# Exploring factors determining the composition of whole cell mRNA expression

Feng Geng

2016/01/17

## 1 Introduction: whether the variance observed is of biological origin or technical origin?

To address the pausibility of mouse as a model organism that highly resembles human being, Lin et al. in mouse ENCODE consortium set off to compare the mRNA profiles of these two species . The conclusion they drew from the data (as made available at http://zenodo.org/record/17606#.Vmr27bFuUaM in [2]) was however conterintuitive in that different tissues in mouse have more similar mRNA expression than orthologous tissues from human and from mouse. (The PCA analysis helps visualise the effect as the first PCA value is predominantly related to species, not tissue. 3D PCA plot also revealed clustering by species. A more quantitative analysis is variance decomposition of a linear fit, using a mixed model where effect of species and tissues are nested in a gene frame). Overall, clustering of mRNA expression by species is dominant while clustering by tissue is observable.

Nevertheless, validity of the conclusion above is still at question as Gilad and Mizarhi-Man revealed the significant confounding between sequencing batch and species (figure 1) raising a technical explanation of the observed species-dominated clustering. In other words, batch effect might have biased the study of the clustering pattern as previously known[1]. Furthermore, Gilad and Mizarhi-Man applied a linear model to remove potential batch effect, through which clustering by species is also removed, confirming the confounding between these two variables. It was argued that Lin et al. was unwarranted to claim that species-specific clustering dominates the transcript abundance profiles.

The confounding really requires careful examination. On one hand, confounding between species and sequencing multiplexing raise the concern that observation being a technical artefact instead of a biological phenomena. On the other hand, one simply can't exclude every variable by controlling them as constant. In the case of Shin et al., they chose to account for the variable with larger known effect when unable to fix two at the same time.

| D87PMJN1 (run 253, flow cell D2GUAACXX, lane 7) | D87PMJN1 (run 253, flow cell D2GUAACXX , lane 8) | D4LHBFN1 (run 276, flow cell C2HKJACXX , lane 4) | MONK (run 312, flow cell C2GR3ACXX , lane 6) | HWI-ST373 (run 375, flow cell C3172ACXX , lane 7) |
|---|---|---|---|---|
| heart | adipose | adipose | heart | brain |
| kidney | adrenal | adrenal | kidney | pancreas |
| liver | sigmoid colon | sigmoid colon | liver | brain |
| small bowel | lung | lung | small bowel | spleen |
| spleen | ovary | ovary | testis | Human |
| testis | | pancreas | | Mouse |

Figure 1: Inferred and confirmed sequencing design in Lin et al. [3], adapted from [2]

Traditional approaches were focused on decomposing variance in gene expression where ANOVA is restricted in that gene interaction can't be incorporated in a lack of replicates for each variety. Using PCA analysis, another choice to represent variance, it is tricky to fully represent the clustering in the subspace containing multiple PCA's (6 or more) without an intuitively plot at such dimensionality.

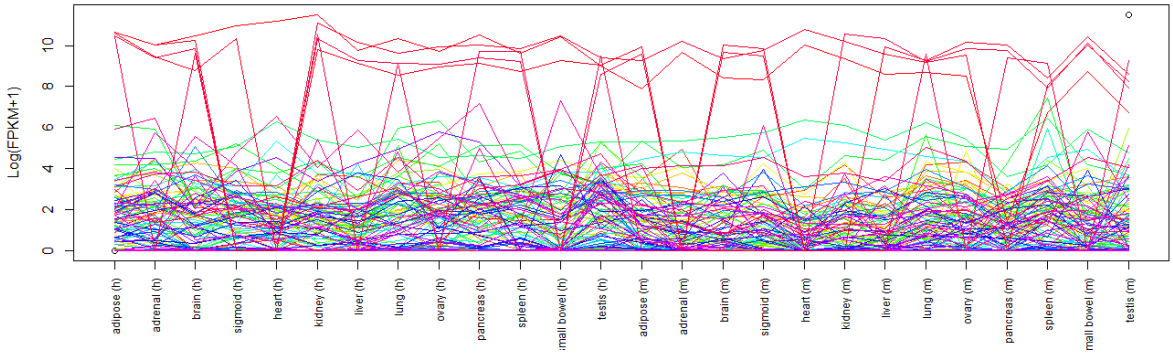
Mike in [6] developed a novel approach (NACC) to meausre expression conservation of a single gene to compare samples from various biological origin. I don't exactly understand how this could avert the artefact introduced in batch manipulation but combined with Gene Ontology (GO) annotation, NACC confirmed that categories of genes are differentially conserved. Unfortunately I did not have the chance to realise an NACC analysis in this paper.

