Graph Complexity Analysis Identifies an ETV5 Tumor-Specific Network in Human and Murine Low-Grade Glioma

Christina Duron, Jo Hardin, and Ami Radunskaya

January 23, 2018

```
knitr::opts_chunk$set(echo = TRUE, fig.height = 3.5, size = "small", cache=TRUE)
options(stringsAsFactors = FALSE, digits=3)

require(tidyr)
require(ggplot2)
require(scales)
require(readr)
require(dplyr)
require(xtable)
require(readxl)
# source("https://bioconductor.org/biocLite.R")
# biocLite("illuminaHumanv4.db")  # install Illumina database
# biocLite("hgu133plus2.db")  # install Affymetrix database
# biocLite("DESeq2")
# biocLite("goseq")
# biocLite("org.Mm.eg.db")
require(illuminaHumanv4.db)
require(hgu133plus2.db)
require(DESeq2)
library(gdata)
library(Matrix)
library(igraph)
library(car)
library(goseq)
library(org.Mm.eg.db)
library(GO.db)
```

```
sessionInfo()

## R version 3.4.2 (2017-09-28)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.2
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
```

```
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] GO.db_3.5.0                org.Mm.eg.db_3.5.0
##  [3] goseq_1.30.0               geneLenDataBase_1.14.0
##  [5] BiasedUrn_1.07             car_2.1-6
##  [7] igraph_1.1.2               Matrix_1.2-12
##  [9] gdata_2.18.0               DESeq2_1.18.1
## [11] SummarizedExperiment_1.8.1 DelayedArray_0.4.1
## [13] matrixStats_0.52.2         GenomicRanges_1.30.1
## [15] GenomeInfoDb_1.14.0        hgu133plus2.db_3.2.3
## [17] illuminaHumanv4.db_1.26.0  org.Hs.eg.db_3.5.0
## [19] AnnotationDbi_1.40.0       IRanges_2.12.0
## [21] S4Vectors_0.16.0           Biobase_2.38.0
## [23] BiocGenerics_0.24.0        readxl_1.0.0
## [25] xtable_1.8-2               dplyr_0.7.4
## [27] readr_1.1.1                scales_0.5.0
## [29] ggplot2_2.2.1              tidyr_0.7.2
## [31] knitr_1.18
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-131             bitops_1.0-6
##  [3] pbkrtest_0.4-7           bit64_0.9-7
##  [5] httr_1.3.1               progress_1.1.2
##  [7] RColorBrewer_1.1-2       tools_3.4.2
##  [9] backports_1.1.2          R6_2.2.2
## [11] rpart_4.1-12             mgcv_1.8-23
## [13] Hmisc_4.1-1              DBI_0.7
## [15] lazyeval_0.2.1           colorspace_1.3-2
## [17] nnet_7.3-12              prettyunits_1.0.2
## [19] RMySQL_0.10.13           gridExtra_2.3
## [21] bit_1.1-12               compiler_3.4.2
## [23] quantreg_5.34            htmlTable_1.11.1
## [25] SparseM_1.77             rtracklayer_1.38.2
## [27] checkmate_1.8.5          genefilter_1.60.0
## [29] Rsamtools_1.30.0         stringr_1.2.0
## [31] digest_0.6.14            foreign_0.8-69
## [33] minqa_1.2.4              XVector_0.18.0
## [35] base64enc_0.1-3          pkgconfig_2.0.1
## [37] htmltools_0.3.6          lme4_1.1-15
## [39] highr_0.6                htmlwidgets_0.9
## [41] rlang_0.1.6              rstudioapi_0.7
## [43] RSQLite_2.0              bindr_0.1
## [45] BiocParallel_1.12.0      gtools_3.5.0
## [47] acepack_1.4.1            RCurl_1.95-4.10
## [49] magrittr_1.5             GenomeInfoDbData_1.0.0
## [51] Formula_1.2-2            Rcpp_0.12.14
## [53] munsell_0.4.3            stringi_1.1.6
## [55] MASS_7.3-48              zlibbioc_1.24.0
## [57] plyr_1.8.4               grid_3.4.2
## [59] blob_1.1.0               lattice_0.20-35
```

```
## [61] Biostrings_2.46.0        splines_3.4.2
## [63] GenomicFeatures_1.30.0   annotate_1.56.1
## [65] hms_0.4.0                locfit_1.5-9.1
## [67] pillar_1.1.0             biomaRt_2.34.1
## [69] geneplotter_1.56.0       XML_3.98-1.9
## [71] glue_1.2.0               evaluate_0.10.1
## [73] latticeExtra_0.6-28      data.table_1.10.4-3
## [75] nloptr_1.0.4             MatrixModels_0.4-1
## [77] cellranger_1.1.0         gtable_0.2.0
## [79] purrr_0.2.4              assertthat_0.2.0
## [81] survival_2.41-3          tibble_1.4.1
## [83] GenomicAlignments_1.14.1 memoise_1.1.0
## [85] bindrcpp_0.2             cluster_2.0.6
```

1 Normalizing the Data

1.1 Data

Data from Peter Sims, 5/21/16. All columns have been included (data from previous analysis as well as newer samples). We continue to us FF_FMC_matrixv2.xlx to get the REFSEQ_ID gene information.

- C57 = C57 black 6 mice
- F18C = germline F18C mutation in NF1
- F21C = germline F21C mutation in NF1
- FF = flox/flox control, 3 month-old (control for all mutant mice except FMOC)
- FF_6mo = flox/flox control, 6 month-old (control for FMOC mice)
- FMC_carbo = Gfap-Cre NF1 mutant mice treated with carboplatin
- FMC = Gfap-Cre NF1 mutant mice
- FMC_Min = Gfap-Cre NF1 mutant mice treated with minocycline
- FMC_rapa = Gfap-Cre NF1 mutant mice treated with rapamycin
- FMOC = Olig2-Cre NF1 mutant mice
- FMPC = Gfap-Cre NF1/PTEN mutant mice
- FMPC_NVP = Gfap-Cre NF1/PTEN mutant mice treated with NVP-BKM120

```
micefinal <- read.delim("BTECfinal_cts.txt")
micefinal <- micefinal %>% dplyr::select(gene, dplyr::starts_with("FF_fe"), dplyr::starts_with("FF_ma"),
                                dplyr::starts_with("FMC_fe"), dplyr::starts_with("FMC_ma"))

geneinfo <- read_excel("FF_FMC_matrixv2.xlsx",
                       col_names=c("gene", "REFSEQ_ID", paste("FF", 1:5, sep=""),
                                   paste("FMC",1:5, sep=""), "X", paste("FFk",1:5,sep=""),
                                   paste("FMCk",1:5,sep="")))
geneinfo <- geneinfo %>% dplyr::select(gene, REFSEQ_ID)

micefinal <- micefinal %>% dplyr::left_join(geneinfo, by="gene")
```

1.2 DESeq2 Normalization

```
cond <- factor(c(rep("FF", 4), rep("FMC", 11)))
dds <- DESeq2::DESeqDataSetFromMatrix(micefinal[,-c(1,17)], DataFrame(cond), ~ cond)
dds <- DESeq2::DESeq(dds, betaPrior=TRUE, fitType = "parametric")
res <- results(dds)  # Diff Exp results if we want/need the p-values
dds.data <- counts(dds, normalized=TRUE)
miceout <- data.frame(gene=toupper(micefinal$gene), REFSEQ_ID=micefinal$REFSEQ_ID, dds.data)
miceps <- data.frame(gene=toupper(micefinal$gene), REFSEQ_ID=micefinal$REFSEQ_ID, res)
```

2 Network Betweeness

2.1 Cleaning the Data

We analyze the data taken from mice in both the normal and tumor groups. Note that the normal data contains only 4 data points, while the tumor data contains 11 data points. These results come from the code written in the R files miceDESeqnorm.R and FindBetweeness.R.

After reading in the data and network, the gene names are converted to all uppercase to ease the analysis and the interactome network was set to be undirected.

```
network = read.table("gbm-sun-interactome.txt", header = TRUE,sep = "\t")

# Read in data.  Use the file without all of the header stuff
geodat = read.table("miceDESeqnorm.txt",header=TRUE,sep="\t")
geodat = geodat[,-2]
uppernames = toupper(geodat[,1]) #
geodat[,1] = uppernames
rn <-geodat[,1]
geodat <-geodat[,2:length(geodat[1,])]
row.names(geodat)=rn


# Split into normal and tumor
tumor_data = geodat[,5:length(geodat[1,])]
norm_data = geodat[,1:4]

library('igraph')
ghuman = graph.data.frame(network[,1:2], directed = FALSE)
ghuman = simplify(ghuman,remove.multiple=TRUE)


## Complexity Analysis
norm_zero_count <-as.matrix(apply(norm_data == 0, 1, sum)) # number of zeros in normal data
tumor_zero_count <- as.matrix(apply(tumor_data == 0, 1, sum)) # number of zeros in tumor data
```

Some of the data points of the genes contained zeros, and so, those genes that contained a certain number of zeros in either the normal and tumor groups were removed. Genes that had more than 2 zeros in the normal data or more than 5 zeros in the tumor data. Note that if one gene had two 2 zeros in the normal data, but less than 5 zeros in the tumor data, it was still removed, and vice versa. Therefore, in order for the gene to remain, it has 3 or 4 normal nonzero data points or 6 to 11 tumor nonzero data points.

```
norm_data <- norm_data[norm_zero_count <= 2 & tumor_zero_count <= 5,]
tumor_data <- tumor_data[norm_zero_count <= 2 & tumor_zero_count <= 5,]
```

In addition, any genes that are not connected are removed from both networks. Upon simplifying the network and cleaning up the data, both the normal and tumor sets reduce from 23420 to 11055 genes. Note that each set contains the same 11055 genes.

In biological networks, measures of node centrality are useful in detecting genes with critical functional roles, as it can indicate which genes occupy critical positions in the network. What is deemed "critical" determines which measure, or measures, one employs.

2.2 Betweeness

In the following analysis, the betweeness centrality is used to identify potential biologically meaningful genes, while other centrality measures - closeness, degree, and weighted node connectivity - are utilized for further validation.

The **betweenness centrality**, b_i, is a measurement of the number of shortest paths connecting any two nodes, j and k, which pass through node i, where node i can be thought of as a "bridging" node in the network. It can be calculated as

$$b_i = \sum_{i \neq j \neq k} \frac{n_{j,k}(i)}{n_{j,k}}$$

where $n_{j,k}$ is the total number of shortest paths connecting nodes j and k, and $n_{j,k}(i)$ is the number of shortest paths that pass through node i. Note that the distance of each path, d_{ij}, is one minus the absolute value of the correlation coefficient between genes i and j, where the correlation coefficient is taken to be the smaller value between the Spearman and Pearson coefficients. This measure quantifies the control of a node on the communication, or information flow, between other nodes, so nodes with large betweenness values are often called "high traffic" nodes and can be thought of as "bottleneck" genes. The genes with subtantially larger betweenness values in the tumor set as compared to those in the normal set are discussed in this analysis.

```
source('FindBetweeness.R')

## Betweeness
miceDESeqComplex = FindBetweeness(norm_data, tumor_data, ghuman,0) # run for original 11 tumor samples

# grab info from normal and tumor network (using original data of 4 normal and 11 tumor)

NB <- as.matrix(V(miceDESeqComplex$normalg)$between)
rownames(NB) = as.matrix(V(miceDESeqComplex$normalg)$name)
colnames(NB) = c('Normal Betweenness')

NC <- as.matrix(V(miceDESeqComplex$normalg)$close)
rownames(NC) = as.matrix(V(miceDESeqComplex$normalg)$name)
colnames(NC) = c('Normal Closeness')

#----#

TB <- as.matrix(V(miceDESeqComplex$tumorg)$between)
rownames(TB) = as.matrix(V(miceDESeqComplex$tumorg)$name)
colnames(TB) = c('Tumor Betweenness')

TC <- as.matrix(V(miceDESeqComplex$tumorg)$close)
rownames(TC) = as.matrix(V(miceDESeqComplex$tumorg)$name)
colnames(TC) = c('Tumor Closeness')
```

Histograms of the betweeness are plotted in Figure 1. The left-hand plot is for the normal data and the right-hand plot is for the tumor data.

```
par(mfrow=c(1,2))
hist(NB, xlab = "Normal Betweenness", main = NA, breaks = 100)
hist(TB, xlab = "Tumor Betweenness", main = NA, breaks = 100)
```

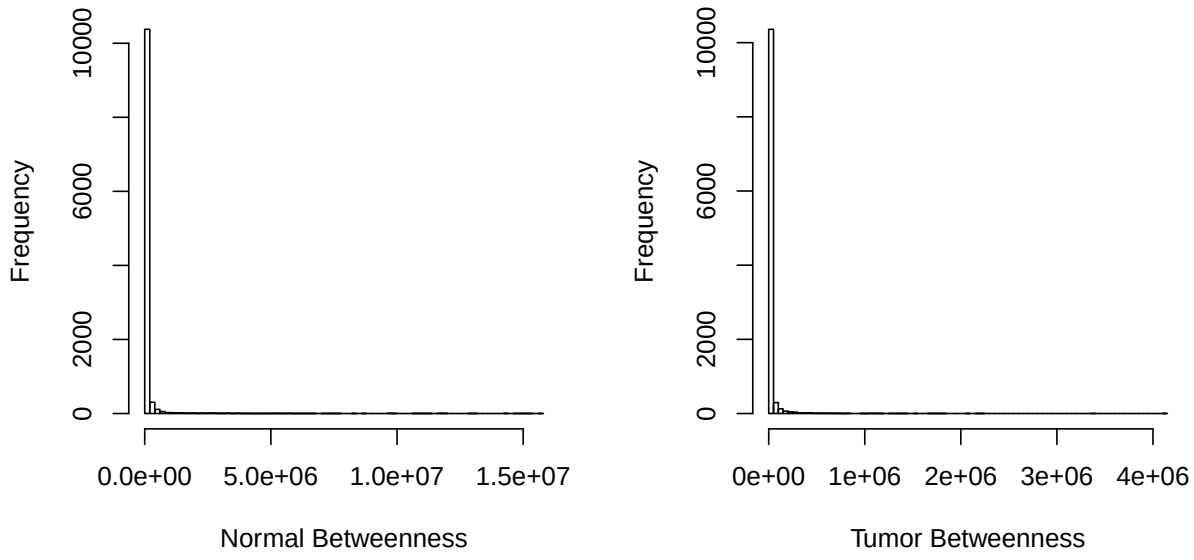


Figure 1a and 1b (left to right): Histograms of the betweenness centrality values for the normal and tumor data. Both are drastically skewed right.

