mirTarRnaSeq

26 October 2021

Contents

1 Introduction . . . 3
  1.1 Data upload . . . 3

2 Part1 - miRNA mRNA regressions across sample cohorts . . . 3
  2.1 Uploading data into the application. The example data can be found in the test folder under package. . . . 3
  2.2 Get miRanda file . . . 3
  2.3 Select miRNA . . . 4
  2.4 Combine the mRNA and miRNA file and define boundaries . . . 4
  2.5 Run a one to one miRNA/mRNA gaussian regression model . . . 4
  2.6 Running Gausian model over all individual miRNA mRNA models . . . 4
  2.7 Running poisson model over all individual miRNA mRNA models. . . 5
  2.8 Running negative binomial model over all individual miRNA mRNA models . . . 6
  2.9 Running zero inflated negative binomial model over all individual miRNA mRNA models . . . 7
  2.10 Running zero inflated poisson binomial model over all individual miRNA mRNA models . . . 7
  2.11 Including Plots for all models to decide which to use . . . 8
  2.12 The user can decide to use runModels() with glm_multi(). . . . 9
  2.13 The user can decide to use runModels() with specific models . . . 9
  2.14 Running all miRNA and mRNA combinations . . . 9

3 Part2 - Identify miRNA mRNA correlations across 3 or more time points. . . . 10
  3.1 Get mRNAs . . . 11
  3.2 Get mRNAs with particular fold change . . . 11
  3.3 Get all miRNAs. . . . 11
  3.4 Get mRNA miRNA correlation. . . . 11
  3.5 Make a background distribution correlation . . . 11
  3.6 Plot density plots . . . 11
  3.7 Get correlations below threshold . . . 12

3.8 Get mouse miRanda data 12
3.9 mRNA miRNA correlation heatmap 12
3.10 get intersection of miRanda 12

4 Part3 - Identify significant miRNA mRNA relationships for 2 time points 13

4.1 Import data 13
4.2 Only look for time point difference 0-5 13
4.3 Get fold changes above thereshold 13
4.4 Estimate miRNA mRNA differences based on Fold Change 14
4.5 Make background distribution 14
4.6 miRanda data import 14
4.7 Identify relationships below threshold 14
4.8 miRanda intersection with results 14
4.9 Make dataframe and plots 14
4.10 mRNA miRNA heatmap of miRNA mRNA FC differences 15

1 Introduction

mirTarRnaSeq is a package for miRNA and mRNA interaction analysis through regression and correlation approaches supporting various modeling approaches (gaussian, poisson, negative binomial, zero inflated poisson or negative binomial models for the data). miRNA and mRNA seq data are analysed from the same experiment (condition or time point data) and mRNA targets of the miRNAs from same experiment will be identified using statistical approaches.

The example data set for the first approach is 25 matching miRNA and mRNA EBV positive samples from TCGA identified as high EBV load based on Movassagh et al, Scientific Reports, 2019 paper. We attempt to identify the EBV miRNA targets on EBV genome (part1).

The second example set set is the simulated mouse fibroblast differentiated to muscle cells in three time points. Here, we try to identify mRNA targets of miRNA expressed at various time points (parts 2 and 3).

1.1 Data upload

1.1.1 mirTarRnaSeq accepts data in dataframe or table formats

- For the first approach (Part1) we use a table of expressed mRNA genes in EBV from TCGA stomach cancer samples with high levels of EBV miRNA expression.
- Next we use a list of normalized (tpm) EBV miRNA expression data from the same samples.The user has the option to use count data and model accordingly or use tzTrans() function for zscore normalization and then model.
- For the second part (Part2, Part3) of the experiment we are using two tables of differentially expressed mRNA and miRNA sequencing fold change results from mouse time point specific differentiation experiments.
- The example data is available at https://doi.org/10.5281/zenodo.5234278 .