# 2 Data Visualisation and Quantitative Analysis: Is the data really telling a batch effect or anything else?

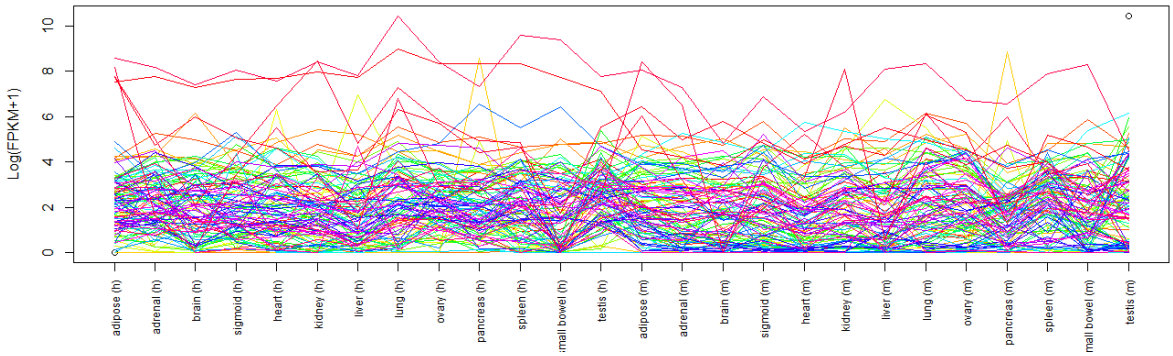
## 2.1 Intial Processing

For each gene ,we calculated log(FPKM+1) to indicate its expression level in order to normalise its distribution and avoid log(0) consistently during this analysis.

To get a overview of our data profiles of first 100 genes are plotted (figure 2). Some genes exhibit extreme vibration with a change of 10 on the log-scaled. Tracing these genes, we confirmed that mitochondrial genes were overrepresented in the result [2]. Thus we used his filtration script to improve the data.




