Analysis work-flow for the article *Targeted expression profiling by RNA-Seq improves detection of cellular dynamics during pregnancy and identifies a role for T cells in term parturition*

Adi L Tarca and Vincent J. Carey

November 29, 2018

Introduction

This document illustrates the analysis workflow included in the manuscript **Targeted expression profiling by RNA-Seq improves detection of cellular dynamics during pregnancy and identifies a role for T cells in term parturition** (Tarca et al. 2018). The goal of the analysis is to compare three omics platforms (Affymetrix HTA 2.0 microarrays, Illumina RNA-Seq, and targeted profiling by DriverMap) to detect changes with gestational age and with labor at term in whole blood samples collected from pregnant women.

Loading required packages

To access the data needed for analysis, we install and load the *pregnomics* package that includes all expression data sets and relevant metadata. You may also need to install the *devtools* package:

```r
library(devtools)
if(!require(pregnomics)){
install_github("atarca/pregnomics")
}else{library(pregnomics)}
```

Additional packages, including several from Bioconductor (Gentleman et al. 2004), needed for analysis, are also loaded. Note the version of annotation packages needed to reproduce the results described in (Tarca et al. 2018).

```
library(hta20sttranscriptcluster.db) #hta20sttranscriptcluster.db_8.3.1
library(org.Hs.eg.db)  #org.Hs.eg.db_3.2.3
library(EnsDb.Hsapiens.v75) # EnsDb.Hsapiens.v75_2.99.0
library(annotate)   #annotate_1.48.0
library(limma)
library(DESeq2)
library(UpSetR)
library(epiR)
library(pROC)
library(ROCR)
library(gplots)
library(Heatplus)
library(marray)
library(lme4)
library(splines)
```

Study Design

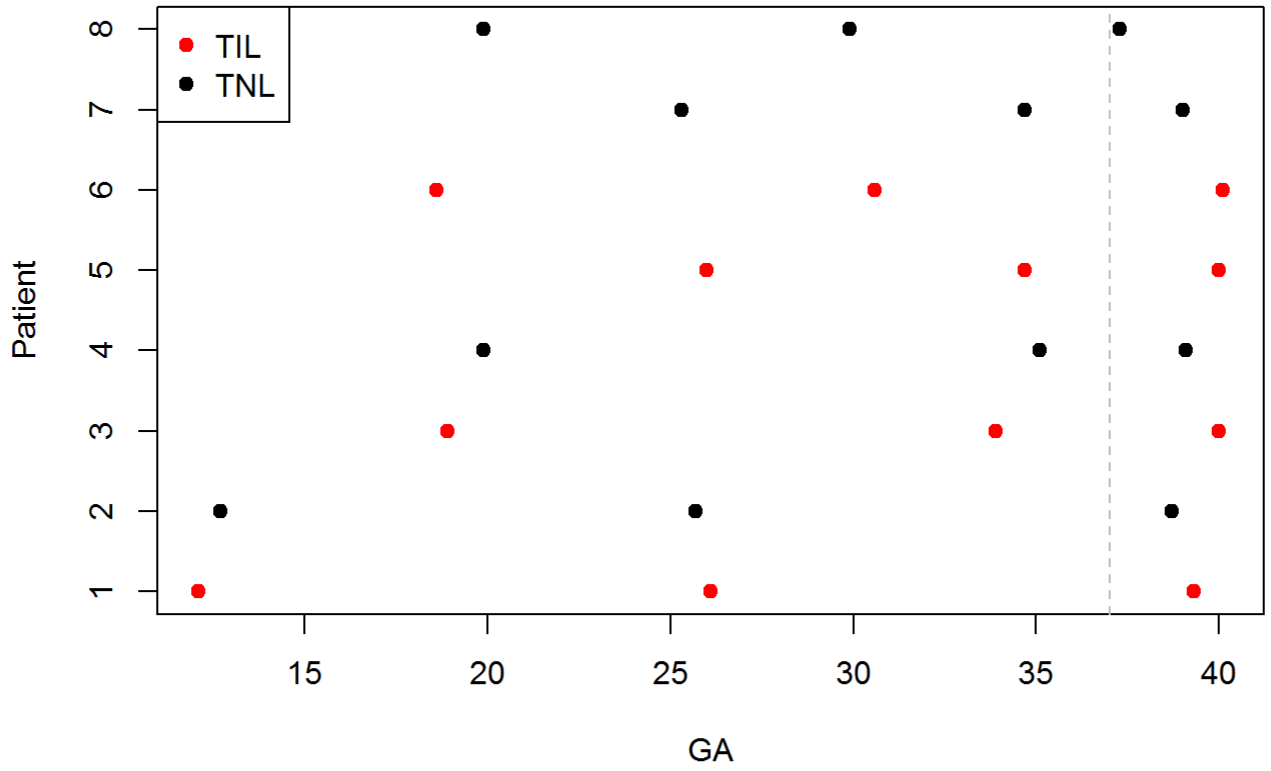
The characteristics of the 32 blood samples are provided in the *ano32* table which corresponds to Table S1 in (Tarca et al. 2018).

```
data(ano32)
ano32$T=factor(ifelse(ano32$GA<37,"Preterm","Term")) #define gestational age interval
anoALL<-ano32
head(anoALL,n=3)
```

```
##            SampleID IndividualID   GA Group RIN StorageMonths GAAnalysis
## Sample_10 Sample_10        mi548 26.0   TIL 6.3      47.43333          1
## Sample_11 Sample_11        mi548 34.7   TIL 6.4      45.40000          1
## Sample_12 Sample_12        mi548 40.0   TIL 6.8      44.16667          1
##           LaborAnalysis       T
## Sample_10             0 Preterm
## Sample_11             0 Preterm
## Sample_12             1    Term
```

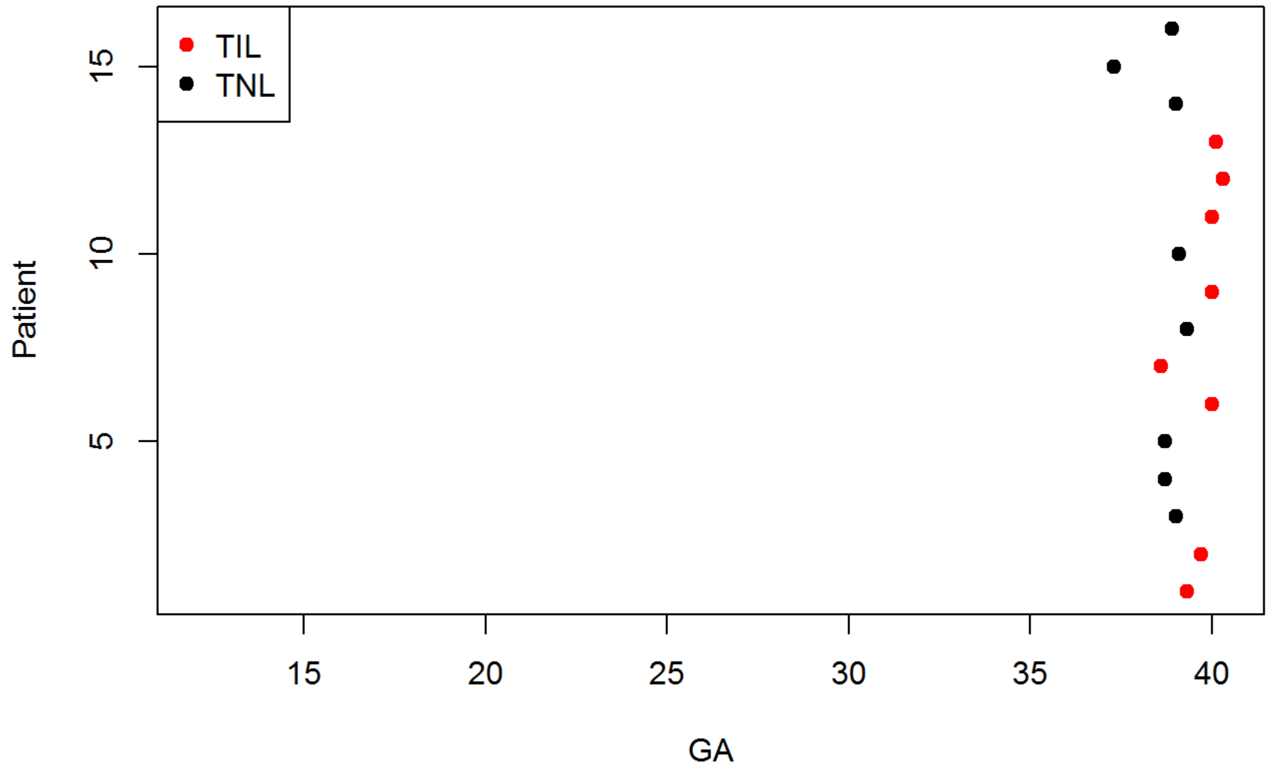
The analysis studying the effect of gestational age (GA) (term vs preterm gestation) is based on data from both women that had a spontaneous labor at term (TIL) and those that delivered by cesarean section (TNL) and had 3 longitudinal samples collected during gestation:

```
anoGA=anoALL[anoALL$GAAnalysis==1,]
plot(as.numeric(as.factor(IndividualID))~GA,data=anoGA,col=ifelse(anoGA$Group=="TIL","red","blac
k"),pch=19,xlab="GA",ylab="Patient")
legend("topleft",legend=c("TIL","TNL"),pch=c(19,19),col=c("red","black"))
abline(v=37,lty=2,col="grey")
```



The analysis studying the effect of labor at term (TIL vs TNL) is based on all samples collected at delivery:

```
anoLabor=anoALL[anoALL$LaborAnalysis==1,]
plot(as.numeric(as.factor(IndividualID))~GA,data=anoLabor,col=ifelse(anoLabor$Group=="TIL","red"
,"black"),pch=19,xlab="GA",ylab="Patient",xlim=range(ano32$GA))
legend("topleft",legend=c("TIL","TNL"),pch=c(19,19),col=c("red","black"))
```



Gene expression data

Gene expression data for HTA microarrays, RNA-Seq, DriverMap and qRT-PCR are availble loading the respective matrices:

```
data(package="pregnomics",list=c("esetHTA","Rcount","Ccount","esetPCR"))
```

Of note *esetHTA* and *esetPCR* data is on a log2 scale and can be reasonably assumed to be normally distributed. Therefore the *limma* package (Ritchie et al. 2015) will be leveraged to create functions that return the differential expression results for these datasets. In turn, *Rcount* and *Ccount* data obtained via sequencing are count data and hence will be analyzed using negative binomial models using the *DESeq2* package(Love, Huber, and Anders 2014).

Differential expression analysis for normally distributed data (microarray and qRT-PCR)

We define below a function that fits log2 gene expression data as a function of the gestational age interval (term vs preterm gestation, variable *T*) and uses a fixed effect for each woman (*IndividualID*) so that we obtain estimates of within subject changes with gestation:

```
analyzeGA_limma=function(ano,eset){
  ano$ID=factor(ano$IndividualID)
  design <- model.matrix(~0+T+IndividualID,ano)
  eset=eset[,rownames(ano)]
  colnames(design)<-substr(colnames(design),2,100)
  fit <- lmFit(eset, design)
  cont.matrix <- makeContrasts( contrasts="Term-Preterm",levels=design)
  fit2 <- contrasts.fit(fit, cont.matrix)
  fit2 <- eBayes(fit2)
  aT1<-topTable(fit2,coef=1, number=1000000, adjust="fdr")
  aT1$FC=2^abs(aT1$logFC)*sign(aT1$logFC) #signed linear fold change
  aT1$ID=rownames(aT1)
  aT1
}
```

Similarly, we define a function that perfroms the unpaired analysis between TIL and TNL groups, where *Group* is the variable defining the TIL vs TNL status:

```
analyzeLabor_limma=function(ano,eset){
design <- model.matrix(~0+Group,ano)
colnames(design)<-gsub("Group","",colnames(design))
fit <- lmFit(eset, design)
cont.matrix <- makeContrasts( contrasts="TIL-TNL",levels=design)
fit2 <- contrasts.fit(fit, cont.matrix)
fit2 <- eBayes(fit2)
aT1<-topTable(fit2,coef=1, number=1000000, adjust="fdr")
aT1$FC=2^abs(aT1$logFC)*sign(aT1$logFC) #signed linear fold change
aT1$ID=rownames(aT1)
aT1
}
```

Differential expression with HTA microarray data

We define here the annotation package that will be used to map Affymetrix transcript cluster IDs to gene symbols, and initialize a list to sore *limma* top tables for HTA data:

```
anpack="hta20sttranscriptcluster"
HTA=list() #will store DE results with GA and Labor
```

We call the *analyzeGA_limma* function created above passing the sample annotation data frame for this analysis *ano* and the HTA expression data for the corresponding samples *eset*:

```
#GA effect
aT1=analyzeGA_limma(anoGA,esetHTA[,rownames(anoGA)])
head(aT1,n=3)
```

```
##                      logFC  AveExpr        t      P.Value   adj.P.Val
## 2828012_st      1.7529332 4.806876 8.596907 6.341048e-08 0.002908073
## TC09000134.hg.1 0.5616826 5.784031 8.422414 8.635773e-08 0.002908073
## TC09000335.hg.1 0.9812865 5.763134 8.222084 1.237074e-07 0.002908073
##                        B       FC              ID
## 2828012_st      7.890442 3.370431      2828012_st
## TC09000134.hg.1 7.635034 1.475990 TC09000134.hg.1
## TC09000335.hg.1 7.335857 1.974225 TC09000335.hg.1
```

Next we add gene annotation and remove transcript clusters without a valid *ENTREZ* identifier:

```
#add gene annotation to top table
aT1$SYMBOL<-unlist((lookUp(aT1$ID, anpack, 'SYMBOL')))
aT1$ENTREZ<-unlist((lookUp(aT1$ID, anpack, 'ENTREZID')))
aT1=aT1[!is.na(aT1$ENTREZ),]
```

Then we retain transcript clusters for which at least one probset was deemed expressed (see Methods section in the paper) and then recalculate the adjusted p-values since we would have not retained transcripts 1) without a valid gene identifer or 2) if they were not expressed, regardless of the p-value from the differential expression test:

```
data(npspge) # based on detection above background from Affymetrix Transcriptome Analysis Consol
e
expressed=names(npspge)[npspge>0]
aT1=aT1[rownames(aT1)%in%expressed,]
aT1$adj.P.Val=p.adjust(aT1$P.Value,"fdr") #
HTA[["GAEffect"]]<-aT1
head(aT1,n=3)
```

```
##                      logFC  AveExpr        t      P.Value   adj.P.Val
## TC09000134.hg.1 0.5616826 5.784031 8.422414 8.635773e-08 0.001840767
## TC09000335.hg.1 0.9812865 5.763134 8.222084 1.237074e-07 0.001840767
## TC02003335.hg.1 0.3966638 4.724841 7.854297 2.425972e-07 0.002406564
##                        B       FC              ID SYMBOL ENTREZ
## TC09000134.hg.1 7.635034 1.475990 TC09000134.hg.1 DNAJA1   3301
## TC09000335.hg.1 7.335857 1.974225 TC09000335.hg.1  ANXA1    301
## TC02003335.hg.1 6.769782 1.316460 TC02003335.hg.1 MTHFD2  10797
```

A similar approach is used to generate a top table of differential expression statistics for the labor effect analysis. For both analyses the output is saved in a list (*HTA*) that will be used later.

```
aT1=analyzeLabor_limma(anoLabor,esetHTA[,rownames(anoLabor)])
aT1$SYMBOL<-unlist((lookUp(aT1$ID, anpack, 'SYMBOL')))
aT1$ENTREZ<-unlist((lookUp(aT1$ID, anpack, 'ENTREZID')))
aT1=aT1[!is.na(aT1$ENTREZ),]
aT1=aT1[rownames(aT1)%in%expressed,]
aT1$adj.P.Val=p.adjust(aT1$P.Value,"fdr")
HTA[["LaborEffect"]]<-aT1
head(aT1,n=3)
```

```
##                      logFC  AveExpr        t      P.Value adj.P.Val
## TC08002250.hg.1 0.3308649 6.195091 6.068226 1.194673e-05 0.2928539
## TC20000660.hg.1 0.7743451 7.595816 5.817761 1.968104e-05 0.2928539
## TC02001416.hg.1 0.7629459 8.596203 5.402295 4.585129e-05 0.3126317
##                         B       FC              ID  SYMBOL ENTREZ
## TC08002250.hg.1 1.5690964 1.257767 TC08002250.hg.1   CCAR2  57805
## TC20000660.hg.1 1.2907149 1.710413 TC20000660.hg.1 SNORD17 692086
## TC02001416.hg.1 0.8039668 1.696952 TC02001416.hg.1 SCARNA5 677775
```

Differential expression with qRT-PCR data

The gold standard of differential expression for a subset of 86 genes selected for validation, was defined using the same analysis models used for microarray data but using surrogates for log2 gene expression (-Delta CT values), for changes with gestational age:

```
PCR<-list()
aT1=analyzeGA_limma(anoGA,esetPCR[,rownames(anoGA)])
aT1$SYMBOL=rownames(aT1)
aT1$Sig=(aT1$P.Value<0.05)
PCR[["GAEffect"]]<-aT1
head(aT1,n=3)
```

```
##          logFC   AveExpr        t      P.Value    adj.P.Val        B
## ANXA1 1.070440 -3.787994 6.804643 6.119777e-07 5.263009e-05 6.103114
## IFIT1 1.082369 -8.421636 6.004645 4.010999e-06 1.099591e-04 4.236220
## RPS24 1.121194 -4.023192 5.935027 4.740271e-06 1.099591e-04 4.070444
##             FC    ID SYMBOL  Sig
## ANXA1 2.100074 ANXA1  ANXA1 TRUE
## IFIT1 2.117511 IFIT1  IFIT1 TRUE
## RPS24 2.175269 RPS24  RPS24 TRUE
```

and changes with labor:

```
aT1=analyzeLabor_limma(anoLabor,esetPCR[,rownames(anoLabor)])
aT1$SYMBOL=rownames(aT1)
aT1$Sig=(aT1$P.Value<0.05)
PCR[["LaborEffect"]]<-aT1
head(aT1,n=3)
```

```
##           logFC   AveExpr        t      P.Value  adj.P.Val          B
## KLRF1  1.271014 -6.722440 4.575224 0.0001754571 0.01508931  0.8369098
## AKR1C3 1.087162 -5.823647 4.262154 0.0003668047 0.01577260  0.1298058
## KLRC2  2.337048 -7.138628 4.077333 0.0006189696 0.01774380 -0.3332619
##              FC     ID SYMBOL  Sig
## KLRF1  2.413311  KLRF1  KLRF1 TRUE
## AKR1C3 2.124557 AKR1C3 AKR1C3 TRUE
## KLRC2  5.052676  KLRC2  KLRC2 TRUE
```

