OICR Module 9 Data Integration and Survival Workshop

Lauren Erdman, Goldenberg Lab

NOTE: THIS CODE REQUIRES THE FOLLOWING PACKAGES TO BE INSTALLED

SNFtool

RColorBrewer

survival

rms

First install packages and set working directory

```
# install.packages(pkgs = c("SNFtool","RColorBrewer","survival","rms"))

setwd("C:/Users/Owner/Desktop/Goldenberg Lab/OICR Workshop Materials/Data/")
```

Load the data for the module

```
load("OICR-Survival-Workshop-Data-revised-6-3-2016.RData")
```

Similarity Network Fusion

Importing SNF library and set parameters for SNF

```
library('SNFtool')
## First, set all the parameters:
K = 20;     # number of neighbors, usually (10~30)
alpha = 0.5;     # hyperparameter, usually (0.3~0.8)
T = 20;   # Number of Iterations, usually (10~20)
```

TRANSPOSE AND STANDARDIZE DATA GENOMIC DATA BEING USED

```
std.kirc.list <- lapply(X = list(methyl = t(methyl.kirc),
                                 mirna = t(mirna.kirc),
                                 mrna = t(mrna.kirc)),
                        standardNormalization)
```

GENERATE DISTANCE MATRICES USING EUCLIDEAN DISTANCE

```
dist.kirc.matrices <- lapply(X = std.kirc.list,
                             function(x){dist2(as.matrix(x),as.matrix(x))})
```

GENERATE AFFINITY MATRICES

```
affinity.kirc.matrices <- lapply(X = dist.kirc.matrices,
                                 function(x){affinityMatrix(x,K,alpha)})
```

CLUSTER INDIVIDUAL DATA TYPES

```
(n.clusters.estimated <- lapply(X = affinity.kirc.matrices,
                                function(x){estimateNumberOfClustersGivenGraph(x)[[1]]}))
```

```
## $methyl
## [1] 2
##
## $mirna
## [1] 2
##
## $mrna
## [1] 3
```

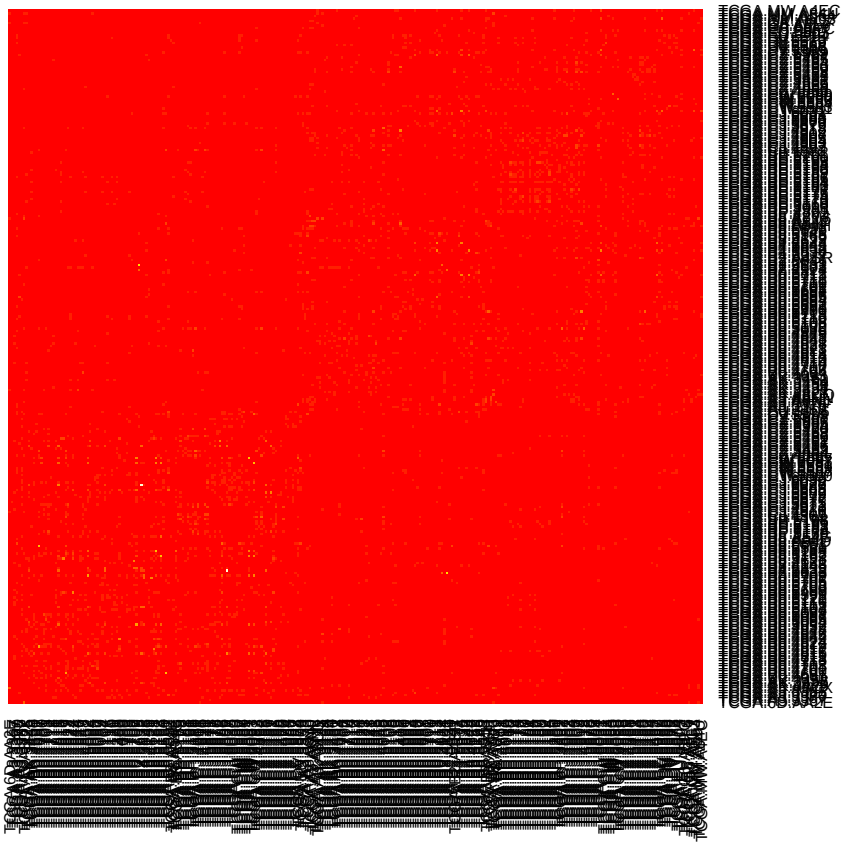
```
clustered.groups <- sapply(X = seq(1,3),
                           function(x){spectralClustering(affinity = affinity.kirc.matrices[[x]],
                                                          K = n.clusters.estimated[[x]])})


colnames(clustered.groups) <- names(affinity.kirc.matrices)

## Looking at distribution of group assignment
apply(clustered.groups,2,table)
```

```
## $methyl
##
##   1   2
## 120 164
##
## $mirna
##
##   1   2
## 174 110
##
## $mrna
##
##   1   2   3
## 137 144   3
```

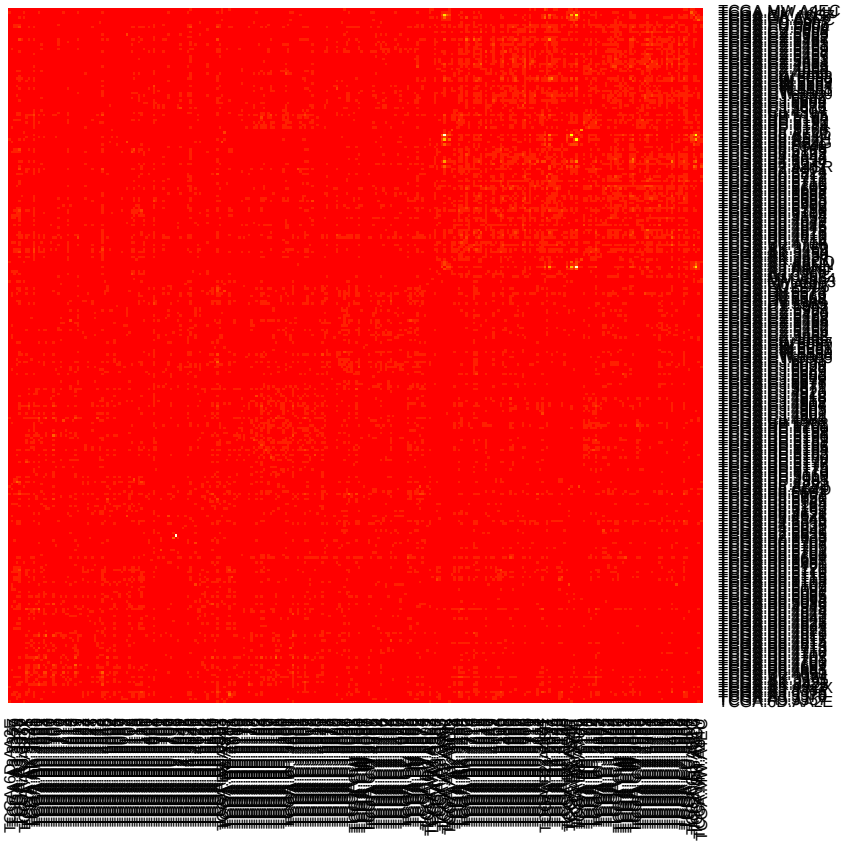
```
    ## Weird! Looks like there is a group of outliers in miRNA and mRNA
```

Using heatmap to look at clusters:

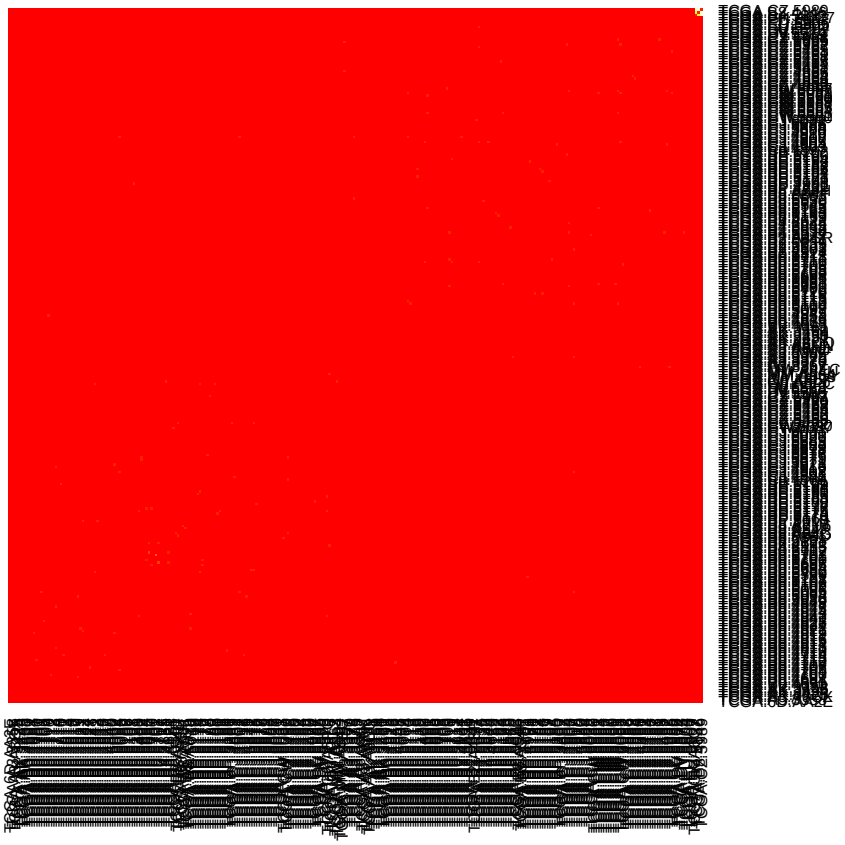
```
displayClustersWithHeatmap(affinity.kirc.matrices[['methyl']],group = clustered.groups[,'methyl'])
```



```
displayClustersWithHeatmap(affinity.kirc.matrices[['mirna']],group = clustered.groups[,'mirna'])
```



