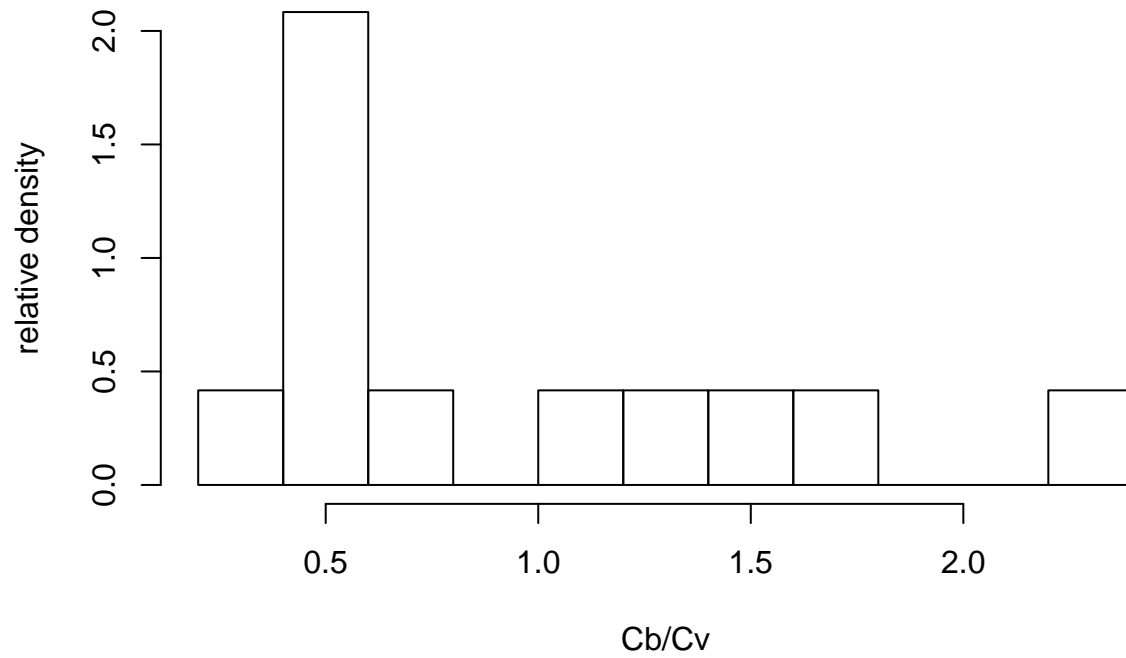
```

```

### side by side bar-plots of Tg/Tc Cb/Cv
mystep=0.2
my.breaks = seq( 0.2, round(max( c( tb3$Cb.vs.Cv, tb3$Tg.vs.Tc )) + 0.2, 1) ,by= mystep );
# my.breaks3 = seq( 0.1, round(max( c( tb3$Cb.vs.Cv, tb3$Tg.vs.Tc )) + 0.1, 1)) ,by= mystep );
h.H202 <- hist( tb3$Cb.vs.Cv, br= my.breaks, xlab = "Cb/Cv", ylab = "relative density", freq=F ) ;