Below are the summaries for the normal and tumor betweenness values.

```
summary(NB)

##  Normal Betweenness
##  Min.   :       0
##  1st Qu.:       0
##  Median :       0
##  Mean   :  104182
##  3rd Qu.:    6757
##  Max.   :15702357

summary(TB)

##  Tumor Betweenness
##  Min.   :      0
##  1st Qu.:     15
##  Median :    327
##  Mean   :  17577
##  3rd Qu.:   3613
##  Max.   :4116666
```

The normal data has a much larger betweenness value than the tumor data. Out of 11055 genes, 7500 genes have a betweenness value of 0 in the normal data, and 2130 genes have that value in the tumor data. This implies that 67.8% and 19.3% of the genes in the normal and tumor data can be made in this network without the aid of any intermediary gene. Additionally, 75% of genes have a betweenness value less than 5598 and 3556 in the normal and tumor data. Therefore, only a few genes have betweenness value that tend towards the maximum values, which imply that these are the "high-trafic" nodes in the network.

Figure 1 is a plot of the normal versus tumor betweenness values.

```r
# creating a dataframe that has both expression & betweeness data
# both sets have the same genes, so names should match up

norm_central = cbind(NB) # grab betweeness
tumor_central = cbind(TB) # grab betweeness for tumor
centrality_data = cbind(norm_central, tumor_central) # hold betweeness for norm and tumor


indx <- which(row.names(norm_data) %in% row.names(norm_central)) # grab indices of genes in both sets
norm_avg <- as.matrix(rowMeans(norm_data[indx,])) # average the values across the row for each gene
colnames(norm_avg) <- c('Normal Avg')
rownames(norm_avg) = row.names(norm_data)[indx]


indx <- which(row.names(tumor_data) %in% row.names(tumor_central)) # grab indices of genes in both sets
tumor_avg <- as.matrix(rowMeans(tumor_data[indx,])) # average the values across the row for each gene
colnames(tumor_avg) <- c('Tumor Avg')
rownames(tumor_avg) = row.names(tumor_data)[indx]


# re-arrange to match the same gene order as the centrality data
norm_avg  = as.matrix(norm_avg[row.names(centrality_data),])
tumor_avg = as.matrix(tumor_avg[row.names(centrality_data),])
colnames(norm_avg) <- c('Normal Avg')
colnames(tumor_avg) <- c('Tumor Avg')


# calculate fold-change values
fc <- tumor_avg/norm_avg
colnames(fc) <- c('FC Value')


# calculate ratio of tumor to normal betweenness
ratio_b <- as.matrix(centrality_data[,2]/centrality_data[,1])
colnames(ratio_b) <- c('T_B/N_B')

# store centrality measures, rpkm, and fc values in one place
data_plot = cbind(centrality_data, norm_avg, tumor_avg, fc, ratio_b)

# set threshold #1
nb_thresh = 1000000
tb_thresh = 1000000
indx <- which(data_plot[,1] >= nb_thresh | data_plot[,2] >= tb_thresh) # grab genes who meet one of the cutoff mark
data_thresh = as.matrix(data_plot[indx,]) # grab genes


# set threshold #2
indx <- which(data_thresh[,6] >= 1.1) # grab indices of genes whose ratio is >= 2
data_thresh = as.matrix(data_thresh[indx,]) # grab genes

tiff("Fig3.tiff", width = 6, height = 5, units = 'in', res = 800)
plot(data_plot[,1], data_plot[,2], xlab = "Normal Betweenness", ylab = "Tumor Betweenness", col = 'black', pch = 1,
par(new=TRUE)
plot(data_thresh[,1], data_thresh[,2], xlab = "Normal Betweenness", ylab = "Tumor Betweenness", col = 'red', pch =
dev.off()
```

```
## pdf
##   2

jpeg("Fig3.jpeg", width = 6, height = 5, units = 'in', res = 800)
plot(data_plot[,1], data_plot[,2], xlab = "Normal Betweenness", ylab = "Tumor Betweenness", col = 'black', pch = 1,
par(new=TRUE)
plot(data_thresh[,1], data_thresh[,2], xlab = "Normal Betweenness", ylab = "Tumor Betweenness", col = 'red', pch =
dev.off()

## pdf
##   2

indx = which(rownames(NC) %in% rownames(data_thresh))

tiff("Fig4.tiff", width = 6, height = 5, units = 'in', res = 800)
plot(NC[,1], TC[,1], xlab = "Normal Closeness", ylab = "Tumor Closeness", col = 'black', pch = 1, xlim = c(0,0.0025
par(new=TRUE)
plot(NC[indx,1], TC[indx,1], xlab = "Normal Closeness", ylab = "Tumor Closeness", col = 'red', pch = 16,  main = NA
dev.off()

## pdf
##   2

jpeg("Fig4.jpeg", width = 6, height = 5, units = 'in', res = 800)
plot(NC[,1], TC[,1], xlab = "Normal Closeness", ylab = "Tumor Closeness", col = 'black', pch = 1, xlim = c(0,0.0025
par(new=TRUE)
plot(NC[indx,1], TC[indx,1], xlab = "Normal Closeness", ylab = "Tumor Closeness", col = 'red', pch = 16,  main = NA
dev.off()

## pdf
##   2
```

We are interested in the genes which have a betweenness ratio of 1.1:1, meaning its tumor betweenness value is at least 1.1 as big as its normal betweenness value. The genes we are concerned are colored red in Figure 3 and are

```
row.names(data_thresh)

##  [1] "CEBPZ"   "ETV5"    "SPEN"    "ZCCHC14" "CAMTA1"  "CHD5"    "CERS2"
##  [8] "HNRNPAB" "ILF2"    "ZCCHC17" "ZC3H15"  "TULP4"   "PURB"    "RPL7"
## [15] "TCF3"    "TEAD1"   "CNBP"    "PRDM2"   "SARNP"   "ZRANB2"  "GCSH"
## [22] "IFT74"   "MYL12B"
```

Those 23 genes are colored in red in the plots in Figure 1, which highlights the betweenness values of the normal and tumor sets. The same 23 genes are colored in red in Figure 2, which highlights the WNC values (I think WNC just means closeness, but I'm not sure... ???) of the normal and tumor sets. Note that the scale in Figure 2 are adjusted by multiplying the WNC values by the standard error of each set, $\sqrt{(4)}$ for the normal data and $\sqrt{(11)}$ for the tumor data.

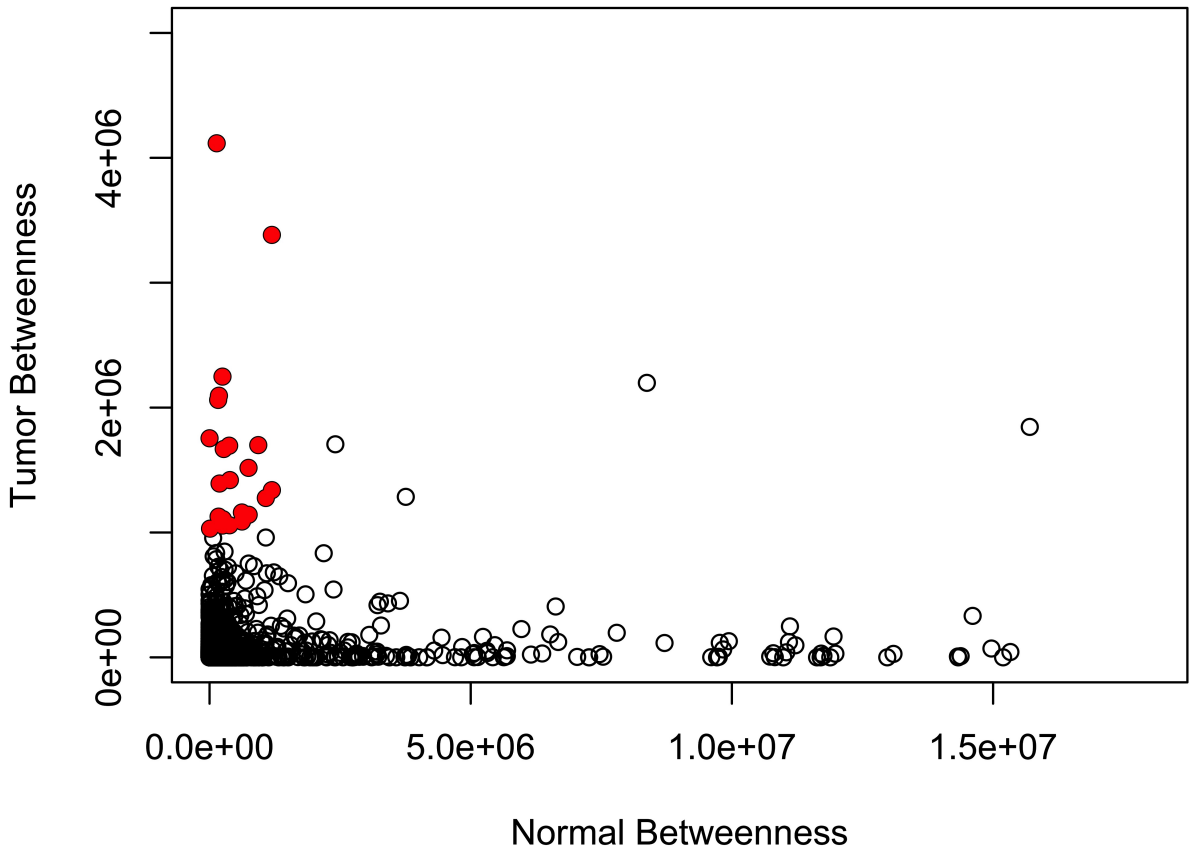


Figure 1: Betweenness plot of the genes.

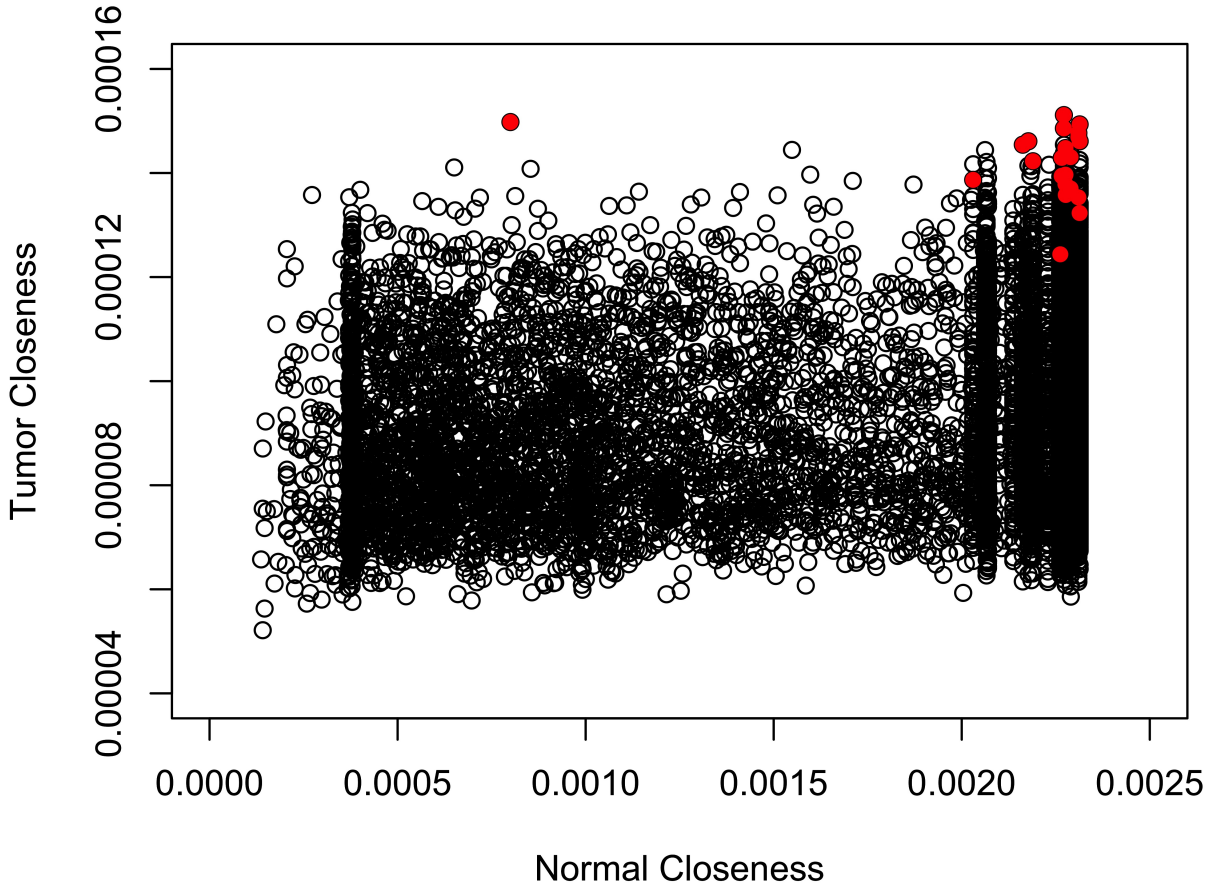


Figure 2: Closeness plot of the genes.

3 ETV5 Differential Expression

The goal of this analysis is to understand the differential expression of the genes which are *regulated* by genes identified in a previous analysis. At this stage, we are using only one dataset: Optic Glioma. There are 4 normal mice samples and 11 tumor mice samples.

3.1 Previously identified genes

The process for identifying regulator genes is as follows:

1. Use interactome network as provided by Peter Sims (binary and directional relationships between 13089 genes).
2. For the weights along the interactome network, use 1-abs(cor) for a dissimilarity metric (using expression data). Find the new weighted network for each of the tumor and normal datasets.
3. Extract a sub-network using genes which are highest according to the betweeness complexity measure. For each dataset, there will be 2 sub-networks (one normal and one tumor). [Be sure to intersect the genes in the two lists for an apples to apples comparison.]
4. Compare normal and tumor sub-networks in each dataset.