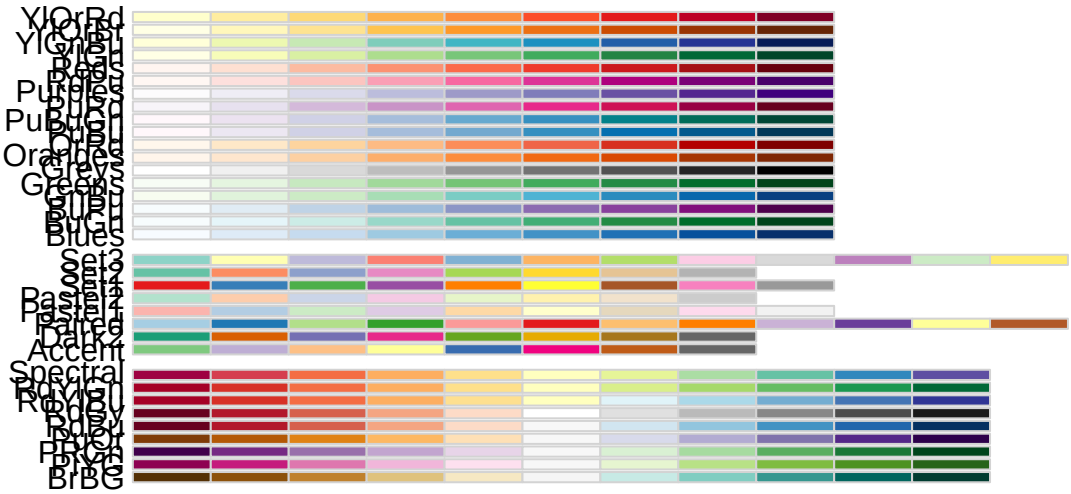
```
displayClustersWithHeatmap(affinity.kirc.matrices[['mrna']],group = clustered.groups[,'mrna'])
```



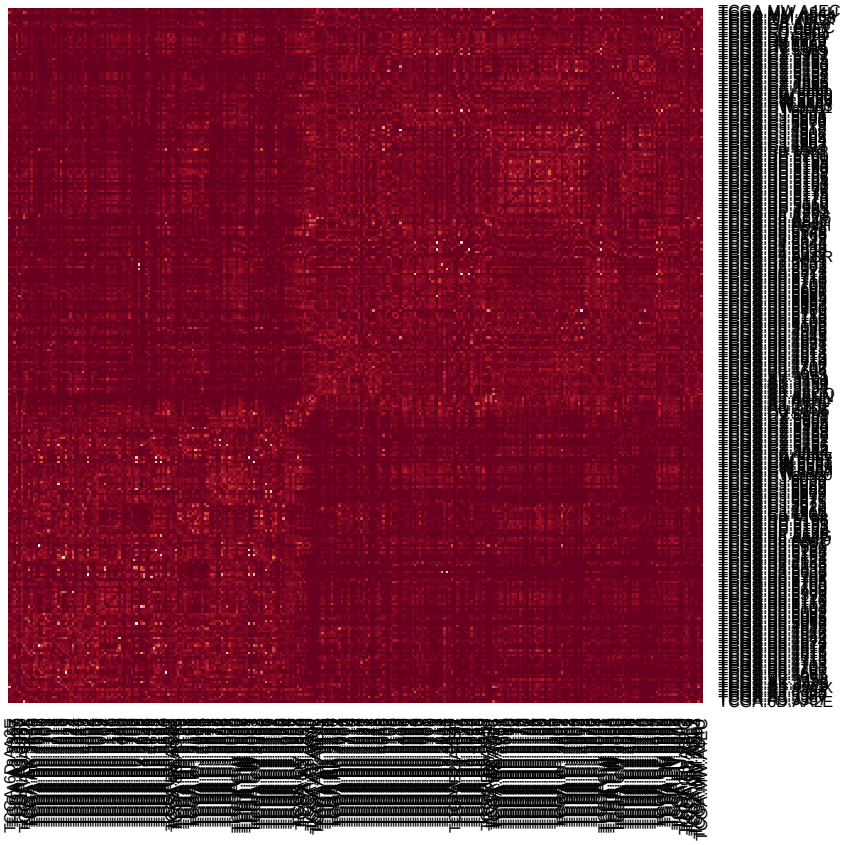
```
## This color isn't the best, let's revise the heatmap colors using RColorBrewer
```

CHANGING HEATMAP COLORS USING RColorBrewer

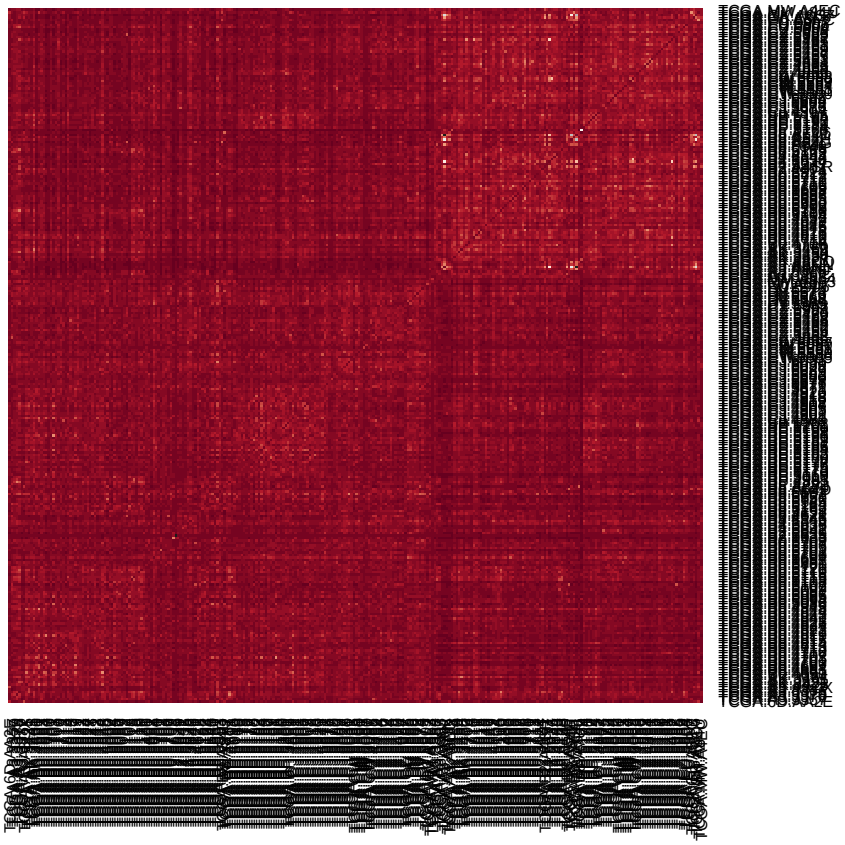
```
library("RColorBrewer")
rd.gy = colorRampPalette(brewer.pal(n = 11,name = "RdGy"))(50)

display.brewer.all()
```



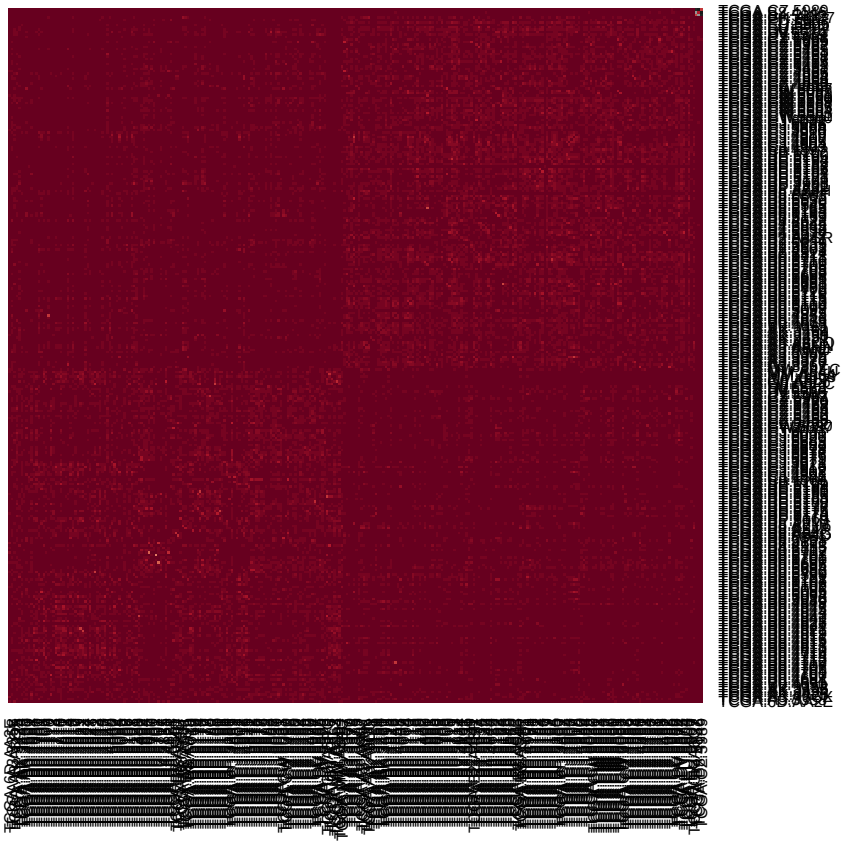

```
displayClustersWithHeatmap(affinity.kirc.matrices[['methyl']],group = clustered.groups[,'methyl'],col =
```



```
displayClustersWithHeatmap(affinity.kirc.matrices[['mirna']],group = clustered.groups[,'mirna'],col = re
```



```
displayClustersWithHeatmap(affinity.kirc.matrices[['mrna']],group = clustered.groups[,'mrna'],col = rd.
```



```
## those groups of 3 seem to be ruining our signal elsewhere - let's see what removing them gets us
```