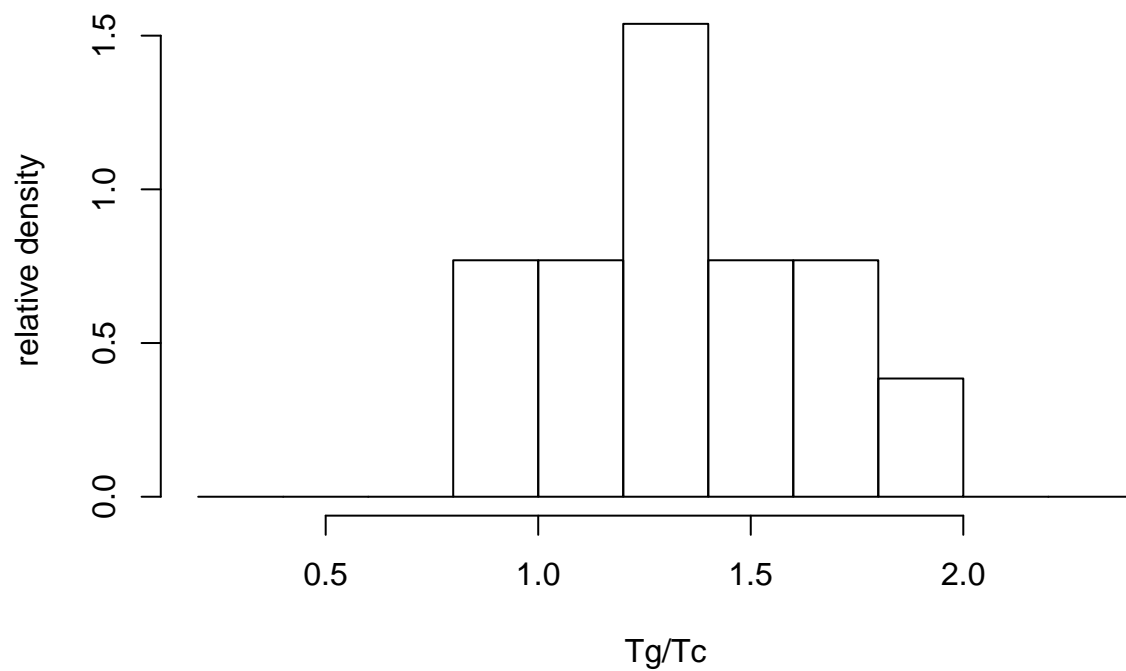
```

**Histogram of tb3\$Cb.vs.Cv**



```
# h.H202 <- hist( tb3$Cb.vs.Cv, br= 10, xlab = "Cb/Cv", ylab = "relative density", freq=F ) ;  
h.aging <- hist(tb$Tg.vs.Tc, br= my.breaks, xlab = "Tg/Tc", ylab = "relative density", freq=F ) ;
```

**Histogram of tb\$Tg.vs.Tc**



```

#generate the comparison table
bins <- data.frame( rbind(h.H2O2$density,h.aging$density) ) ;
# bins3 <- data.frame( rbind(h.H2O2$density,h.aging$density, h.H2O2Mean$density) ) ;
my.mids = my.breaks[-length(my.breaks)] + mystep/2
#my.mids
names( bins ) <- my.mids
row.names(bins) <- c( "H2O2", "Chronological Aging" )
# row.names(bins3) <- c( "H2O2", "Chronological Aging", "H2O2 Mean" )
bins

```

```

##              0.3      0.5      0.7      0.9      1.1
## H2O2          0.416667 2.083333 0.416667 0.000000 0.416667
## Chronological Aging 0.000000 0.000000 0.000000 0.7692308 0.7692308
##              1.3      1.5      1.7      1.9 2.1      2.3
## H2O2          0.416667 0.416667 0.416667 0.000000 0 0.416667
## Chronological Aging 1.5384615 0.7692308 0.7692308 0.3846154 0 0.0000000

```

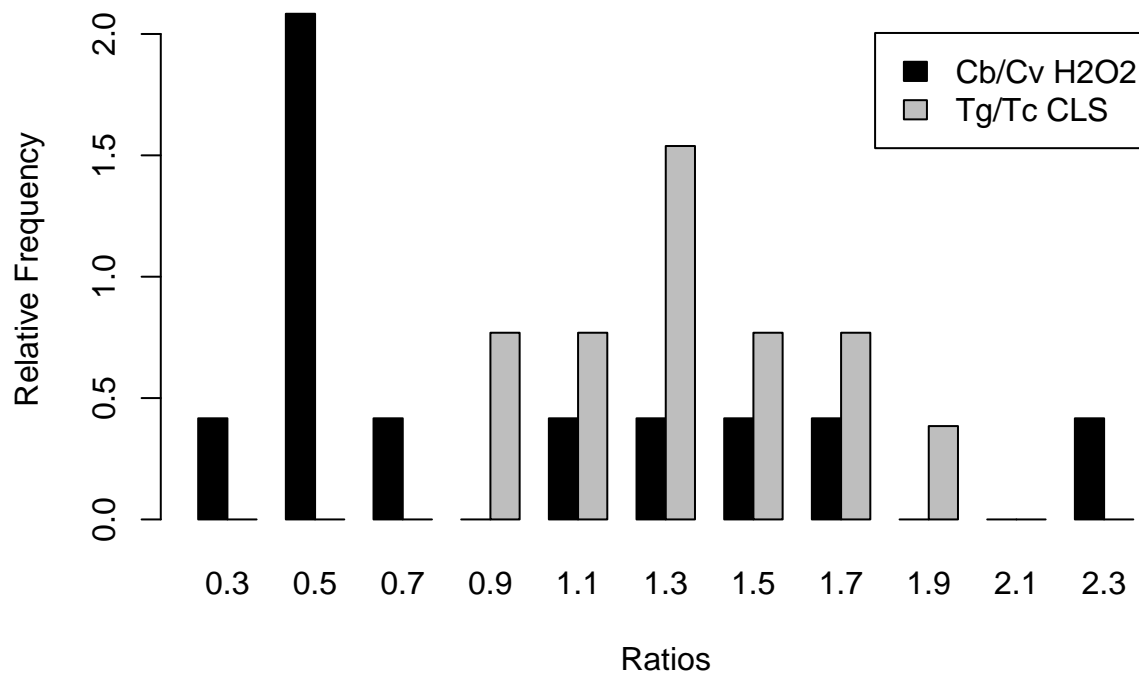
```