(a) Before filtering

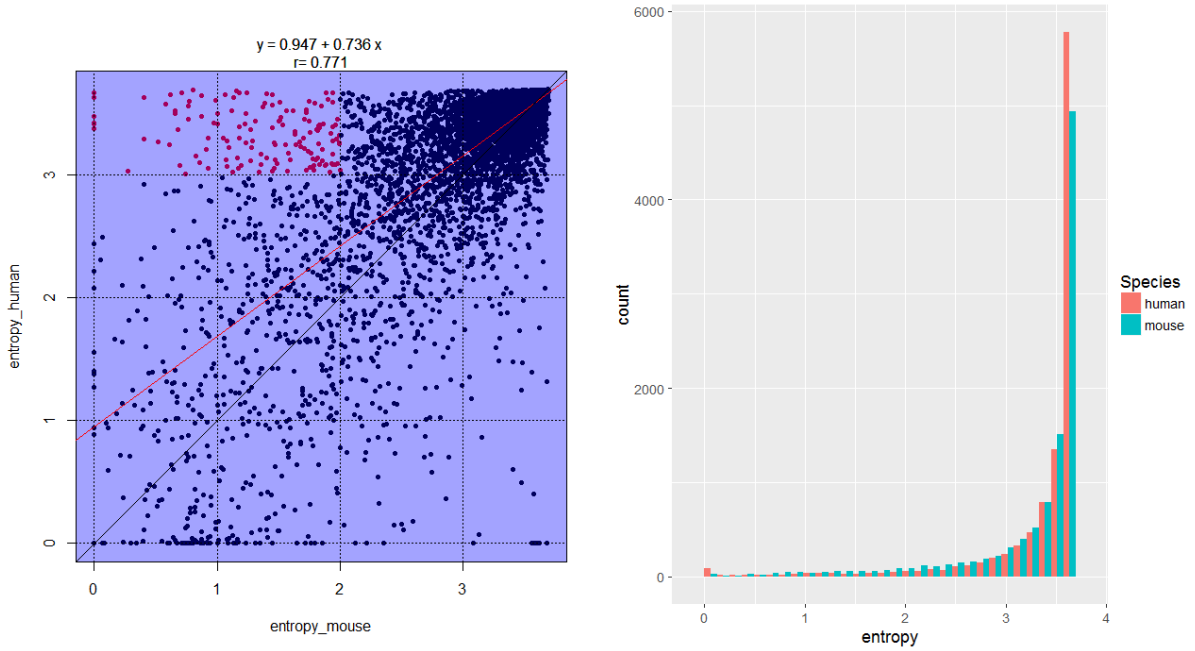



(b) After filtering

Figure 2: log(FPKM) across biosamples of first 100 genes in the dataset.

Following Lin et al., Shannon Entropy is ($H$) calculated for each gene in each species separately to estimate its tissue specificity [5]. The lower the $H$, the more specificly a gene is expressed. We found that apparent gene specificities in the two species do highly correlate (r=0.770), while human genes in general appear less specific(intercept=0.95±0.02,p<2E(-16), human regressed on mouse, see figure 4a). Patches show up at top-left and bottom-right of the scatter plot, indicating a change in specificity. As shown in figure 4b, there are more human genes to the higher end and lower end of entropy.

To further account for this increased number of unspecific genes in human, we plotted entropies of tissue-specific genes ($H$<2) of their orhologues in the opposite species (figure 3c). More human orthologues of such switched genes acquired a profound generalisation. With a criteria H>3 in human and H<2 in mouse, we identify 144 genes (see figure 5). Clearly their pattern of expression differ between species. Noticeably, loads of such expression cluster at mouse brain and mouse sigmoid. For tissue-specific genes in each species, we define the tissue where its expression is maximum as native. As seen in figure 6, mouse brain, mouse spleen and mouse testis are enriched with such native genes.GO analysis revealed overrepresentation in both ganetogenesis and cellular connection. Noticeably, mouse




(a) Paired plot with a least square fit (red) and an invariant line (black). Red dots are genes that lose specificity in human

(b) Histogram

Figure 4: Shannon Entropies in human and in mouse

brain and mouse spleen are sequenced in the same batch (see figure 1), which might provide a technical explanation.

## 2.2 Single gene level

To decompose variance at a single gene level, we first considered the model

$$x_{i,jk} = \mu_i + \alpha_{ij} + \beta_{ik} + \epsilon_i$$

Here expression of a specific gene (numbered i) under a specific condition (tissue j, species k) is considered to have a random baseline expression ($\mu$) as estimated by the average expression ,upon which tissue and species assert fixed effects ($\alpha$ and $\beta$ ) specific to this gene. A random variation for this gene across categories ($\epsilon_i$) is also included.

One should notice interaction between tissue and gene is not included here since expression in each category is only measured once, so that we could not estimate the random error when tissue and species