IDENTIFYING GROUPS OF 3 OUTLIERS IN miRNA and mRNA

```
(mirna.group3.ids <- colnames(mirna.kirc)[which(clustered.groups[,'mirna'] == 3)])
```

```
## character(0)
```

```
(mrna.group3.ids <- colnames(mrna.kirc)[which(clustered.groups[,'mrna'] == 3)])
```

```
## [1] "TCGA.B4.5832" "TCGA.B8.4146" "TCGA.CZ.5989"
```

```
    ## 3 outliers the same - let's remove these thress individuals and see what we get
```

REMOVING INDIVIDUALS FROM EACH DATASET

```
# First, let's make a vector of IDs we'd like to keep
ids.to.keep <- colnames(mirna.kirc)[which(clustered.groups[,'mirna'] != 3)]

sum(clinic.kirc$revised.ids %in% ids.to.keep) ## number of individuals we expect in each dataset
```

```
## [1] 284
```

```
####
#   SUBSETTING OUR DATASETS
####

## We handle the clinic data separately since the structure is different
##    than the genomics data

## Clinic data
clinic.kirc.sub <- clinic.kirc[match(ids.to.keep,clinic.kirc$revised.ids),]

## Genomic data
genomic.sub.list <- lapply(list(methyl=t(methyl.kirc),
                                mirna = t(mirna.kirc),
                                mrna = t(mrna.kirc)),
                           function(x){x[match(ids.to.keep,rownames(x)),]})
```

DOUBLE CHECKING THAT OUR DATA IS NAMED CORRECTLY AND HAS THE CORRECT DIMENSIONS

```
names(genomic.sub.list)
```

```
## [1] "methyl" "mirna"  "mrna"
```

```
lapply(genomic.sub.list,dim)
```

```
## $methyl
## [1]   284 20914
##
## $mirna
## [1] 284 853
##
## $mrna
## [1]   284 20248
```

```
str(genomic.sub.list)
```

```
## List of 3
##  $ methyl: num [1:284, 1:20914] 0.369 0.395 0.342 0.356 0.37 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:284] "TCGA.6D.AA2E" "TCGA.A3.3357" "TCGA.A3.3358" "TCGA.A3.3367" ...
##   .. ..$ : chr [1:20914] "WASH5P" "OR4F5" "XK" "MIR1977" ...
##  $ mirna : num [1:284, 1:853] 13.4 11.4 13.7 11.8 11.6 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:284] "TCGA.6D.AA2E" "TCGA.A3.3357" "TCGA.A3.3358" "TCGA.A3.3367" ...
##   .. ..$ : chr [1:853] "hsa-let-7a-1" "hsa-let-7a-2" "hsa-let-7a-3" "hsa-let-7b" ...
##  $ mrna  : num [1:284, 1:20248] 0 0 0.398 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:284] "TCGA.6D.AA2E" "TCGA.A3.3357" "TCGA.A3.3358" "TCGA.A3.3367" ...
##   .. ..$ : chr [1:20248] "100130426" "100133144" "100134869" "10357" ...
```

RE-RUNNING DATA-SPECIFIC AFFINITY MATRIX CLUSTERING WITH SUBSET DATA

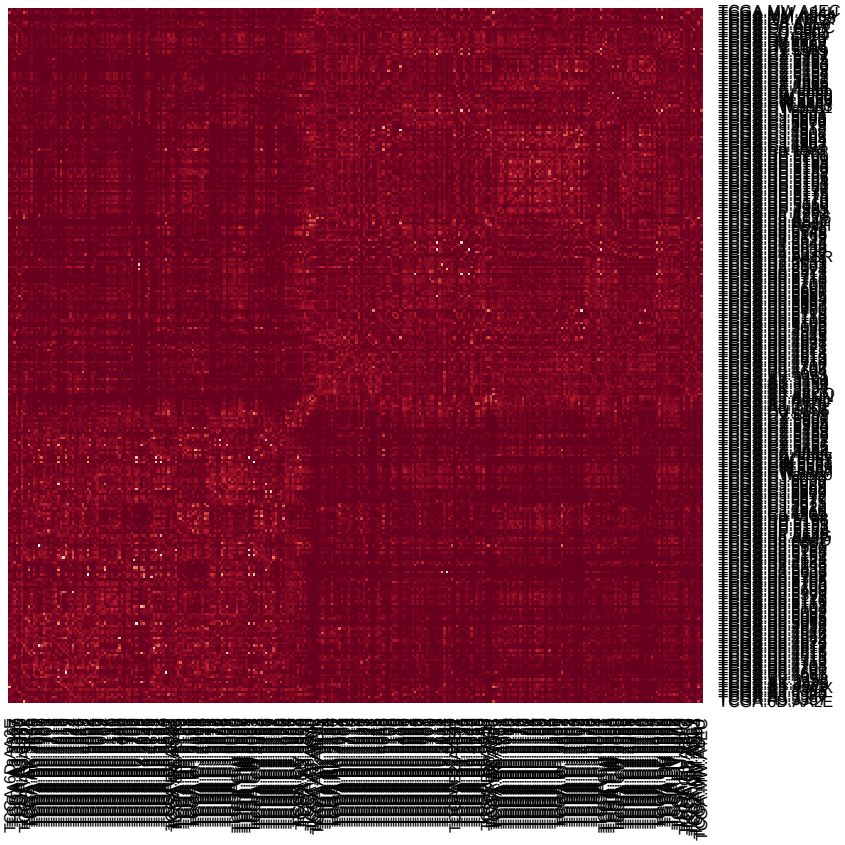
```
#   STANDARD NORMALIZE DATA GENOMIC DATA BEING USED
std.kirc.list.sub <- lapply(X = genomic.sub.list,
                            standardNormalization)

#   GENERATE DISTANCE MATRICES USING EUCLIDEAN DISTANCE
dist.kirc.matrices.sub <- lapply(X = std.kirc.list.sub,
                                 function(x){dist2(as.matrix(x),as.matrix(x))})

#   GENERATE AFFINITY MATRICES
affinity.kirc.matrices.sub <- lapply(X = dist.kirc.matrices.sub,
                                     function(x){affinityMatrix(x,K,alpha)})

#   CLUSTER INDIVIDUAL DATA TYPES
n.clusters.estimated.sub <- lapply(X = affinity.kirc.matrices.sub,
                                   function(x){estimateNumberOfClustersGivenGraph(x)[[1]]})
clustered.groups.sub <- sapply(X = seq(1,3),
                               function(x){spectralClustering(affinity = affinity.kirc.matrices.sub[[x]]
                                                              K = n.clusters.estimated.sub[[x]])})
colnames(clustered.groups.sub) <- names(affinity.kirc.matrices.sub)

## Looking at distribution of group assignment
apply(clustered.groups.sub,2,table)
```

```
## $methyl
##
##   1   2
## 120 164
##
## $mirna
##
##   1   2
## 174 110
##
## $mrna
##
##   1   2   3
## 137 144   3
```