Differential expression analysis for count data (sequencing platforms)

For the count data generated by the two sequencing based platfroms (RNASeq and DriverMap), we first define the two functions that will use negative binomial models implemented in *DESeq2* package to identify genes that change with gestational age and with labor as follows:

```
analyzeGA_DESeq=function(ano,anoall,countM){
  dds<- DESeqDataSetFromMatrix(countData= countM[,rownames(ano)],colData= ano,design=~T+IndividualID)
  dds<- DESeq(dds)
  res<-results(dds,contrast=c("T","Term","Preterm"),independentFiltering=FALSE)
  res=as.data.frame(res)
  expressed=rownames(countM)[apply(countM[,rownames(anoall)]>=5,1,sum)>5]
  res=res[rownames(res)%in%expressed,]
  res=res[!is.na(res$log2FoldChange),]
  res$logFC=res$log2FoldChange
  names(res)[names(res)=="pvalue"]<-"P.Value"
  names(res)[names(res)=="padj"]<-"adj.P.Val"
  res
}

analyzeLabor_DESeq=function(ano,anoall,countM){
  dds<- DESeqDataSetFromMatrix(countData= countM[,rownames(ano)],colData= ano,design=~Group)
  dds<- DESeq(dds)
  res<-results(dds,contrast=c("Group","TIL","TNL"),independentFiltering=FALSE)
  res=as.data.frame(res)
  expressed=rownames(countM)[apply(countM[,rownames(anoall)]>=5,1,sum)>5]
  res=res[rownames(res)%in%expressed,]
  res=res[!is.na(res$log2FoldChange),]
  res$logFC=res$log2FoldChange
  names(res)[names(res)=="pvalue"]<-"P.Value"
  names(res)[names(res)=="padj"]<-"adj.P.Val"
  res
}
```

In both functions above, after genes are tested and p-values are computed, we drop genes for which a fold change could not be estimated (mostly due to 0 counts) and genes which were not expressed (do not have a count >=5 in at least 5 of the 32 samples). Before applying the two functions we retrieve ENSEMBLE gene annotation so that gene symbols can be assigned to RNASeq differential expression results:

```
edb <- EnsDb.Hsapiens.v75
Tx.ensemble <- transcripts(edb, columns = c("tx_id", "gene_id", "gene_name"),
                           return.type = "DataFrame")
```

Differential expression for RNASeq data

The two count data differential expression functions are applied to the RNASeq data and results are stored in a list called *RNASeq* as follows :

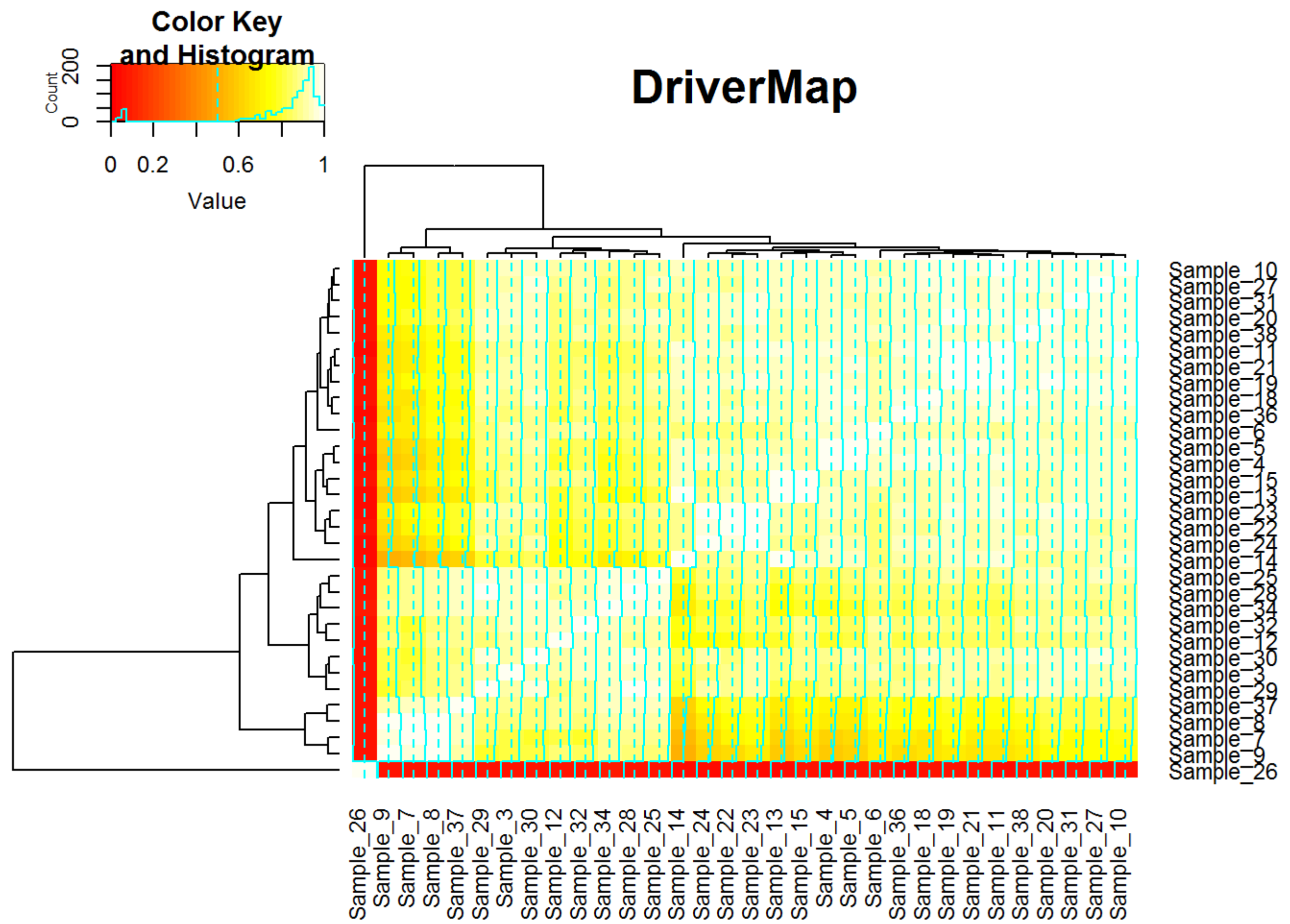
```
RNASeq<-list()
#GA effect
res=analyzeGA_DESeq(ano=anoGA,anoall=anoALL,countM=Rcount)
res$SYMBOL=Tx.ensemble[match(rownames(res),Tx.ensemble[,2]),3]
RNASeq[["GAEffect"]]<-res

#Labor effect
res=analyzeLabor_DESeq(ano=anoLabor,anoall=anoALL,countM=Rcount)
res$SYMBOL=Tx.ensemble[match(rownames(res),Tx.ensemble[,2]),3]
RNASeq[["LaborEffect"]]<-res
head(res,n=3)
```

```
##                  baseMean log2FoldChange     lfcSE      stat    P.Value
## ENSG00000000003  20.82937      0.1719070 0.4523755 0.3800094 0.70393841
## ENSG00000000419 175.82392      0.3368373 0.2285902 1.4735422 0.14060489
## ENSG00000000457 139.47841      0.2353901 0.1149515 2.0477336 0.04058611
##                 adj.P.Val     logFC SYMBOL
## ENSG00000000003         1 0.1719070 TSPAN6
## ENSG00000000419         1 0.3368373   DPM1
## ENSG00000000457         1 0.2353901  SCYL3
```

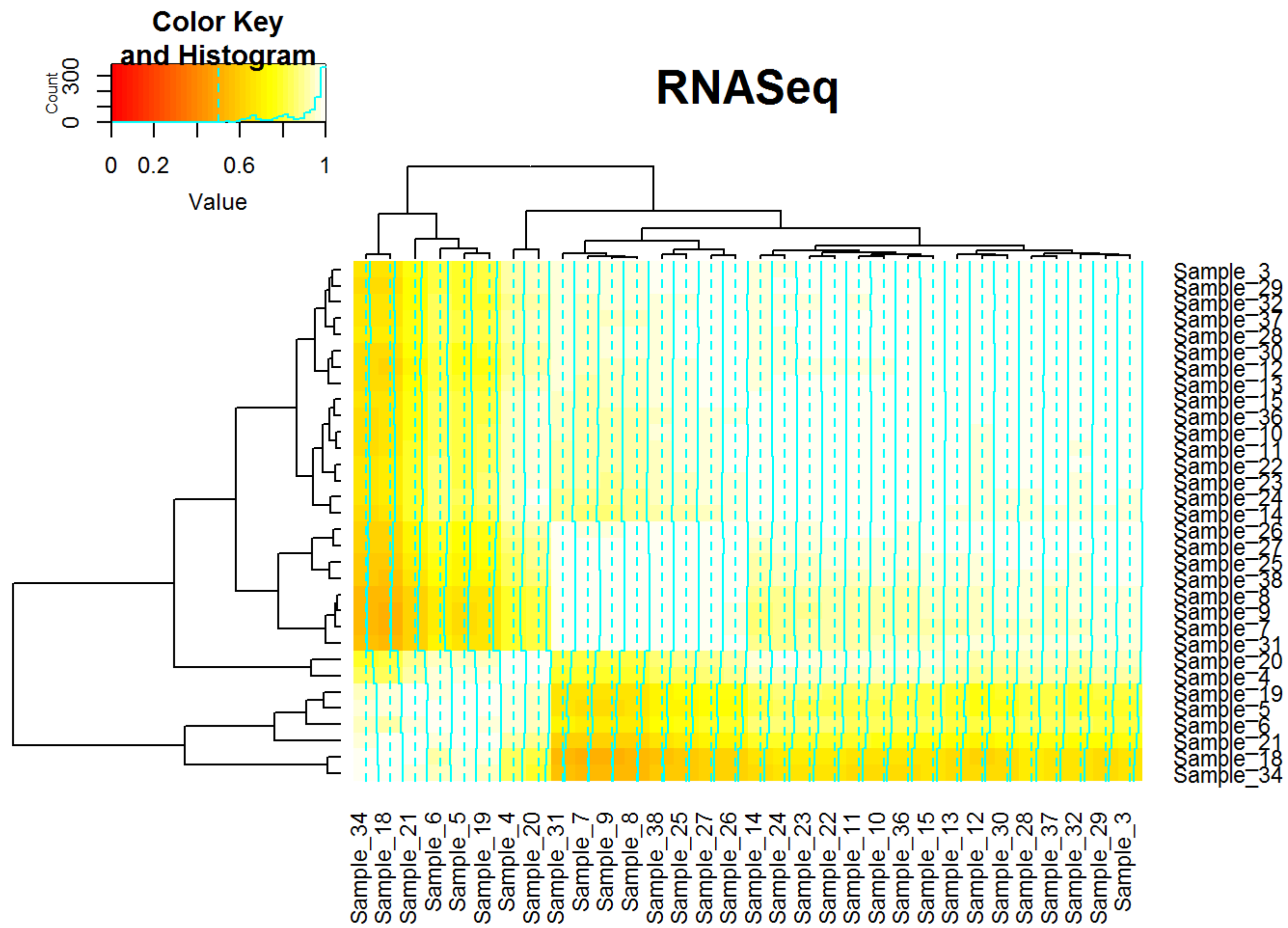
Differential expression for DriverMap

The same functions used for RNASeq data above are applied to DriverMap derived count data, except that Sample_26 (involved on in the labor effect analysis) needs to be removed first due to contamination. As seen below, the correlation between expression profiles for this sample is unexpectedly low for DriverMap due to contamination:

```
bk =seq(0, 1, by=0.025)
heatmap.2(cor(Ccount),breaks=bk, main="DriverMap")
```



```
heatmap.2(cor(Rcount),breaks=bk, main="RNASeq")
```



Note also that no gene annotation is needed, as the rows in the DriverMap cout data (*Ccount*) correspond already gene symbols:

```
CELLECTA=list()
#GA effect
anoall=anoALL
anoall=anoall[anoall$SampleID!="Sample_26",] #remove the contaminated sample
ano=anoGA
ano=ano[ano$SampleID!="Sample_26",]
res=analyzeGA_DESeq(ano,anoall,countM=Ccount)
res$SYMBOL=rownames(res)
CELLECTA[["GAEffect"]]<-res

#labor effect
res=analyzeLabor_DESeq(ano=anoLabor,anoall,countM=Ccount)
res$SYMBOL=rownames(res)
CELLECTA[["LaborEffect"]]<-res
head(res,n=3)
```

```
##           baseMean log2FoldChange     lfcSE      stat      P.Value
## A3GALT2   1.643093      0.8924494 0.8462079  1.054646 2.915875e-01
## A4GALT   46.324790     -4.1202976 0.8751259 -4.708234 2.498719e-06
## AAAS    176.140550     -0.2447221 0.1220811 -2.004586 4.500729e-02
##           adj.P.Val      logFC  SYMBOL
## A3GALT2 0.813937815  0.8924494 A3GALT2
## A4GALT  0.003103766 -4.1202976  A4GALT
## AAAS    0.484145468 -0.2447221    AAAS
```

Preparing expression matrices for downsteam ploting and gene set signature analysis

The gene expression matrices for platforms are next given shorter names, and prepared for downstream analyses by organizing the columns so that they correspond to the same samples. For count data, normalization is also applied to account for different library sizes. This is not needed for HTA data since it is already quantile normalized, while the qRT-PCR data used reference genes for normalization.

```
Hr=esetHTA[,rownames(anoALL)]
Pr=esetPCR[,rownames(anoALL)]

dds<- DESeqDataSetFromMatrix(countData= Rcount[,rownames(anoALL)],colData= anoALL,design=~Indivi
dualID)
dds=estimateSizeFactors(dds)
Rr=counts(dds, normalized=TRUE)

dds<- DESeqDataSetFromMatrix(countData= Ccount[,rownames(anoALL)],colData= anoALL,design=~Indivi
dualID)
dds=estimateSizeFactors(dds)
Cr=counts(dds, normalized=TRUE)
```

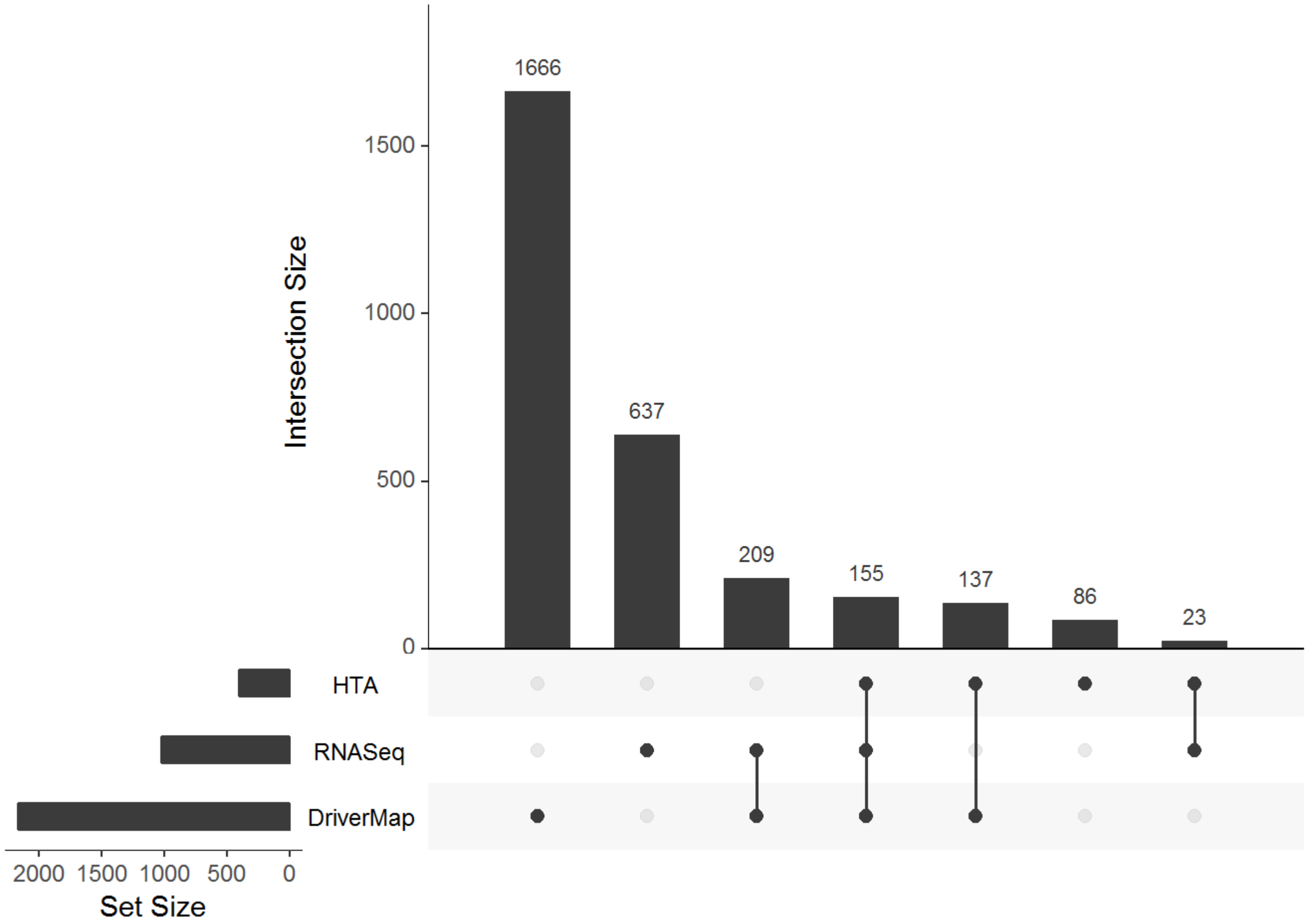
Next, for microarray data, one transcript cluster expression profile is retained for a given unique gene symbol, while for RNASeq, one ENSEMBLE gene expression profile is retained for each gene symbol. Finally, the normalized count data matrices are added 0.5 count to enable log2 transformation.