2 Part1 - miRNA mRNA regressions across sample cohorts

2.1 Uploading data into the application. The example data can be found in the test folder under package.

```
DiffExp<-read.table("test/EBV_mRNA.txt", as.is=TRUE, header=TRUE, row.names=1)
miRNAExp<-read.table("test/EBV_miRNA.txt", as.is=TRUE, header=TRUE, row.names=1)
```

2.2 Get miRanda file

We currently support miRanda runs (potential miRNA target parsing by score,interaction energy, and interaction or miRNA length) on seven species ("Human", "Mouse", "Drosophila", "C.elegans", "Epstein_Barr" (EBV), "Cytomegalovirus" (CMV) and "Kaposi_Sarcoma" (KSHV)). We also support the viral miRNAs targeting human genes for EBV "Epstein_Barr_Human", CMV "CMV_Human" and KSHV "KSHV_Human". 1) Here we first import the relevant miRanda file . 2) We only keep targets which are also targets of EBV miRNAs based on our EBV miRanda file.

```
miRanda <- getInputSpecies("Epstein_Barr", threshold = 140)
DiffExpmRNASub <- miRanComp(DiffExp, miRanda)
```

2.3 Select miRNA

```
miRNA_select<-c("ebv-mir-bart9-5p")
```

2.4 Combine the mRNA and miRNA file and define boundaries

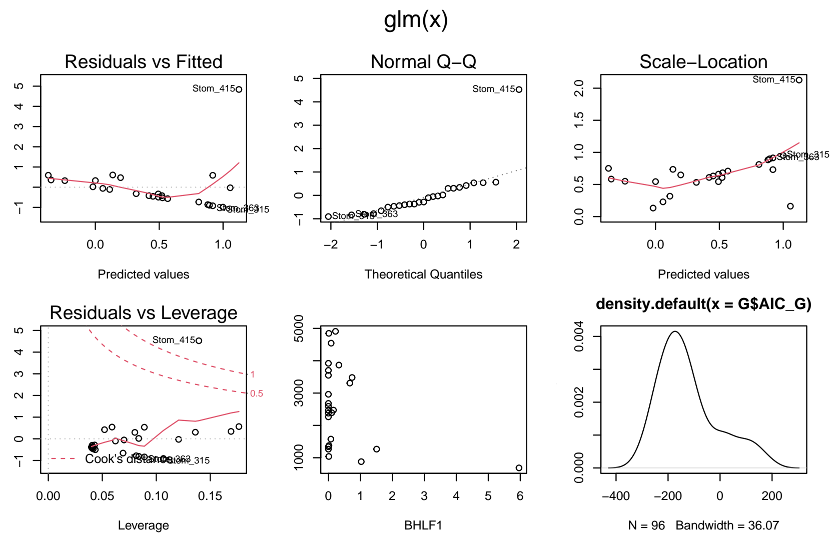
```
Combine <- combiner(DiffExp, miRNAExp, miRNA_select)
geneVariant <- geneVari(Combine, miRNA_select)
```

2.5 Run a one to one miRNA/mRNA gaussian regression model

```
j <- runModel(`LMP-1` ~ `ebv-mir-bart9-5p`,
              Combine, model = glm_poisson(),
              scale = 100)
# Association between between mRNA and miRNA
print(modelTermPvalues(j))
#> `ebv-mir-bart9-5p`
#>       9.344338e-06
```

2.6 Running Gausian model over all individual miRNA mRNA models

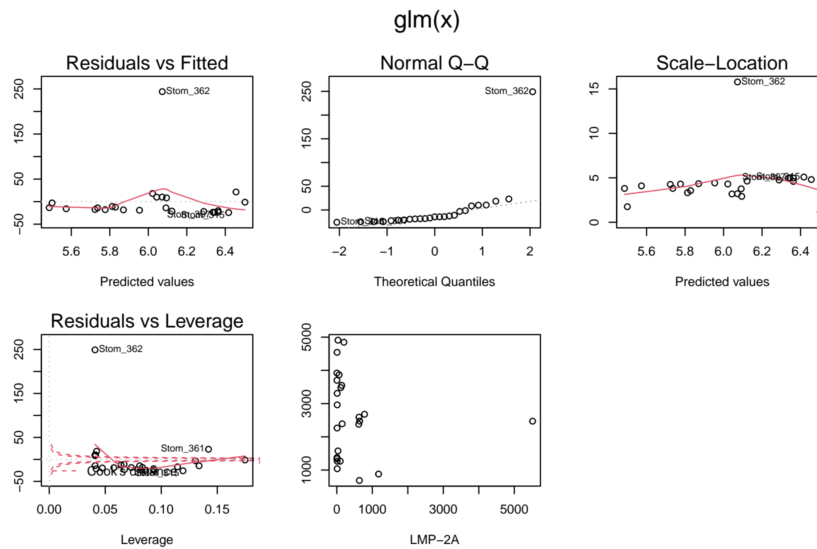
```
blaGaus <- runModels(Combine,
                     geneVariant, miRNA_select,
                     family = glm_gaussian(),
                     scale = 100)
par(oma=c(2,2,2,2))
par(mfrow=c(2,3),mar=c(4,3,3,2))
plot(blaGaus[["all_models"]][["BHLF1"]])
plot(modelData(blaGaus[["all_models"]][["BHLF1"]]))
#To test AIC model performance
G <- do.call(rbind.data.frame, blaGaus[["AICvalues"]])
names(G) <- c("AIC_G")
#Low values seems like a reasonable model
plot(density(G$AIC_G))
```




```
GM <- melt(G)
#> No id variables; using all as measure variables
```