# bins3

#pdf("plots/Figure_sideBYside20131209.pdf", width=8, height=5)
# tiff("plots/Figure_sideBYside20131209.tif", width=480, height=480)
barplot( as.matrix(bins), beside=T, col=c("black","gray"), ylab="Relative Frequency", xlab="Ratios",
         legend= c( "Cb/Cv H2O2", "Tg/Tc CLS" ) );
title(main="H2O2 and CLS trigger LOH at different modes" )

```

## H2O2 and CLS trigger LOH at different modes



```

# dev.off();

ks.test(tb3$Cb.vs.Cv, tb3$Tg.vs.Tc) #p=0.031

```

```
##
## Two-sample Kolmogorov-Smirnov test
##
## data:  tb3$Cb.vs.Cv and tb3$Tg.vs.Tc
## D = 0.58333, p-value = 0.03144
## alternative hypothesis: two-sided

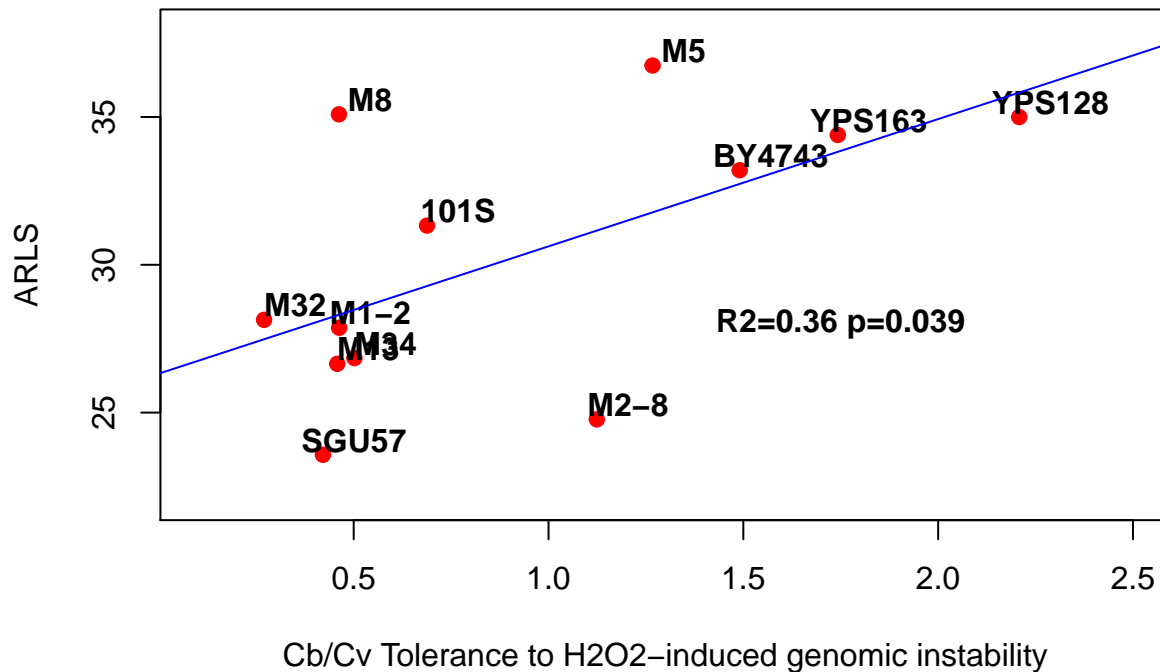
# tiff("plots/ARLS-CbCv-20131209.tif",width=480,height=480)
par(font=2)
plot( tb3$ARLS ~ tb3$Cb.vs.Cv , pch=19, col="red", main="H2O2-LOH ~ ARLS, 20131209", ylim=c(22,38), xlab=
      , ylab='ARLS',xlab='Cb/Cv Tolerance to H2O2-induced genomic instability')
text( tb3$Cb.vs.Cv+0.08, tb3$ARLS+0.5, tb3$strain)
m = lm(tb3$ARLS ~ tb3$Cb.vs.Cv )
abline( m, col="blue")
summary(m)