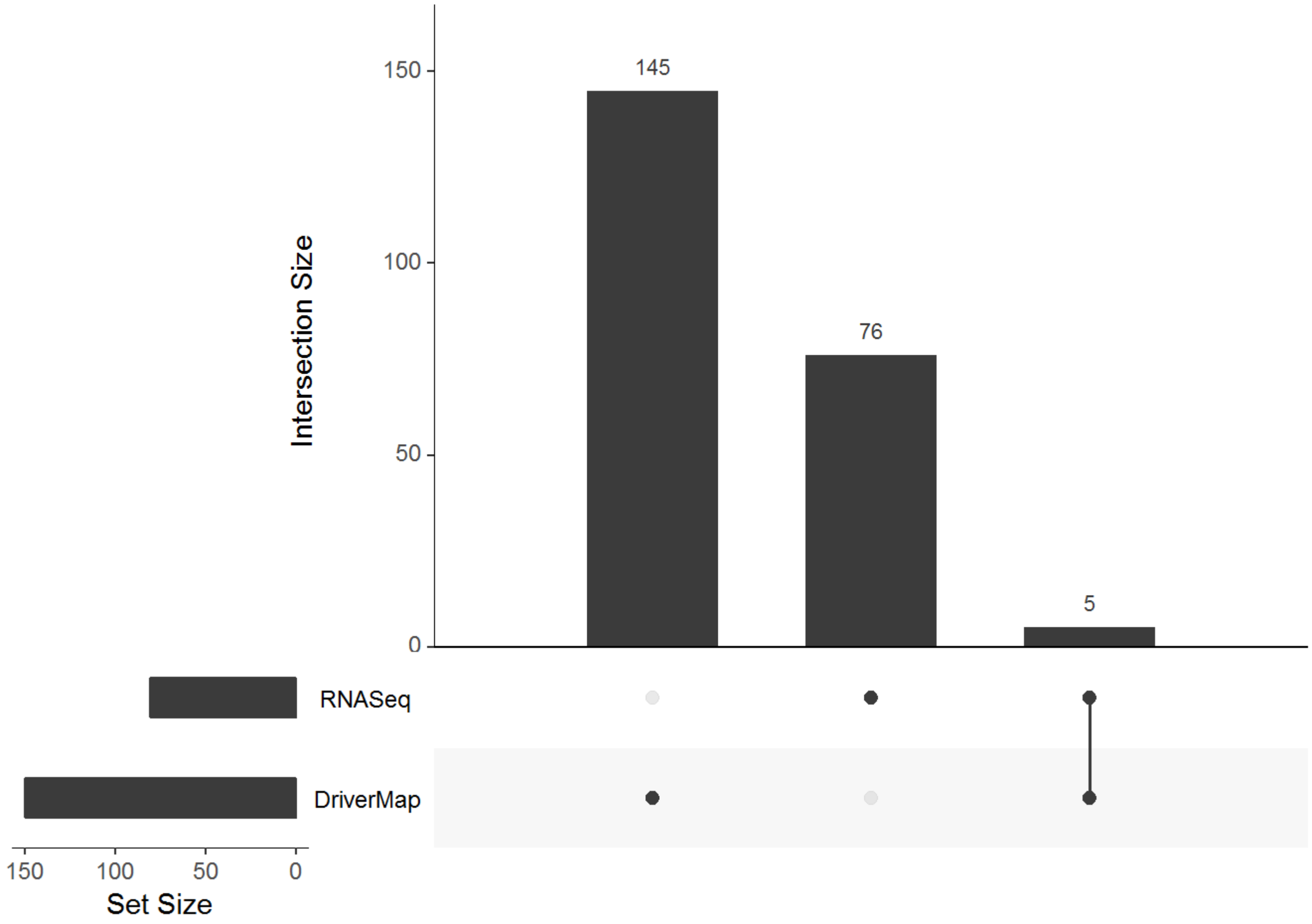
```
a=HTA[[2]] #
Hr=Hr[rownames(Hr)%in%rownames(a),]
Hr=Hr[rownames(a),]
Hr=Hr[!duplicated(a$SYMBOL),]
rownames(Hr)=a[rownames(Hr),"SYMBOL"]

a=RNASeq[[2]]
Rr=Rr[rownames(Rr)%in%rownames(a),]
Rr=Rr[rownames(a),]
Rr=Rr[!duplicated(a$SYMBOL),]
rownames(Rr)=a[rownames(Rr),"SYMBOL"]

Rr=log2(Rr+0.5)
Cr=log2(Cr+0.5)
# Hr, Rr, Cr, Pr are the four expression sets (one row per gene symbol)
```

Assessing the validation rates and differential expression overlap among platforms

To calculate the qRT-PCR validation rates (positive predicted values) for both comparisons (with gestation and and labor) for each of the three transcriptomics platform and create the differential expression overlap UpSet plots (Figure 1 in (Tarca et al. 2018)) we use the following:

```r
effs=c("GAEffect","LaborEffect")
platforms=c("HTA","RNASeq","DriverMap")
DEPile=list(HTA=HTA,RNASeq=RNASeq,DriverMap=CELLECTA)
ddPile<-NULL

for(eff in effs){
  vaT1t=PCR[[eff]];#gold standard differential expression
  DESymb=list()
  for(platf in names(DEPile)){
  x<-DEPile[[platf]][[eff]]
  x=x[!is.na(x$SYMBOL),]
  x$adj.P.Val<-p.adjust(x$P.Value,"fdr")
  x=x[order(x$P.Value),]
  x=x[!duplicated(x$SYMBOL),]
  if(sum(x$adj.P.Val<0.1)>=1){
  x=x[x$adj.P.Val<0.1,]
  x$Tested=x$SYMBOL%in%vaT1t[,"ID"]
  x$Validated=x$SYMBOL%in%vaT1t[vaT1t$Sig,"ID"]&sign(x$logFC)==sign(vaT1t$logFC[match(x$SYMBOL,vaT1t$ID)])
  x$Method=platf
  x$comp=eff
  ddPile=rbind(ddPile,x[,c("SYMBOL","P.Value","adj.P.Val","logFC","Tested","Validated","Method","comp")])
  DESymb[[platf]]<-paste(x$SYMBOL,ifelse(x$logFC>0,"+","-"))
  }
  }
  upset(fromList(DESymb), order.by = "freq",text.scale = 1.3)

}
```


Intersection Size
1666
637
209
155
137
86
23
1500
1000
500
0
HTA
RNASeq
DriverMap
2000 1500 1000 500 0
Set Size


Intersection Size
145
76
5
150
100
50
0
RNASeq
DriverMap
150 100 50 0
Set Size

Note that in the code above, duplicated gene symbols are first removed (if any), and a gene is considered validated if it was positive by the omics platform and significant by qRT-PCR with a matching direction of change. The differential expression expression statistics (both comparisons) for each platform (one row per gene symbol) will be stored in data frames *ddH*, *ddR* and *ddC* for HTA, RNASeq and DriverMap (Cellecta), respectively:

```
ddH=ddPile[ddPile$Method=="HTA",]
ddR=ddPile[ddPile$Method=="RNASeq",]
ddC=ddPile[ddPile$Method=="DriverMap",]
```

The qRT-PCR validation status of genes tested by each omics platfrom is now availble in the *ddPile* data frame. To summarize validation rates for each method we use the code below:

```
#genes present on all 4 platforms
comg=table(c(unique(rbind(HTA[[1]],HTA[[2]])$SYMBOL),unique(rbind(RNASeq[[1]],RNASeq[[2]])$SYMBO
L),
             unique(rbind(CELLECTA[[1]],CELLECTA[[2]])$SYMBOL),PCR[[1]]$SYMBOL))
comg=names(comg[comg==4])
comg66=comg

#validation rates
a=ddPile[,c("SYMBOL","Tested","Validated","Method","comp")]
b=a[a$Tested&a$SYMBOL%in%comg,]
valTab=aggregate(b[,c("Tested","Validated")],by=list(Method=b$Method,Comp=b$comp),sum)
valTab$Ratio=round(valTab$Validated/valTab$Tested*100,0)
print(valTab)
```

```
##      Method        Comp Tested Validated Ratio
## 1 DriverMap    GAEffect     49        43    88
## 2       HTA    GAEffect     35        34    97
## 3    RNASeq    GAEffect     31        29    94
## 4 DriverMap LaborEffect     24        23    96
## 5    RNASeq LaborEffect      7         4    57
```

The last column in the table above are the percentage validation rates for each platform and each comparison.

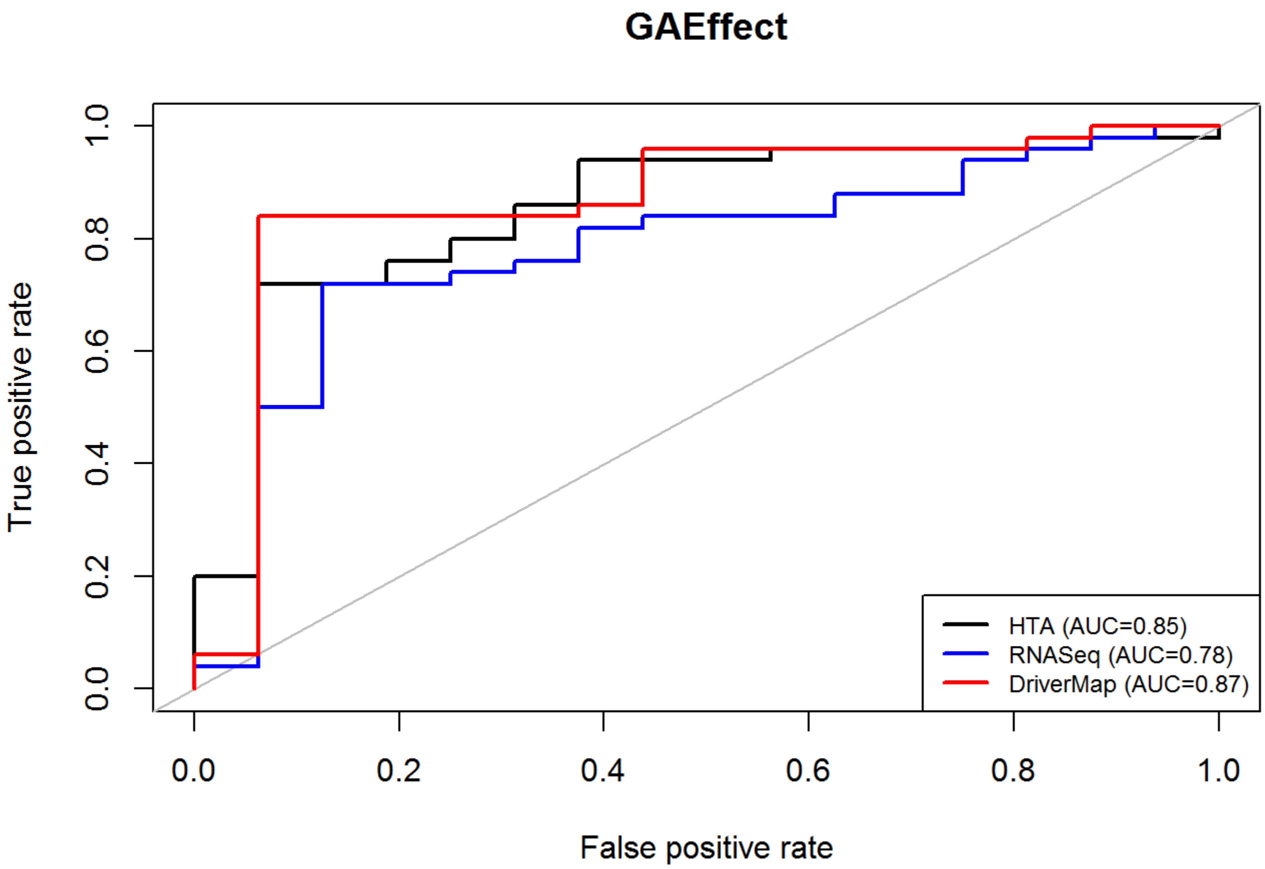
ROC curves analysis

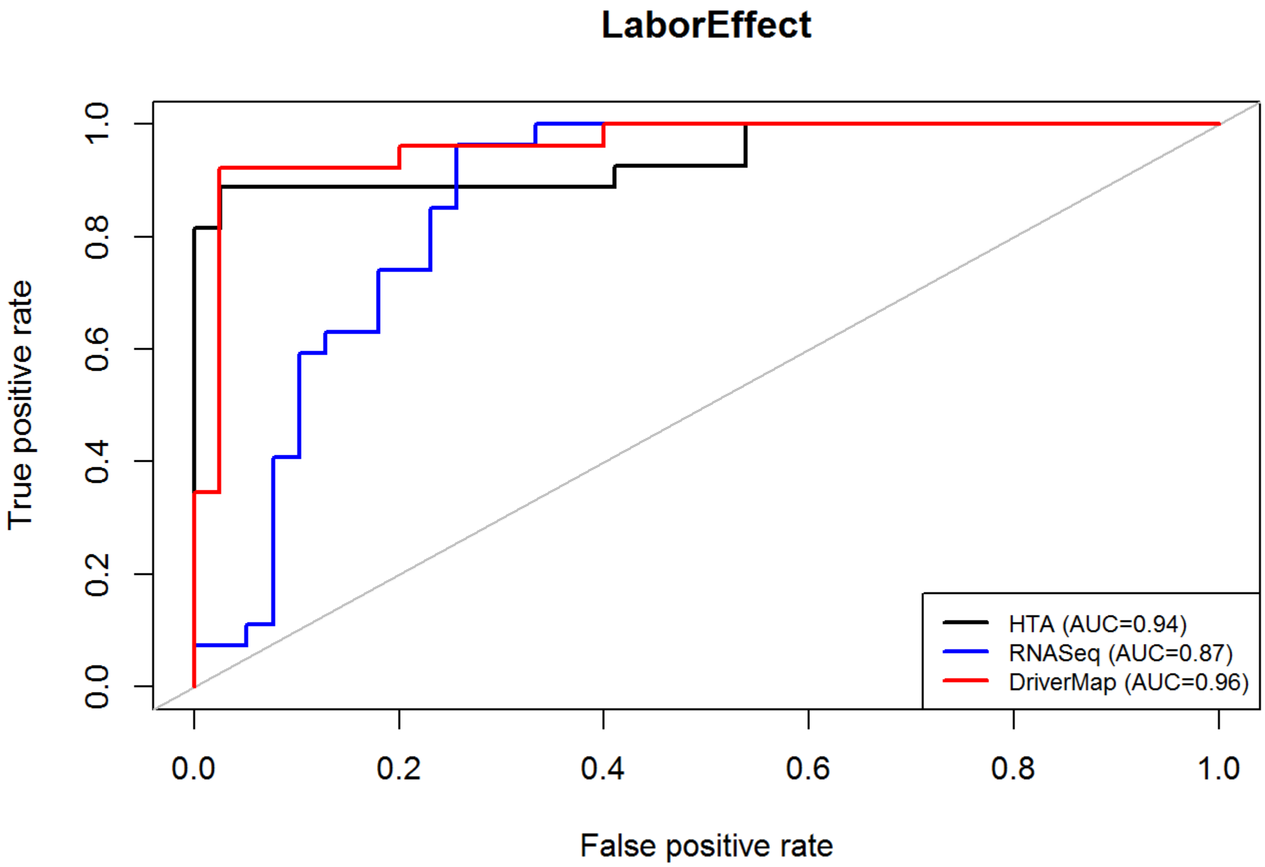
The calculation of gene validation rates required to define which gene is positive with a given platform. However, even though the p-values obtained with HTA microarrays for changes with labor may be meaningful, no gene was significant after multiple testing correction. The ROC curve analysis avoids the need for choosing significance cut-offs. The status of each gene (TRUE of FALSE positive) for each platform and each method is first determined as above, yet now we retain data for all genes (regardless wether or not they were significant by omics platforms):

```
ddPile<-NULL
for(eff in effs){
  #gold standard
  vaT1t=PCR[[eff]];
  for(platf in names(DEPile)){
    #gold standard
    x<-DEPile[[platf]][[eff]]
    x=x[!is.na(x$SYMBOL),]
    x$adj.P.Val<-p.adjust(x$P.Value,"fdr")
    x=x[order(x$P.Value),]
    x=x[!duplicated(x$SYMBOL),]
      x$Tested=x$SYMBOL%in%vaT1t[,"ID"]
      x$Validated=x$SYMBOL%in%vaT1t[vaT1t$Sig,"ID"]&sign(x$logFC)==sign(vaT1t$logFC[match(x$SYMBOL,vaT1t$ID)])
      x$Method=platf
      x$comp=eff
      ddPile=rbind(ddPile,x[,c("SYMBOL","P.Value","adj.P.Val","logFC","Tested","Validated","Method","comp")])
  }
}
```

and then the ROC curves are created using:

```
mycols=c("black","blue","red")
names(mycols)<-platforms
for(eff in effs){
b=ddPile[ddPile$SYMBOL%in%comg&ddPile$comp==eff,]
b$Validated=as.numeric(b$Validated)
AUCs=NULL
 for(platf in platforms){
  tmp=b[b$Method==platf,]
  pred <- prediction(1-tmp$P.Value, tmp$Validated)
  perf <- performance(pred,measure="tpr", x.measure="fpr")
  AUCs=c(AUCs,round(performance(pred,"auc")@y.values[[1]],2))
  if(platf=="HTA"){
  plot(perf,lwd=2,col=mycols[platf],main=eff)
  abline(c(0,0),c(1,1),col="grey")
  }else{
  points(perf@x.values[[1]],perf@y.values[[1]],lwd=2,col=mycols[platf],type="l")
  }
 }
  legend("bottomright",lwd=c(2,2,2),col=mycols,
         legend=paste(platforms," (AUC=",AUCs,")",sep=""),cex=0.75)
}
```