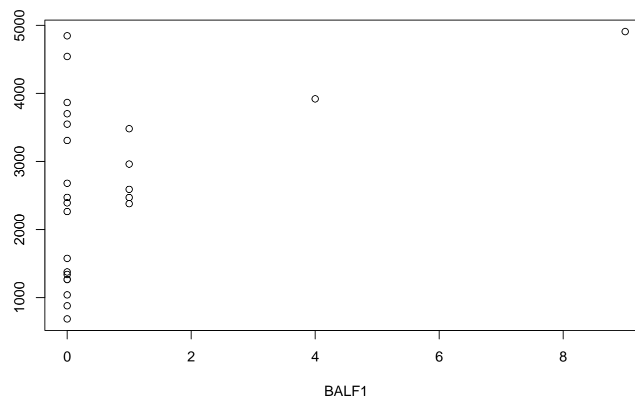
2.7 Running poisson model over all individual miRNA mRNA models

```
blaPois <- runModels(Combine,
                     geneVariant, miRNA_select,
                     family = glm_poisson(),
                     scale = 100)
par(oma=c(2,2,2,2))
par(mfrow=c(2,3),mar=c(4,3,3,2))
plot(blaPois[["all_models"]][["LMP-2A"]])
plot(modelData(blaPois[["all_models"]][["LMP-2A"]]))
P <- do.call(rbind.data.frame, blaPois[["AICvalues"]])
names(P) <- c("AIC_Po")
PM <- melt(P)
#> No id variables; using all as measure variables
```



2.8 Running negative binomial model over all individual miRNA mRNA models

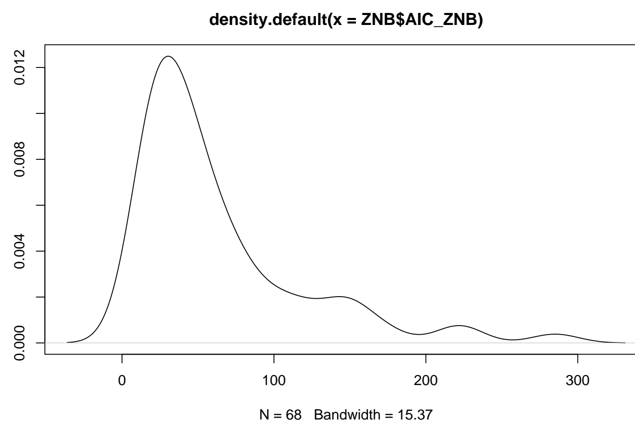
```
blaNB <- runModels(Combine,
                   geneVariant, miRNA_select,
                   family = glm_nb(), scale = 100)
par(mar=c(4,3,3,2))
plot(modelData(blaNB[["all_models"]][["BALF1"]]))
```




```
B <- do.call(rbind.data.frame, blaNB[["AICvalues"]])
names(B) <- c("AIC_NB")
BM <- melt(B)
#> No id variables; using all as measure variables
```