The genes which were identified as most promising are the following:

```
row.names(data_thresh)
```

```
##  [1] "CEBPZ"   "ETV5"    "SPEN"    "ZCCHC14" "CAMTA1"  "CHD5"    "CERS2"
##  [8] "HNRNPAB" "ILF2"    "ZCCHC17" "ZC3H15"  "TULP4"   "PURB"    "RPL7"
## [15] "TCF3"    "TEAD1"   "CNBP"    "PRDM2"   "SARNP"   "ZRANB2"  "GCSH"
## [22] "IFT74"   "MYL12B"
```

3.2 Finding target genes

As mentioned, the original interactome network from Peter Sims contains the relationships between genes. For each of the regulators, we find the list of upstream genes.

```
possibleRegulators = row.names(data_thresh)
```

```
# Read in network.
intNet=read.table("gbm-sun-interactome.txt",header=TRUE,sep="\t")

tarGenes = list()
for(i in 1:length(possibleRegulators)){
  tarGenes[[i]] = t(data.frame(lapply(intNet[intNet[,1]==possibleRegulators[i],2],as.character),
                               stringsAsFactors = FALSE))
 if(nrow(tarGenes[[i]]) > 0) {colnames(tarGenes[[i]]) = c("GENE")
 tarGenes[[i]] = data.frame(tarGenes[[i]])}
}
names(tarGenes) = as.vector(possibleRegulators)
```

```
# The number of target genes for each of the possible regulators:
lapply(tarGenes,nrow)
```

```
## $CEBPZ
## [1] 416
##
## $ETV5
## [1] 504
##
## $SPEN
## [1] 443
##
## $ZCCHC14
## [1] 402
##
## $CAMTA1
## [1] 1210
##
## $CHD5
## [1] 950
##
## $CERS2
## [1] 570
##
## $HNRNPAB
## [1] 608
##
## $ILF2
## [1] 613
##
## $ZCCHC17
## [1] 286
##
## $ZC3H15
## [1] 735
##
## $TULP4
## [1] 460
##
## $PURB
## [1] 783
##
## $RPL7
## [1] 280
##
## $TCF3
## [1] 732
##
## $TEAD1
## [1] 498
##
## $CNBP
## [1] 522
##
## $PRDM2
## [1] 730
##
## $SARNP
## [1] 317
##
## $ZRANB2
## [1] 559
##
## $GCSH
## [1] 0
##
## $IFT74
## [1] 0
##
## $MYL12B
## [1] 0
```

```
# the number of genes described in the interactome network:
length(unique(intNet[,1]))  # number of unique regulators

## [1] 1265
```

```r
length(unique(intNet[,2]))  # number of unique targets
```

```
## [1] 11824
```

```r
length(unique(union(intNet[,1], intNet[,2]))) # number of unique transcriptomes total
```

```
## [1] 13089
```

For good measure, it is interesting to see which of the regulators are also differentially expressed. (Only ETV5! The genes are sorted based on the adjusted p-value.)

```r
DEanalysis = miceps # the DE analysis from normalization above
DEps = DEanalysis[,c("gene", "pvalue", "padj", "log2FoldChange")]
DEps = DEps %>% mutate(GENE=gene) %>% dplyr::select(-gene)
#head(DEps)

regGenesP <- left_join(data.frame(GENE = possibleRegulators), DEps, by="GENE")
regGenesP %>% arrange(padj)
```

```
##       GENE   pvalue     padj log2FoldChange
## 1     ETV5 1.34e-09 3.87e-07         1.4474
## 2   MYL12B 6.81e-04 1.61e-02        -0.5850
## 3   ZC3H15 4.03e-03 5.12e-02        -0.5603
## 4   CAMTA1 4.22e-03 5.24e-02        -0.4705
## 5     SPEN 4.61e-03 5.55e-02         0.5714
## 6    TULP4 6.11e-03 6.48e-02         0.4845
## 7    SARNP 7.36e-03 7.24e-02        -0.7022
## 8  ZCCHC17 9.34e-03 8.26e-02        -0.6699
## 9    IFT74 9.66e-03 8.43e-02        -0.5475
## 10    TCF3 1.29e-02 9.89e-02         0.4928
## 11 ZCCHC14 3.82e-02 1.80e-01         0.3660
## 12    RPL7 3.83e-02 1.80e-01        -0.5100
## 13    CNBP 4.16e-02 1.88e-01        -0.3724
## 14  ZRANB2 6.71e-02 2.47e-01        -0.3515
## 15   TEAD1 1.40e-01 3.70e-01         0.2794
## 16    ILF2 1.51e-01 3.83e-01        -0.3077
## 17   CEBPZ 1.60e-01 3.96e-01        -0.3255
## 18    GCSH 3.92e-01 6.38e-01        -0.1581
## 19   CERS2 5.36e-01 7.49e-01        -0.0923
## 20 HNRNPAB 6.23e-01 8.06e-01         0.0784
## 21    PURB 6.80e-01 8.43e-01         0.0573
## 22   PRDM2 8.75e-01 9.45e-01         0.0207
## 23    CHD5 8.98e-01 9.57e-01        -0.0334
```

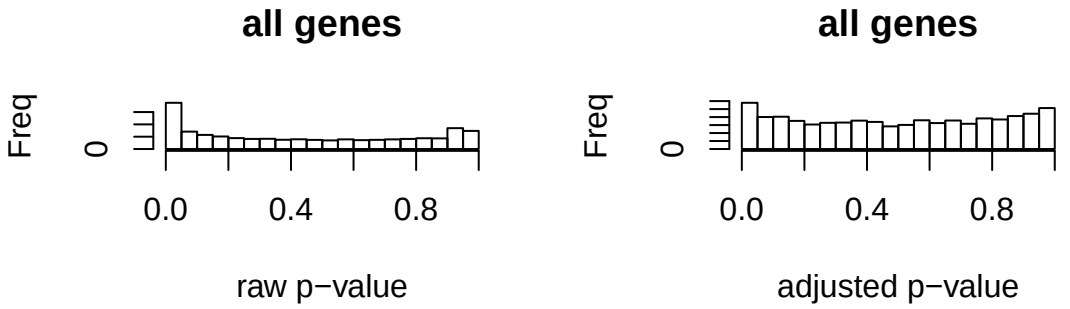
DE for target genes

A DESeq anlaysis was done on the optic glioma dataset using DESeq2.

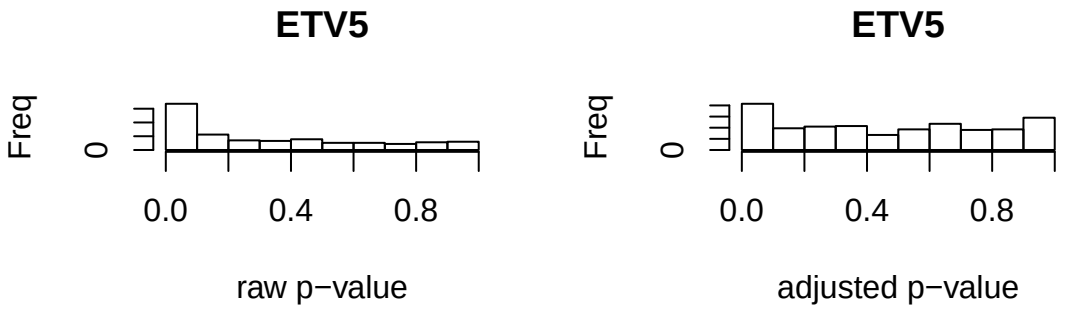
```
tarGenesP = list()
for(i in 1:length(possibleRegulators)){
  temp = tarGenes[[i]]
  if(nrow(temp) > 0) tarGenesP[[i]] <- left_join(temp, DEps, by= "GENE")
  if(nrow(temp) == 0) tarGenesP[[i]] <- data.frame(GENE=NA, pvalue=NA, padj=NA, log2FoldChange=NA)
}
names(tarGenesP) = as.vector(possibleRegulators)

siglevel = 0.01
numsig = sapply(tarGenesP, function(x) sum(x$padj < siglevel, na.rm=T))
totalps = sapply(tarGenesP, function(x) sum(!is.na(x$padj)))
percsig = sapply(tarGenesP, function(x) sum(x$padj < siglevel, na.rm=T)) / sapply(tarGenesP, function(x) sum(!is.na

all = data.frame(numsig=sum(DEps$padj < siglevel, na.rm=T),
                 totalps = sum(!is.na(DEps$padj)),
                 percsig = sum(DEps$padj < siglevel, na.rm=T)/sum(!is.na(DEps$padj)))
TargetPs = cbind(GENE=c(names(numsig),"ALL"),rbind(data.frame(numsig,totalps,percsig), all=all))
```

3.2.1 Visualizing p-values

```
numReg = length(possibleRegulators)
par(mfrow=c(1,2))
hist(DEps$pvalue, main="all genes", xlab="raw p-value", ylab="Freq")
hist(DEps$padj, main="all genes", xlab="adjusted p-value", ylab="Freq")
```




```
# only printing for ETV5, uncomment to see all sig genes for all regulators
# for(i in 1:numReg){
#  if(sum(!is.na(tarGenesP[[i]]$pvalue)) > 0){
#  hist(tarGenesP[[i]]$pvalue, main=possibleRegulators[i], xlab="raw p-value", ylab="Freq")
#  hist(tarGenesP[[i]]$padj, main=possibleRegulators[i], xlab="adjusted p-value", ylab="Freq")}
#}

hist(tarGenesP["ETV5"]$ETV5$pvalue, main="ETV5", xlab="raw p-value", ylab="Freq")
hist(tarGenesP["ETV5"]$ETV5$padj, main="ETV5", xlab="adjusted p-value", ylab="Freq")
```




```
#prints summary of pvalues for all regulators of interest:
#lapply(tarGenesP,function(x) summary(x[,-1]))

#significance level is 0.01 (for adjusted p-value)
#ordered by the percent of significant targets, ETV5 is highest
TargetPs  %>% arrange(desc(percsig))
```

```
##       GENE numsig totalps percsig
## 1     ETV5     31     449  0.0690
## 2    CERS2     32     496  0.0645
## 3    SARNP     15     255  0.0588
## 4  ZCCHC14     16     361  0.0443
## 5     TCF3     28     640  0.0437
## 6    TEAD1     19     443  0.0429
## 7   ZC3H15     27     652  0.0414
## 8    TULP4     17     412  0.0413
## 9     RPL7     10     247  0.0405
## 10    PURB     27     678  0.0398
## 11     ALL    529   14926  0.0354
## 12 HNRNPAB     19     543  0.0350
## 13    CNBP     15     458  0.0328
## 14  CAMTA1     35    1093  0.0320
## 15    CHD5     27     852  0.0317
## 16 ZCCHC17      8     257  0.0311
## 17   PRDM2     20     652  0.0307
## 18    SPEN     12     392  0.0306
## 19   CEBPZ     10     379  0.0264
## 20    ILF2     13     555  0.0234
## 21  ZRANB2     11     486  0.0226
## 22    GCSH      0       0     NaN
## 23   IFT74      0       0     NaN
## 24  MYL12B      0       0     NaN
```

3.2.2 Lists of Significant Genes

```
# only printing for ETV5, uncomment to see all sig genes for all regulators
# lapply(tarGenesP, function(x) x[c(x£padj < siglevel & !is.na(x£padj)),])
tarGenesP$ETV5[c(tarGenesP$ETV5$padj < siglevel & !is.na(tarGenesP$ETV5$padj)),]
```

```
##         GENE   pvalue     padj log2FoldChange
## 16     SPRY2 1.13e-08 2.41e-06          0.916
```

```
## 38    DNAJB4 2.63e-05 1.65e-03         -0.457
## 64    COL2A1 1.07e-04 4.43e-03          1.235
## 91    SPRED1 1.04e-05 7.78e-04          0.800
## 102    DUSP6 1.50e-04 5.54e-03          0.637
## 104    S1PR1 6.84e-17 9.28e-14          1.358
## 110      AK4 5.38e-05 2.77e-03          0.959
## 115    FABP5 1.39e-09 3.93e-07          1.087
## 116    FABP7 6.22e-05 3.02e-03          0.956
## 127   RSBN1L 2.16e-04 7.15e-03         -0.428
## 132    BTBD3 7.12e-09 1.62e-06          0.755
## 172    GAP43 1.94e-05 1.27e-03         -0.797
## 183     GJA1 3.64e-10 1.26e-07          0.600
## 188     GLDC 6.60e-06 5.29e-04          1.161
## 201   KCNIP1 3.06e-04 9.09e-03         -0.797
## 212   IGFBP6 3.07e-04 9.10e-03         -0.942
## 232     LRP4 5.38e-05 2.77e-03          0.671
## 239    MMP15 2.11e-04 7.07e-03          1.003
## 255     NT5E 1.18e-04 4.76e-03         -0.744
## 260  PCDHGC3 2.12e-06 2.09e-04          0.932
## 278    TPPP3 6.01e-05 2.94e-03         -0.912
## 291     SHC3 2.71e-04 8.35e-03          0.903
## 295    NLGN3 1.82e-05 1.21e-03          0.705
## 300   SPATA6 1.62e-04 5.86e-03         -0.552
## 302   ELOVL2 1.62e-11 9.28e-09          1.908
## 437    SPRY4 5.77e-05 2.87e-03          1.104
## 466    SOCS2 1.77e-04 6.28e-03         -0.814
## 483 SLC9A3R1 1.41e-06 1.51e-04          0.722
## 486    CHST2 2.28e-11 1.26e-08          1.163
## 490   CXCL14 4.88e-17 7.29e-14          1.425
## 496    DOCK4 2.88e-05 1.73e-03          0.642
```

4 Public Data - Human

4.1 Data

The following analysis is further investigation into ETV5 using two human datasets.

4.1.1 GSE42656 Study

https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE42656
http://cancerres.aacrjournals.org/content/73/18/5834.long#sec-2
Henriquez NV et al., "Comparative expression analysis reveals lineage relationships between human and murine gliomas and a dominance of glial signatures during tumor propagation in vitro.", **Cancer Res**, 2013 Jul 25;73(18):5834-44

From NCBI data repository
"We analysed gene expression in paediatric brain tumours as compared to normal adult brain in order to understand the molecular profiles. Our cohort included 14 pilocytic astrocytomas, 3 diffuse astrocytomas, 2 anaplastic astrocytomas, 5 glioblastomas, 14 ependymomas, 9 medulloblastomas, 5 atypical teratoid/rhabdoid tumours, 4 choroid plexus papillomas, 1 papillary glioneuronal, 8 adult brain and 8 foetal brain controls."

Article Abstract:
Brain tumors are thought to originate from stem/progenitor cell populations that acquire specific genetic mutations. Although current preclinical models have relevance to human pathogenesis, most do not recapitulate the histogenesis of the human disease. Recently, a large series of human gliomas and medulloblastomas were analyzed for genetic signatures of prognosis and therapeutic response. Using a mouse model system that generates three distinct types of intrinsic brain tumors, we correlated RNA and protein expression levels with human brain tumors. A combination of genetic mutations and cellular environment during tumor propagation defined the incidence and phenotype of intrinsic murine tumors. Importantly, in vitro passage of cancer stem cells uniformly promoted a glial expression profile in culture and in brain tumors. Gene expression profiling revealed that experimental gliomas corresponded to distinct subclasses of human glioblastoma, whereas experimental supratentorial primitive neuroectodermal tumors (sPNET) correspond to atypical teratoid/rhabdoid tumor (AT/RT), a rare childhood tumor.

Methods: Human samples using Ilumina arrays (Illumina HT12_v3).

4.1.2 GSE12907 Study

Need to fill in the references / experimental design / methods for the GSE12907 dataset.