GAEffect

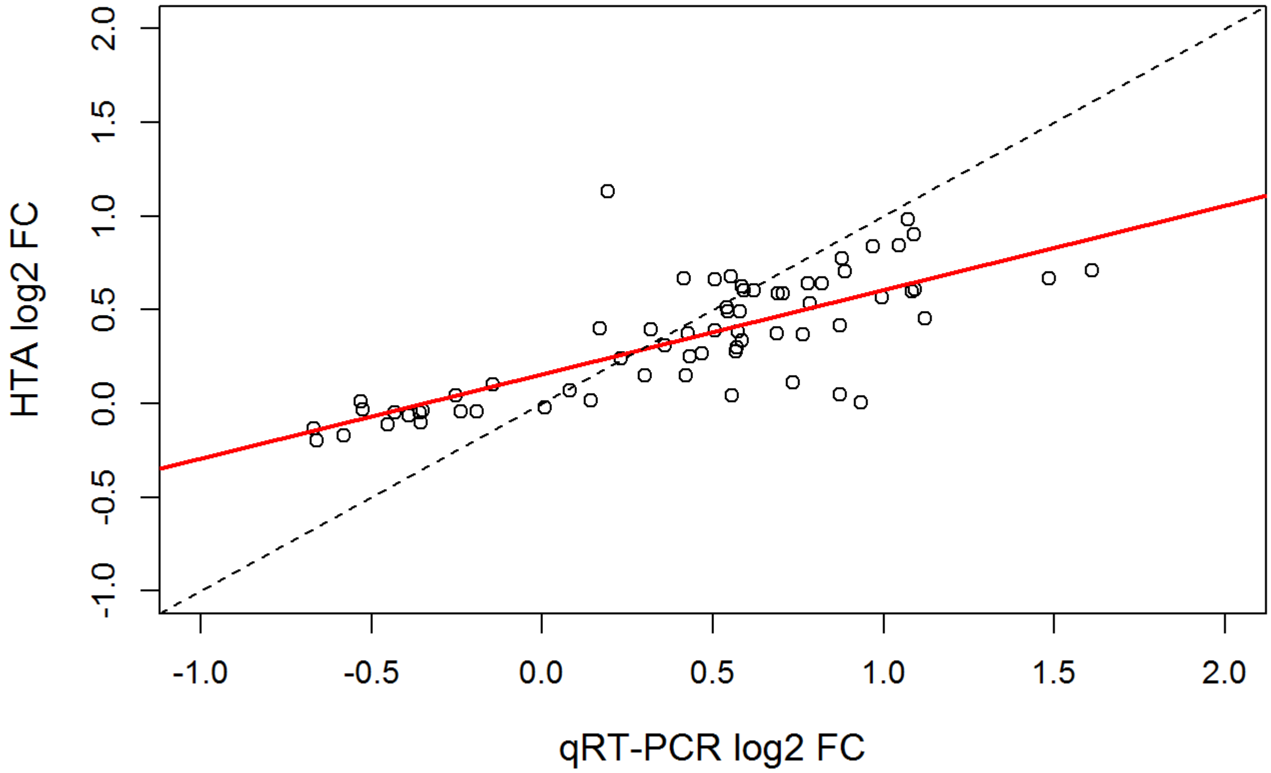

LaborEffect

Correlation analysis of fold changes between omics platfroms and qRT-PCR

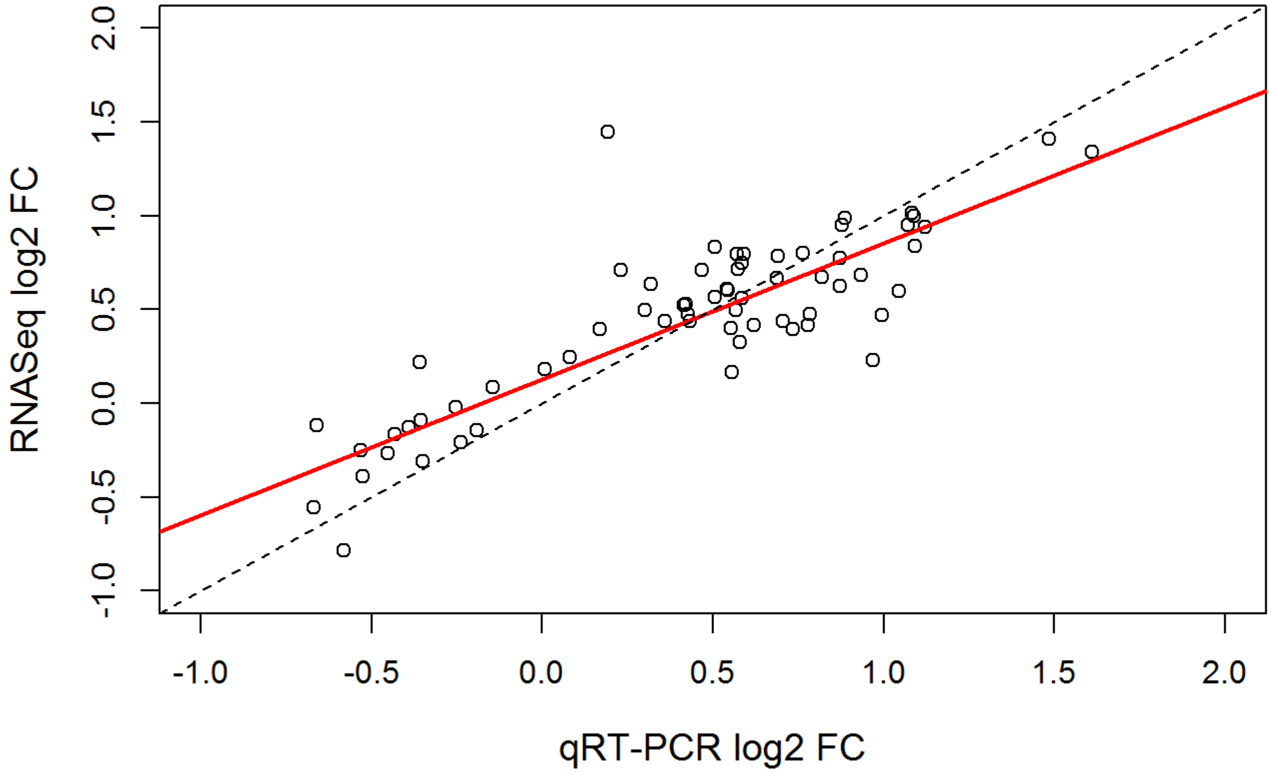
Next, for the the 66 genes profiled on all four platforms, we determine the correlation of log2 fold changes between each omics platfrom and qRT-PCR (gold standard) for both comparions (GA and Labor) (Figure 3 in (Tarca et al. 2018)).

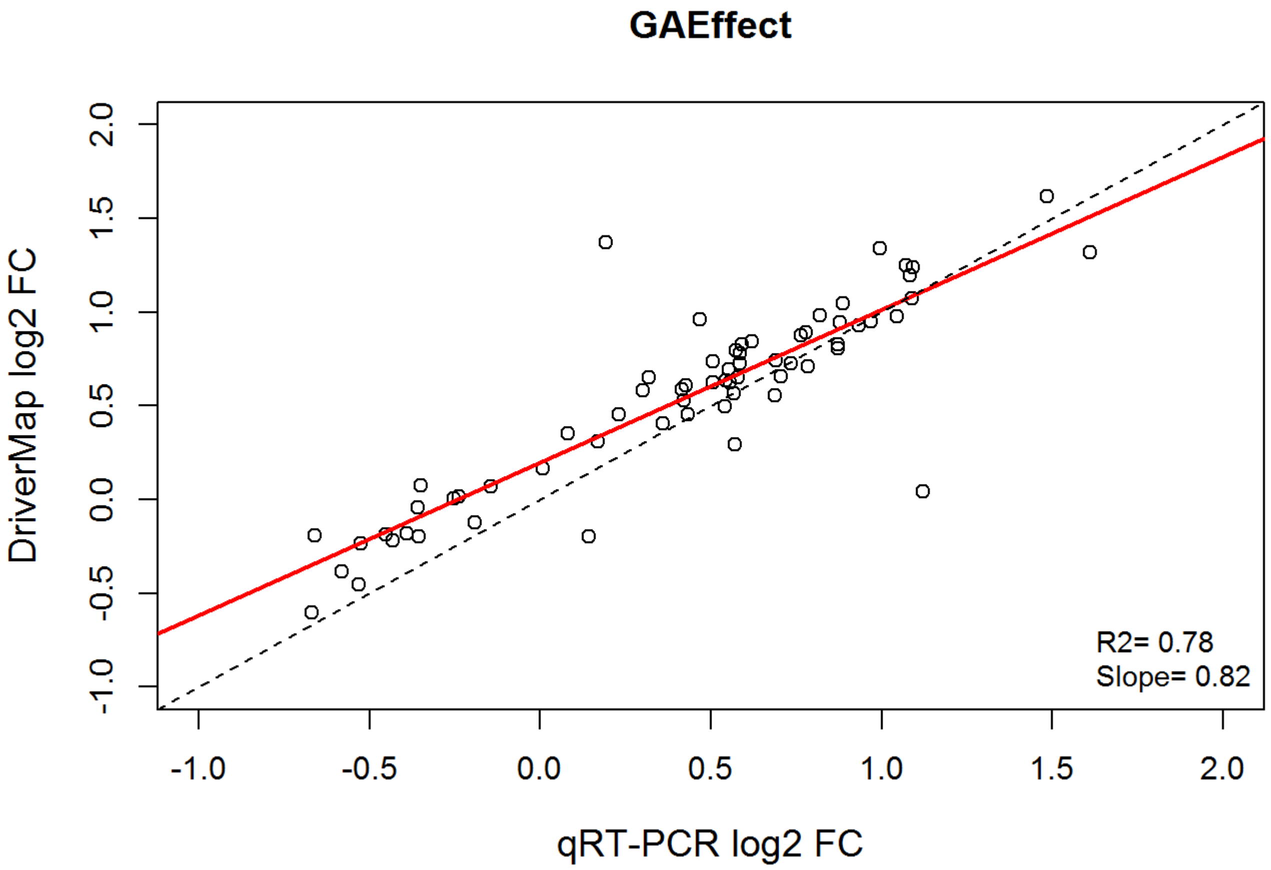
```
for(eff in c("GAEffect","LaborEffect")){
if(eff=="GAEffect"){lims=c(-1,2)}else{lims=c(-4,3)}
for(platf in platforms){
m1=DEPile[[platf]][[eff]];m2=PCR[[eff]]
tm=data.frame(x=m2[match(comg,m2$SYMBOL),"logFC"],y=m1[match(comg,m1$SYMBOL),"logFC"])
plot(y~x,tm,xlab="qRT-PCR log2 FC",ylab=paste(platf,"log2 FC"),xlim=lims,ylim=lims,cex.lab=1.2,main=eff)
mo=lm(y~x,data=as.data.frame(tm))
abline(mo$coef,col="red",lwd=2)
abline(0,1,lty=2)
}
legend("bottomright",c(paste("R2=",round(summary(mo)$r.squared,2)),paste("Slope=",round(mo$coef[2],2))),cex=0.9,bty="n")
}
```

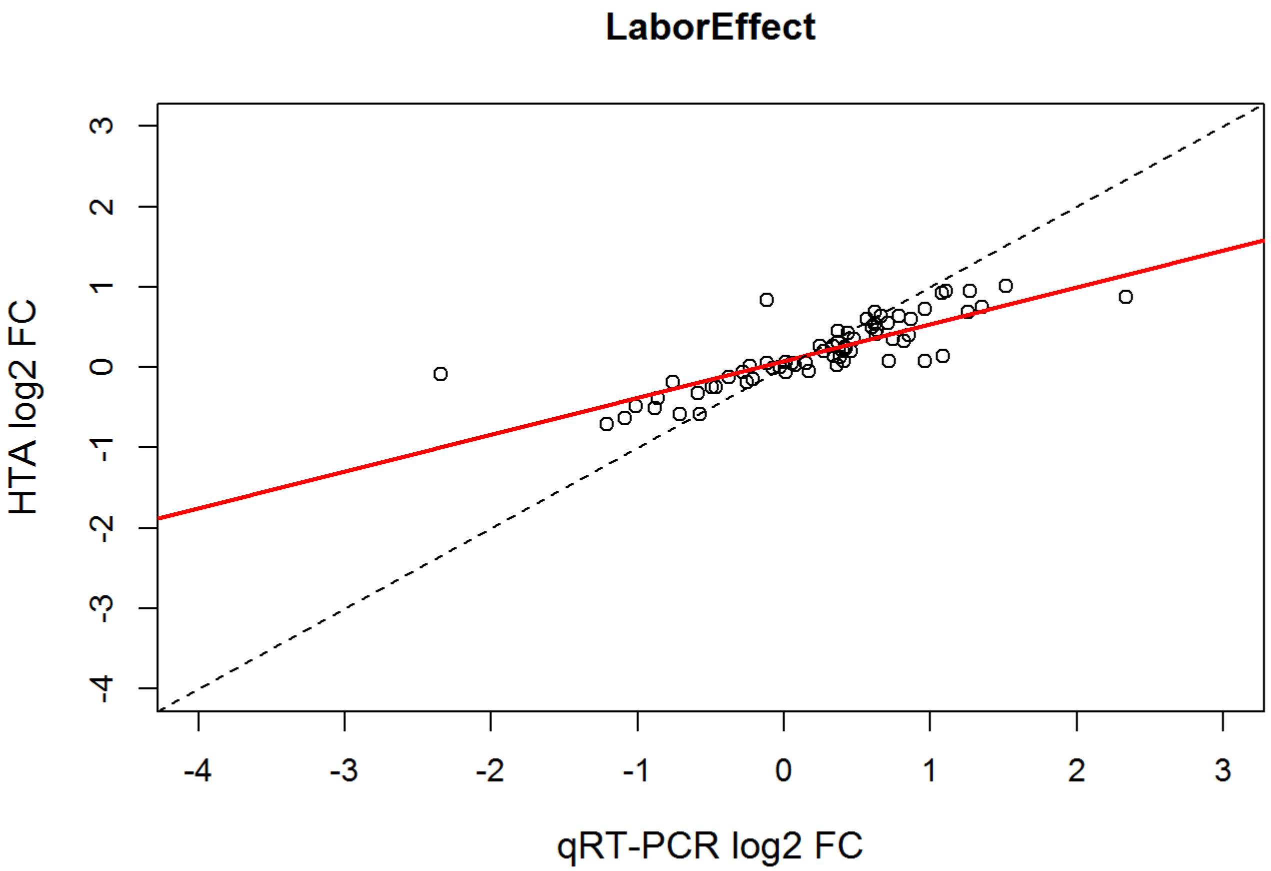
GAEffect

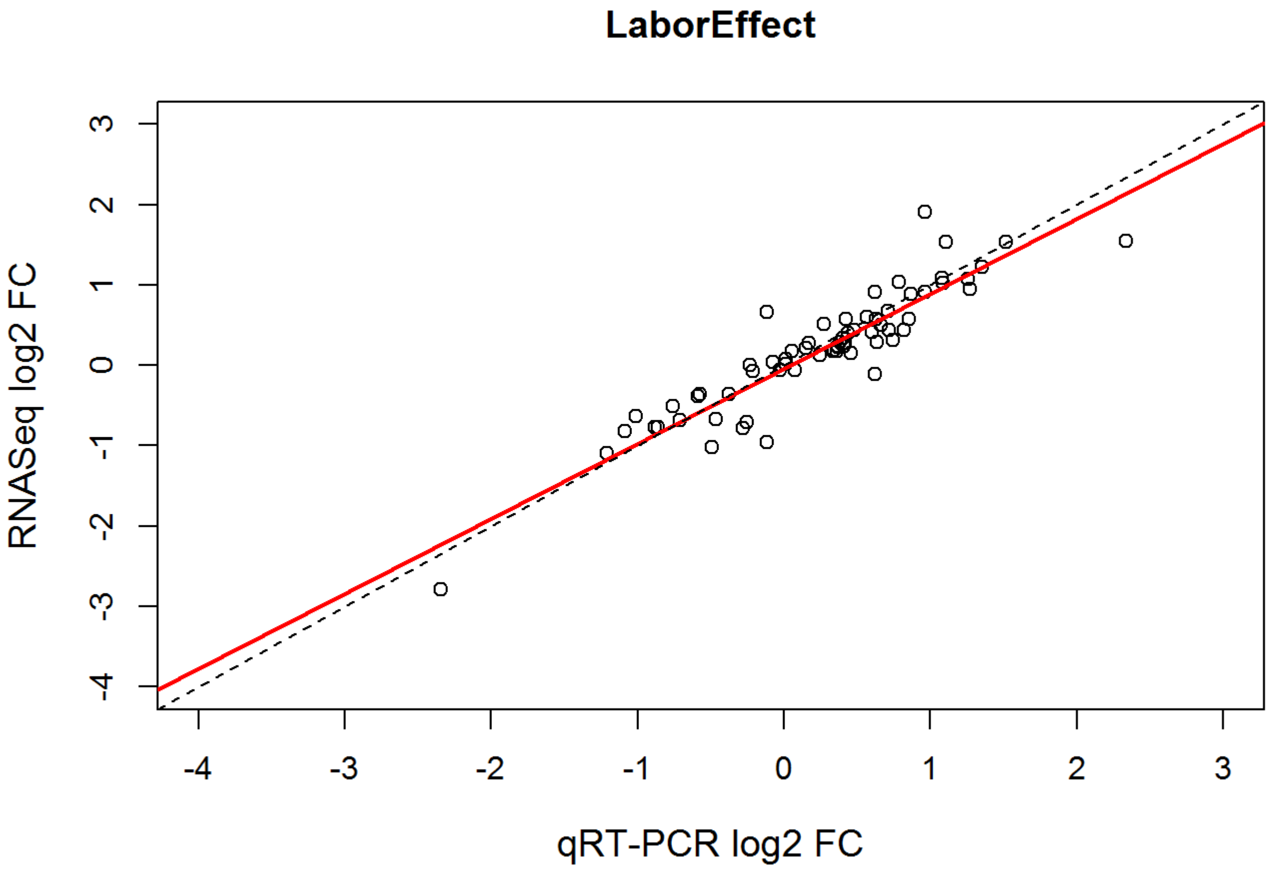


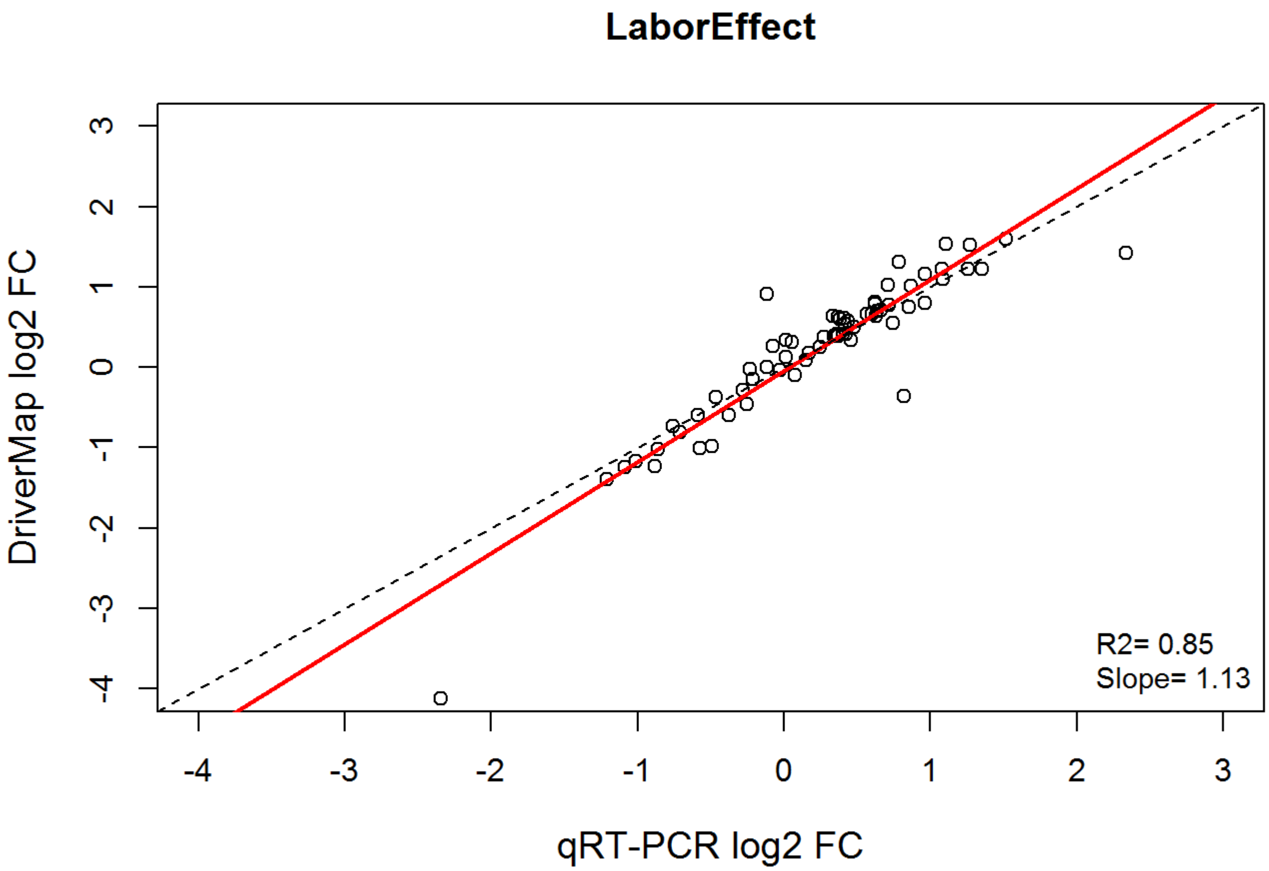
GAEffect




GAEffect
DriverMap log2 FC
qRT-PCR log2 FC
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
R2= 0.78
Slope= 0.82