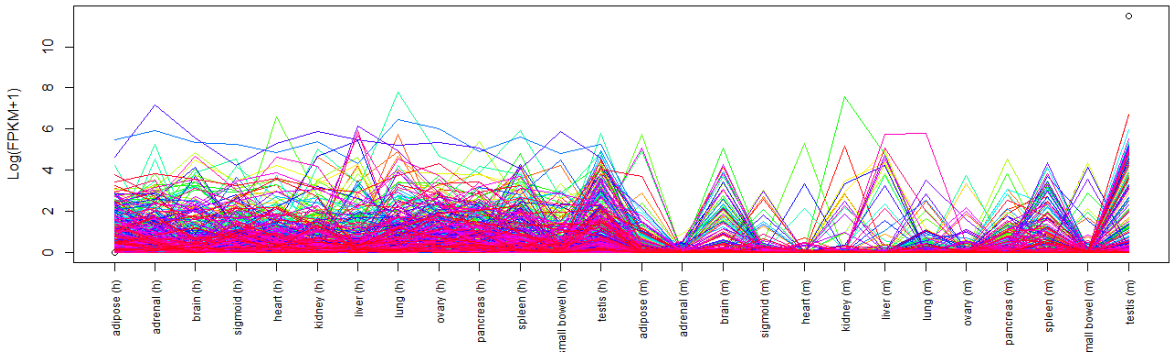



Figure 5: Profiles of genes that lose specificity in human

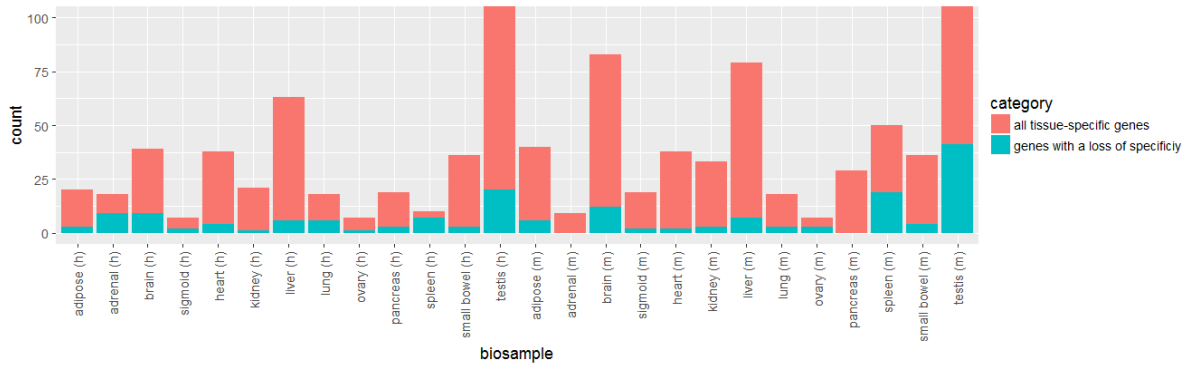



Figure 6: Number of genes native to each tissue

are both fixed. In other words, we don't have enough DOF to include the interaction term. This linear model is then fitted for each of the gene, on which an ANOVA is done. Confusingly, tissue's predictive power (as measured with its contribution to variance) is distributed with an increased variance as itself increases (see figure 7a). Intuitively tissue effect and species effect should depend on each other. To test the interaction term, however, we would need to include multiple points under a single catrgory so that differences between categorical averages could be distinguished from random error. With only one point under each category here, we are essentially forced to exclude the interaction term and assume its variance included in residual variance, i.e. the larger the residual variance, the more interaction between the tissue and species.

We also fitted a gene-specific random model instead using lme4 package in R, where it appears about half of the genes have most of their variance unexplained, while other genes adopt either tissue-specific expression or species-specific expression. Since I don't see how random model would fit our situation here, I continue with fixed effect model.

(I also tried to incorporate Shin et al. resequencing data[4], but my inexperience in RNA-seq doesn't allowed me to process the raw data to a stage suitable for analysis in R).

If we put seqBatch (sequence batch) in front of species, more variance is explained by seqBatch, vice versa. This can be explained by the confounding between the two variables, namely that their effects can't be distinguished from each other since there is little difference between their distribution.

Another concern in this regression is the nature of the effect, whether random or fixed. One can intrepret the mRNA profile difference between 2 biosamples as fixed, since we can specify the exact type of biosample to examine. Species effect and tissue effect should not be random since species and tissue type are definitely controlled by the experimentor. Nevertheless, these effects are always confounded with the highly random nature of biological samples in that biological replicates bearing same genotype could diverge in their phenotype to a random degree. Nevertheless, the distinguishment between random effect and fixed effect is variable.