##
## Call:
## lm(formula = tb3$ARLS ~ tb3$Cb.vs.Cv)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.3870 -1.6373  0.0113  1.0047  6.7836
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    26.317      2.005   13.129 1.25e-07 ***
## tb3$Cb.vs.Cv     4.307      1.816    2.371  0.0392 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.791 on 10 degrees of freedom
## Multiple R-squared:  0.3599, Adjusted R-squared:  0.2958
## F-statistic: 5.622 on 1 and 10 DF, p-value: 0.03921

text(1.75, 28, "R2=0.36 p=0.039")
```



## H2O2-LOH ~ ARLS, 20131209



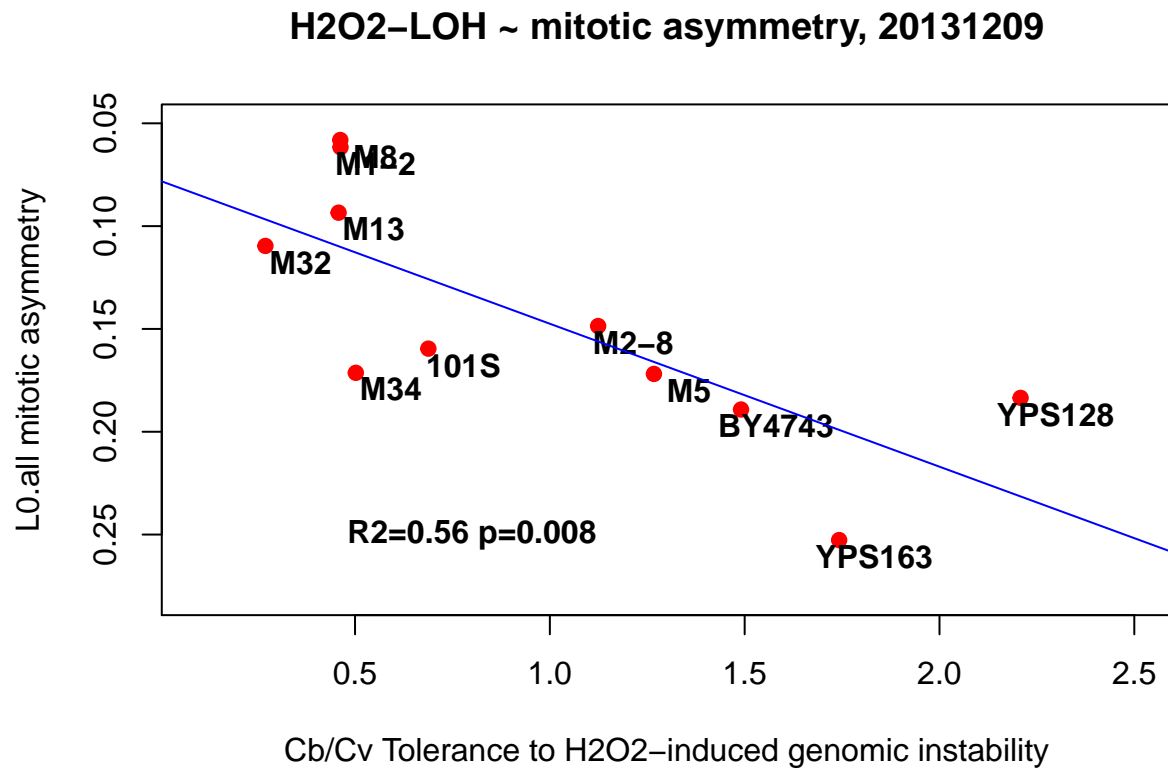
```
# dev.off()

# tiff("plots/L0-CbCv-20131209.tif",width=480,height=480)
par(font=2)
plot( tb3$L0.all ~ tb3$Cb.vs.Cv , pch=19, col="red", main="H2O2-LOH ~ mitotic asymmetry, 20131209"
      ,xlim=c(0.1,2.5),ylim=c(0.28, 0.05)
      , ylab='L0.all mitotic asymmetry',xlab='Cb/Cv Tolerance to H2O2-induced genomic instability')
text( tb3$Cb.vs.Cv+0.09, tb3$L0.all+0.008,tb3$strain)
m = lm( tb3$L0.all ~ tb3$Cb.vs.Cv )
abline( m, col="blue")
summary(m)

##
## Call:
## lm(formula = tb3$L0.all ~ tb3$Cb.vs.Cv)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.051976 -0.032127  0.005828  0.023344  0.058495
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.07795    0.02344   3.325  0.00886 **
## tb3$Cb.vs.Cv  0.06953    0.02046   3.398  0.00789 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04132 on 9 degrees of freedom
```

```
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.562, Adjusted R-squared: 0.5133
## F-statistic: 11.55 on 1 and 9 DF, p-value: 0.007895
```

```
text(0.8, 0.25, "R2=0.56 p=0.008")
```



```
# dev.off()
```

```
### Cv/Cb or Cb/Cv ~ robustness? I need a positive proxy
#summary(lm( tb3$Cv.vs.Cb ~ tb3$G )) #positive, p=0.20,
#summary(lm( tb3$Cb.vs.Cv ~ tb3$G )) #negative, p=0.25
```