LaborEffect
HTA log2 FC
qRT-PCR log2 FC
-4
-3
-2
-1
0
1
2
3


LaborEffect
RNASeq log2 FC
qRT-PCR log2 FC
-4
-3
-2
-1
0
1
2
3



Overlap of differential expression among platforms with findings from other studies

To determine the overlap of differentially expressed genes with gestational age with reports from previous studies, we first retain genes present on all three omics platfroms and extract statistics for his comparison:

```
comg=table(c(unique(rbind(HTA[[1]],HTA[[2]])$SYMBOL),unique(rbind(RNASeq[[1]],RNASeq[[2]])$SYMBO
L),rownames(Cr)))
comg=names(comg[comg==3])

ddH=ddH[ddH$comp=="GAEffect"&ddH$SYMBOL%in%comg,]
ddR=ddR[ddR$comp=="GAEffect"&ddR$SYMBOL%in%comg,]
ddC=ddC[ddC$comp=="GAEffect"&ddC$SYMBOL%in%comg,]
```

The top tables of genes changing with gestation by (Al-Garawi et al. 2016) and (Heng et al. 2016) are next filtered and duplicates are removed:

```
data(heng)
data(algarawi)

heng=heng[heng$adj.P.Val<0.05,]
heng=heng[!duplicated(heng$SYMBOL),]

algarawi$SYMBOL=algarawi$Gene.symbol
algarawi=algarawi[!duplicated(algarawi$SYMBOL),]
algarawi=algarawi[algarawi$adj.P.Val<0.05,]
```

The erichment analysis preseneted in Table 1 of (Tarca et al. 2018) are obtained as follows:

```
mygslist=list(HengEtAl=heng$SYMBOL,AlGarawiEtAl=algarawi$SYMBOL)
des=list(ddH,ddR,ddC)
names(des)<-c("HTA","RNASeq","DriverMap")
respile=NULL

for(me in names(des)){
  mygns=des[[me]]$SYMBOL
  gns<-OR<-pv<-ns<-NULL
  for(gs in 1:length(mygslist)){
    path=intersect(mygslist[[gs]],comg)
    noMy=length(intersect(mygns,path))
    gns=c(gns,paste(intersect(mygns,path),collapse=";"))
    if(noMy>=1){
      q = noMy ; m = length(path); n = length(comg) -length(path); k = length(mygns)
      pv=c(pv,phyper(q = noMy - 1, m = length(path), n = length(comg) - length(path), k = length
(mygns), lower.tail = FALSE))
      OR=c(OR,fisher.test(matrix(c(q,k-q,m-q,n-k+q),2,2))$est)
      ns=c(ns,noMy)
    }else{pv=c(pv,NA);OR=c(OR,NA);ns=c(ns,0)}
  }
  res=data.frame(Method=me,ID=names(mygslist),N=ns,OR=OR)
  res$P=pv;
  respile=rbind(respile,res)
}
print(respile)
```

```
##      Method           ID   N       OR            P
## 1       HTA     HengEtAl  73 1.719833 7.504718e-05
## 2       HTA AlGarawiEtAl 142 2.097606 5.835060e-11
## 3    RNASeq     HengEtAl  97 1.419795 1.868809e-03
## 4    RNASeq AlGarawiEtAl 198 1.773165 7.198327e-10
## 5 DriverMap     HengEtAl 335 1.362412 2.676037e-06
## 6 DriverMap AlGarawiEtAl 711 1.843533 3.542383e-31
```

In the table above, p-values and odds-ratios quantify the extent of the overlap between genes changing with gestation in this study and previous reports.

Analysis of gene set signature expression

In this last section, we will present analysis of gene set expression, where gene sets are defined as those specific to different cell types using single cell experiments (Tsang et al. 2017). In these analyses the expression over a given gene set will be averaged within a given sample and then associations with gestational age and labor will be tested. In these analyses we will use gene expression matrices in which rows correspond to an unique gene symbol as described above, and consider only genes present on all three platforms.

```
comg=c(rownames(Hr),rownames(Rr),rownames(Cr))
comg=names(table(comg)[table(comg)==3])

data(SCGeneSets)
SCGeneSets=SCGeneSets[SCGeneSets$Symbol%in%comg,]
head(SCGeneSets)
```

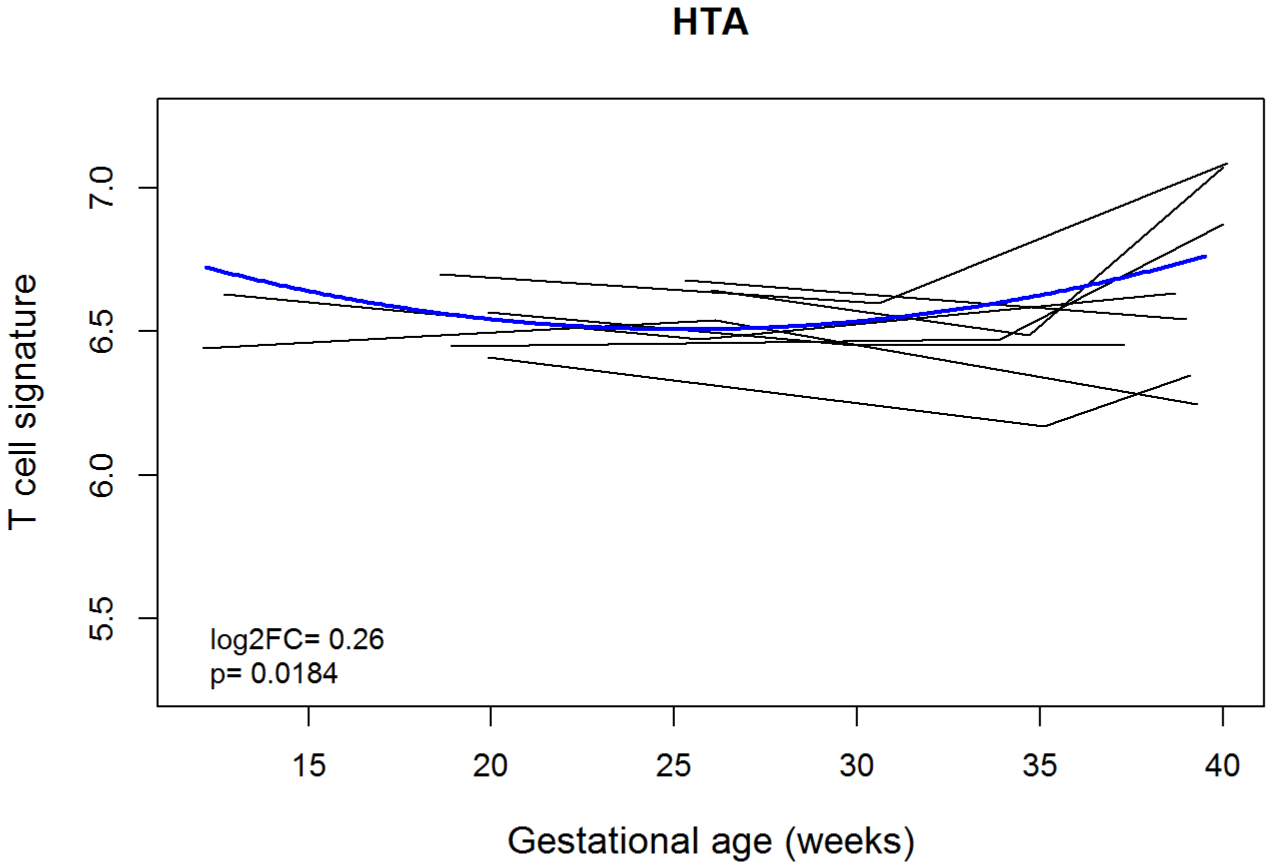
```
##      Symbol          Type
## 3    SLC7A8 Decidual cell
## 5    PRUNE2 Decidual cell
## 6     VEGFA Decidual cell
## 11   SPOCK1 Decidual cell
## 12   APCDD1 Decidual cell
## 13 PDZK1IP1 Decidual cell
```

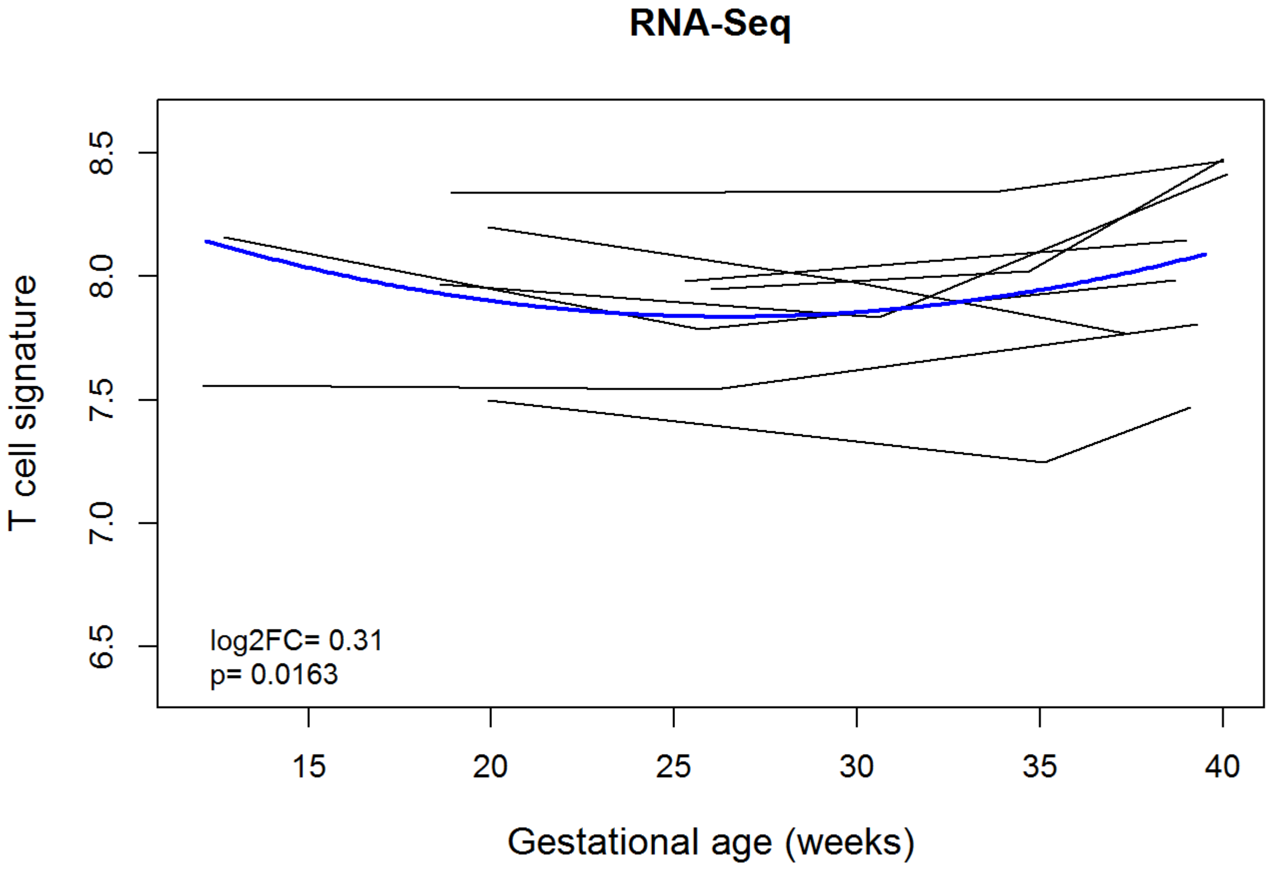
```
table(SCGeneSets$Type)
```

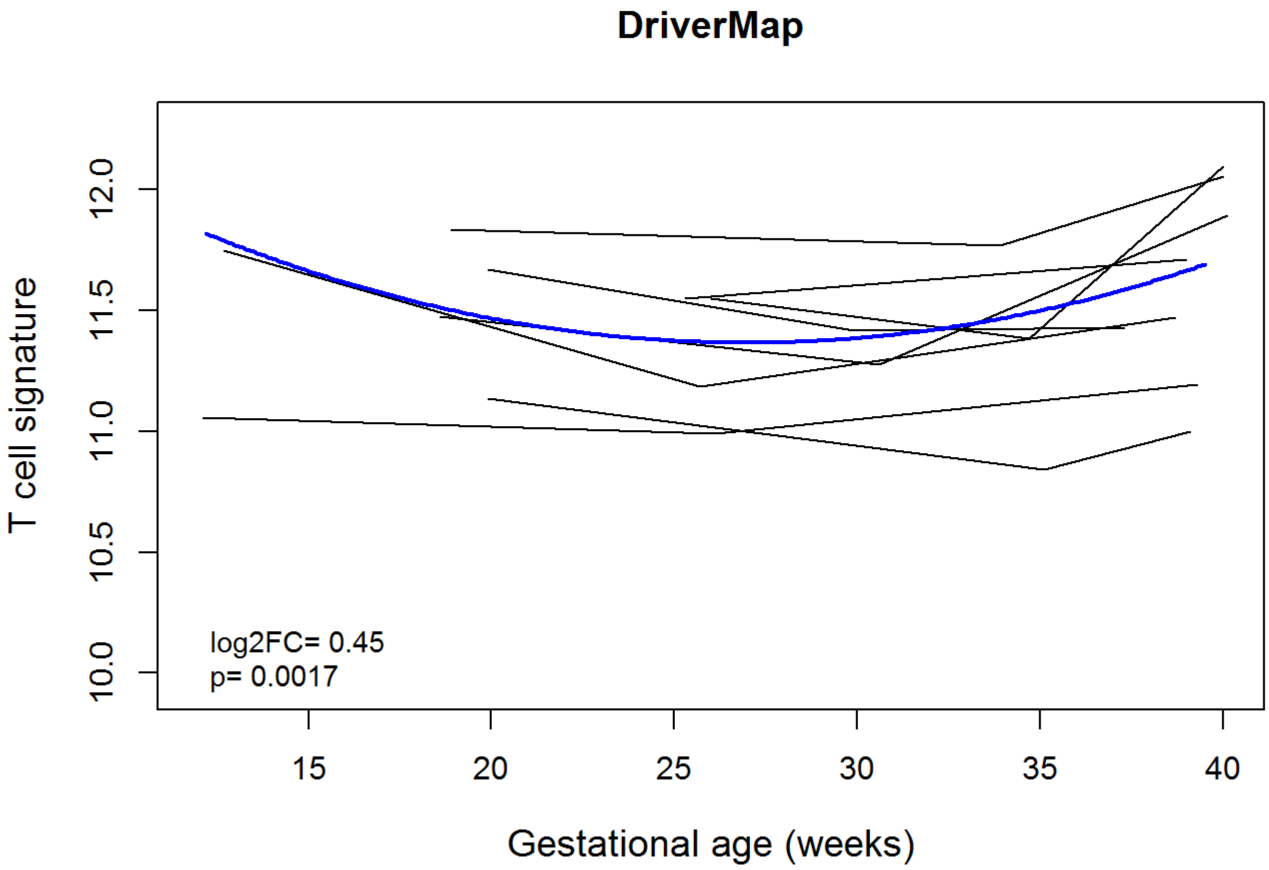
```
## 
##                      B cell             Cytotrophoblast 
##                          12                           2 
##               Decidual cell              Dendritic cell 
##                          11                           1 
##            Endothelial cell                 Erythrocyte 
##                          11                           3 
##    Extravillous trophoblast               Hofbauer cell 
##                           9                           8 
##                    Monocyte                Stromal cell 
##                          12                          10 
##         Syncytiotrophoblast                      T cell 
##                          14                          17 
## Vascular smooth muscle cell 
##                           7 
```