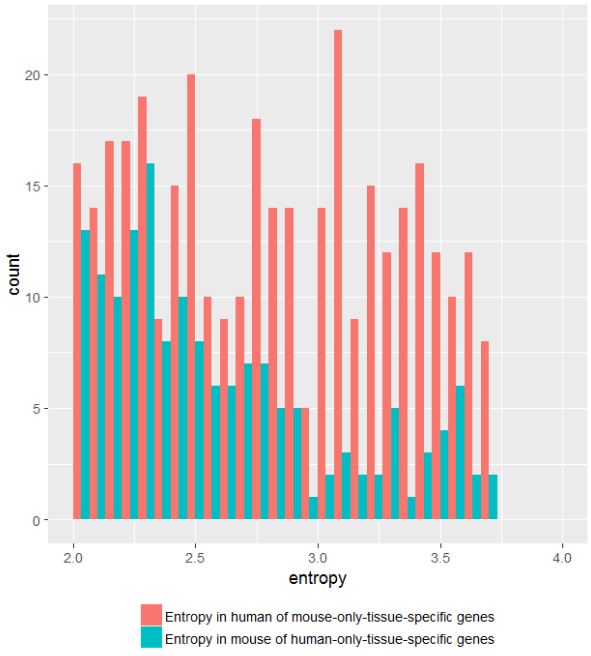
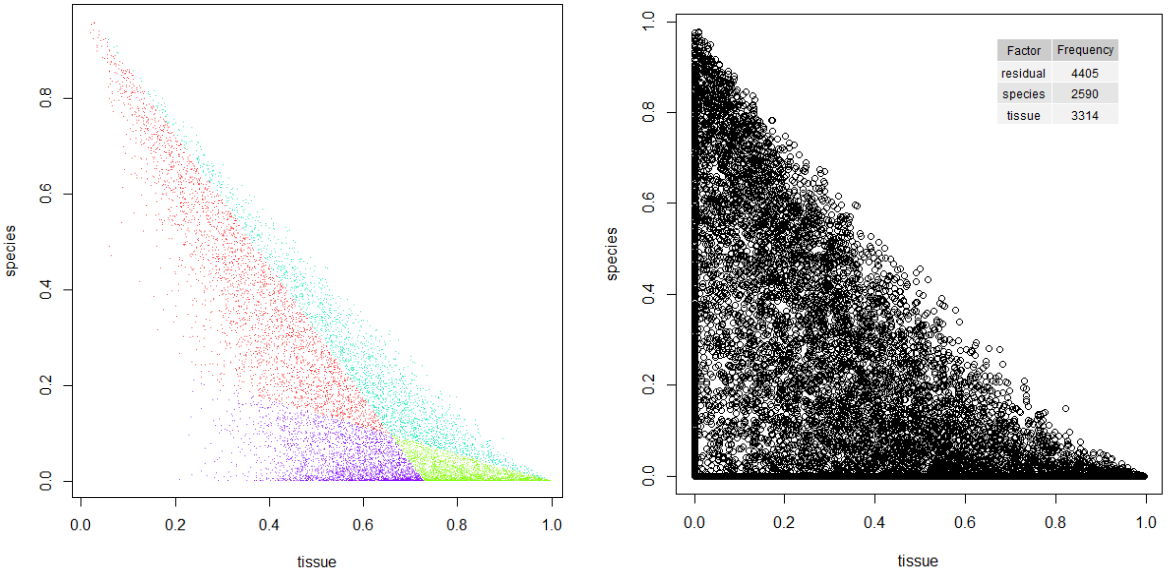



Figure 3: The higher entropy of genes with a change in specificity status

## 2.3 ANOVA with gene as a fixed effect

To assess the effect of sequence batch on the data as a whole and on the excess group that changed their specifcities (see 3), we construct models where gene name and its interaction with all other terms were




(a) Tissue and species as a fixed model

(b) Tissue and species as a fixed model, table showing the factor contributing the most variance

Figure 7: Fraction of variance conrtibuted by different variables

included to account for gene-to-gene difference. We then compare likelihood of a null model $H_0$:

$$x_{i,jk} = (\mu_i + \alpha_j + \beta_k) + \delta_{ij} + \sigma_{ik} + \epsilon$$

where $\delta$and $\sigma$ denote the interaction between tissue and gene, and between species and gene respectively, and a hypothised model $H_1$:

$$x_{i,jk} = (\mu_i + \alpha_j + \beta_k + \gamma_l) + \delta_{ij} + \sigma_{ik} + \xi_{il} + \epsilon$$

where $\gamma$ and $\xi$ denote independent and interacted effects of sequence batch.