2.9 Running zero inflated negative binomial model over all individual miRNA mRNA models

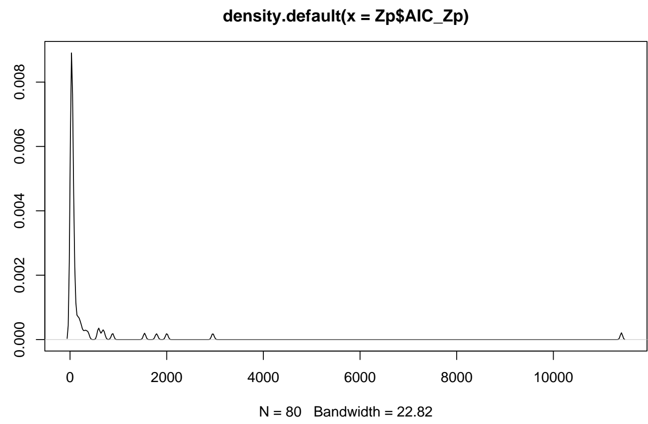
```
blazeroinflNB <- runModels(Combine, geneVariant,
                            miRNA_select,
                            family = glm_zeroinfl(dist = "negbin"),
                            scale = 100)
# To test AIC model performance
ZNB <- do.call(rbind.data.frame, blazeroinflNB[["AICvalues"]])
names(ZNB) <- c("AIC_ZNB")
par(mar=c(4,3,3,2))
plot(density(ZNB$AIC_ZNB))
```




```
ZNBM<-melt(ZNB)
#> No id variables; using all as measure variables
```

2.10 Running zero inflated poisson binomial model over all individual miRNA mRNA models

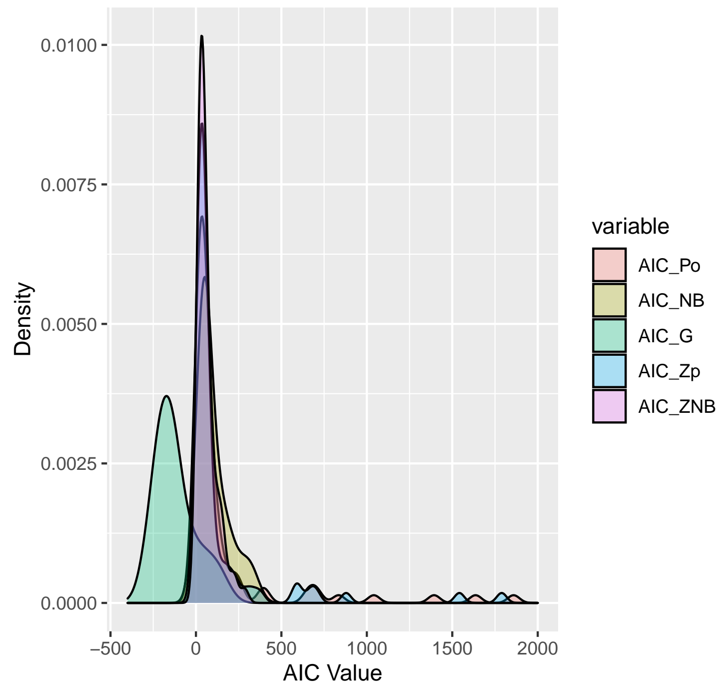
```
blazeroinfl <- runModels(Combine, geneVariant,
                          miRNA_select,
                          family = glm_zeroinfl(),
                          scale = 100)
# To test AIC model performance
Zp <- do.call(rbind.data.frame, blazeroinfl[["AICvalues"]])
names(Zp) <- c("AIC_Zp")
par(mar=c(4,3,3,2))
plot(density(Zp$AIC_Zp))
```




```
ZpM <- melt(Zp)
```

2.11 Including Plots for all models to decide which to use

```
bindM <- rbind(PM, BM, GM, ZpM, ZNBM)
p2 <- ggplot(data = bindM, aes(x = value, group = variable,
                               fill = variable)) +
  geom_density(adjust = 1.5, alpha = .3) +
  xlim(-400, 2000)+
  ggtitle("Plot of of AIC for ebv-mir-bart9-5p regressed all mRNAs ")+
  ylab("Density")+ xlab ("AIC Value")
p2
```