This data set has 21 juvenile pilocytic astrocytoma samples (columns 2-22), three samples from normal cerebellum (from humans ages 8, 16 and 63, columns 23-25), and one normal foetal sample (column 26). The first column contains the Probe IDs.

4.1.3 Data Wrangling of both datasets

```
#targets of ETV5 only
tarGenesETV5 <- tarGenesP$ETV5

# inputting the two human datasets
GSE42656 <- read.delim("GSE42656_series_dataonly.txt", sep="\t")
GSE12907 <- read.delim("GSE12907_series_dataonly.txt", sep="\t")
ETV5name <- data.frame(GENE="ETV5")
```

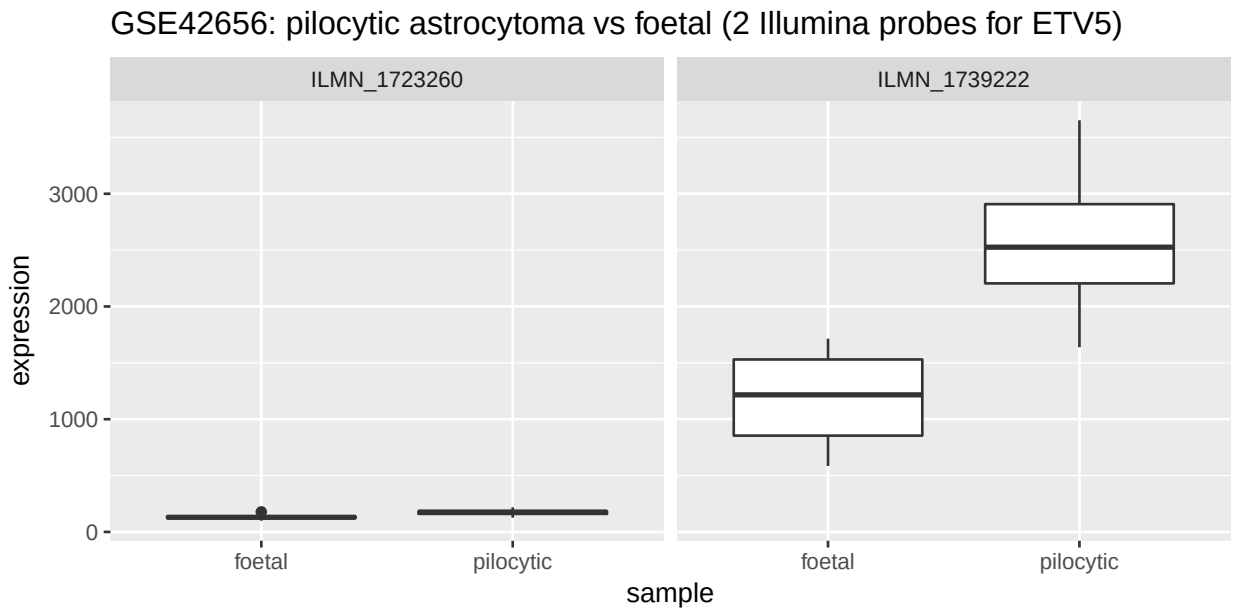
```
PROBES.GSE42656 <- as.character(GSE42656$ID_REF)  # Illumina
PROBES.GSE12907 <- as.character(GSE12907$ID_REF)  # Affymetrix Human Genome U133A

IlluminaIDs <- AnnotationDbi::select(illuminaHumanv4.db, PROBES.GSE42656, c("SYMBOL","PROBEID", "ENTREZID", "GENENA
AffyIDs <- AnnotationDbi::select(hgu133plus2.db, PROBES.GSE12907, c("SYMBOL","PROBEID", "ENTREZID", "GENENAME"))


# cleaning up GSE42656
pilocytic <- c("GSM1047395", "GSM1047396", "GSM1047397", "GSM1047399", "GSM1047400",
               "GSM1047401", "GSM1047402", "GSM1047403", "GSM1047404", "GSM1047405",
               "GSM1047406", "GSM1047407", "GSM1047408", "GSM1047409")

foetal <- c("GSM1047452",       "GSM1047453",   "GSM1047454",   "GSM1047455",   "GSM1047456",
            "GSM1047457",       "GSM1047458",   "GSM1047459")

# ETV5_GSE42656 - FOETAL
ETV5_GSE42656 <- GSE42656 %>%
  dplyr::select(one_of(c("ID_REF", pilocytic, foetal))) %>%
  left_join(IlluminaIDs, by = c("ID_REF" = "PROBEID")) %>%
  right_join(ETV5name, by=c("SYMBOL"= "GENE")) %>%
  dplyr::select(-ENTREZID, -GENENAME)

ETV5_GSE42656_tidy <- ETV5_GSE42656 %>%
  tidyr::gather(sampleID, expression, -c(ID_REF,SYMBOL)) %>%
  mutate(expression = parse_number(expression)) %>%
  mutate(sample = ifelse(sampleID %in% pilocytic, "pilocytic", "foetal"))


AffyIDs <- AnnotationDbi::select(hgu133plus2.db, PROBES.GSE12907,  c("SYMBOL", "ENTREZID", "GENENAME"))

CBM = colnames(GSE12907)[23:26]
pilocytic12907 = colnames(GSE12907)[2:22]

# ETV5_GSE12907
ETV5_GSE12907 <- GSE12907 %>%
  left_join(AffyIDs, by = c("ID_REF" = "PROBEID")) %>%
  right_join(ETV5name, by=c("SYMBOL"= "GENE")) %>%
  dplyr::select(-ENTREZID, -GENENAME)

ETV5_GSE12907_tidy <- ETV5_GSE12907 %>%
  tidyr::gather(sampleID, expression, -c(ID_REF,SYMBOL)) %>%
  mutate(expression = parse_number(expression)) %>%
  mutate(sample = ifelse(sampleID %in% CBM, "CBM", "PA"))
```

4.2 t-tests on ETV5 for both datasets

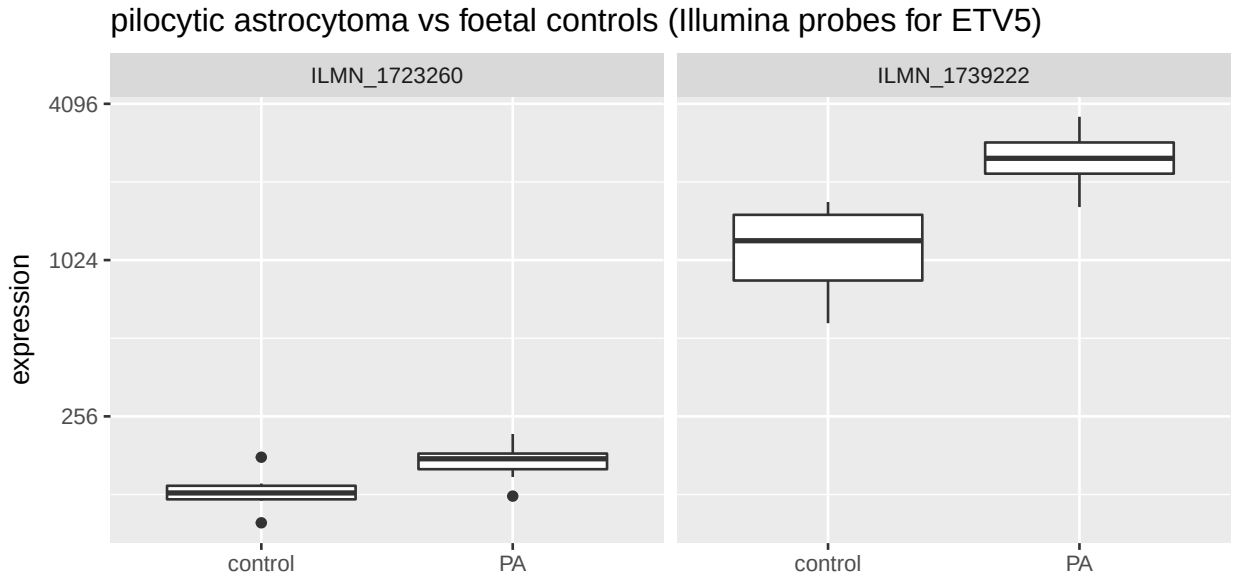
```
#ETV5_GSE42656
for(i in 1:nrow(ETV5_GSE42656)){
print(t.test(ETV5_GSE42656[i,2:15], ETV5_GSE42656[i,16:23]))
}

##
##  Welch Two Sample t-test
```

```
##
## data:  ETV5_GSE42656[i, 2:15] and ETV5_GSE42656[i, 16:23]
## t = 4, df = 20, p-value = 8e-04
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  20.8 64.5
## sample estimates:
## mean of x mean of y
##       175       133
##
##
##  Welch Two Sample t-test
##
## data:  ETV5_GSE42656[i, 2:15] and ETV5_GSE42656[i, 16:23]
## t = 6, df = 20, p-value = 4e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   900 1762
## sample estimates:
## mean of x mean of y
##      2521      1190
```

```
ETV5_GSE42656_tidy <- ETV5_GSE42656_tidy %>%
  mutate(sample2 = ifelse(sample=="foetal", "control", "PA"))

ggplot(ETV5_GSE42656_tidy, aes(y=expression, x=sample)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("GSE42656: pilocytic astrocytoma vs foetal (2 Illumina probes for ETV5)")
```

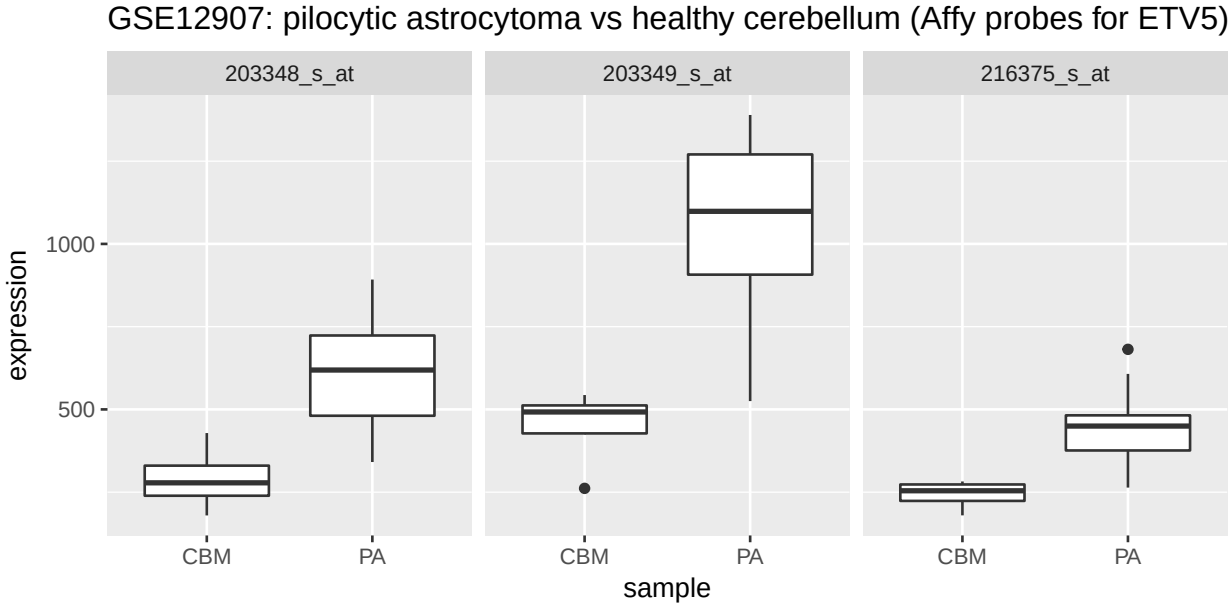



```
ggplot(ETV5_GSE42656_tidy, aes(y=expression, x=sample2)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("pilocytic astrocytoma vs foetal controls (Illumina probes for ETV5)") +
  scale_y_continuous(trans=log2_trans()) + xlab("")
```

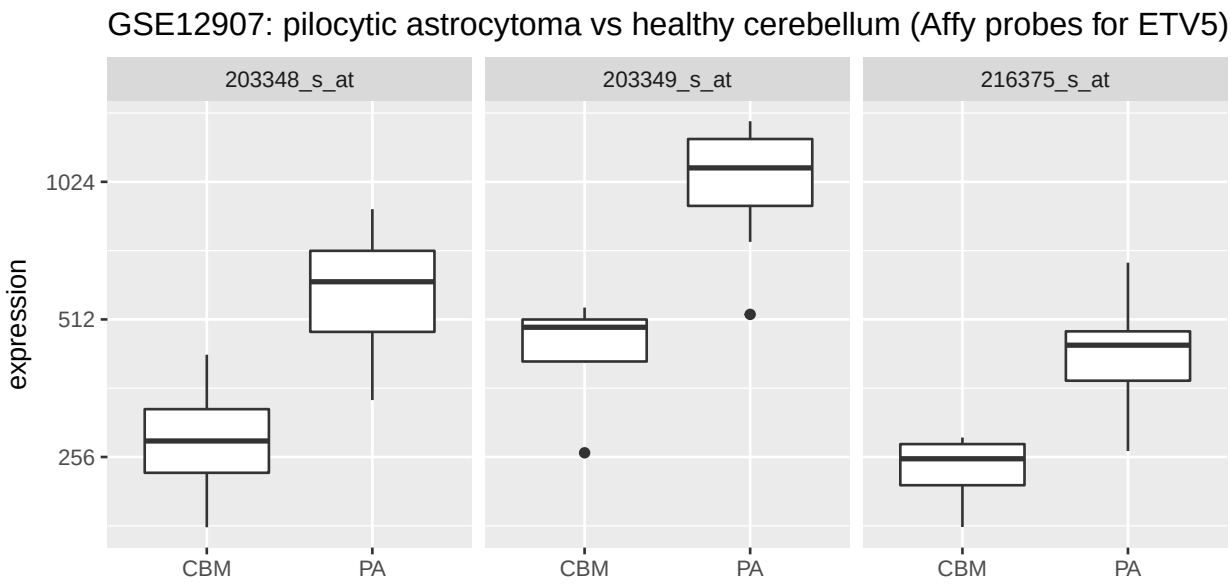



```
# ETV5_GSE12907
ETV5_GSE12907_t <- GSE12907 %>%
  left_join(AffyIDs, by = c("ID_REF" = "PROBEID")) %>%
  dplyr::select(-ENTREZID, -GENENAME)

all_p_12907 <- as.numeric()
for(i in 1:nrow(ETV5_GSE12907_t)){
   all_p_12907[i] <- t.test(ETV5_GSE12907_t[i,2:22], ETV5_GSE12907_t[i,23:26])$p.value
 }
mean(all_p_12907 <= 0.05)

## [1] 0.241

length(all_p_12907)

## [1] 24433

for(i in 1:nrow(ETV5_GSE12907)){
print(t.test(ETV5_GSE12907[i,2:15], ETV5_GSE12907[i,23:26]))
}

##
##  Welch Two Sample t-test
##
## data:  ETV5_GSE12907[i, 2:15] and ETV5_GSE12907[i, 23:26]
## t = 6, df = 6, p-value = 0.001
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  197 502
## sample estimates:
## mean of x mean of y
##       641       291
##
##
```