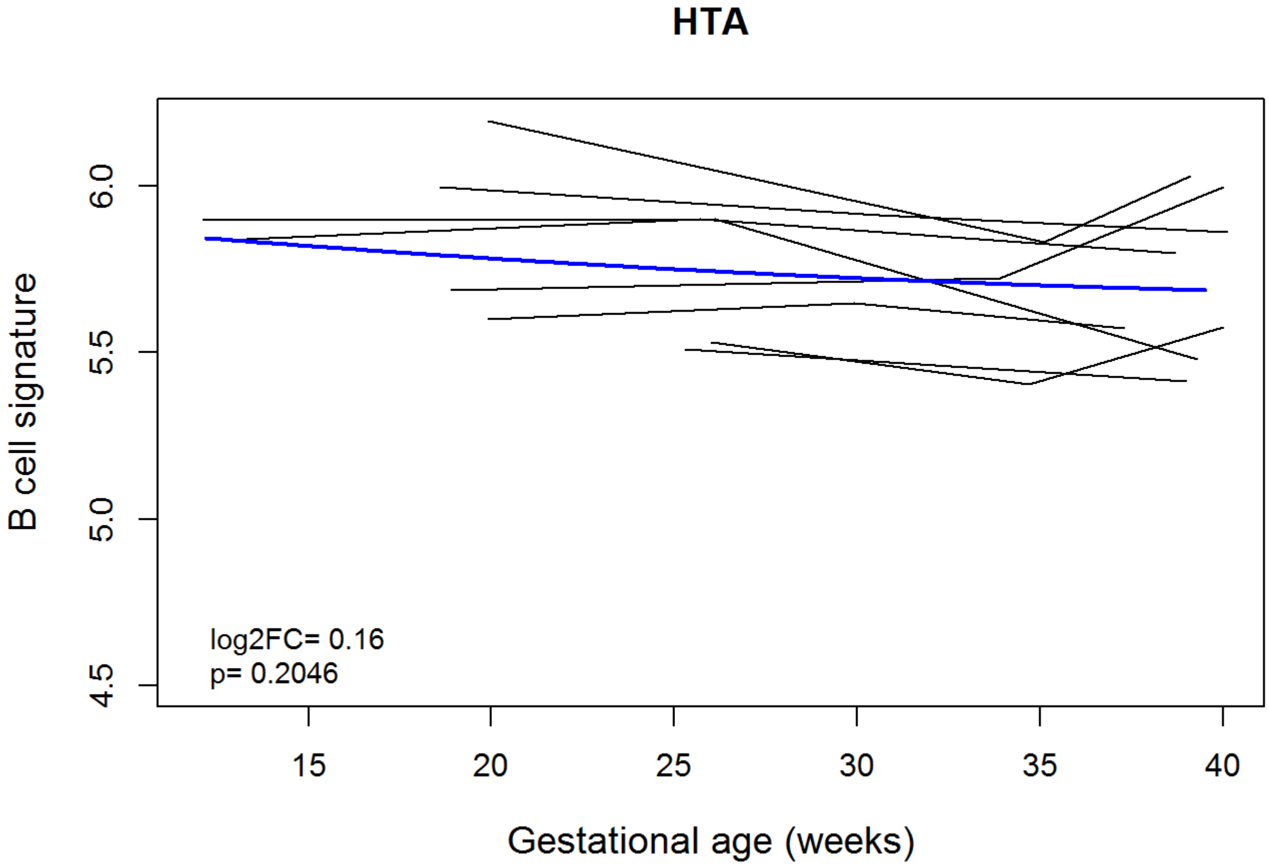
Next, we use linear mixed-effect models with splines to fit gene set expression summaries as a function of gestational age and plot these against the raw data:

```
nms=c("HTA","RNA-Seq","DriverMap")
ys=c("H","R","C")
ano=anoALL
ano=ano[ano$SampleID!="Sample_26",] #remove contaminated sample
z=bs(ano$GA,degree=2,knots=1,intercept=FALSE)
colnames(z)<-paste("t",colnames(z),sep="")
ano=cbind(ano,z)

for( sig in c("T cell","B cell")){
  #simple mean
  ano$H=apply(Hr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)
  ano$R=apply(Rr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)
  ano$C=apply(Cr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)

  for(meths in 1:3){
   ano$Y=ano[,ys[meths]]
  plot(0,0,ylab=paste(sig,"signature"),xlab="Gestational age (weeks)",ylim=c(min(ano$Y)-0.9,max(ano$Y)),xlim=c(12,40),main=nms[meths], cex.lab=1.2)
  for(i in unique(ano$IndividualID)){
    ano2=ano[ano$IndividualID==i,]
   points(Y~GA,ano2,type="l")
  }

mod1=lmer(Y~0+t1+t2+t3+(1|IndividualID),data = ano,control=lmerControl(optimizer="bobyqa"),REML=FALSE)
mod2=lmer(Y~1+(1|IndividualID),data = ano,control=lmerControl(optimizer="bobyqa"),REML=FALSE)

p=anova(mod1,mod2)$"Pr(>Chisq)"[2]
pred=expand.grid(GA=seq(12.2,39.5,by=0.1),IndividualID="newpoint")
tmp=predict(z,pred$GA)
colnames(tmp)<-paste("t",colnames(tmp),sep="")
pred=cbind(pred,tmp)

pred$Y=predict(mod1,pred,allow.new.levels=TRUE)
points(Y~GA,pred,type="l",lwd=2,col="blue")

FC=(max(pred$Y)-min(pred$Y))
legend("bottomleft",c(paste("log2FC=",round(FC,2)),paste("p=",round(p,4))),cex=0.9,bty="n")
}

}
```


HTA
T cell signature
7.0
6.5
6.0
5.5
15
20
25
30
35
40
Gestational age (weeks)
log2FC= 0.26
p= 0.0184

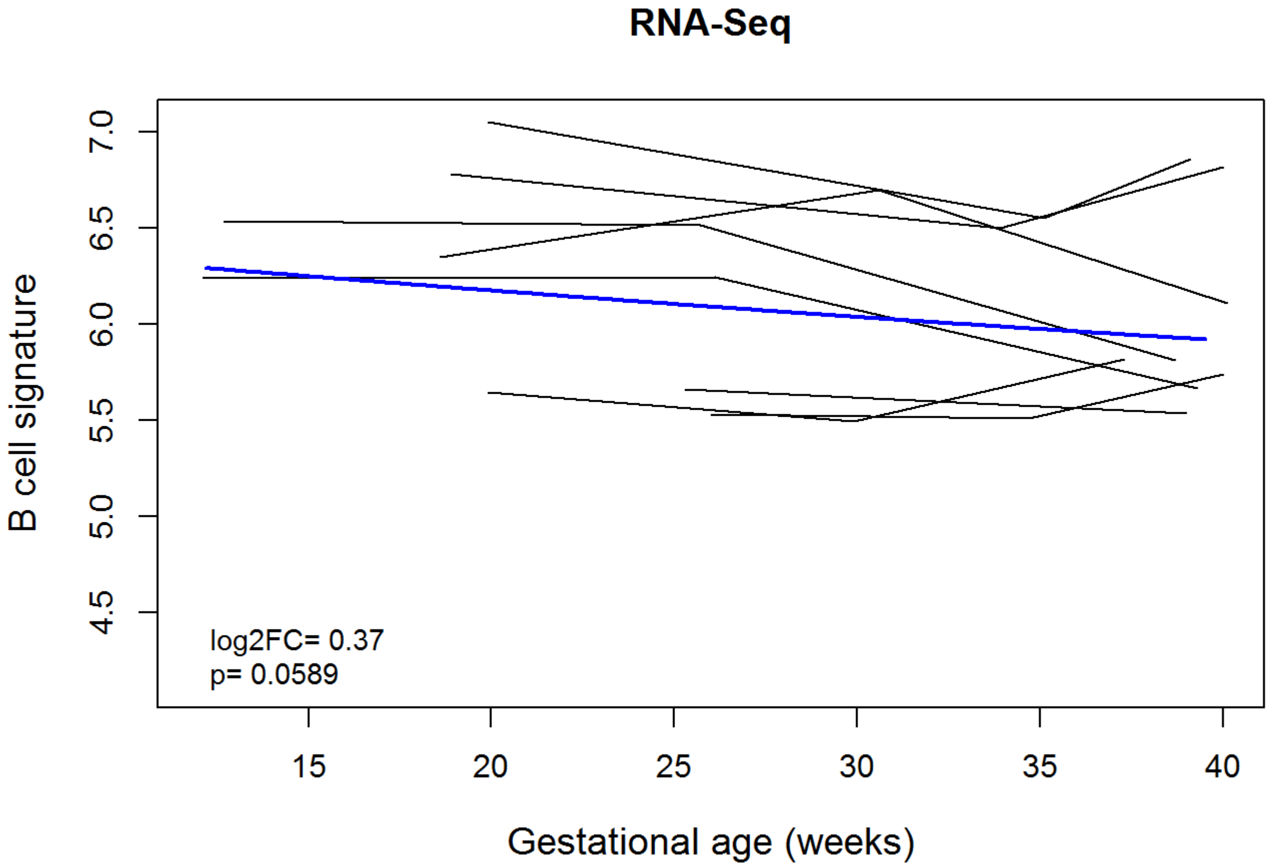

RNA-Seq
T cell signature
8.5
8.0
7.5
7.0
6.5
15
20
25
30
35
40
Gestational age (weeks)
log2FC= 0.31
p= 0.0163


DriverMap
T cell signature
10.0
10.5
11.0
11.5
12.0
15
20
25
30
35
40
Gestational age (weeks)
log2FC= 0.45
p= 0.0017



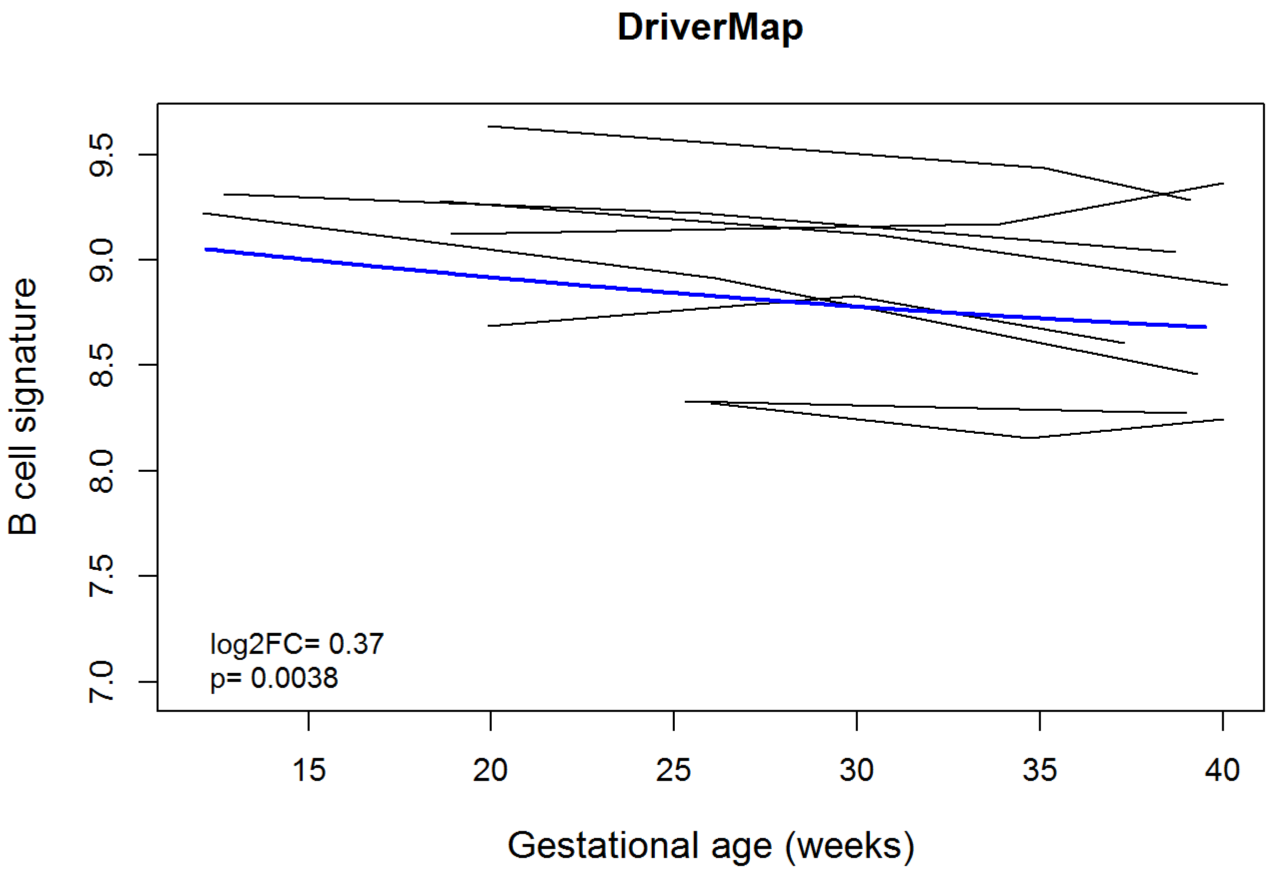
HTA

B cell signature
4.5
5.0
5.5
6.0
15
20
25
30
35
40
Gestational age (weeks)
log2FC= 0.16
p= 0.2046



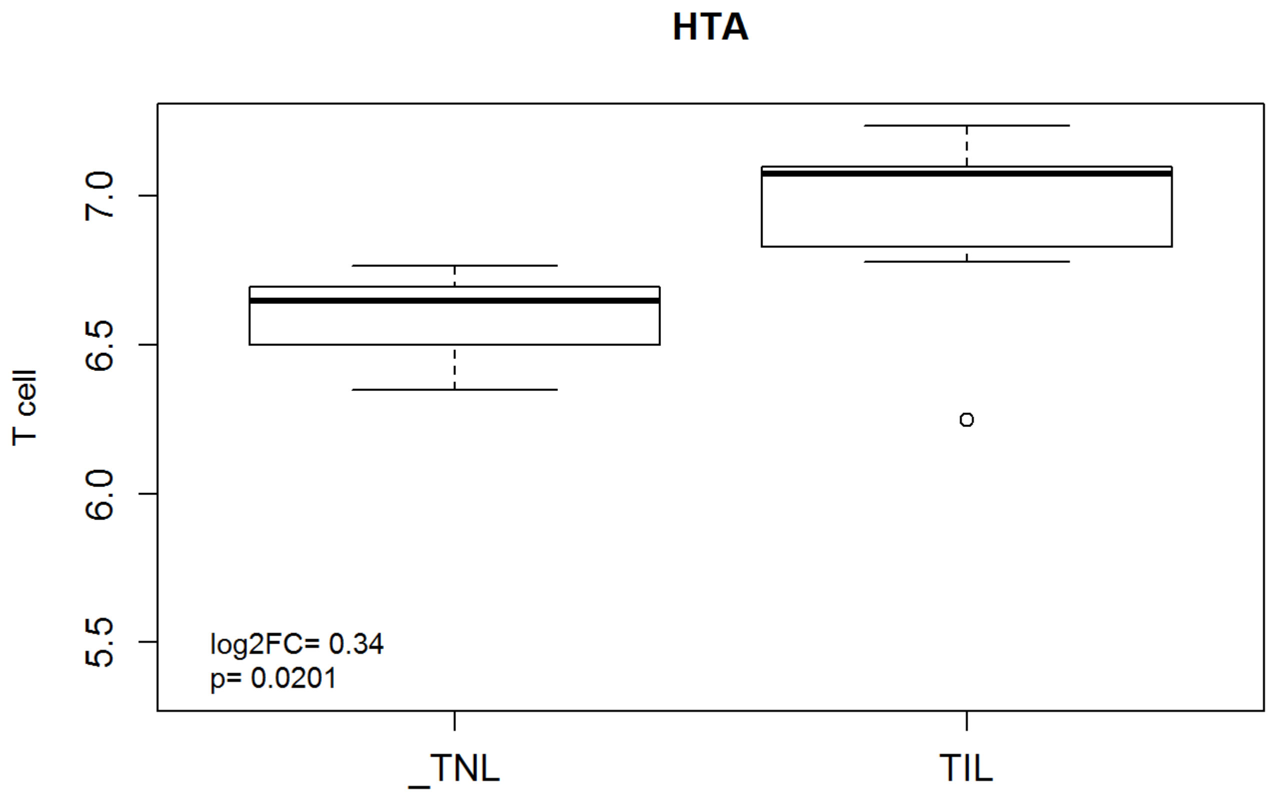
RNA-Seq

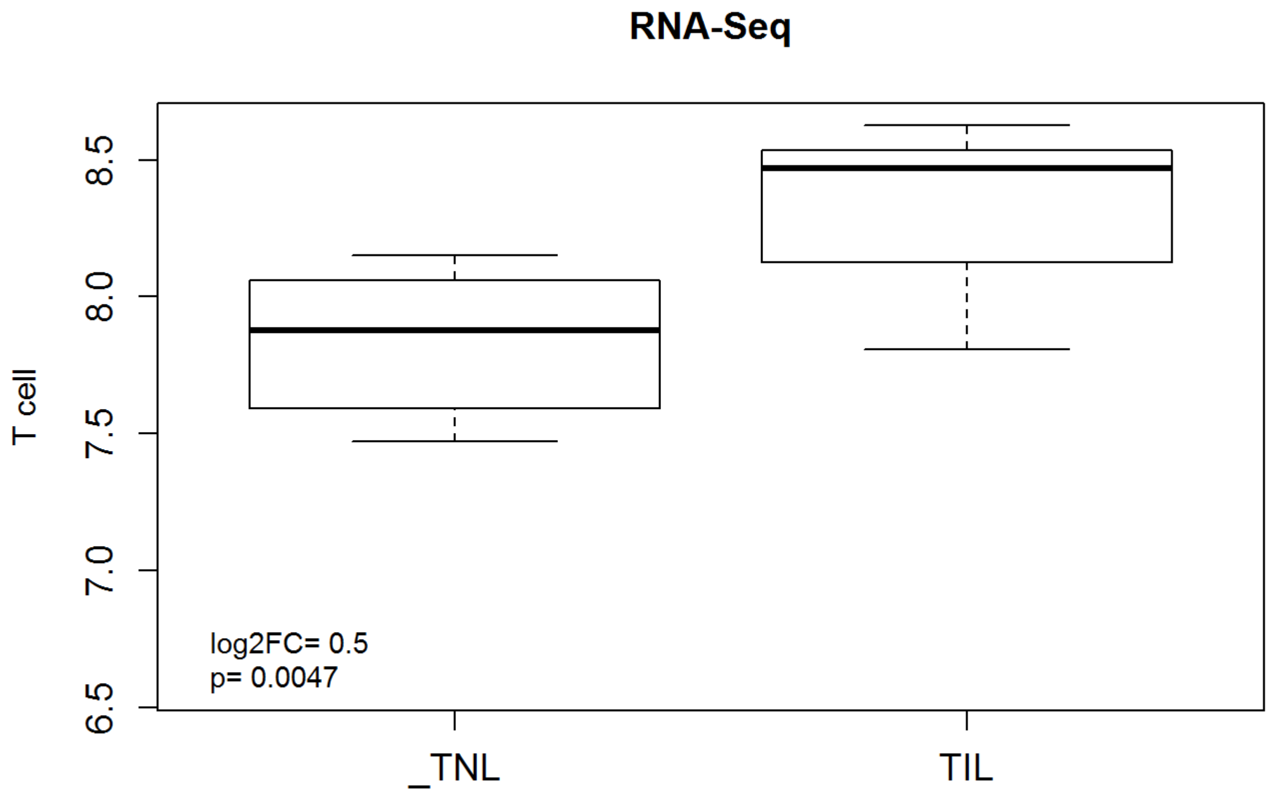
B cell signature
4.5
5.0
5.5
6.0
6.5
7.0
15
20
25
30
35
40
Gestational age (weeks)
log2FC= 0.37
p= 0.0589