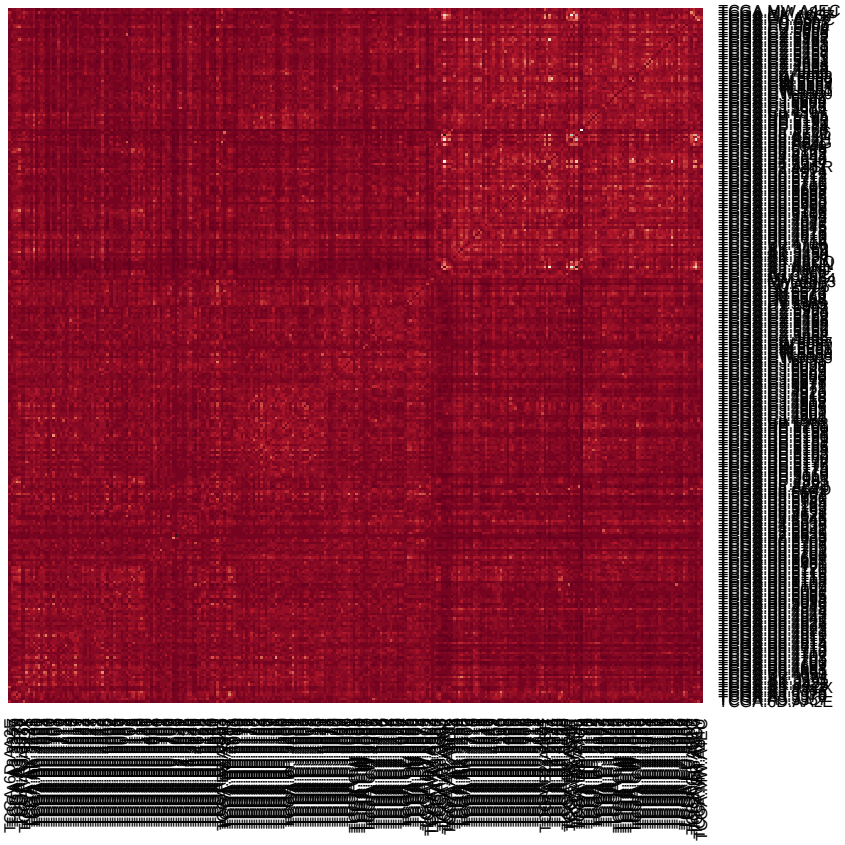
```
## No obvious outliers! :)
```

Using heatmap to look at clusters:

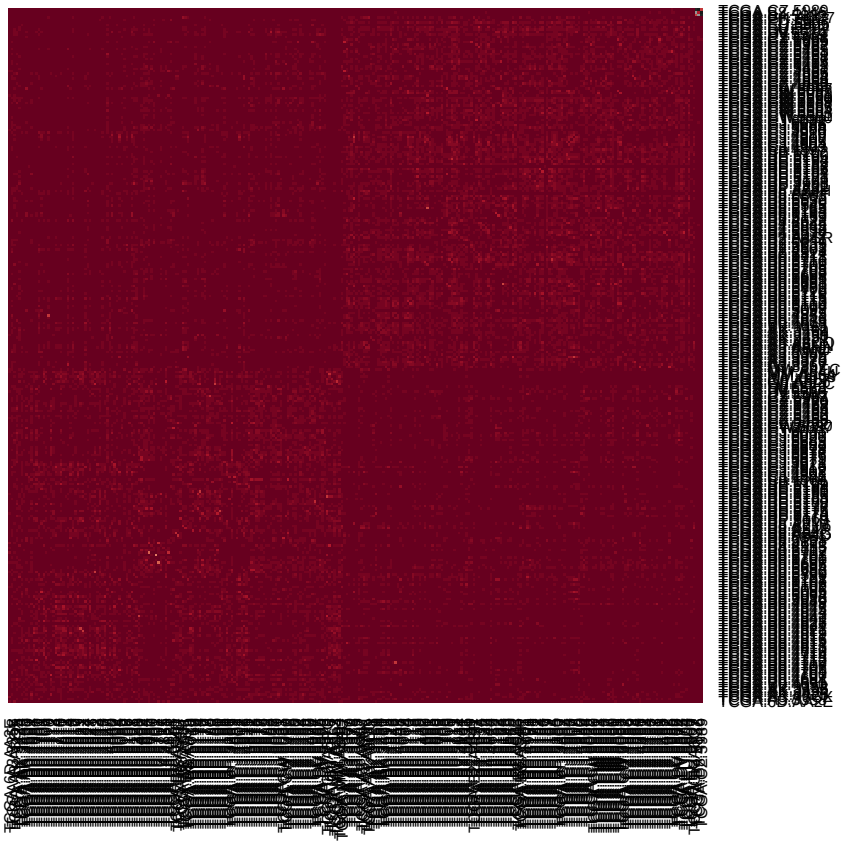
```
displayClustersWithHeatmap(affinity.kirc.matrices.sub[['methyl']],
                           group = clustered.groups.sub[,'methyl'],col = rd.gy)
```



```
displayClustersWithHeatmap(affinity.kirc.matrices.sub[['mirna']],
                           group = clustered.groups.sub[,'mirna'],col = rd.gy)
```



```
displayClustersWithHeatmap(affinity.kirc.matrices.sub[['mrna']],
                           group = clustered.groups.sub[,'mrna'],col = rd.gy)
```



```
## Much nicer clusters!
```

RUN SNF

```
kirc.snf <- SNF(affinity.kirc.matrices.sub,K = K,t = T)

## Naming names and columns
colnames(kirc.snf) <- rownames(kirc.snf) <- rownames(genomic.sub.list[[1]])
str(kirc.snf)
```

```
##  num [1:284, 1:284] 0.56492 0.0031 0.0031 0.003 0.00328 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:284] "TCGA.6D.AA2E" "TCGA.A3.3357" "TCGA.A3.3358" "TCGA.A3.3367" ...
##   ..$ : chr [1:284] "TCGA.6D.AA2E" "TCGA.A3.3357" "TCGA.A3.3358" "TCGA.A3.3367" ...
```

Generating fused clusters

```
#   FIND NUMBER OF CLUSTERS
estimateNumberOfClustersGivenGraph(W = kirc.snf,NUMC = 2:5)
```

```
## [[1]]
```

```
## [1] 2
##
## [[2]]
## [1] 4
##
## [[3]]
## [1] 2
##
## [[4]]
## [1] 4
```

```
#   GENERATE GROUP ASSIGNMENTS FROM NUMBER OF CLUSTERS DEFINED ABOVE
###

snf.groups <- spectralClustering(kirc.snf,K = 2)

## LOOK AT GROUP SIZES
table(snf.groups)
```

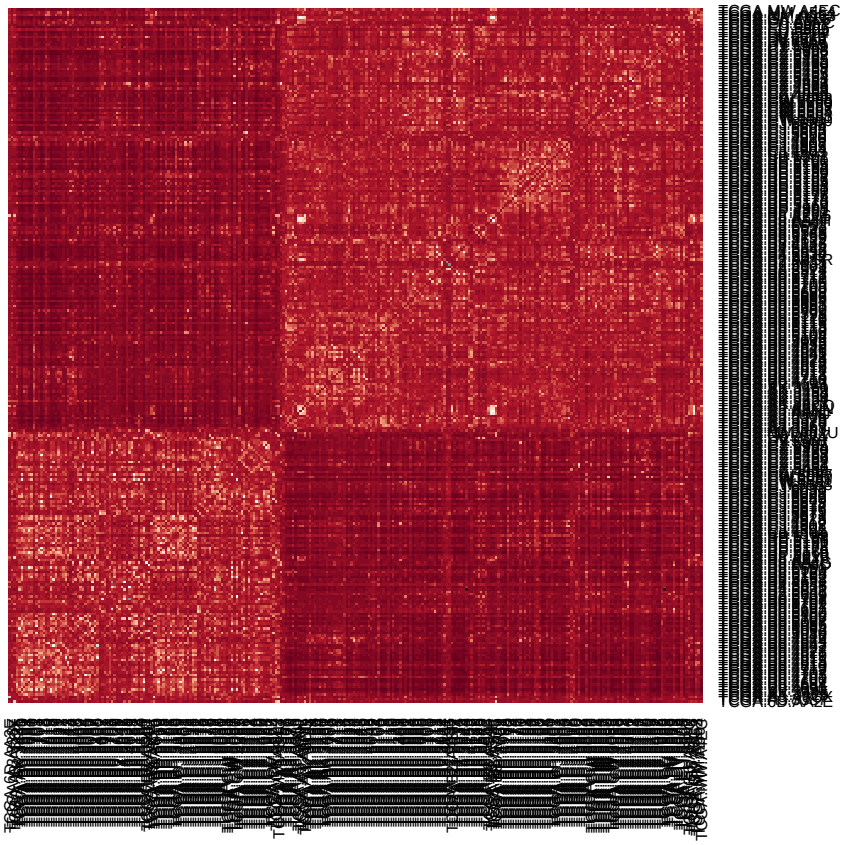
```
## snf.groups
##   1   2
## 111 173
```

```
## SET UP A DATAFRAME WITH GROUP ASSIGNMENT BY ID
##    (WE WILL USE THIS IN THE SURVIVAL ANALYSIS)
ids.groups2 <- data.frame(cbind(colnames(kirc.snf),snf.groups))
names(ids.groups2) <- c("id","group")
head(ids.groups2)
```

```
##             id group
## 1 TCGA.6D.AA2E     1
## 2 TCGA.A3.3357     2
## 3 TCGA.A3.3358     2
## 4 TCGA.A3.3367     2
## 5 TCGA.A3.3370     2
## 6 TCGA.A3.3373     2
```

GENERATE HEATMAP OF RESULTING MATRIX WITH CLUSTERING

```
displayClustersWithHeatmap(W = kirc.snf,group = snf.groups,col = rd.gy)
```