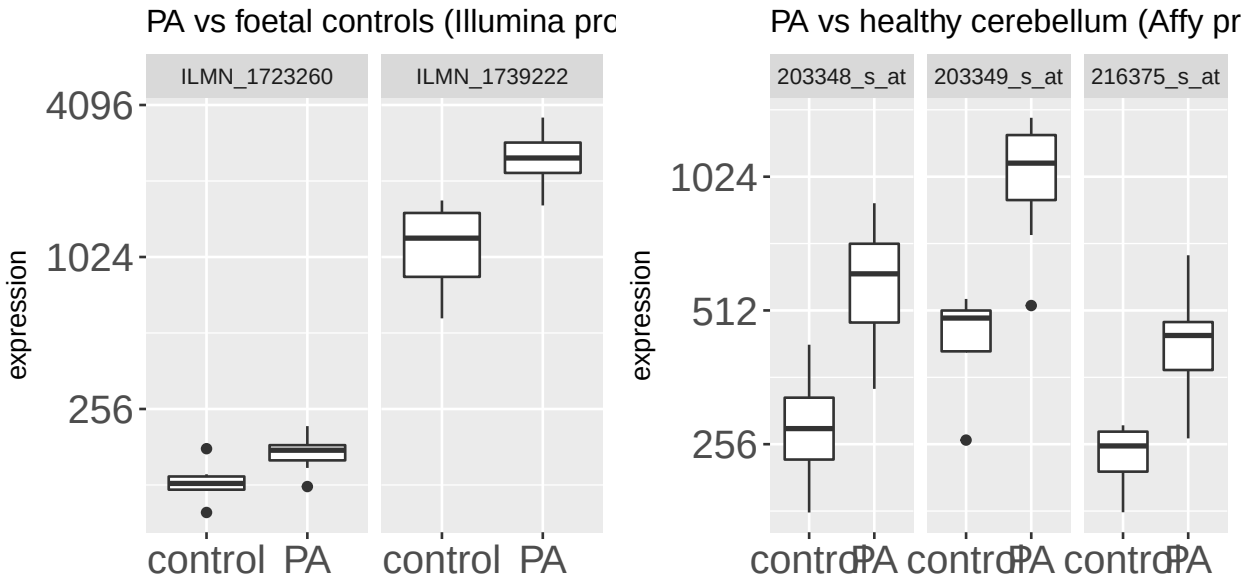
```
##  Welch Two Sample t-test
##
## data:  ETV5_GSE12907[i, 2:15] and ETV5_GSE12907[i, 23:26]
## t = 8, df = 7, p-value = 5e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  490 870
## sample estimates:
## mean of x mean of y
##      1127       447
##
##
##  Welch Two Sample t-test
##
## data:  ETV5_GSE12907[i, 2:15] and ETV5_GSE12907[i, 23:26]
## t = 7, df = 10, p-value = 3e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  157 310
## sample estimates:
## mean of x mean of y
##       476       243

ETV5_GSE12907_tidy <- ETV5_GSE12907_tidy %>%
  mutate(sample2 = ifelse(sample=="CBM", "control", "PA"))

ggplot(ETV5_GSE12907_tidy, aes(y=expression, x=sample)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("GSE12907: pilocytic astrocytoma vs healthy cerebellum (Affy probes for ETV5)")
```




```
ggplot(ETV5_GSE12907_tidy, aes(y=expression, x=sample)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("GSE12907: pilocytic astrocytoma vs healthy cerebellum (Affy probes for ETV5)") +
  scale_y_continuous(trans=log2_trans()) + xlab("")
```




```
p1 <- ggplot(ETV5_GSE42656_tidy, aes(y=expression, x=sample2)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("PA vs foetal controls (Illumina probes for ETV5)") +
  scale_y_continuous(trans=log2_trans()) + xlab("") + theme(axis.text=element_text(size=16))
p2 <- ggplot(ETV5_GSE12907_tidy, aes(y=expression, x=sample2)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle("PA vs healthy cerebellum (Affy probes for ETV5)") +
  scale_y_continuous(trans=log2_trans()) + xlab("") + theme(axis.text=element_text(size=16))
multiplot(p1,p2,cols=2)
```



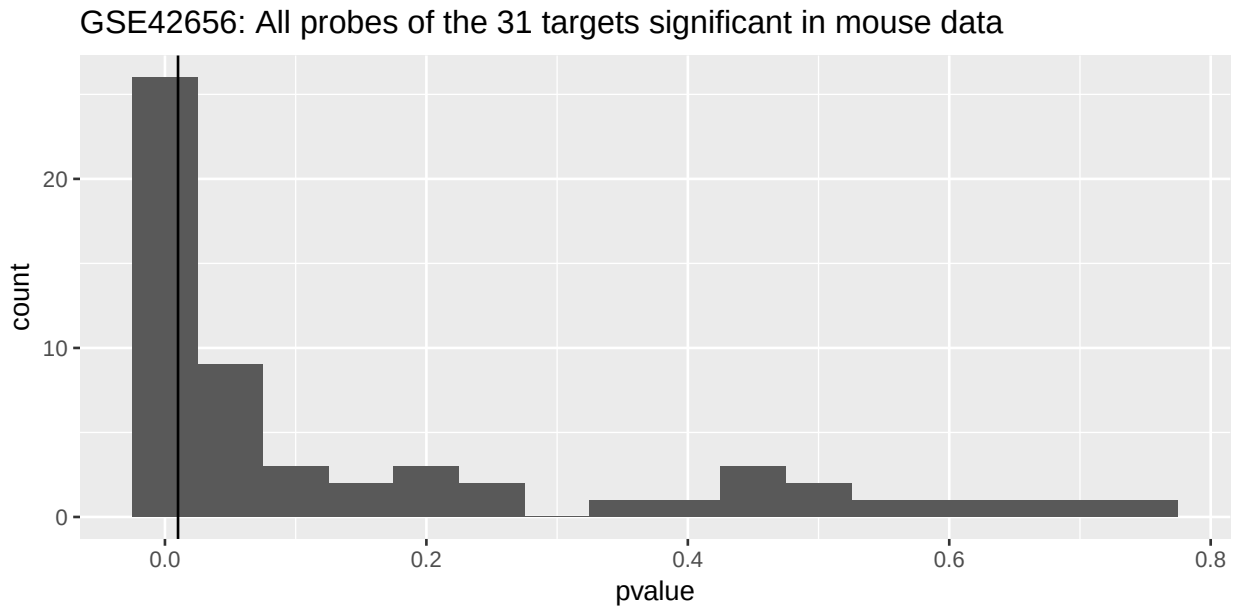
4.3 t-tests on ETV5 *targets* for both datasets

4.3.1 GSE42656

Using the 31 targets which were significant for the mouse data, we notice that the majority of the probes for those targets are significant using the human data as well. Notice that 56.1% of the 31 EVT5 target probes were significantly differentially expressed (not adjusted for multiple comparisons); 20/31 = 64.5% of the ETV5 target genes were significantly differentially expressed.

Listed are the significant gene/probes (20 of 31 of the mouse-signifcant targets were also signficant in the GSE42656 dataset). For example, SPRY4 is significant for two of the four probes.

```
temp <- tarGenesETV5[c(tarGenesETV5$padj < siglevel & !is.na(tarGenesETV5$padj)),]
ETV5_sig <- data.frame(GENE=as.character(temp$GENE))
ETV5_targets <- data.frame(GENE=as.character(tarGenes$ETV5$GENE))


# targets_GSE42656 - FOETAL
targets_GSE42656 <- GSE42656 %>%
  dplyr::select(one_of(c("ID_REF", pilocytic, foetal))) %>%
  left_join(IlluminaIDs, by = c("ID_REF" = "PROBEID")) %>%
  dplyr::mutate(SYMBOL = toupper(SYMBOL)) %>%
  dplyr::filter(SYMBOL %in% ETV5_sig$GENE) %>%
  dplyr::select(-ENTREZID, -GENENAME) %>%
  dplyr::filter(!is.na(ID_REF))


p_targ_GSE42656 <-data.frame(GENE=character(), PROBE=character(),
                             pvalue=double(), statistic=double())

for(i in 1:nrow(targets_GSE42656)){
 temp <- t.test(as.numeric(targets_GSE42656[i,2:15]),
                as.numeric(targets_GSE42656[i,16:23]))
 temp2 <- data.frame(GENE = as.character(targets_GSE42656[i,24]),
                     PROBE = as.character(targets_GSE42656[i,1]),
                     pvalue = temp$p.value,
                     statistic = temp$statistic)
 p_targ_GSE42656 <- p_targ_GSE42656 %>% bind_rows(temp2)
}


p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
#  filter(pvalue <= 0.05) %>%
#  summarize(n_distinct(GENE))
  summarize(proportion.05 = mean(pvalue <= 0.05))

##   proportion.05
## 1         0.561

p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  ggplot(aes(x=pvalue)) + geom_histogram(binwidth = .05) +
  geom_vline(xintercept = siglevel) +
  ggtitle("GSE42656: All probes of the 31 targets significant in mouse data")
```




```
#siglevel/nrow(targets_GSE42656)  # no p-value adustment
p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  #summarize(unique_genes = n_distinct(GENE))
  arrange(pvalue)
```

```
##       GENE        PROBE   pvalue statistic
## 1   SPRED1 ILMN_1804277 2.88e-11     15.03
## 2    SPRY4 ILMN_2086105 3.77e-09     11.41
## 3   KCNIP1 ILMN_1744387 1.00e-07      8.08
## 4    SPRY4 ILMN_1797596 5.42e-07      7.43
## 5     LRP4 ILMN_1675268 1.13e-06      8.02
## 6     NT5E ILMN_1697220 2.29e-06      7.30
## 7    DUSP6 ILMN_2396020 3.28e-06      6.63
## 8    NLGN3 ILMN_1759700 4.67e-06      6.85
## 9   ELOVL2 ILMN_1716843 7.82e-06     -8.11
## 10   TPPP3 ILMN_1797744 8.78e-06      6.28
## 11   DUSP6 ILMN_1677466 2.24e-05      5.74
## 12   SPRY2 ILMN_2089329 4.08e-05      5.24
## 13 PCDHGC3 ILMN_2274355 6.43e-05      5.34
## 14  SPATA6 ILMN_1775926 2.01e-04      5.01
## 15   SOCS2 ILMN_2131861 2.39e-04     -4.54
## 16 PCDHGC3 ILMN_1656955 2.94e-04      4.38
## 17   SOCS2 ILMN_1798926 3.89e-04     -4.38
## 18 PCDHGC3 ILMN_2251963 7.52e-04      3.97
## 19  KCNIP1 ILMN_2368856 8.26e-04      3.94
## 20   FABP7 ILMN_1745299 9.22e-04     -3.90
## 21 PCDHGC3 ILMN_2251961 1.13e-03      3.84
## 22 PCDHGC3 ILMN_2345824 1.53e-03      3.67
## 23 PCDHGC3 ILMN_1675428 4.07e-03      3.44
## 24  RSBN1L ILMN_1712027 6.20e-03     -3.19
## 25     AK4 ILMN_1798249 7.84e-03     -3.52
## 26    GLDC ILMN_1806754 8.09e-03      2.95
```

```
## 27   BTBD3 ILMN_1713964 2.94e-02     -2.72
## 28    SHC3 ILMN_1770905 3.45e-02      2.39
## 29   S1PR1 ILMN_1653504 3.45e-02     -2.46
## 30     AK4 ILMN_1843198 3.46e-02     -2.50
## 31     AK4 ILMN_1764090 3.71e-02     -2.34
## 32     AK4 ILMN_2338038 4.21e-02     -2.41
```

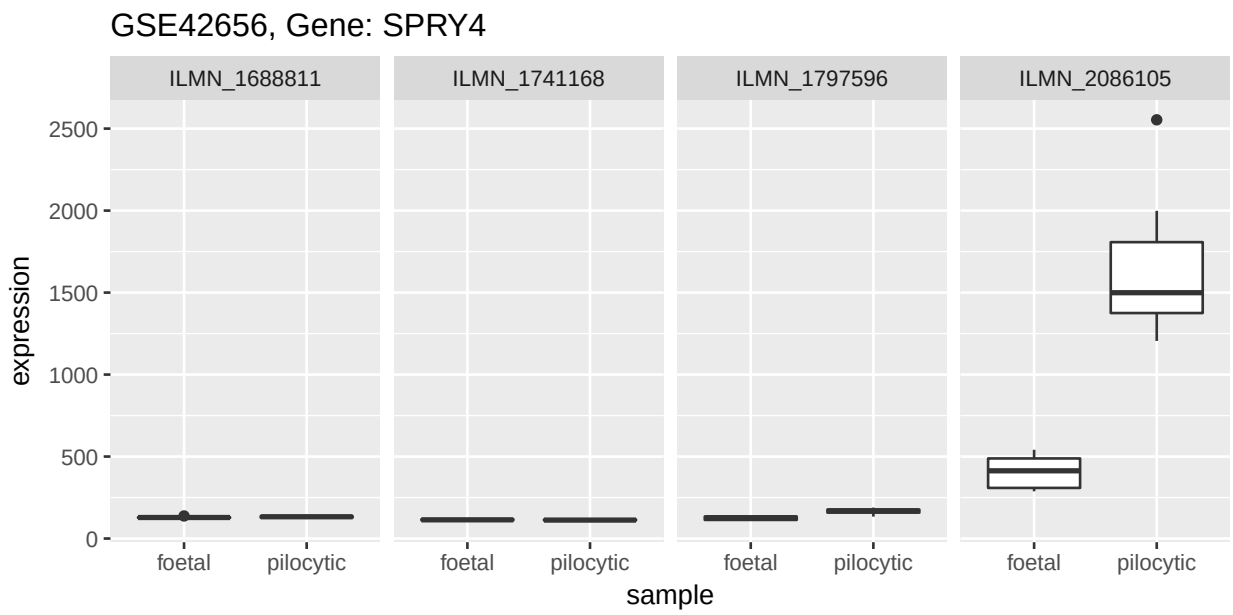
```
p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  summarize(unique_genes = n_distinct(GENE))
```

```
##   unique_genes
## 1           20
```

```
targets_GSE42656_tidy <- targets_GSE42656 %>%
  tidyr::gather(sampleID, expression, -c(ID_REF,SYMBOL)) %>%
  mutate(expression = parse_number(expression)) %>%
  mutate(sample = ifelse(sampleID %in% pilocytic, "pilocytic", "foetal"))


genename <- data.frame(GENE="SPRY4")
onegene_GSE42656 <- GSE42656 %>%
  dplyr::select(one_of(c("ID_REF", pilocytic, foetal))) %>%
  left_join(IlluminaIDs, by = c("ID_REF" = "PROBEID")) %>%
  mutate(SYMBOL = toupper(SYMBOL)) %>%
  right_join(genename, by=c("SYMBOL"= "GENE")) %>%
  dplyr::select(-ENTREZID, -GENENAME)

for(i in 1:nrow(onegene_GSE42656)){
  print(t.test(onegene_GSE42656[i,2:15], onegene_GSE42656[i,16:23]))
}
```

```
##
##  Welch Two Sample t-test
##
## data:  onegene_GSE42656[i, 2:15] and onegene_GSE42656[i, 16:23]
## t = 1, df = 20, p-value = 0.2
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.83  9.49
## sample estimates:
## mean of x mean of y
##       132       128
##
##
##  Welch Two Sample t-test
##
## data:  onegene_GSE42656[i, 2:15] and onegene_GSE42656[i, 16:23]
## t = -0.5, df = 20, p-value = 0.6
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -6.66  3.90
## sample estimates:
## mean of x mean of y
##       112       114
```

```
##
##
##  Welch Two Sample t-test
##
## data:  onegene_GSE42656[i, 2:15] and onegene_GSE42656[i, 16:23]
## t = 7, df = 20, p-value = 5e-07
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  31.0 55.3
## sample estimates:
## mean of x mean of y
##       167       124
##
##
##  Welch Two Sample t-test
##
## data:  onegene_GSE42656[i, 2:15] and onegene_GSE42656[i, 16:23]
## t = 10, df = 20, p-value = 4e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   987 1437
## sample estimates:
## mean of x mean of y
##      1619       407

targets_GSE42656_tidy %>%
  filter(SYMBOL %in% genename) %>%
  ggplot(aes(y=expression, x=sample)) +
  geom_boxplot() + facet_grid(.~ID_REF) +
  ggtitle(paste("GSE42656, Gene:",genename))
```