2.12 The user can decide to use runModels() with glm_multi()

where all available models will be run, the AICs will be compared and the best model will be chosen based on the miRNA-mRNA mocel AIC score. In the example bellow we are using the mode= "multi" option for combination of 2 miRNAs (multivariate model) for interaction model the user can choose the mode= "inter" option.

```
miRNA_select<-c("ebv-mir-bart9-5p","ebv-mir-bart6-3p")
Combine <- combiner(DiffExp, miRNAExp, miRNA_select)
geneVariant <- geneVari(Combine, miRNA_select)
MultiModel <- runModels(Combine, geneVariant,
                        miRNA_select, family = glm_multi(),
                        mode="multi", scale = 10)
#print the name of the models used for the analysis
print(table(unlist(lapply(MultiModel$all_models, modelModelName))))
TRUE
TRUE glm_gaussian       glm_nb
TRUE           95            1
```

2.13 The user can decide to use runModels() with specific models

where all available models will be run, the AICs will be compared and the best model will be chosen based on the miRNA-mRNA mocel AIC score. In the example bellow we are using the mode= "inter" option for combination of 2 miRNAs (multivariate model) for interaction model the user can choose the mode= "multi" option.

```
miRNA_select<-c("ebv-mir-bart9-5p","ebv-mir-bart6-3p")
Combine <- combiner(DiffExp, miRNAExp, miRNA_select)
geneVariant <- geneVari(Combine, miRNA_select)
InterModel <- runModels(Combine,
                        geneVariant, miRNA_select,
                        family = glm_multi(
                          models=list(glm_gaussian,
                          glm_poisson())),mode="inter", scale = 10)
#print the name of the models used for the analysis
print(table(unlist(lapply(InterModel$all_models, modelModelName))))
TRUE
TRUE glm_gaussian
TRUE           96
```

2.14 Running all miRNA and mRNA combinations

Note for "inter" and "multi" mode options we only support cobinations of 2.

```
vMiRNA<-rownames(miRNAExp)
All_miRNAs_run<-runAllMirnaModels(mirnas =vMiRNA[1:5] ,
                                  DiffExpmRNA = DiffExpmRNASub,
                                  DiffExpmiRNA = miRNAExp,
                                  miranda_data = miRanda,prob=0.75,
                                  cutoff=0.05,fdr_cutoff = 0.1, method = "fdr",
                                  family = glm_multi(), scale = 2, mode="multi")
```

```
TRUE 1: ebv-mir-bart1-3p, ebv-mir-bart1-5p
TRUE 2: ebv-mir-bart1-3p, ebv-mir-bart10-3p
TRUE 3: ebv-mir-bart1-3p, ebv-mir-bart10-5p
TRUE 4: ebv-mir-bart1-3p, ebv-mir-bart11-3p
TRUE 5: ebv-mir-bart1-5p, ebv-mir-bart10-3p
TRUE 6: ebv-mir-bart1-5p, ebv-mir-bart10-5p
TRUE 7: ebv-mir-bart1-5p, ebv-mir-bart11-3p
TRUE 8: ebv-mir-bart10-3p, ebv-mir-bart10-5p
TRUE 9: ebv-mir-bart10-3p, ebv-mir-bart11-3p
TRUE 10: ebv-mir-bart10-5p, ebv-mir-bart11-3p

hasgenes <- lapply(All_miRNAs_run, function(x) nrow(x$SigFDRGenes)) > 0
All_miRNAs_run <- All_miRNAs_run[hasgenes]
print(table(unlist(lapply(All_miRNAs_run$all_models, modelModelName))))
TRUE < table of extent 0 >
print(table(unlist(lapply
                   (All_miRNAs_run[[1]][["FDRModel"]][["all_models"]],
                     modelModelName))))
TRUE
TRUE          glm_gaussian                glm_nb glm_zeroinfl_poisson
TRUE                    79                     5                    1
print(
  table(
    unlist(
      lapply(
        All_miRNAs_run[["ebv-mir-bart1-5p and ebv-mir-bart11-3p"]][["FDRModel"]][["all_models"]],
        modelModelName))))
TRUE
TRUE          glm_gaussian                glm_nb  glm_zeroinfl_negbin
TRUE                    79                     3                    2
TRUE glm_zeroinfl_poisson
TRUE                     1
```