Survival Analysis

GENERATING SURVIVAL OUTCOME

```
### getting names columns in clinic data (don't forget - it's subsetted now)
names(clinic.kirc.sub)
```

```
##  [1] "X"
##  [2] "admin.batch_number"
##  [3] "patient.bcr_patient_barcode"
##  [4] "patient.bcr_patient_uuid"
##  [5] "patient.days_to_death"
##  [6] "patient.days_to_last_followup"
##  [7] "patient.days_to_last_known_alive"
##  [8] "patient.vital_status"
##  [9] "patient.age_at_initial_pathologic_diagnosis"
## [10] "patient.days_to_birth"
## [11] "patient.number_pack_years_smoked"
## [12] "patient.gender"
## [13] "patient.white_cell_count_result"
## [14] "patient.tobacco_smoking_history"
## [15] "patient.year_of_tobacco_smoking_onset"
## [16] "patient.race"
```

```
## [17] "patient.number_of_lymphnodes_positive"
## [18] "revised.ids"
## [19] "time.to.event"
## [20] "event"
## [21] "survival.outcome"
```

```
## first take a look at the variables we're going to use
# str(clinic.kirc$patient.age_at_initial_pathologic_diagnosis)
head(clinic.kirc.sub$patient.days_to_death)
```

```
## [1] NA NA NA NA NA NA
```

```
head(clinic.kirc.sub$patient.days_to_last_known_alive)
```

```
## [1]   NA   NA 1307   NA   NA   NA
```

```
head(clinic.kirc.sub$patient.days_to_last_followup)
```

```
## [1]  135 1425 1307 1054  776  334
```

```
head(clinic.kirc.sub$patient.vital_status)
```

```
## [1] "alive" "alive" "alive" "alive" "alive" "alive"
```

```
# generating time to event variable
clinic.kirc.sub$time.to.event <- clinic.kirc.sub$patient.days_to_last_followup
clinic.kirc.sub$time.to.event[is.na(clinic.kirc.sub$patient.days_to_death) == FALSE] <-
  clinic.kirc.sub$patient.days_to_death[is.na(clinic.kirc.sub$patient.days_to_death) == FALSE]

# generating event variable
clinic.kirc.sub$event <- NA
clinic.kirc.sub$event[clinic.kirc.sub$patient.vital_status == "alive"] <- 0
clinic.kirc.sub$event[clinic.kirc.sub$patient.vital_status == "dead"] <- 1

# tying together 'time.to.event' and 'event' in the survival outcome
library("survival")
```

```
## Warning: package 'survival' was built under R version 3.1.3
```

```
clinic.kirc.sub$survival.outcome <- Surv(clinic.kirc.sub$time.to.event,
                                         clinic.kirc.sub$event)
```

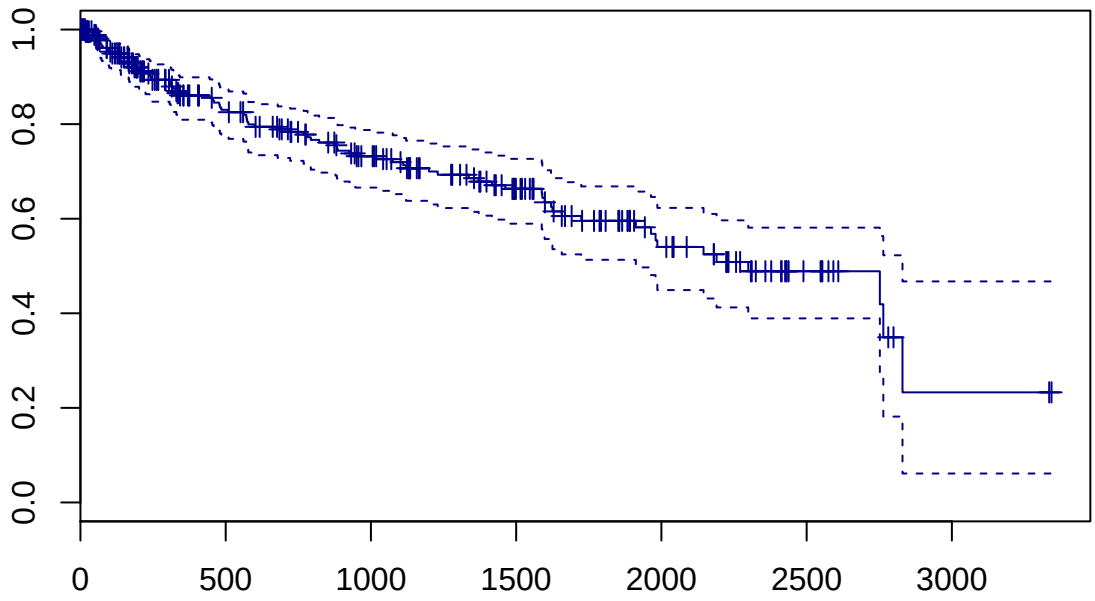
SUMMARIZING SURVIVAL WITHOUT COVARIATE

```
(kirc.survival.fit <- survfit(clinic.kirc.sub$survival.outcome ~ 1,
                              conf.type = "log-log"))
```

```
## Call: survfit(formula = clinic.kirc.sub$survival.outcome ~ 1, conf.type = "log-log")
##
##       n  events  median 0.95LCL 0.95UCL
##     284      84    2299    1912    2830
```

CREATING BASIC KM CURVE

```
plot(kirc.survival.fit,col="blue4")
```



COMPARING SURVIVAL ACROSS GROUPS

```
## merging SNF groups and clinic.data (recall we made 'ids.groups2' when we clustered SNF)
clinic.kirc.snf.group <- merge(x = clinic.kirc.sub, ## clinic dataframe
                               y = ids.groups2, ## SNF group dataframe
                               by.x="revised.ids", ## clinic ID column
                               by.y="id")  ## SNF group ID column

## creating factor variables for cluster and sex
clinic.kirc.snf.group$sex <- factor(clinic.kirc.snf.group$patient.gender,
                                    levels = c("male","female"))
clinic.kirc.snf.group$cluster <- factor(clinic.kirc.snf.group$group,
                                        levels = c(1,2))

###
#   TESTING THE DIFFERENCE IN SURVIVAL TIME USING PETO&PETO MODIVICATION ON THE
#       GEHAN-WILCOXON TEST, USING THE survdiff FUNCTION
###
```

```
## Sex

survdiff(clinic.kirc.snf.group$survival.outcome ~
           clinic.kirc.snf.group$sex, rho=1)
```

```
## Call:
## survdiff(formula = clinic.kirc.snf.group$survival.outcome ~ clinic.kirc.snf.group$sex,
##     rho = 1)
##
##                                   N Observed Expected (O-E)^2/E (O-E)^2/V
## clinic.kirc.snf.group$sex=male   184     44.8     43.7    0.0275    0.0982
## clinic.kirc.snf.group$sex=female 100     21.9     23.0    0.0524    0.0982
##
##  Chisq= 0.1  on 1 degrees of freedom, p= 0.754
```

```
## Cluster Assignment

survdiff(clinic.kirc.snf.group$survival.outcome ~
           clinic.kirc.snf.group$cluster, rho=1)
```

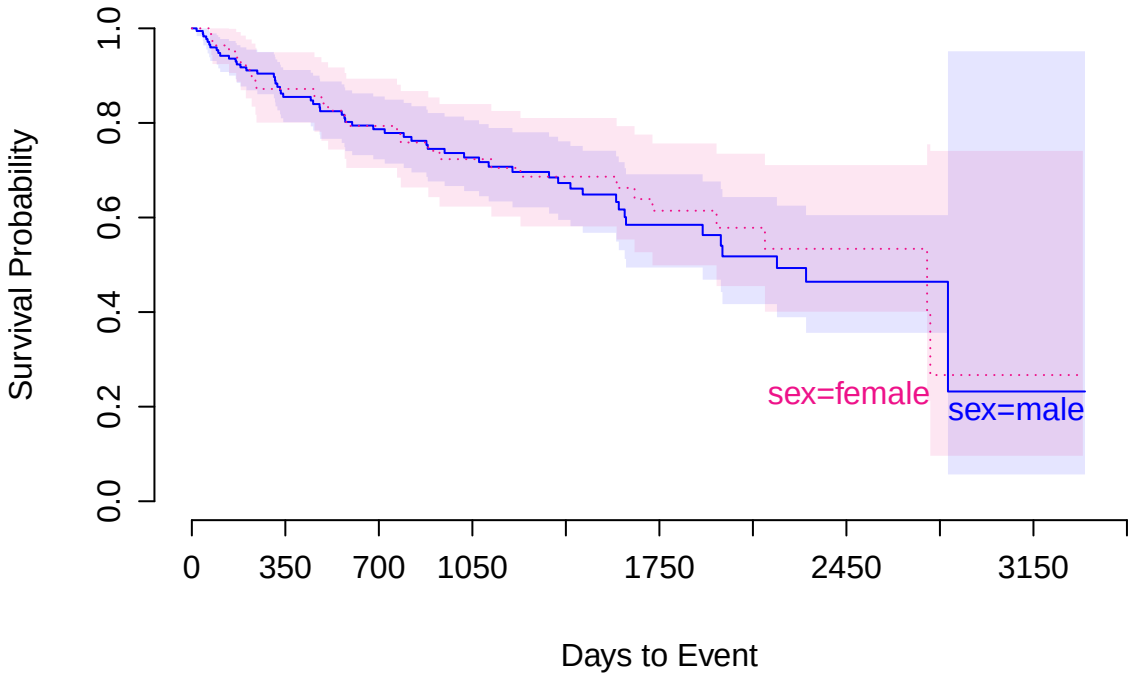
```
## Call:
## survdiff(formula = clinic.kirc.snf.group$survival.outcome ~ clinic.kirc.snf.group$cluster,
##     rho = 1)
##
##                                    N Observed Expected (O-E)^2/E (O-E)^2/V
## clinic.kirc.snf.group$cluster=1 111     39.8     24.8      8.96      17.4
## clinic.kirc.snf.group$cluster=2 173     26.9     41.8      5.32      17.4
##
##  Chisq= 17.4  on 1 degrees of freedom, p= 3.1e-05
```

```
# npsurv (non-parametric survival fit) function is a work around/replacement
#   for survfit since survfit no longer works with survplot which we want to use below

library('rms')
```