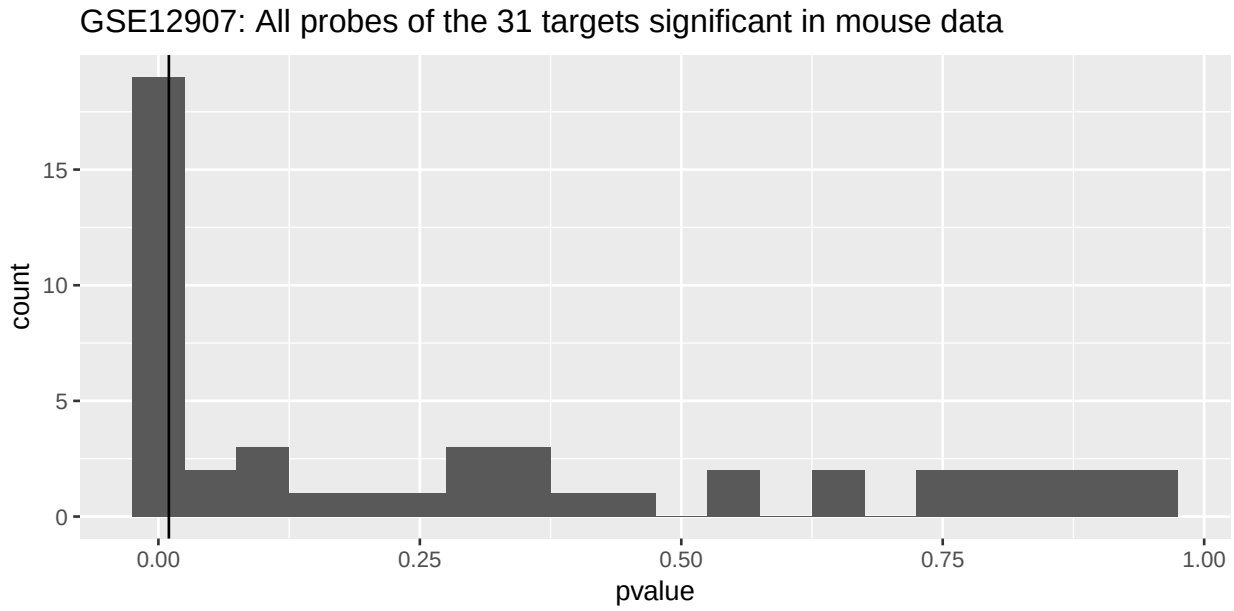
4.3.2 GSE12907

Using the 31 targets which were significant for the mouse data, we notice that the majority of the probes for those targets are significant using the human data as well. Notice that 40.8% of the 31 EVT5 target probes (49 probes) were significantly differentially expressed (not adjusted for multiple comparisons); 12/31 = 38.7% of the ETV5 target genes were significantly differentially expressed.

Listed are the significant gene/probes (12 of 31 of the mouse-signifcant targets were also signficant in the human dataset).

```
# targets_GSE12907
targets_GSE12907 <- GSE12907 %>%
  dplyr::select(one_of(c("ID_REF", pilocytic12907, CBM))) %>%
  left_join(AffyIDs, by = c("ID_REF" = "PROBEID")) %>%
  mutate(SYMBOL = toupper(SYMBOL)) %>%
  filter(SYMBOL %in% ETV5_targets$GENE) %>%
  dplyr::select(-ENTREZID, -GENENAME) %>%
  filter(!is.na(ID_REF))

p_targ_GSE12907 <-data.frame(GENE=character(), PROBE=character(),
                             pvalue=double(), statistic=double())

for(i in 1:nrow(targets_GSE12907)){
 temp <- t.test(targets_GSE12907[i,2:22], targets_GSE12907[i,23:26])
 temp2 <- data.frame(GENE = as.character(targets_GSE12907[i,27]),
                     PROBE = as.character(targets_GSE12907[i,1]),
                     pvalue = temp$p.value,
                     statistic = temp$statistic)
 p_targ_GSE12907 <- p_targ_GSE12907 %>% bind_rows(temp2)
}


p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  summarize(proportion.05 = mean(pvalue <= 0.05))

##   proportion.05
## 1         0.408

#siglevel/nrow(targets_GSE12907)  # no p-value adustment
p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  arrange(pvalue)

##        GENE       PROBE   pvalue statistic
## 1     DUSP6   208891_at 1.01e-10     11.60
## 2     NLGN3   219726_at 1.58e-10     10.98
## 3     DUSP6 208893_s_at 5.93e-10     10.22
## 4    SPATA6 220298_s_at 8.02e-09     10.19
## 5     DUSP6 208892_s_at 2.19e-08      9.01
## 6    SPATA6   220299_at 5.27e-08      9.40
## 7      NT5E   203939_at 1.09e-07      9.70
## 8     SPRY4 221489_s_at 3.06e-07      8.66
## 9      LRP4 212850_s_at 1.21e-05      7.83
## 10   KCNIP1   221307_at 1.76e-05      5.60
## 11  PCDHGC3 211066_x_at 1.07e-04     10.44
```

```
## 12  PCDHGC3 215836_s_at 1.34e-04     10.46
## 13  PCDHGC3 214564_s_at 1.83e-04      4.45
## 14  PCDHGC3 205717_x_at 2.16e-04      9.83
## 15  PCDHGC3 209079_x_at 2.30e-04     10.06
## 16  PCDHGC3 211876_x_at 3.08e-03      3.47
## 17     SHC3 206330_s_at 7.89e-03      4.29
## 18    FABP5 202345_s_at 1.21e-02      3.29
## 19 SLC9A3R1   201349_at 2.13e-02     -3.64
## 20   ELOVL2   213712_at 4.49e-02     -2.93

p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  summarize(unique_genes = n_distinct(GENE))

##   unique_genes
## 1           12

p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  ggplot(aes(x=pvalue)) + geom_histogram(binwidth = .05) +
  geom_vline(xintercept = siglevel) +
  ggtitle("GSE12907: All probes of the 31 targets significant in mouse data")
```



4.3.3 Comparing directionality of DE changes

```
tarGenes31 <- lapply(tarGenesP,
                     function(x) x[c(x$padj < siglevel & !is.na(x$padj)),])$ETV %>%
  arrange(padj)

p_sig31_GSE12907 <- p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <=0.05)
```

```
p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  full_join(p_sig31_GSE12907, by="GENE") %>%
  full_join(tarGenes31, by="GENE") %>%
  mutate(changemouse = ifelse(log2FoldChange > 0, "up", "down")) %>%
  mutate(change42656 = ifelse(statistic.x > 0, "up", "down")) %>%
  mutate(change12907 = ifelse(statistic.y > 0, "up", "down")) %>%
  mutate(pmouse = padj, p42656 = pvalue.x, p12907 = pvalue.y) %>%
  dplyr::select(GENE, changemouse, pmouse,change12907, p12907, change42656, p42656) %>%
  arrange(GENE)
```

```
##        GENE changemouse   pmouse change12907   p12907 change42656   p42656
## 1       AK4          up 2.77e-03        <NA>       NA        down 3.71e-02
## 2       AK4          up 2.77e-03        <NA>       NA        down 7.84e-03
## 3       AK4          up 2.77e-03        <NA>       NA        down 3.46e-02
## 4       AK4          up 2.77e-03        <NA>       NA        down 4.21e-02
## 5     BTBD3          up 1.62e-06        <NA>       NA        down 2.94e-02
## 6     CHST2          up 1.26e-08        <NA>       NA        <NA>       NA
## 7    COL2A1          up 4.43e-03        <NA>       NA        <NA>       NA
## 8    CXCL14          up 7.29e-14        <NA>       NA        <NA>       NA
## 9    DNAJB4        down 1.65e-03        <NA>       NA        <NA>       NA
## 10    DOCK4          up 1.73e-03        <NA>       NA        <NA>       NA
## 11    DUSP6          up 5.54e-03          up 1.01e-10          up 2.24e-05
## 12    DUSP6          up 5.54e-03          up 2.19e-08          up 2.24e-05
## 13    DUSP6          up 5.54e-03          up 5.93e-10          up 2.24e-05
## 14    DUSP6          up 5.54e-03          up 1.01e-10          up 3.28e-06
## 15    DUSP6          up 5.54e-03          up 2.19e-08          up 3.28e-06
## 16    DUSP6          up 5.54e-03          up 5.93e-10          up 3.28e-06
## 17   ELOVL2          up 9.28e-09        down 4.49e-02        down 7.82e-06
## 18    FABP5          up 3.93e-07          up 1.21e-02        <NA>       NA
## 19    FABP7          up 3.02e-03        <NA>       NA        down 9.22e-04
## 20    GAP43        down 1.27e-03        <NA>       NA        <NA>       NA
## 21     GJA1          up 1.26e-07        <NA>       NA        <NA>       NA
## 22     GLDC          up 5.29e-04        <NA>       NA          up 8.09e-03
## 23   IGFBP6        down 9.10e-03        <NA>       NA        <NA>       NA
## 24   KCNIP1        down 9.09e-03          up 1.76e-05          up 1.00e-07
## 25   KCNIP1        down 9.09e-03          up 1.76e-05          up 8.26e-04
## 26     LRP4          up 2.77e-03          up 1.21e-05          up 1.13e-06
## 27    MMP15          up 7.07e-03        <NA>       NA        <NA>       NA
## 28    NLGN3          up 1.21e-03          up 1.58e-10          up 4.67e-06
## 29     NT5E        down 4.76e-03          up 1.09e-07          up 2.29e-06
## 30  PCDHGC3          up 2.09e-04          up 2.16e-04          up 2.94e-04
## 31  PCDHGC3          up 2.09e-04          up 2.30e-04          up 2.94e-04
## 32  PCDHGC3          up 2.09e-04          up 1.07e-04          up 2.94e-04
## 33  PCDHGC3          up 2.09e-04          up 3.08e-03          up 2.94e-04
## 34  PCDHGC3          up 2.09e-04          up 1.83e-04          up 2.94e-04
## 35  PCDHGC3          up 2.09e-04          up 1.34e-04          up 2.94e-04
## 36  PCDHGC3          up 2.09e-04          up 2.16e-04          up 4.07e-03
## 37  PCDHGC3          up 2.09e-04          up 2.30e-04          up 4.07e-03
## 38  PCDHGC3          up 2.09e-04          up 1.07e-04          up 4.07e-03
## 39  PCDHGC3          up 2.09e-04          up 3.08e-03          up 4.07e-03
## 40  PCDHGC3          up 2.09e-04          up 1.83e-04          up 4.07e-03
## 41  PCDHGC3          up 2.09e-04          up 1.34e-04          up 4.07e-03
## 42  PCDHGC3          up 2.09e-04          up 2.16e-04          up 1.13e-03
## 43  PCDHGC3          up 2.09e-04          up 2.30e-04          up 1.13e-03
```

```
## 44  PCDHGC3          up 2.09e-04          up 1.07e-04          up 1.13e-03
## 45  PCDHGC3          up 2.09e-04          up 3.08e-03          up 1.13e-03
## 46  PCDHGC3          up 2.09e-04          up 1.83e-04          up 1.13e-03
## 47  PCDHGC3          up 2.09e-04          up 1.34e-04          up 1.13e-03
## 48  PCDHGC3          up 2.09e-04          up 2.16e-04          up 7.52e-04
## 49  PCDHGC3          up 2.09e-04          up 2.30e-04          up 7.52e-04
## 50  PCDHGC3          up 2.09e-04          up 1.07e-04          up 7.52e-04
## 51  PCDHGC3          up 2.09e-04          up 3.08e-03          up 7.52e-04
## 52  PCDHGC3          up 2.09e-04          up 1.83e-04          up 7.52e-04
## 53  PCDHGC3          up 2.09e-04          up 1.34e-04          up 7.52e-04
## 54  PCDHGC3          up 2.09e-04          up 2.16e-04          up 6.43e-05
## 55  PCDHGC3          up 2.09e-04          up 2.30e-04          up 6.43e-05
## 56  PCDHGC3          up 2.09e-04          up 1.07e-04          up 6.43e-05
## 57  PCDHGC3          up 2.09e-04          up 3.08e-03          up 6.43e-05
## 58  PCDHGC3          up 2.09e-04          up 1.83e-04          up 6.43e-05
## 59  PCDHGC3          up 2.09e-04          up 1.34e-04          up 6.43e-05
## 60  PCDHGC3          up 2.09e-04          up 2.16e-04          up 1.53e-03
## 61  PCDHGC3          up 2.09e-04          up 2.30e-04          up 1.53e-03
## 62  PCDHGC3          up 2.09e-04          up 1.07e-04          up 1.53e-03
## 63  PCDHGC3          up 2.09e-04          up 3.08e-03          up 1.53e-03
## 64  PCDHGC3          up 2.09e-04          up 1.83e-04          up 1.53e-03
## 65  PCDHGC3          up 2.09e-04          up 1.34e-04          up 1.53e-03
## 66   RSBN1L        down 7.15e-03        <NA>       NA        down 6.20e-03
## 67    S1PR1          up 9.28e-14        <NA>       NA        down 3.45e-02
## 68     SHC3          up 8.35e-03          up 7.89e-03          up 3.45e-02
## 69 SLC9A3R1          up 1.51e-04        down 2.13e-02        <NA>       NA
## 70    SOCS2        down 6.28e-03        <NA>       NA        down 3.89e-04
## 71    SOCS2        down 6.28e-03        <NA>       NA        down 2.39e-04
## 72   SPATA6        down 5.86e-03          up 8.02e-09          up 2.01e-04
## 73   SPATA6        down 5.86e-03          up 5.27e-08          up 2.01e-04
## 74   SPRED1          up 7.78e-04        <NA>       NA          up 2.88e-11
## 75    SPRY2          up 2.41e-06        <NA>       NA          up 4.08e-05
## 76    SPRY4          up 2.87e-03          up 3.06e-07          up 5.42e-07
## 77    SPRY4          up 2.87e-03          up 3.06e-07          up 3.77e-09
## 78    TPPP3        down 2.94e-03        <NA>       NA          up 8.78e-06
```

```
best_targets <- p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  full_join(p_sig31_GSE12907, by="GENE") %>%
  full_join(tarGenes31, by="GENE") %>%
  mutate(changemouse = ifelse(log2FoldChange > 0, "up", "down")) %>%
  mutate(change42656 = ifelse(statistic.x > 0, "up", "down")) %>%
  mutate(change12907 = ifelse(statistic.y > 0, "up", "down")) %>%
  mutate(pmouse = padj, p42656 = pvalue.x, p12907 = pvalue.y) %>%
  dplyr::select(GENE, changemouse, change12907, change42656) %>%
  filter(changemouse == change12907 | changemouse == change42656) %>%
  distinct() %>%
  arrange(GENE)


best_targets
```

```
##       GENE changemouse change12907 change42656
## 1    DUSP6          up          up          up
## 2    FABP5          up          up        <NA>
```

```
## 3     GLDC          up        <NA>          up
## 4     LRP4          up          up          up
## 5    NLGN3          up          up          up
## 6  PCDHGC3          up          up          up
## 7   RSBN1L        down        <NA>        down
## 8     SHC3          up          up          up
## 9    SOCS2        down        <NA>        down
## 10  SPRED1          up        <NA>          up
## 11   SPRY2          up        <NA>          up
## 12   SPRY4          up          up          up

xtable(best_targets)

## % latex table generated in R 3.4.2 by xtable 1.8-2 package
## % Tue Jan 23 05:14:29 2018
## \begin{table}[ht]
## \centering
## \begin{tabular}{rllll}
##   \hline
##  & GENE & changemouse & change12907 & change42656 \\
##   \hline
## 1 & DUSP6 & up & up & up \\
##   2 & FABP5 & up & up &  \\
##   3 & GLDC & up &  & up \\
##   4 & LRP4 & up & up & up \\
##   5 & NLGN3 & up & up & up \\
##   6 & PCDHGC3 & up & up & up \\
##   7 & RSBN1L & down &  & down \\
##   8 & SHC3 & up & up & up \\
##   9 & SOCS2 & down &  & down \\
##   10 & SPRED1 & up &  & up \\
##   11 & SPRY2 & up &  & up \\
##   12 & SPRY4 & up & up & up \\
##    \hline
## \end{tabular}
## \end{table}
```