3 Part2 - Identify miRNA mRNA correlations across 3 or more time points

Load files from test directory or you can load individually and feed them in separately in a list: list[(mRNA1,mRNA2,mRNA)]

```
files <- local({
  filenames <- list.files(path="test", pattern="^.*\\.txt$", full.names=TRUE)
  files <- lapply(filenames, read.table, as.is=TRUE, header=TRUE, sep="\t")
  names(files) <- gsub("^.*/(.*)\\.txt$", "\\1", filenames)
  return(files)
})
```

3.1 Get mRNAs

```
mrna_files <- files[grep("^mRNA", names(files))]
```

3.2 Get mRNAs with particular fold change

```
mrna_files <- files[grep("^mRNA", names(files))]
mrna <- one2OneRnaMiRNA(mrna_files, pthreshold = 0.05)$foldchanges
```

3.3 Get all miRNAs

```
mirna_files <- files[grep("^miRNA", names(files))]
mirna <- one2OneRnaMiRNA(mirna_files)$foldchanges
```

3.4 Get mRNA miRNA correlation

```
corr_0 <- corMirnaRna(mrna, mirna,method="pearson")
```

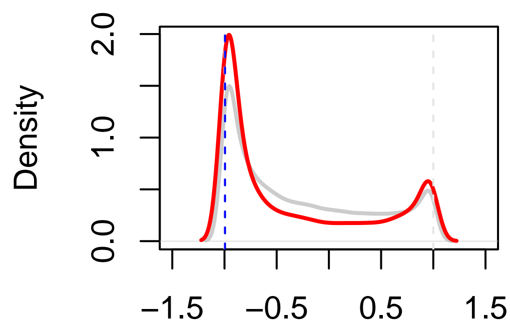
3.5 Make a background distribution correlation

```
outs <- sampCorRnaMirna(mrna, mirna,method="pearson",
                        Shrounds = 100, Srounds = 1000)
```

3.6 Plot density plots

Density plot for background and corrs in our data. Note grey is the background distribution and red is the actual data.

```
#Draw density plot
mirRnaDensityCor(corr_0, outs)
```



3.7 Get correlations below threshold

```
#Identify significant correlation
sig_corrs <- threshSig(corr_0, outs,pvalue = 0.05)
```

3.8 Get mouse miRanda data

```
#Import concordant miRanda file
miRanda <- getInputSpecies("Mouse", threshold = 150)
```

3.9 mRNA miRNA correlation heatmap

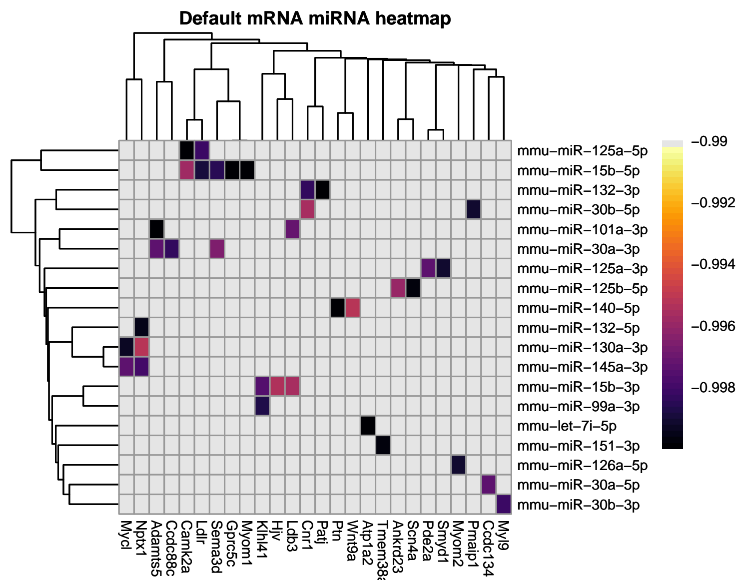
Correlation heatmap for cor equal or less than -0.7. Note upperbound for heatmap should be always less than the correlation threshold.

```
#Extract your target correlations based on miRanda and correlation threshold.
newcorr <- corMirnaRnaMiranda(mrna, mirna, -0.7, miRanda)
mirRnaHeatmap(newcorr,upper_bound = -0.6)
```