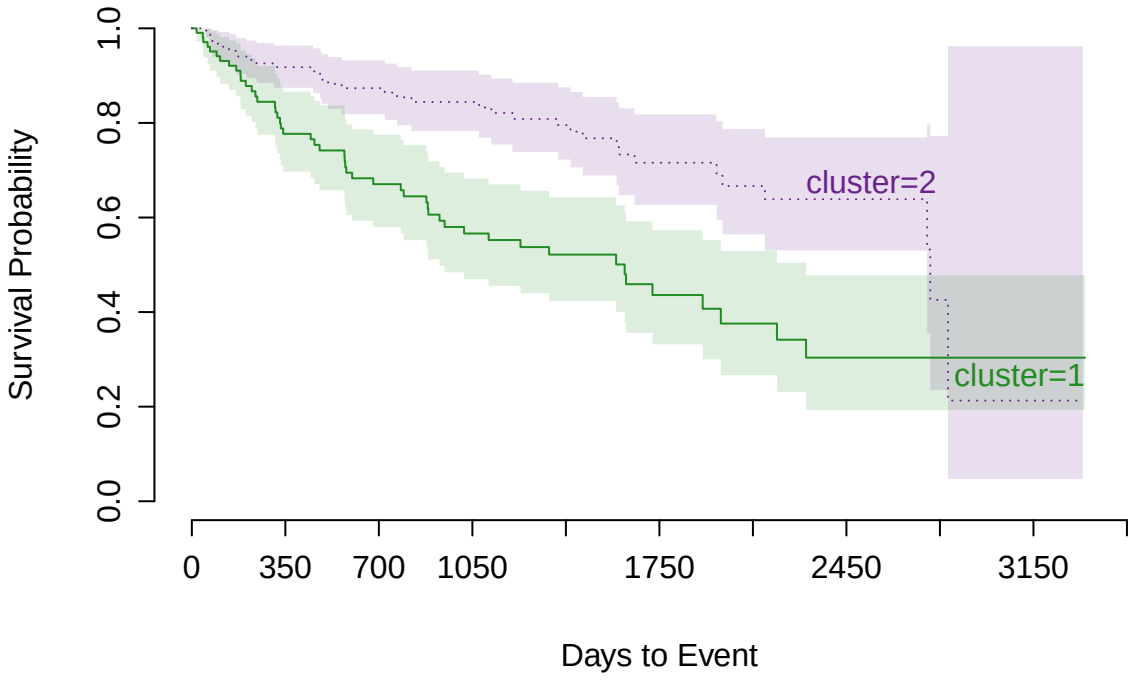
```
## Warning: package 'rms' was built under R version 3.1.3

## Loading required package: Hmisc

## Warning: package 'Hmisc' was built under R version 3.1.3

## Loading required package: grid
## Loading required package: lattice
## Loading required package: Formula

## Warning: package 'Formula' was built under R version 3.1.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.1.3
```

```
##
## Attaching package: 'Hmisc'
##
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
##
## Loading required package: SparseM

## Warning: package 'SparseM' was built under R version 3.1.3

##
## Attaching package: 'SparseM'
##
## The following object is masked from 'package:base':
##
##     backsolve
```

```
## looking at marginal survival difference by sex

kirc.survival.fit.by.sex <- npsurv(survival.outcome ~ sex,
                                   data = clinic.kirc.snf.group)

## looking at marginal survival difference by SNF generated cluster

kirc.survival.fit.by.snf.group <- npsurv(survival.outcome ~ cluster,
                                         data = clinic.kirc.snf.group)

####
#   ADDING CONFIDENCE BOUNDS AND COLORS TO KM CURVES PLOTTED BY GROUP
####

survplot(fit = kirc.survival.fit.by.sex,col=c('blue','deeppink2'),
         col.fill = sapply(c('blue','deeppink2'),function(x){adjustcolor(x, alpha.f = 0.1)}),
         xlab="Days to Event")
```




```
survplot(fit = kirc.survival.fit.by.snf.group,col=c('forestgreen','darkorchid4'),
         col.fill = sapply(c('forestgreen','darkorchid4'),function(x){adjustcolor(x, alpha.f = 0.15)}),
         xlab="Days to Event")
```



Cox proportional hazards analysis

```
# Want to fit model with survival as an outcome,
  # analyzing cluster assignment while controling for sex as covariates

## Efron method
(coxph.fit <- coxph(survival.outcome ~
                      cluster + sex + patient.age_at_initial_pathologic_diagnosis,
                    data = clinic.kirc.snf.group,method = "efron"))
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "efron")
##
##
##                                             coef exp(coef) se(coef)
## cluster2                                 -0.90425   0.40485  0.23077
## sexfemale                                 0.08923   1.09333  0.24175
## patient.age_at_initial_pathologic_diagnosis  0.02773   1.02811  0.00927
##                                                z       p
## cluster2                                 -3.92 8.9e-05
## sexfemale                                 0.37  0.7121
## patient.age_at_initial_pathologic_diagnosis  2.99  0.0028
##
```

```
## Likelihood ratio test=25.7  on 3 df, p=1.09e-05
## n= 284, number of events= 84
```

```
summary(coxph.fit)
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "efron")
##
##   n= 284, number of events= 84
##
##                                             coef exp(coef)  se(coef)
## cluster2                                -0.904247  0.404846  0.230769
## sexfemale                                0.089229  1.093331  0.241748
## patient.age_at_initial_pathologic_diagnosis  0.027725  1.028113  0.009266
##                                             z Pr(>|z|)
## cluster2                                -3.918 8.91e-05 ***
## sexfemale                                0.369  0.71205
## patient.age_at_initial_pathologic_diagnosis  2.992  0.00277 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                         exp(coef) exp(-coef) lower .95
## cluster2                                   0.4048     2.4701    0.2575
## sexfemale                                  1.0933     0.9146    0.6807
## patient.age_at_initial_pathologic_diagnosis    1.0281     0.9727    1.0096
##                                         upper .95
## cluster2                                   0.6364
## sexfemale                                  1.7560
## patient.age_at_initial_pathologic_diagnosis    1.0470
##
## Concordance= 0.664  (se = 0.035 )
## Rsquare= 0.087   (max possible= 0.942 )
## Likelihood ratio test= 25.72  on 3 df,   p=1.091e-05
## Wald test            = 25.13  on 3 df,   p=1.447e-05
## Score (logrank) test = 26.43  on 3 df,   p=7.736e-06
```

```
## Exact method
(coxph.fit <- coxph(survival.outcome ~
                      cluster + sex + patient.age_at_initial_pathologic_diagnosis,
                    data = clinic.kirc.snf.group,method = "exact"))
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "exact")
##
##
##                                             coef exp(coef) se(coef)
## cluster2                                -0.90435   0.40481  0.23078
## sexfemale                                0.08923   1.09333  0.24177
## patient.age_at_initial_pathologic_diagnosis  0.02773   1.02812  0.00927
##                                             z       p
## cluster2                                -3.92 8.9e-05
```

```
## sexfemale                                      0.37  0.7121
## patient.age_at_initial_pathologic_diagnosis  2.99  0.0028
##
## Likelihood ratio test=25.7  on 3 df, p=1.09e-05
## n= 284, number of events= 84
```

```
summary(coxph.fit)
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "exact")
##
##   n= 284, number of events= 84
##
##                                                 coef exp(coef)  se(coef)
## cluster2                                    -0.904345  0.404807  0.230785
## sexfemale                                    0.089226  1.093328  0.241768
## patient.age_at_initial_pathologic_diagnosis  0.027731  1.028119  0.009267
##                                                 z Pr(>|z|)
## cluster2                                    -3.919 8.91e-05 ***
## sexfemale                                    0.369  0.71208
## patient.age_at_initial_pathologic_diagnosis  2.992  0.00277 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                             exp(coef) exp(-coef) lower .95
## cluster2                                       0.4048     2.4703    0.2575
## sexfemale                                      1.0933     0.9146    0.6807
## patient.age_at_initial_pathologic_diagnosis    1.0281     0.9726    1.0096
##                                             upper .95
## cluster2                                       0.6363
## sexfemale                                      1.7561
## patient.age_at_initial_pathologic_diagnosis    1.0470
##
## Rsquare= 0.087   (max possible= 0.942 )
## Likelihood ratio test= 25.72  on 3 df,   p=1.089e-05
## Wald test            = 25.14  on 3 df,   p=1.445e-05
## Score (logrank) test = 26.44  on 3 df,   p=7.724e-06
```

```
## Breslow method
(coxph.fit <- coxph(survival.outcome ~
                      cluster + sex + patient.age_at_initial_pathologic_diagnosis,
                    data = clinic.kirc.snf.group,method = "breslow"))
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "breslow")
##
##
##                                                 coef exp(coef) se(coef)
## cluster2                                    -0.90424   0.40485  0.23077
## sexfemale                                    0.08921   1.09331  0.24175
## patient.age_at_initial_pathologic_diagnosis  0.02773   1.02812  0.00927
```

```
##                                              z       p
## cluster2                                 -3.92 8.9e-05
## sexfemale                                 0.37  0.7121
## patient.age_at_initial_pathologic_diagnosis  2.99  0.0028
##
## Likelihood ratio test=25.7  on 3 df, p=1.09e-05
## n= 284, number of events= 84
```

```
summary(coxph.fit)
```

```
## Call:
## coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_pathologic_diagnosis,
##     data = clinic.kirc.snf.group, method = "breslow")
##
##   n= 284, number of events= 84
##
##                                                 coef exp(coef)  se(coef)
## cluster2                                    -0.904238  0.404850  0.230772
## sexfemale                                    0.089211  1.093311  0.241751
## patient.age_at_initial_pathologic_diagnosis  0.027728  1.028116  0.009266
##                                                 z Pr(>|z|)
## cluster2                                    -3.918 8.92e-05 ***
## sexfemale                                    0.369  0.71211
## patient.age_at_initial_pathologic_diagnosis  2.992  0.00277 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                             exp(coef) exp(-coef) lower .95
## cluster2                                       0.4049     2.4701    0.2575
## sexfemale                                      1.0933     0.9147    0.6807
## patient.age_at_initial_pathologic_diagnosis    1.0281     0.9727    1.0096
##                                             upper .95
## cluster2                                       0.6364
## sexfemale                                      1.7560
## patient.age_at_initial_pathologic_diagnosis    1.0470
##
## Concordance= 0.664  (se = 0.035 )
## Rsquare= 0.087   (max possible= 0.942 )
## Likelihood ratio test= 25.72  on 3 df,   p=1.091e-05
## Wald test            = 25.14  on 3 df,   p=1.447e-05
## Score (logrank) test = 26.43  on 3 df,   p=7.734e-06
```