4.3.4 Note:

To confirm that the data human datasets were appropriate to use in comparison to the mouse data, we did a few analyses. We first checked that the data were normalized, indeed, quantile normalization was used with GSE42656. Additionally, to scale the differential expression, we looked at the global differential expression rates. In GSE42656, of all the probes, approximately 1/3 were differentially expressed; of the probes associated with the 31 signficant ETV5 target genes, more than half were differentially expressed in the human data. In GSE12907, of all the probes, approximately 24% were differentially expressed; of the probes associated with the 31 signficant ETV5 target genes, 41% were differentially expressed.

4.4 GO Analysis, GSE42656 only

Below are the top categories for the 31 targets which are differentially expressed. The category "negative regulation of response to stimulus" is over-represented in our 31 genes (with 12 of 31 being involved in the category).

```
assayed.genes <- DEanalysis$gene
de.genes <- ETV5_sig$GENE
target.genes <- ETV5_targets$GENE

gene.vector=as.integer(assayed.genes%in%de.genes)
names(gene.vector)=assayed.genes

de.pwf = nullp(gene.vector, genome='mm9', id='geneSymbol', plot.fit=FALSE)  #prob weighting function

gopvals = goseq(de.pwf, genome='mm9', id='geneSymbol', test.cats=c("GO:BP"))
gopvals[p.adjust(gopvals$over_represented_pvalue, method = "BH") < 0.05,]

##         category over_represented_pvalue under_represented_pvalue
## 8283  GO:0048585               7.96e-07                         1
## 6901  GO:0043407               3.60e-06                         1
## 10455 GO:0070373               4.22e-06                         1
## 6903  GO:0043409               4.79e-06                         1
##       numDEInCat numInCat                                      term
## 8283          12     1382  negative regulation of response to stimulus
## 6901           4       65   negative regulation of MAP kinase activity
## 10455          4       66 negative regulation of ERK1 and ERK2 cascade
## 6903           5      152          negative regulation of MAPK cascade
##       ontology
## 8283        BP
## 6901        BP
## 10455       BP
## 6903        BP

xtable(gopvals[p.adjust(gopvals$over_represented_pvalue, method = "BH") < 0.05,c(1,6)])

## % latex table generated in R 3.4.2 by xtable 1.8-2 package
## % Tue Jan 23 05:14:43 2018
## \begin{table}[ht]
## \centering
## \begin{tabular}{rll}
##   \hline
##  & category & term \\
##   \hline
## 8283 & GO:0048585 & negative regulation of response to stimulus \\
##   6901 & GO:0043407 & negative regulation of MAP kinase activity \\
##   10455 & GO:0070373 & negative regulation of ERK1 and ERK2 cascade \\
##   6903 & GO:0043409 & negative regulation of MAPK cascade \\
##    \hline
## \end{tabular}
## \end{table}

enriched.GO = gopvals$category[p.adjust(gopvals$over_represented_pvalue, method = "BH") < 0.05]

## To find what the GO categories are:
for (go in enriched.GO){
  print(GOTERM[[go]])
  cat("--------------------------------------------\n")
  }

## GOID: GO:0048585
## Term: negative regulation of response to stimulus
```

```
## Ontology: BP
## Definition: Any process that stops, prevents, or reduces the
##     frequency, rate or extent of a response to a stimulus.
##     Response to stimulus is a change in state or activity of a
##     cell or an organism (in terms of movement, secretion, enzyme
##     production, gene expression, etc.) as a result of a stimulus.
## Synonym: down regulation of response to stimulus
## Synonym: down-regulation of response to stimulus
## Synonym: downregulation of response to stimulus
## Synonym: inhibition of response to stimulus
## ----------------------------------------------
## GOID: GO:0043407
## Term: negative regulation of MAP kinase activity
## Ontology: BP
## Definition: Any process that stops, prevents, or reduces the
##     frequency, rate or extent of MAP kinase activity.
## Synonym: down regulation of MAPK activity
## Synonym: down-regulation of MAPK activity
## Synonym: downregulation of MAPK activity
## Synonym: inhibition of MAPK activity
## Synonym: negative regulation of mitogen activated protein kinase
##     activity
## Synonym: negative regulation of mitogen-activated protein kinase
##     activity
## ----------------------------------------------
## GOID: GO:0070373
## Term: negative regulation of ERK1 and ERK2 cascade
## Ontology: BP
## Definition: Any process that stops, prevents, or reduces the
##     frequency, rate or extent of signal transduction mediated by
##     the ERK1 and ERK2 cascade.
## Synonym: down regulation of ERK1 and ERK2 cascade
## Synonym: down-regulation of ERK1 and ERK2 cascade
## Synonym: downregulation of ERK1 and ERK2 cascade
## Synonym: inhibition of ERK1 and ERK2 cascade
## Synonym: negative regulation of ERK cascade
## Synonym: negative regulation of ERK1 and ERK2 signaling pathway
## Synonym: negative regulation of ERK1 and ERK2 signalling pathway
## Synonym: negative regulation of ERK1 cascade
## Synonym: negative regulation of ERK1/2 cascade
## Synonym: negative regulation of ERK2 cascade
## Synonym: negative regulation of MAPK1 cascade
## Synonym: negative regulation of MAPK3 cascade
## ----------------------------------------------
## GOID: GO:0043409
## Term: negative regulation of MAPK cascade
## Ontology: BP
## Definition: Any process that stops, prevents, or reduces the
##     frequency, rate or extent of signal transduction mediated by
##     the MAPKKK cascade.
## Synonym: down regulation of MAPK cascade
## Synonym: down regulation of MAPKKK cascade
## Synonym: down-regulation of MAPK cascade
## Synonym: down-regulation of MAPKKK cascade
## Synonym: downregulation of MAPK cascade
```

```
## Synonym: downregulation of MAPKKK cascade
## Synonym: inhibition of MAPK cascade
## Synonym: inhibition of MAPKKK cascade
## Synonym: negative regulation of MAP kinase cascade
## Synonym: negative regulation of MAP kinase kinase kinase cascade
## Synonym: negative regulation of MAPKKK cascade
## Synonym: negative regulation of mitogen activated protein kinase
##     cascade
## Synonym: negative regulation of mitogen activated protein kinase
##     kinase kinase cascade
## Synonym: negative regulation of mitogen-activated protein kinase
##     cascade
## Synonym: negative regulation of mitogen-activated protein kinase
##     kinase kinase cascade
## -------------------------------------------
```

5 New Mouse Data

On November 22, 2016, Peter Sims gave us additional data. The additional data consist of three 6-week FF and three 6-week FMC mouse optic nerve RNA-seq samples. We will use the observations to identify if ETV5 is still significant and also to see if the same targets are differentially expressed (and in what direction).

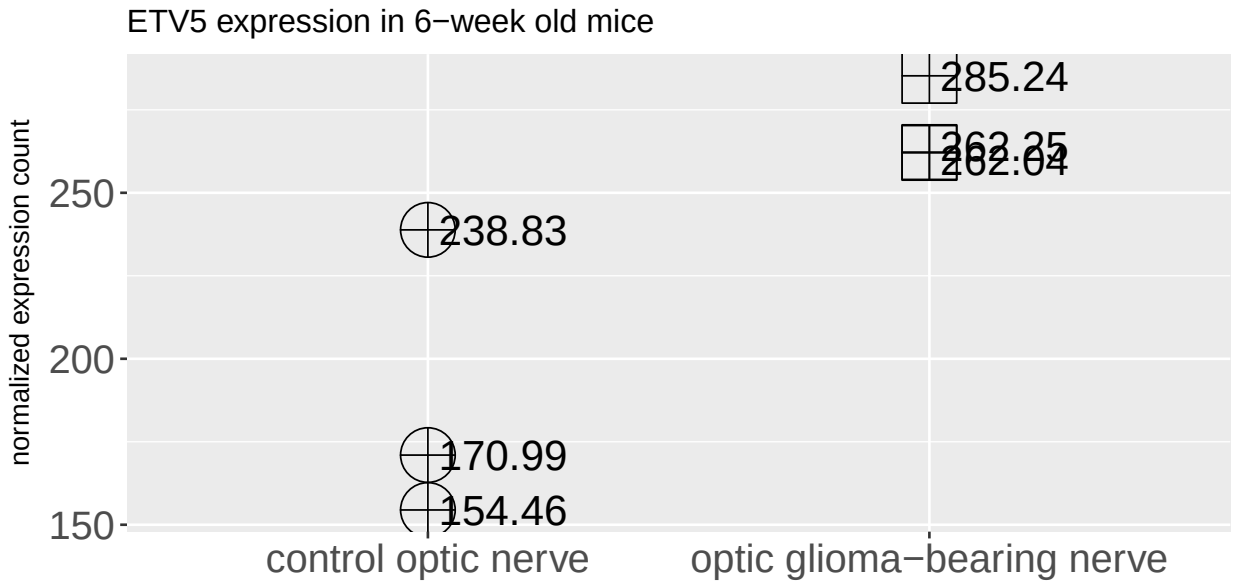
5.1 Normalizing and DE for young data

```
micefinalyoung <- read.delim("FF6wks_FMC6wks.cts.txt")
colnames(micefinalyoung) <- c("gene", paste("FFY", 1:3, sep=""),
                              paste("FMCY", 1:3, sep=""))

geneinfo <- read_excel("FF_FMC_matrixv2.xlsx",
                       col_names=c("gene", "REFSEQ_ID", paste("FF", 1:5, sep=""),
                                   paste("FMC",1:5, sep=""), "X", paste("FFk",1:5,sep=""),
                                   paste("FMCk",1:5,sep="")))
geneinfo <- geneinfo %>% dplyr::select(gene, REFSEQ_ID)

micefinalyoung <- micefinalyoung %>% left_join(geneinfo, by="gene")

condyoung <- factor(c(rep("FFY", 3), rep("FMCY", 3)))
ddsyoung <- DESeq2::DESeqDataSetFromMatrix(micefinalyoung[,-c(1,8)],
                                           DataFrame(condyoung), ~ condyoung)
ddsyoung <- DESeq2::DESeq(ddsyoung)
resyoung <- results(ddsyoung)  # Diff Exp results if we want/need the p-values
dds.datayoung <- counts(ddsyoung, normalized=TRUE)
miceoutyoung <- data.frame(gene=toupper(micefinalyoung$gene),
                           REFSEQ_ID=micefinalyoung$REFSEQ_ID, dds.datayoung)
micepsyoung <- data.frame(gene=toupper(micefinalyoung$gene),
                          REFSEQ_ID=micefinalyoung$REFSEQ_ID, resyoung)
```