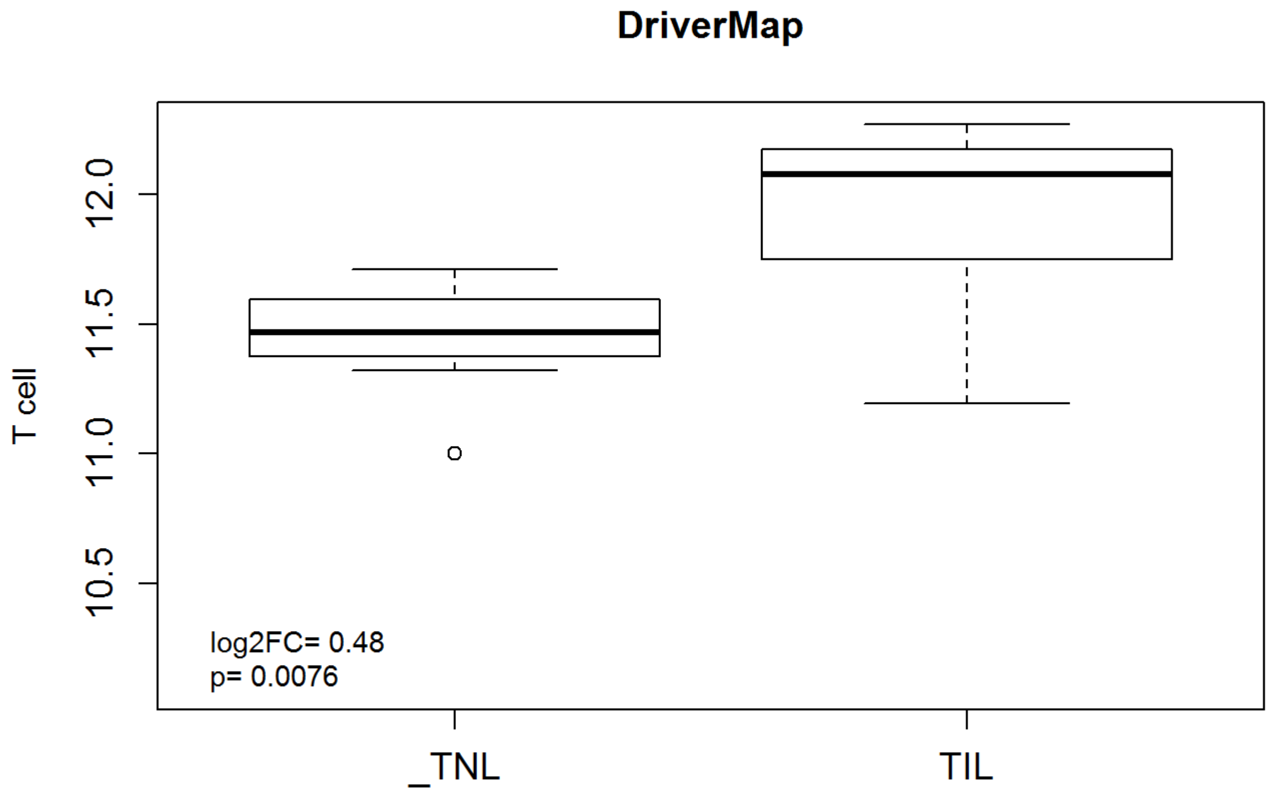


Next we compare the gene set expression for T cell between women in labor (TIL) and those not in Labor (TNL):

```
#Labor effect on signature summary
ano=anoLabor
ano$Group0=factor(ifelse(ano$Group=="TIL","TIL","_TNL"))
sig="T cell"
  ano$H=apply(Hr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)
  ano$R=apply(Rr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)
  ano$C=apply(Cr[SCGeneSets[SCGeneSets$Type==sig,"Symbol"],ano$SampleID],2,mean)

  for(meths in 1:3){
  ano$Y=ano[,ys[meths]]
  lgFC=mean(ano$Y[ano$Group=="TIL"])-mean(ano$Y[ano$Group=="TNL"])
  pv=t.test(ano$Y[ano$Group=="TIL"],ano$Y[ano$Group=="TNL"])$p.value
  boxplot(Y~Group0,ano,ylab=sig,ylim=c(min(ano$Y)-0.9,max(ano$Y)),main=nms[meths],cex.axis=1.2)
  legend("bottomleft",c(paste("log2FC=",round(lgFC,2)),paste("p=",round(pv,4))),cex=0.9,bty="n")
  }
```

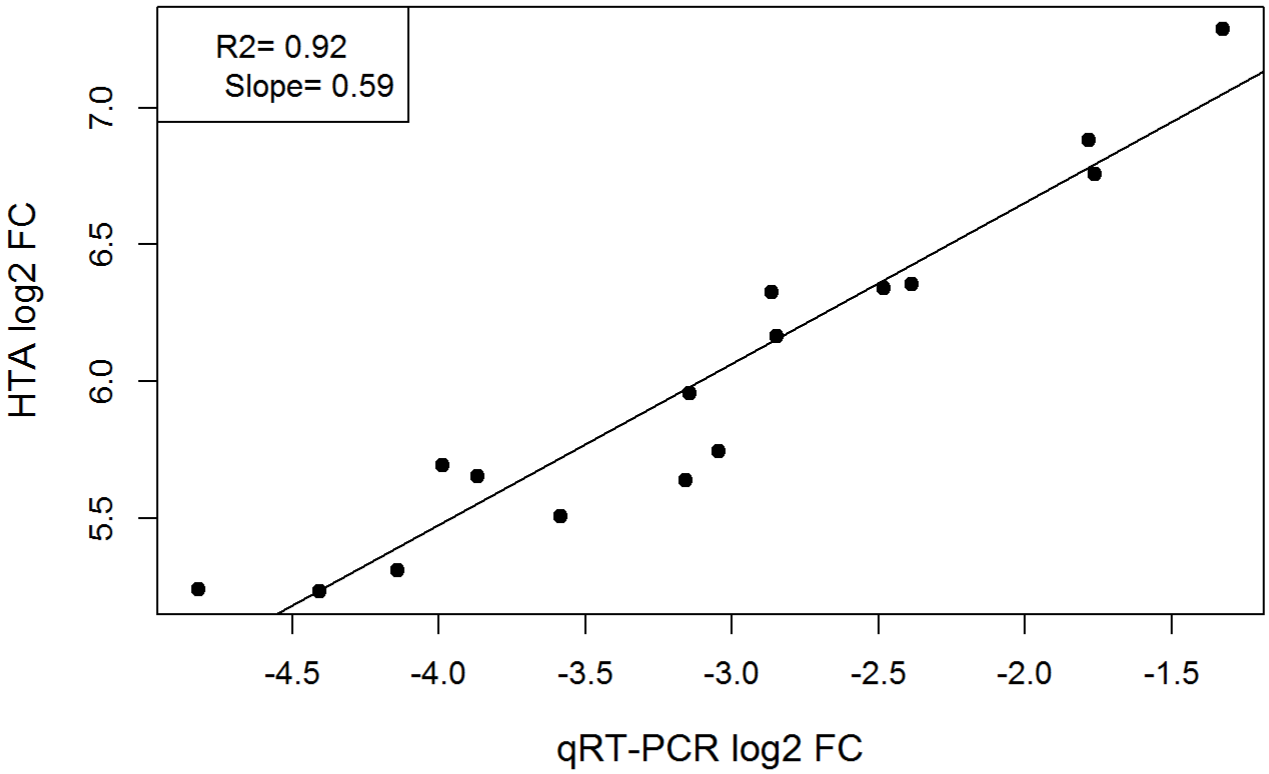

HTA
T cell
7.0
6.5
6.0
5.5
log2FC= 0.34
p= 0.0201
_TNL
TIL

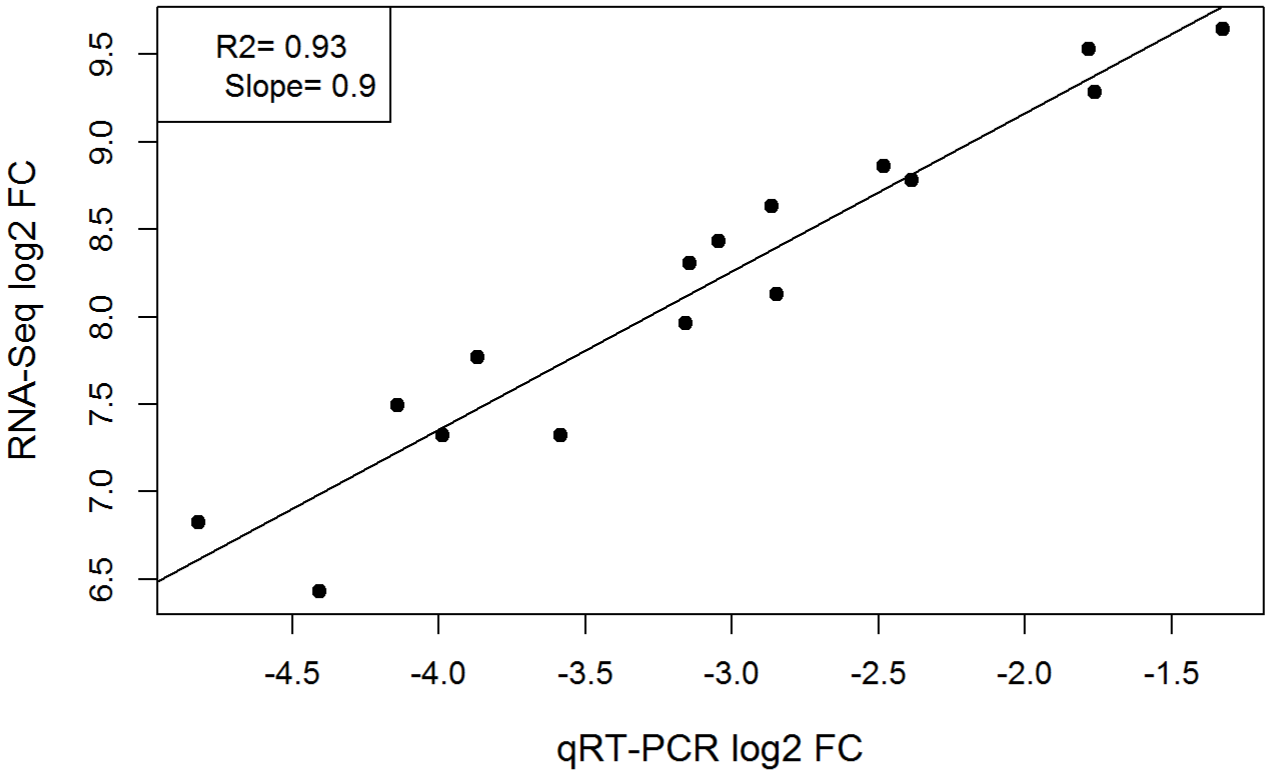

RNA-Seq
T cell
8.5
8.0
7.5
7.0
6.5
log2FC= 0.5
p= 0.0047
_TNL
TIL

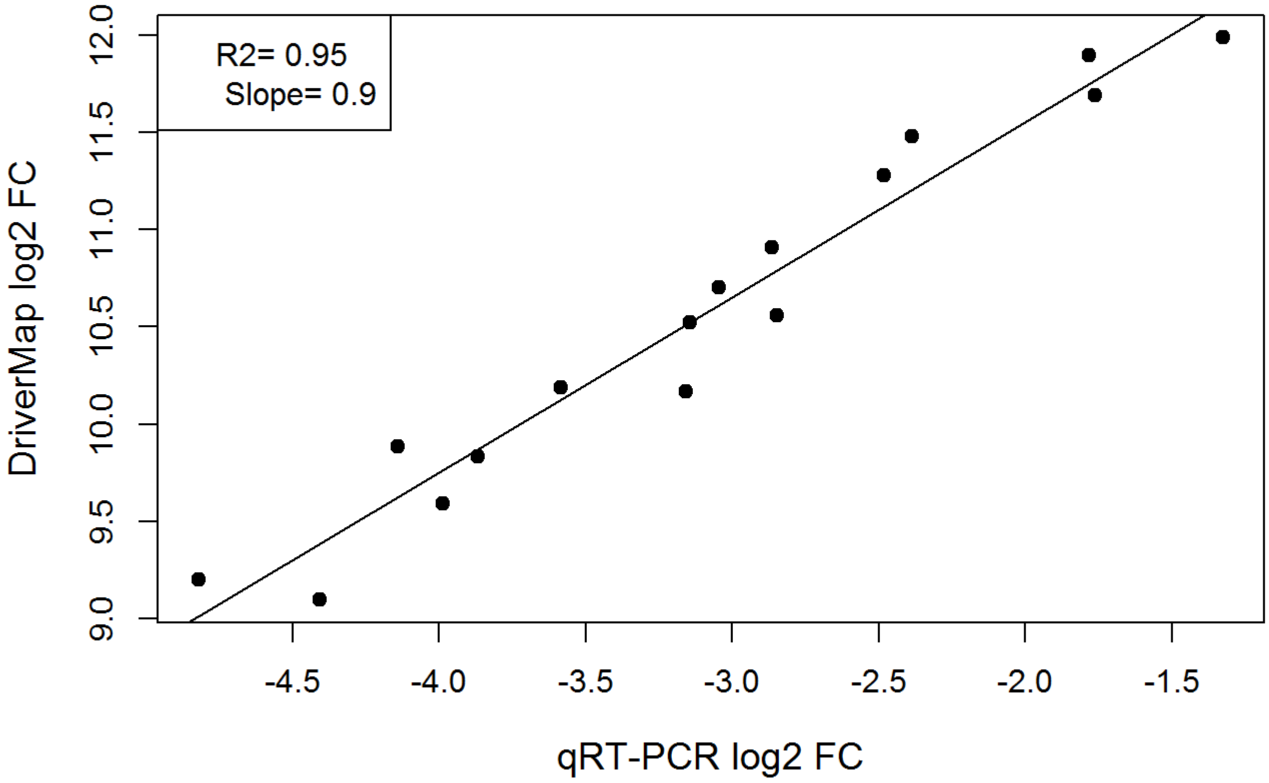


Since some of the genes profiled by qRT-PCR were part of the T cell signature, we compare the correlation between the T cell signature expression between each omics platform and qRT-PCR:

```
ano=anoLabor

data(SCGeneSets)
SCGeneSets=SCGeneSets[SCGeneSets$Symbol%in%rownames(Pr),]

Ps=apply(Pr[rownames(Pr)%in%SCGeneSets$Symbol,ano$SampleID],2,mean) #PCR
Hs=apply(Hr[rownames(Hr)%in%SCGeneSets$Symbol,ano$SampleID],2,mean) #HTA
Rs=apply(Rr[rownames(Rr)%in%SCGeneSets$Symbol,ano$SampleID],2,mean) #RNAseq
Cs=apply(Cr[rownames(Cr)%in%SCGeneSets$Symbol,ano$SampleID],2,mean) #DriverMap
ano$Ps=Ps

ys=list(Hs,Rs,Cs)
names(ys)<-c("HTA","RNA-Seq","DriverMap")

x=Ps
for( k in 1:length(ys)){
y=ys[[k]]
m=lm(y~x)
plot(x,y,xlab="qRT-PCR log2 FC",pch=19,ylab=paste(names(ys)[k],"log2 FC"),cex.lab=1.2)
abline(m$coef)
legend("topleft",c(paste("R2=",round(summary(m)$r.squared,2)),paste(" Slope=",round(m$coef[2],2
))))
}
```

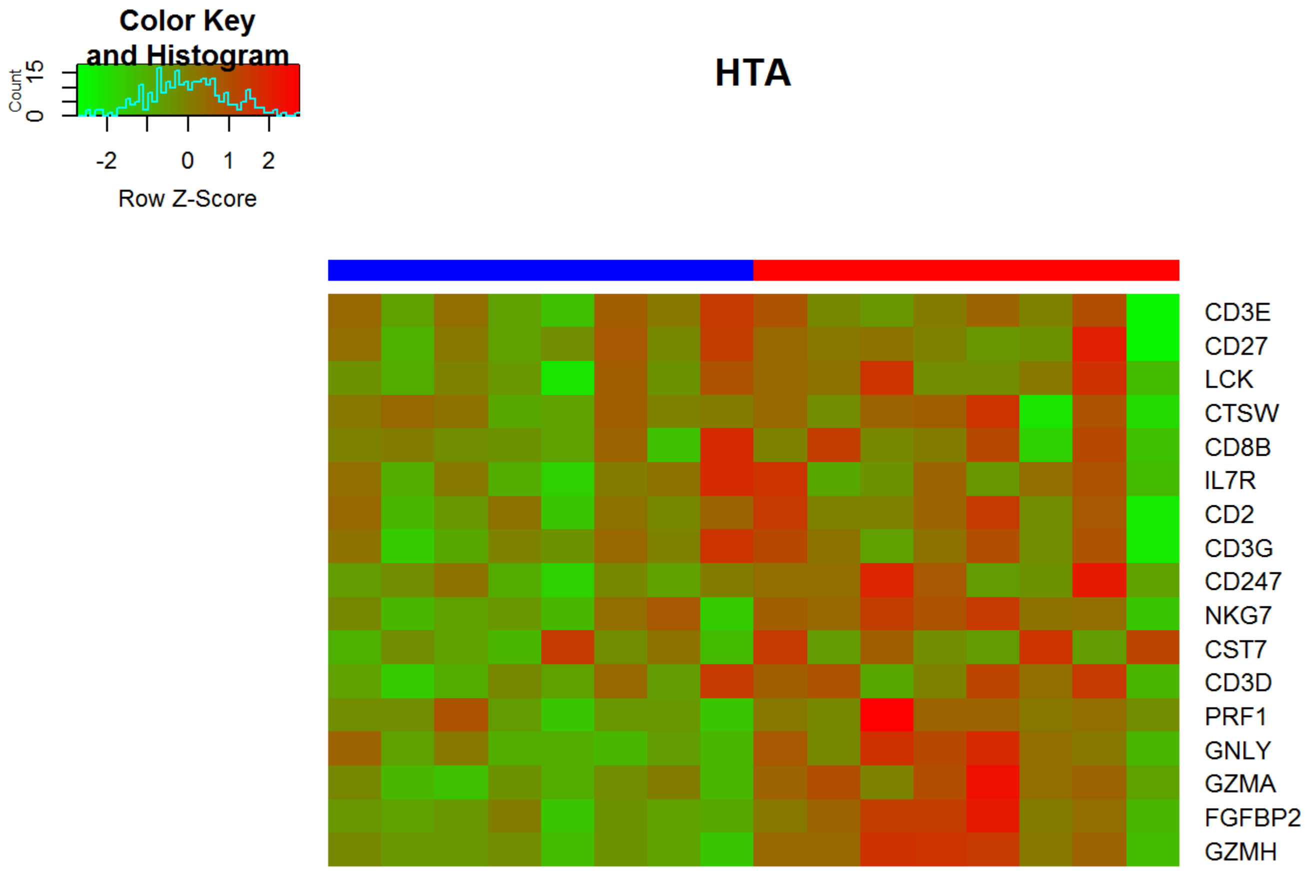

R2= 0.92
Slope= 0.59
HTA log2 FC
7.0
6.5
6.0
5.5
-4.5
-4.0
-3.5
-3.0
-2.5
-2.0
-1.5
qRT-PCR log2 FC