Extracting results

```
str(summary(coxph.fit))
```

```
## List of 14
##  $ call        : language coxph(formula = survival.outcome ~ cluster + sex + patient.age_at_initial_
##  $ fail        : NULL
##  $ na.action   : NULL
##  $ n           : int 284
```

```
##  $ loglik      : num [1:2] -404 -391
##  $ nevent      : num 84
##  $ coefficients: num [1:3, 1:5] -0.9042 0.0892 0.0277 0.4049 1.0933 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "cluster2" "sexfemale" "patient.age_at_initial_pathologic_diagnosis"
##   .. ..$ : chr [1:5] "coef" "exp(coef)" "se(coef)" "z" ...
##  $ conf.int    : num [1:3, 1:4] 0.405 1.093 1.028 2.47 0.915 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "cluster2" "sexfemale" "patient.age_at_initial_pathologic_diagnosis"
##   .. ..$ : chr [1:4] "exp(coef)" "exp(-coef)" "lower .95" "upper .95"
##  $ logtest     : Named num [1:3] 2.57e+01 3.00 1.09e-05
##   ..- attr(*, "names")= chr [1:3] "test" "df" "pvalue"
##  $ sctest      : Named num [1:3] 2.64e+01 3.00 7.73e-06
##   ..- attr(*, "names")= chr [1:3] "test" "df" "pvalue"
##  $ rsq         : Named num [1:2] 0.0866 0.9419
##   ..- attr(*, "names")= chr [1:2] "rsq" "maxrsq"
##  $ waldtest    : Named num [1:3] 2.51e+01 3.00 1.45e-05
##   ..- attr(*, "names")= chr [1:3] "test" "df" "pvalue"
##  $ used.robust : logi FALSE
##  $ concordance : Named num [1:2] 0.664 0.0351
##   ..- attr(*, "names")= chr [1:2] "concordance.concordant" "se.std(c-d)"
##  - attr(*, "class")= chr "summary.coxph"
```

```
(coxph.coefs <- summary(coxph.fit)$coef)
```

```
##                                                coef exp(coef)
## cluster2                                  -0.90423839 0.4048501
## sexfemale                                  0.08921104 1.0933114
## patient.age_at_initial_pathologic_diagnosis  0.02772762 1.0281156
##                                              se(coef)          z
## cluster2                                  0.230771849 -3.9183219
## sexfemale                                 0.241751038  0.3690203
## patient.age_at_initial_pathologic_diagnosis 0.009266363  2.9922870
##                                                Pr(>|z|)
## cluster2                                  8.916757e-05
## sexfemale                                 7.121126e-01
## patient.age_at_initial_pathologic_diagnosis 2.768958e-03
```

```
(coxph.confint <- summary(coxph.fit)$conf.int)
```

```
##                                           exp(coef) exp(-coef) lower .95
## cluster2                                  0.4048501  2.4700500 0.2575496
## sexfemale                                 1.0933114  0.9146525 0.6807145
## patient.age_at_initial_pathologic_diagnosis 1.0281156  0.9726533 1.0096118
##                                           upper .95
## cluster2                                  0.6363962
## sexfemale                                 1.7559927
## patient.age_at_initial_pathologic_diagnosis 1.0469586
```

```
(coxph.results <- cbind(coxph.coefs,coxph.confint))
```

```
##                                                coef exp(coef)
```

```
## cluster2                                       -0.90423839 0.4048501
## sexfemale                                       0.08921104 1.0933114
## patient.age_at_initial_pathologic_diagnosis     0.02772762 1.0281156
##                                                   se(coef)          z
## cluster2                                       0.230771849 -3.9183219
## sexfemale                                      0.241751038  0.3690203
## patient.age_at_initial_pathologic_diagnosis 0.009266363  2.9922870
##                                                    Pr(>|z|) exp(coef)
## cluster2                                       8.916757e-05 0.4048501
## sexfemale                                      7.121126e-01 1.0933114
## patient.age_at_initial_pathologic_diagnosis 2.768958e-03 1.0281156
##                                                exp(-coef) lower .95 upper .95
## cluster2                                        2.4700500 0.2575496 0.6363962
## sexfemale                                       0.9146525 0.6807145 1.7559927
## patient.age_at_initial_pathologic_diagnosis     0.9726533 1.0096118 1.0469586
```

```
colnames(coxph.results)
```

```
## [1] "coef"       "exp(coef)"  "se(coef)"   "z"          "Pr(>|z|)"
## [6] "exp(coef)"  "exp(-coef)" "lower .95"  "upper .95"
```

```
write.csv(coxph.results[,c("coef","exp(coef)","se(coef)","Pr(>|z|)","lower .95","upper .95" )],
          file = "Cox-PH-model-results.csv")
```

Using cox.zph to test for covariate-specific

and global proportional hazards as well as

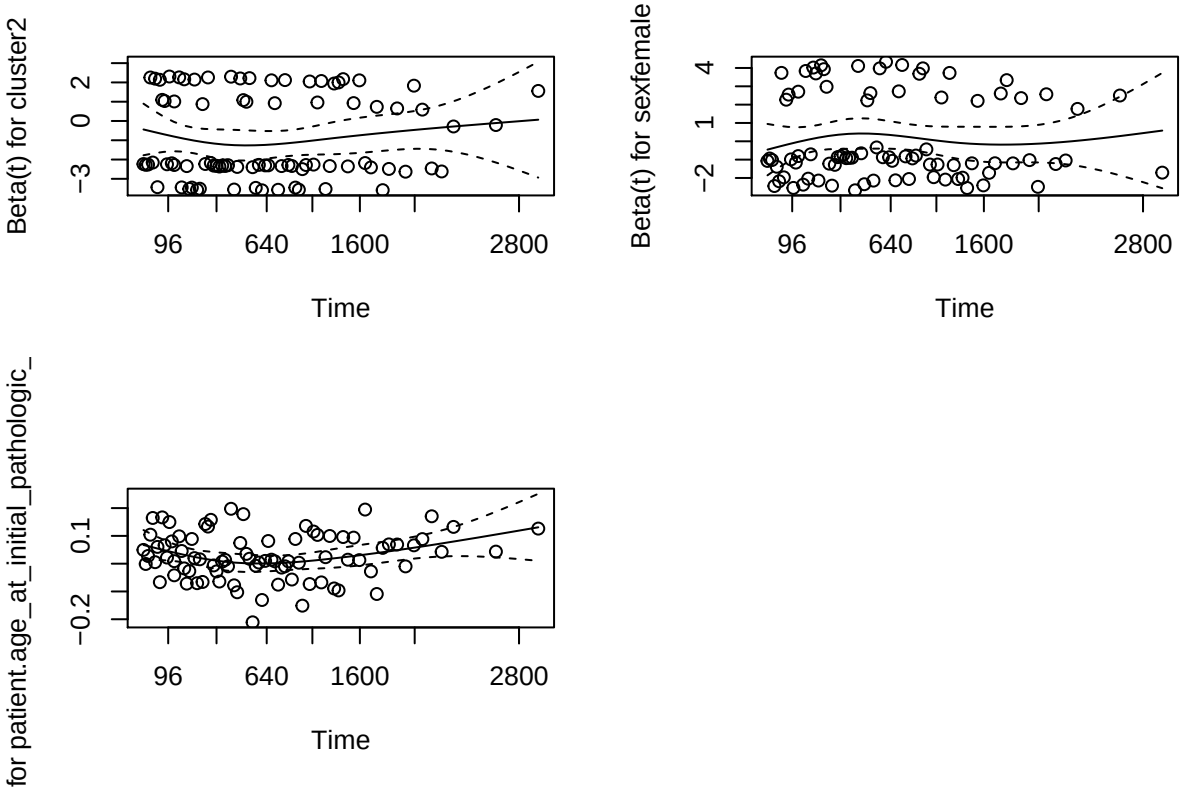
plotting scho residuals to check for

```
cox.zph(fit = coxph.fit)
```