5.2 Significance of ETV5 and its targets

```
# the 504 target genes of ETV5 with the ORIGINAL DE p-values
ETV5andtargets <- data.frame(GENE = c("ETV5", tarGenesETV5[,1]))
head(tarGenesETV5)

##          GENE pvalue  padj log2FoldChange
## 1        ABI1 0.3099 0.565         0.1080
## 2      KHDC1L     NA    NA             NA
## 3 ZSCAN16-AS1     NA    NA             NA
## 4     BCL2L11 0.8686 0.943        -0.0344
## 5       ABCB6 0.8503 0.934        -0.0401
## 6       TSSC4 0.0513 0.211         0.4401

# now investigating the new p-values with the 6-week mice data
DEpsyoung = micepsyoung[,c("gene", "pvalue", "padj", "log2FoldChange")]
DEpsyoung = DEpsyoung %>% mutate(GENE=gene) %>% dplyr::select(-gene)

micefinalyoung %>% filter(gene == "Etv5") # raw data

##   gene FFY1 FFY2 FFY3 FMCY1 FMCY2 FMCY3 REFSEQ_ID
## 1 Etv5  223  172  152   332   260   249 NM_023794
```

```
miceoutyoung %>% filter(gene == "ETV5") # normalized data

##   gene REFSEQ_ID FFY1 FFY2 FFY3 FMCY1 FMCY2 FMCY3
## 1 ETV5 NM_023794  239  154  171   285   262   262

DEpsyoung %>% filter(GENE == "ETV5") # DE results

##   pvalue  padj log2FoldChange GENE
## 1 0.0113 0.298          0.523 ETV5

norm6wkdata <- miceoutyoung %>% filter(gene == "ETV5") %>% dplyr::select(starts_with("F"))

normdataplot <- data.frame(data6wk = unlist(norm6wkdata), sample=c(rep("FFY",3), rep("FMCY",3)))

normdataplot <- normdataplot %>%
  mutate(sample2 = ifelse(sample == "FFY", "control optic nerve", "optic glioma-bearing nerve")) %>%
  mutate(labelpos = ifelse(normdataplot$sample=='FMCY',
                           ifelse(normdataplot$data6wk < 262.2, -2, ifelse(normdataplot$data6wk < 265, 2, 0)),0))

ggplot(normdataplot, aes(x=sample2, y=as.numeric(data6wk),
                         label=round(as.numeric(data6wk),2), shape=sample2)) +
       ylab("normalized expression count") +
       xlab("") + theme(axis.text=element_text(size=16)) +
       geom_point(size=10)  + theme(legend.position="none") +
  scale_shape_manual(values=c(10,12))+
  geom_text(nudge_y=normdataplot$labelpos, nudge_x=0.15, size=6) +
  ggtitle("ETV5 expression in 6-week old mice")
```



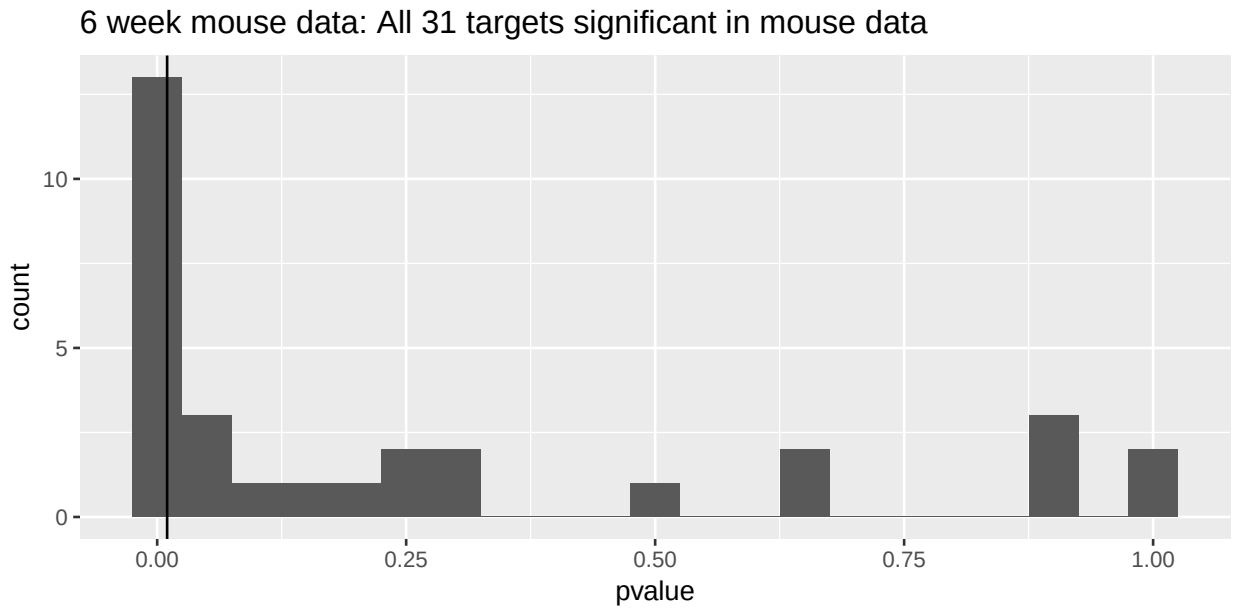

```
# the first 10 of all 504 targets of ETV5
left_join(ETV5andtargets, DEpsyoung, by= "GENE") %>%
  arrange(padj) %>%
  head(10)

##        GENE   pvalue     padj log2FoldChange
```

```
## 1     FABP7 2.31e-16 8.40e-13          1.245
## 2     FABP5 1.19e-12 1.32e-09          1.208
## 3  SLC39A12 1.29e-07 3.58e-05          0.693
## 4    ELOVL2 2.44e-07 5.79e-05          2.047
## 5    SPRED1 1.41e-05 1.91e-03          0.555
## 6     S1PR1 1.76e-05 2.25e-03          0.739
## 7    LHFPL3 3.68e-05 4.07e-03          0.778
## 8   FAM181B 4.59e-05 4.85e-03          1.349
## 9     CHST2 4.84e-05 5.08e-03          0.655
## 10    ACSL3 8.56e-05 7.89e-03          0.547
```

```
# the 31 targets which were DE in the original data
left_join(ETV5andtargets, DEpsyoung, by= "GENE") %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  arrange(padj)
```

```
##        GENE   pvalue     padj log2FoldChange
## 1     FABP7 2.31e-16 8.40e-13        1.24487
## 2     FABP5 1.19e-12 1.32e-09        1.20821
## 3    ELOVL2 2.44e-07 5.79e-05        2.04672
## 4    SPRED1 1.41e-05 1.91e-03        0.55473
## 5     S1PR1 1.76e-05 2.25e-03        0.73925
## 6     CHST2 4.84e-05 5.08e-03        0.65504
## 7   PCDHGC3 8.94e-05 8.14e-03        0.58749
## 8      GJA1 5.60e-04 3.76e-02        0.59708
## 9      LRP4 7.34e-04 4.72e-02        0.64095
## 10    MMP15 1.28e-03 7.19e-02        0.57434
## 11    SPRY4 1.42e-02 3.47e-01        0.70673
## 12 SLC9A3R1 1.60e-02 3.74e-01        0.37267
## 13   CXCL14 2.20e-02 4.43e-01        0.53143
## 14    SOCS2 3.31e-02 5.47e-01       -0.69123
## 15    SPRY2 4.04e-02 5.96e-01        0.54697
## 16    TPPP3 6.76e-02 7.26e-01       -0.33613
## 17     GLDC 8.45e-02 7.87e-01        0.62879
## 18    BTBD3 1.49e-01 9.39e-01        0.33526
## 19   DNAJB4 9.94e-01 1.00e+00       -0.00113
## 20   COL2A1 2.06e-01 1.00e+00        0.44377
## 21    DUSP6 9.92e-01 1.00e+00        0.00201
## 22      AK4 2.74e-01 1.00e+00        0.27608
## 23   RSBN1L 4.84e-01 1.00e+00        0.10420
## 24    GAP43 6.74e-01 1.00e+00       -0.08051
## 25   KCNIP1 9.09e-01 1.00e+00        0.04114
## 26   IGFBP6 9.16e-01 1.00e+00        0.02516
## 27     NT5E 8.93e-01 1.00e+00        0.02881
## 28     SHC3 2.41e-01 1.00e+00        0.34651
## 29    NLGN3 3.23e-01 1.00e+00        0.19639
## 30   SPATA6 6.31e-01 1.00e+00       -0.13586
## 31    DOCK4 2.81e-01 1.00e+00        0.20033
```

```
left_join(ETV5andtargets, DEpsyoung, by= "GENE") %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  arrange(padj) %>%
  ggplot(aes(x=pvalue)) +
  geom_histogram(binwidth=0.05) +
  geom_vline(xintercept=siglevel) +
  ggtitle("6 week mouse data: All 31 targets significant in mouse data")
```



5.2.1 Adding the young data to up/down comparison

```
best_targets <- p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  filter(pvalue <= 0.05)  %>%
  full_join(p_sig31_GSE12907, by="GENE") %>%
  full_join(tarGenes31, by="GENE") %>%
  full_join(DEpsyoung, by="GENE") %>%
  mutate(changemouse = ifelse(log2FoldChange.x > 0, "up", "down")) %>%
  mutate(changeyoung = ifelse(log2FoldChange.y > 0, "up", "down")) %>%
  mutate(change42656 = ifelse(statistic.x > 0, "up", "down")) %>%
  mutate(change12907 = ifelse(statistic.y > 0, "up", "down")) %>%
  mutate(pmouse = padj.x, pyoung = padj.y, p42656 = pvalue.x, p12907 = pvalue.y) %>%
  dplyr::select(GENE, changemouse, changeyoung, change12907, change42656) %>%
  filter(changemouse == change12907 | changemouse == change42656) %>%
  distinct() %>%
  arrange(GENE)


best_targets
```

```
##       GENE changemouse changeyoung change12907 change42656
## 1    DUSP6          up          up          up          up
## 2    FABP5          up          up          up        <NA>
## 3     GLDC          up          up        <NA>          up
## 4     LRP4          up          up          up          up
## 5    NLGN3          up          up          up          up
## 6  PCDHGC3          up          up          up          up
## 7   RSBN1L        down          up        <NA>        down
## 8     SHC3          up          up          up          up
## 9    SOCS2        down        down        <NA>        down
## 10  SPRED1          up          up        <NA>          up
## 11   SPRY2          up          up        <NA>          up
```

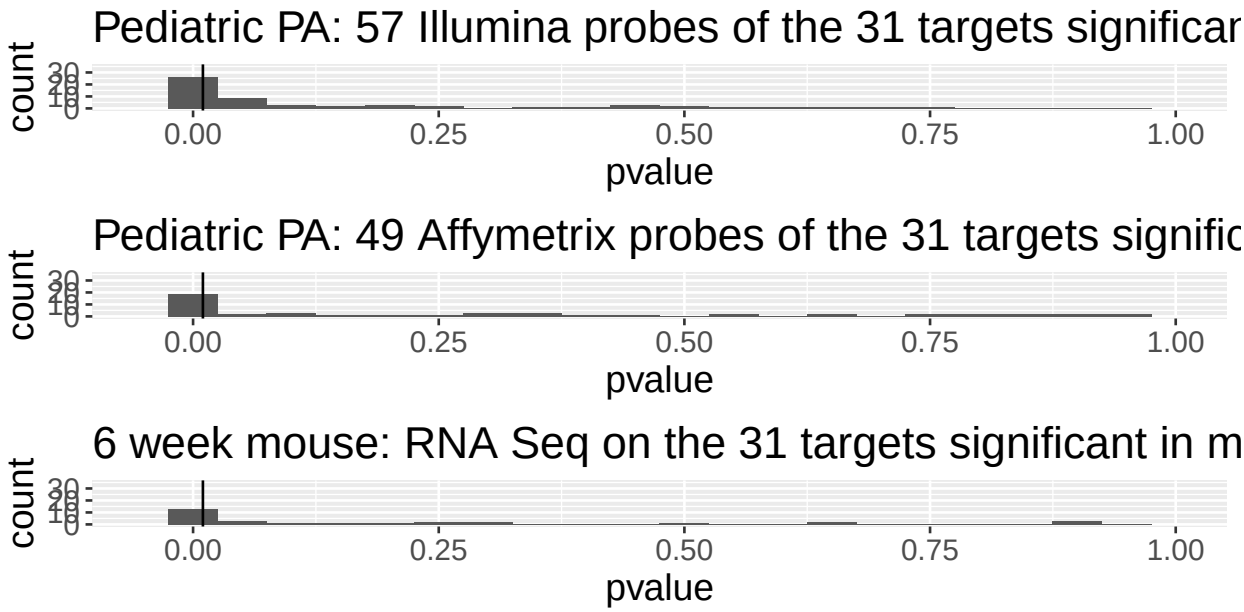
```
## 12   SPRY4          up          up          up          up
```

5.3 Figure 7

```
#install.packages("gridExtra")
library("gridExtra")

##
## Attaching package:  'gridExtra'
## The following object is masked from 'package:gdata':
##
##    combine
## The following object is masked from 'package:Biobase':
##
##    combine
## The following object is masked from 'package:BiocGenerics':
##
##    combine
## The following object is masked from 'package:dplyr':
##
##    combine
```

```
hist1 <- p_targ_GSE42656 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  ggplot(aes(x=pvalue)) + geom_histogram(binwidth = .05) +
  geom_vline(xintercept = siglevel) + xlim(c(-0.05,1)) + ylim(c(0,35)) +
  theme(text = element_text(size = 15)) +
  ggtitle("Pediatric PA: 57 Illumina probes of the 31 targets significant in mouse data")

hist2 <- p_targ_GSE12907 %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  ggplot(aes(x=pvalue)) + geom_histogram(binwidth = .05) +
  geom_vline(xintercept = siglevel) + xlim(c(-.05,1)) + ylim(c(0,35)) +
  theme(text = element_text(size = 15)) +
  ggtitle("Pediatric PA: 49 Affymetrix probes of the 31 targets significant in mouse data")


tempdata <- left_join(ETV5andtargets, DEpsyoung, by= "GENE") %>%
  filter(GENE %in% ETV5_sig$GENE) %>%
  arrange(padj)

hist3 <-  ggplot(tempdata, aes(x=pvalue)) +
  geom_histogram(binwidth=0.05) +
  geom_vline(xintercept=siglevel) +xlim(c(-0.05,1)) + ylim(c(0,35)) +
  theme(text = element_text(size = 15)) +
  ggtitle("6 week mouse: RNA Seq on the 31 targets significant in mouse data")

multiplot(hist1, hist2, hist3, cols=1)
```

```
## Warning:  Removed 1 rows containing missing values (geom_bar).
## Warning:  Removed 1 rows containing missing values (geom_bar).
## Warning:  Removed 1 rows containing missing values (geom_bar).
```


Pediatric PA: 57 Illumina probes of the 31 targets significar
count
30
20
10
0
0.00
0.25
0.50
0.75
1.00
pvalue
Pediatric PA: 49 Affymetrix probes of the 31 targets signific
count
30
20
10
0
0.00
0.25
0.50
0.75
1.00
pvalue
6 week mouse: RNA Seq on the 31 targets significant in m
count
30
20
10
0
0.00
0.25
0.50
0.75
1.00
pvalue