R2= 0.93
Slope= 0.9
RNA-Seq log2 FC
9.5
9.0
8.5
8.0
7.5
7.0
6.5
-4.5
-4.0
-3.5
-3.0
-2.5
-2.0
-1.5
qRT-PCR log2 FC

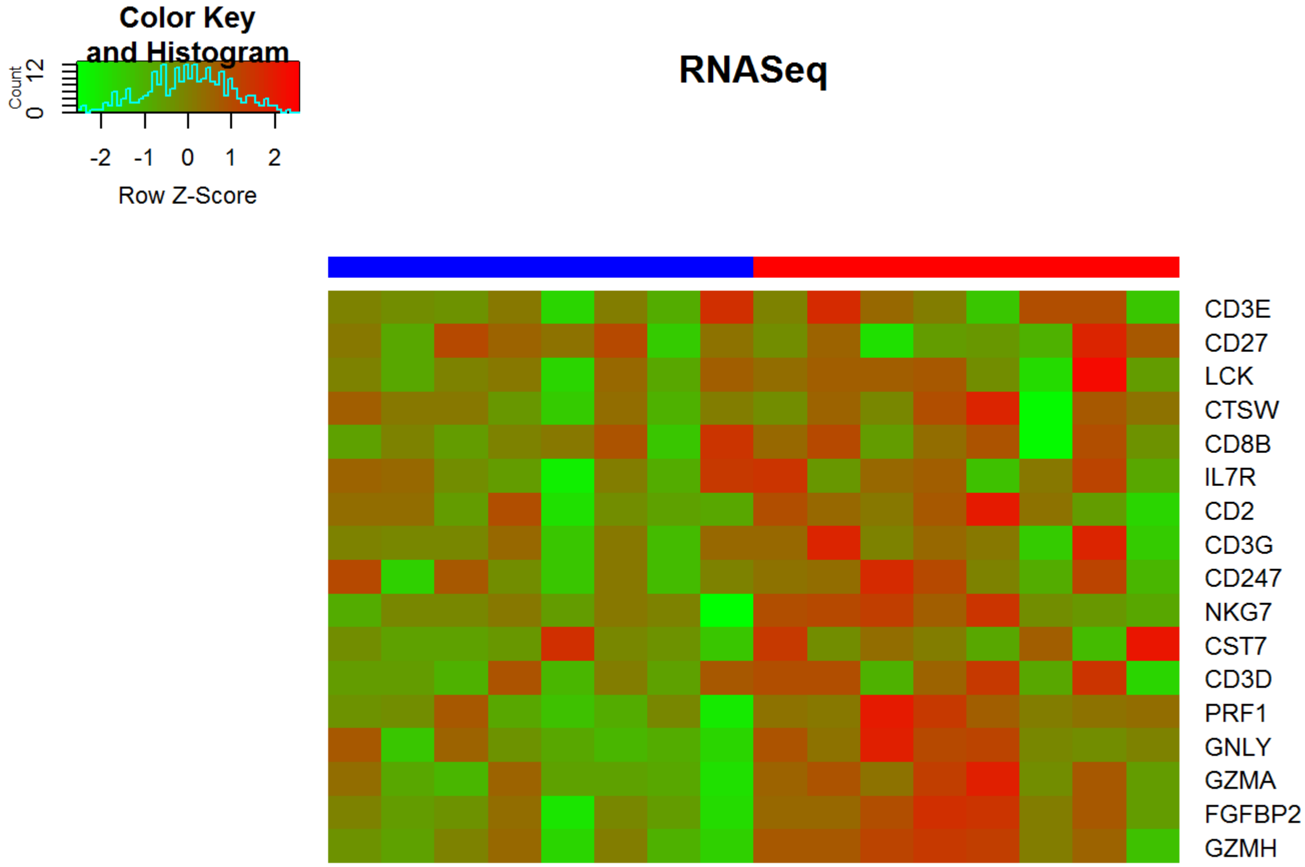


To obtain a heatmap representation of the gene expression for all genes part of the T cell signature in samples collected from women in labor and those not in labor at term, we select first the genes present on all three platfroms that are part of this signature and sort them by the log2 fold change of one of the platforms:

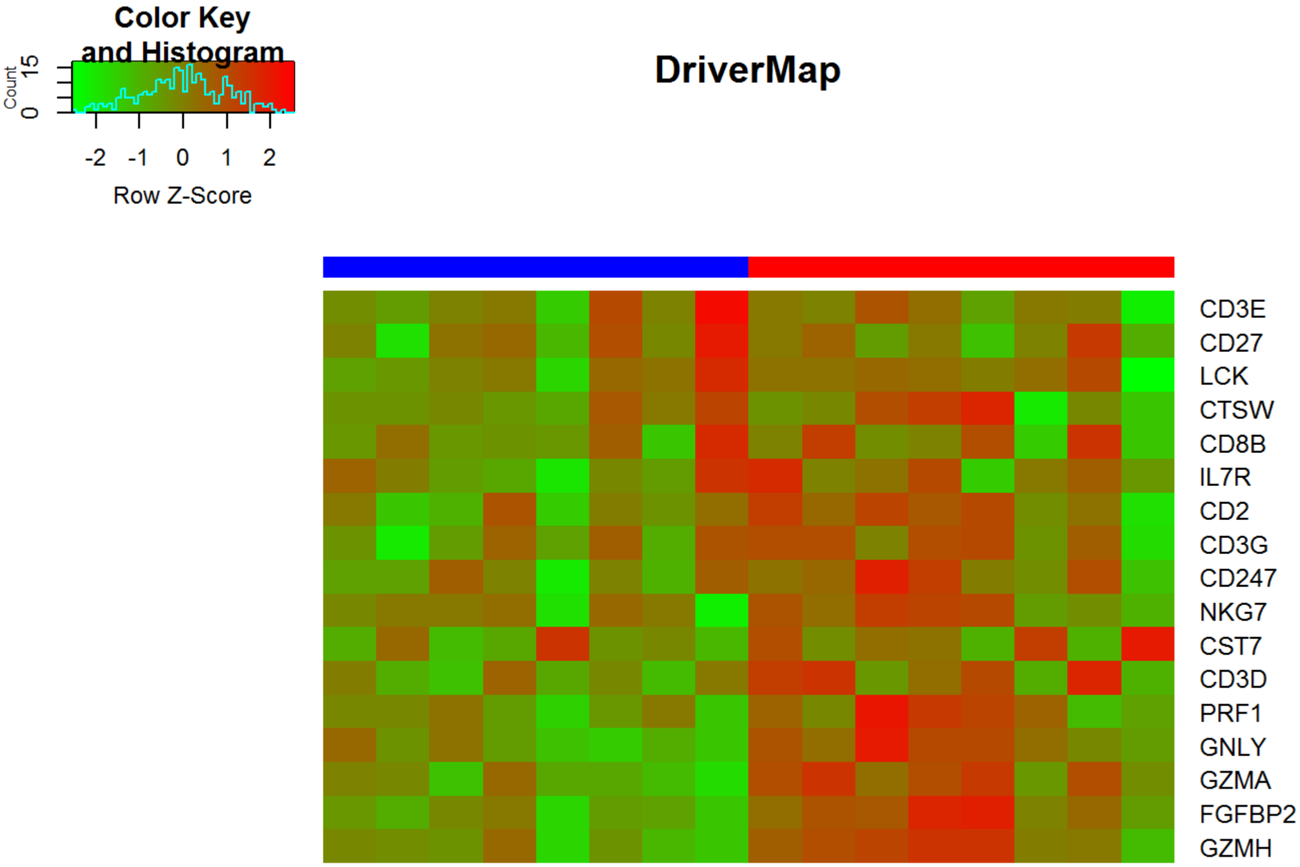
```
data(SCGeneSets)
SCGeneSets=SCGeneSets[SCGeneSets$Symbol%in%comg,]

tg1=CELLECTA[[2]]
tg1=tg1[tg1$SYMBOL%in%SCGeneSets[SCGeneSets$Type=="T cell","Symbol"],]
tg1=tg1[order(tg1$logFC),]

ano=anoLabor
ano=ano[order(ano$Group,decreasing=TRUE),]
coms=as.character(ano$SampleID)
gr=as.character(ano$Group)

heatmap.2(Hr[tg1$SYMBOL,coms],col=maPalette(low = "green", high = "red", k = 50),Colv=FALSE,Rowv
=FALSE,ColSideColors=ifelse(gr=="TIL","red","blue"),cexRow=1.2,
          scale="row",margins =c(4,5),trace="none",labCol = FALSE,main="HTA",dendrogram="none")
```



```
heatmap.2(Rr[tg1$SYMBOL,coms],col=maPalette(low = "green", high = "red", k = 50),Colv=FALSE,Rowv
=FALSE,ColSideColors=ifelse(gr=="TIL","red","blue"),cexRow=1.2,
          scale="row",margins =c(4,5),trace="none",labCol = FALSE,main="RNASeq",dendrogram="non
e")
```



```
heatmap.2(Cr[tg1$SYMBOL,coms],col=maPalette(low = "green", high = "red", k = 50),Colv=FALSE,Rowv
=FALSE,ColSideColors=ifelse(gr=="TIL","red","blue"),cexRow=1.2,
          scale="row",margins =c(4,5),trace="none",labCol = FALSE,main="DriverMap",dendrogram="n
one")
```



The point made with the heatmaps above is that genes part of the T cell signature defined based on single cell analysis tend to have higher expression (more red color) in women in the TIL group.

R session info

The details of the R session that generated these results are:

```
sessionInfo()
```

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
## 
## Matrix products: default
## 
## locale:
## [1] LC_COLLATE=English_United States.1252 
## [2] LC_CTYPE=English_United States.1252   
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C                          
## [5] LC_TIME=English_United States.1252    
## 
## attached base packages:
##  [1] splines   parallel  stats4    stats     graphics  grDevices utils    
##  [8] datasets  methods   base     
## 
## other attached packages:
##  [1] lme4_1.1-19                       Matrix_1.2-15                    
##  [3] marray_1.56.0                     Heatplus_2.24.0                  
##  [5] ROCR_1.0-7                        gplots_3.0.1                     
##  [7] pROC_1.13.0                       epiR_0.9-99                      
##  [9] survival_2.43-1                   UpSetR_1.3.3                     
## [11] DESeq2_1.18.1                     SummarizedExperiment_1.8.1       
## [13] DelayedArray_0.4.1                matrixStats_0.54.0               
## [15] limma_3.34.9                      annotate_1.48.0                  
## [17] XML_3.98-1.16                     EnsDb.Hsapiens.v75_2.99.0        
## [19] ensembldb_2.2.2                   AnnotationFilter_1.2.0           
## [21] GenomicFeatures_1.30.3            GenomicRanges_1.30.3             
## [23] GenomeInfoDb_1.14.0               hta20sttranscriptcluster.db_8.3.1
## [25] org.Hs.eg.db_3.2.3                AnnotationDbi_1.40.0             
## [27] IRanges_2.12.0                    S4Vectors_0.16.0                 
## [29] Biobase_2.38.0                    BiocGenerics_0.24.0              
## [31] pregnomics_1.0                    usethis_1.4.0                    
## [33] devtools_2.0.1                   
## 
## loaded via a namespace (and not attached):
##   [1] backports_1.1.2               Hmisc_4.1-1                  
##   [3] AnnotationHub_2.10.1          plyr_1.8.4                   
##   [5] lazyeval_0.2.1                BiocParallel_1.12.0          
##   [7] ggplot2_3.1.0                 digest_0.6.18                
##   [9] BiocInstaller_1.28.0          htmltools_0.3.6              
##  [11] gdata_2.18.0                  magrittr_1.5                 
##  [13] checkmate_1.8.5               memoise_1.1.0                
##  [15] cluster_2.0.7-1               remotes_2.0.2                
##  [17] Biostrings_2.46.0             prettyunits_1.0.2            
##  [19] colorspace_1.3-2              blob_1.1.1                   
##  [21] BiasedUrn_1.07                dplyr_0.7.8                  
##  [23] callr_3.0.0                   crayon_1.3.4                 
##  [25] RCurl_1.95-4.11               genefilter_1.60.0            
##  [27] bindr_0.1.1                   glue_1.3.0                   
##  [29] gtable_0.2.0                  zlibbioc_1.24.0              
##  [31] XVector_0.18.0                pkgbuild_1.0.2               
##  [33] scales_1.0.0                 DBI_1.0.0                   
##  [35] Rcpp_1.0.0                   xtable_1.8-3                
##  [37] progress_1.2.0               htmlTable_1.12              
##  [39] foreign_0.8-71               bit_1.1-14                  
##  [41] Formula_1.2-3                htmlwidgets_1.3             
##  [43] httr_1.3.1                   RColorBrewer_1.1-2          
##  [45] acepack_1.4.1                pkgconfig_2.0.2             
##  [47] nnet_7.3-12                  locfit_1.5-9.1              
##  [49] labeling_0.3                 tidyselect_0.2.5            
##  [51] rlang_0.3.0.1                later_0.7.5                 
##  [53] munsell_0.5.0                tools_3.4.3                 
##  [55] cli_1.0.1                    RSQLite_2.1.1               
##  [57] evaluate_0.12                stringr_1.3.1               
##  [59] yaml_2.2.0                   processx_3.2.0              
##  [61] knitr_1.20                   bit64_0.9-7                 
##  [63] fs_1.2.6                     caTools_1.17.1.1            
##  [65] purrr_0.2.5                  bindrcpp_0.2.2              
##  [67] nlme_3.1-137                 mime_0.6                    
##  [69] biomaRt_2.34.2               compiler_3.4.3              
##  [71] rstudioapi_0.8               curl_3.2                    
##  [73] interactiveDisplayBase_1.16.0 tibble_1.4.2               
##  [75] geneplotter_1.56.0           stringi_1.2.4               
##  [77] ps_1.2.1                     desc_1.2.0                  
##  [79] lattice_0.20-38              ProtGenerics_1.10.0         
##  [81] nloptr_1.2.1                 pillar_1.3.0                
##  [83] data.table_1.11.8            bitops_1.0-6                
##  [85] httpuv_1.4.5                 rtracklayer_1.38.3          
##  [87] R6_2.3.0                     latticeExtra_0.6-28         
##  [89] RMySQL_0.10.15               promises_1.0.1              
##  [91] KernSmooth_2.23-15           gridExtra_2.3               
##  [93] sessioninfo_1.1.1            MASS_7.3-51.1               
##  [95] gtools_3.8.1                 assertthat_0.2.0            
##  [97] pkgload_1.0.2                rprojroot_1.3-2             
##  [99] withr_2.1.2                  GenomicAlignments_1.14.2    
## [101] Rsamtools_1.30.0             GenomeInfoDbData_1.0.0      
## [103] hms_0.4.2                    grid_3.4.3                  
## [105] rpart_4.1-13                 minqa_1.2.4                 
## [107] rmarkdown_1.10               shiny_1.2.0                 
## [109] base64enc_0.1-3
```

References

Al-Garawi, A., V. J. Carey, D. Chhabra, H. Mirzakhani, J. Morrow, J. Lasky-Su, W. Qiu, N. Laranjo, A. A. Litonjua, and S. T. Weiss. 2016. “The Role of Vitamin D in the Transcriptional Program of Human Pregnancy.” *PLoS ONE* 11 (10): e0163832.

Gentleman, R. C., V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, et al. 2004. “Bioconductor: open software development for computational biology and bioinformatics.” *Genome Biol.* 5 (10): R80.

Heng, Y. J., C. E. Pennell, S. W. McDonald, A. E. Vinturache, J. Xu, M. W. Lee, L. Briollais, et al. 2016. “Maternal Whole Blood Gene Expression at 18 and 28 Weeks of Gestation Associated with Spontaneous Preterm Birth in Asymptomatic Women.” *PLoS ONE* 11 (6): e0155191.

Love, M. I., W. Huber, and S. Anders. 2014. “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.” *Genome Biol.* 15 (12): 550.

Ritchie, M. E., B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth. 2015. “limma powers differential expression analyses for RNA-sequencing and microarray studies.” *Nucleic Acids Res.* 43 (7): e47.

Tarca, A. L., R. Romero, Z. Xu, N. Gomez-Lopez, O. Erez, Hsu C.D., Hassan S.S., and V. J. Carey. 2018. “Targeted expression profiling by RNA-Seq improves detection of cellular dynamics during pregnancy and identifies a role for T cells in term parturition.” *Scientific Reports* accepted.

Tsang, J. C. H., J. S. L. Vong, L. Ji, L. C. Y. Poon, P. Jiang, K. O. Lui, Y. B. Ni, et al. 2017. “Integrative single-cell and cell-free plasma RNA transcriptomics elucidates placental cellular dynamics.” *Proc. Natl. Acad. Sci. U.S.A.* 114 (37): E7786–E7795.