non-proportional hazards – significance implies non-proportionality

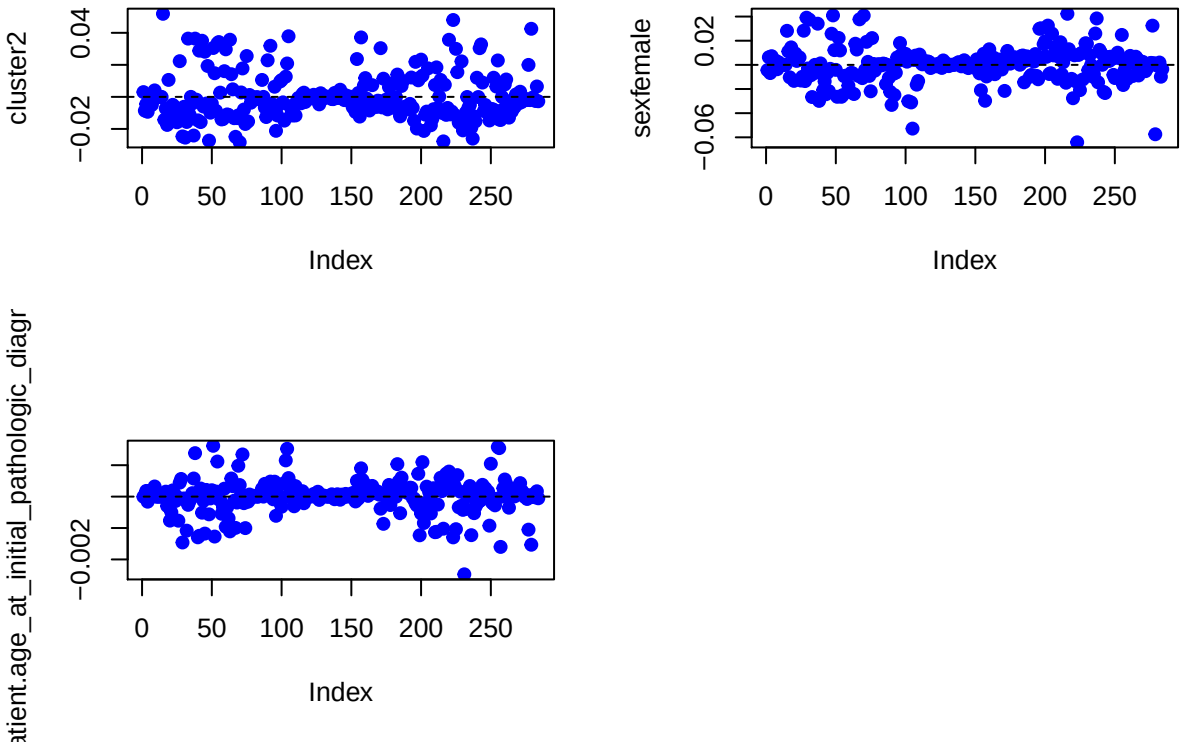
```
##                                                    rho    chisq     p
## cluster2                                       0.055079 2.71e-01 0.603
## sexfemale                                      0.000917 7.58e-05 0.993
## patient.age_at_initial_pathologic_diagnosis 0.082149 5.58e-01 0.455
## GLOBAL                                              NA 8.57e-01 0.836
```

```
par(mfrow=c(2,2))
plot(cox.zph(fit = coxph.fit))
par(mfrow=c(1,1))
```



Checking for influential observations (outliers)

```
dfbeta <- residuals(coxph.fit, type = 'dfbeta') ## Dataframe of change in coefficients as each individua

par(mfrow=c(2,2))
for(j in 1:3){
  plot(dfbeta[,j],ylab=names(coef(coxph.fit))[j],
       pch=19,col='blue')
  abline(h=0,lty=2)
}
par(mfrow=c(1,1))
```




```
##  No terribly influential points
```

Checking for linearity in the covariates using plots of

martingale residuals against the individual covariates

NOTE: This is not necessary for binary variables

so we only check it in our age of initial diagnosis

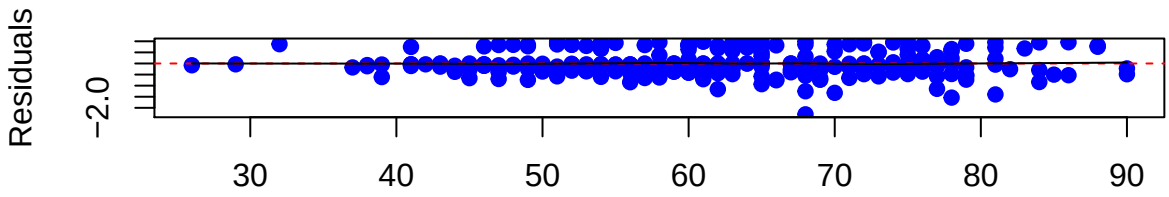
```
martingale.resids <- residuals(coxph.fit,type = 'martingale')
seq(1,nrow(clinic.kirc.snf.group))[!(seq(1,nrow(clinic.kirc.snf.group)) %in% as.numeric(names(martingale
```

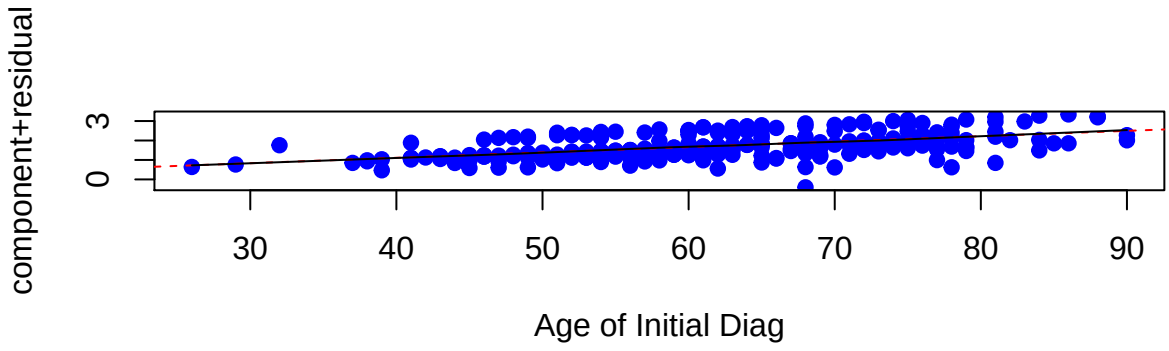
covariate

```
## integer(0)
```

```
  ## checking that there are no missing residual values using the indices of the martingale residuals
    # (the value of 'integer(0)' being returned tells us we haven't missed anything)

par(mfrow=c(2,1))
```

```
## Null plot for residuals:
plot(y = martingale.resids,
     x = clinic.kirc.snf.group$patient.age_at_initial_pathologic_diagnosis,
     ylab = 'Residuals', xlab = 'Age of Initial Diag', pch= 19, col = 'blue')
abline(h=0,lty=2,col='red')
lines(lowess(x = clinic.kirc.snf.group$patient.age_at_initial_pathologic_diagnosis,
             y = martingale.resids, iter = 0))

## Component-plus-residual plot


b <- coef(coxph.fit)[3]
x <- clinic.kirc.snf.group$patient.age_at_initial_pathologic_diagnosis
plot(x, b*x + martingale.resids,
     xlab='Age of Initial Diag',
     ylab="component+residual",
     pch = 19, col = 'blue')
abline(lm(b*x + martingale.resids ~ x), lty=2, col = 'red')
lines(lowess(x, b*x + martingale.resids, iter = 0))
```






```
  ## deviation of lowess line from 0-line and fit slope are
    # extremely small therefore linearity seems to hold

par(mfrow=c(1,1))
```

BONUS MATERIAL

Making predictions using our Cox PH model

```
# Suppose you now have a new individual you'd like to predict the survival of:
individual_new <- data.frame(cluster=factor(2),sex="male",patient.age_at_initial_pathologic_diagnosis=2
      ### Note the way each input variable has to be named the EXACT way it was fit in our cox model of
                           ## and the cluster value has to be in the "factor" form (set using the facto

predict(coxph.fit,individual_new,type="risk")
```

```
##         1
## 0.2202758
```

```
## this is the risk of your 21 year old group 2 male patient
            ## relative to the average of your sample
```