3.10 get intersection of miRanda

Get miRanda intersection and significant miRNA and mRNA interactions and the plot it.

```
#Make final results file for significant
#correlations intersecting with miRanda file
results <- miRandaIntersect(sig_corrs, outs, mrna, mirna, miRanda)
#Draw correlation heatmap
p<- mirRnaHeatmap(results$corr,upper_bound =-0.99)
p
```



4 Part3 - Identify significant miRNA mRNA relationships for 2 time points

4.1 Import data

```
files <- local({
  filenames <- list.files(path="test", pattern="^.*\\.txt$", full.names=TRUE)
  files <- lapply(filenames, read.table, as.is=TRUE, header=TRUE, sep="\t")
  names(files) <- gsub("^.*/(.*)\\.txt$", "\\1", filenames)
  return(files)
})
```

4.2 Only look for time point difference 0-5

```
mirna_files <- files[grep("^miRNA0_5", names(files))]
mrna_files <- files[grep("^mRNA0_5", names(files))]
```

4.3 Get fold changes above thereshold

```
# Parse Fold Change Files for P value and Fold Change.
mrna <- one2OneRnaMiRNA(mrna_files, pthreshold = 0.05)$foldchanges
mirna <- one2OneRnaMiRNA(mirna_files)$foldchanges
```

4.4 Estimate miRNA mRNA differences based on Fold Change

```
# Estimate the miRNA mRNA FC differences for your dataset
inter0 <- twoTimePoint(mrna, mirna)
```

4.5 Make background distribution

```
#Make a background distribution for your miRNA mRNA FC differences
outs <- twoTimePointSamp(mrna, mirna,Shrounds = 10 )
```

4.6 miRanda data import

```
#Import concordant miRanda file
miRanda <- getInputSpecies("Mouse", threshold = 140)
```

4.7 Identify relationships below threshold

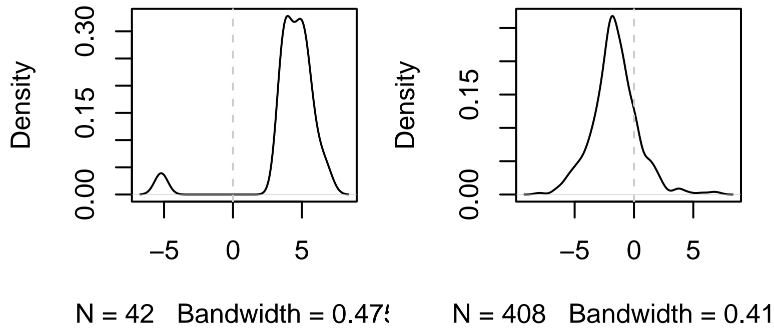
```
#Identify miRNA mRNA relationships bellow a P value threshold, default is 0.05
sig_InterR <- threshSigInter(inter0, outs)
```

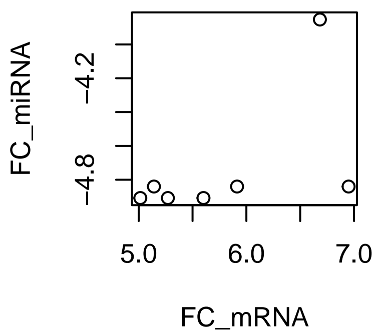
4.8 miRanda intersection with results

```
#Intersect the mirRanda file with your output results
results <- mirandaIntersectInter(sig_InterR, outs, mrna, mirna, miRanda)
```

4.9 Make dataframe and plots

```
#Create a results file for heatmap
final_results <- finInterResult(results)
#Draw plots of miRNA mRNA fold changes for your results file
par(mar=c(4,4,2,1))
drawInterPlots(mrna,mirna,final_results)
```





4.10 mRNA miRNA heatmap of miRNA mRNA FC differences

Heatmap for p value significant miRNA mRNA fold change differences when compared to backgound

```
CorRes<-results$corrs
#Draw heatmap for miRNA mRNA significant differences
#Note: you do not have to use the upper_bound function unless you want
#investigate a particular range for miRNA mRNA differences/relationships
mirRnaHeatmapDiff(CorRes,upper_bound = 9.9)
```