ANOVA concerning the mouse genes that lose their specificities in human (see 3) justifies sequence batch as an effective predictor (F=1.159, p=0.0143), concording with the overrepresentation of genes specific to fifth batch in the group (see 1). To examine the statistical significance of this p-value, we took random subsets of genes and calculate the likelihood of $H_1$over $H_0$. As it appears in 8, sequence batch exmaine significant amout of variance in nearly all of the subsets, the p-value from the subset of interest only lies at the higher end of the distribution thus not enough to conclude the overrepresentation. We rationalise the predicting power of sequence batch as resulted from its equivalency to specific combination of species and tissue and thus partially accounting for intuitively significant species-tissue interaction. Of notice, we did not do such ANOVA on the all 10309 filtered genes since my R environment can't handle a linear model of this size.

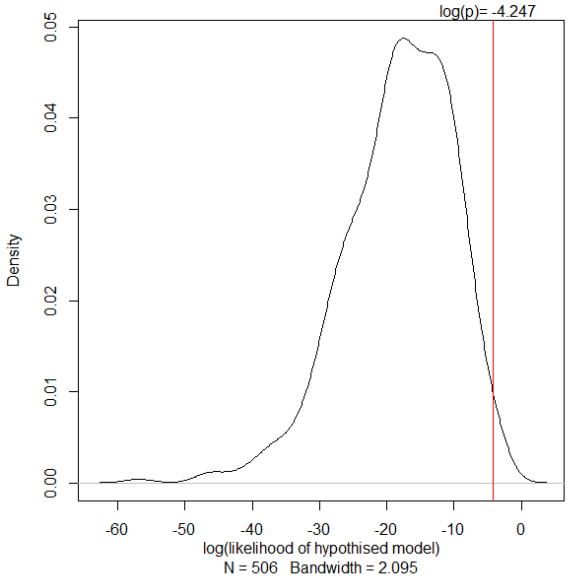
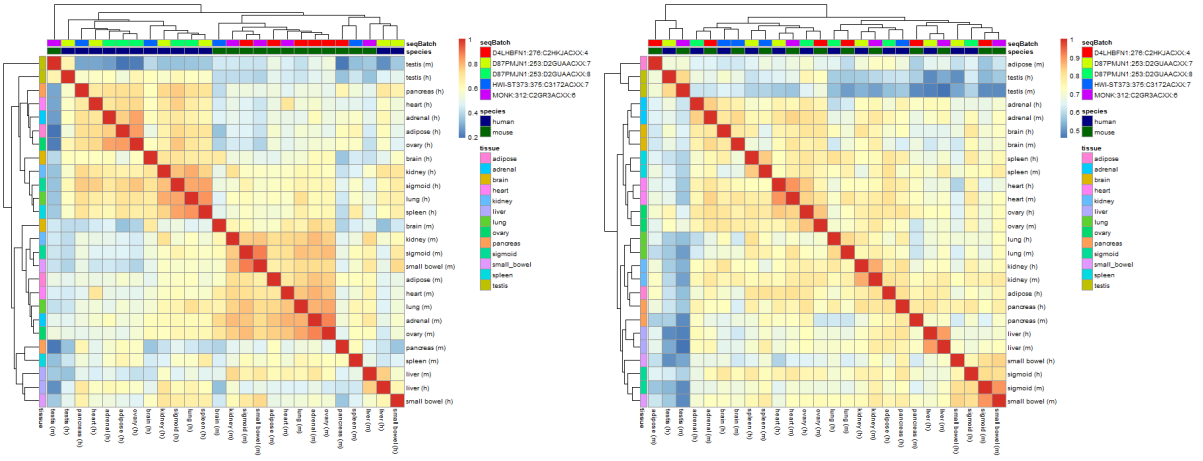



Figure 8: statistical significance of likelihood in determining overrepresentation

## 2.4 Hierarchial clustering:

To further address the source of variablility between samples we performed various clustering, visualised with pheatmap package in R. Before ComBat removal of potential batch effect, species clearly dominates the clustering. After ComBat removal, tissue appear to be more dominant. This is expectable since removal of batch effect would partially remove its confounding partner, which is species, as Gilad argued.

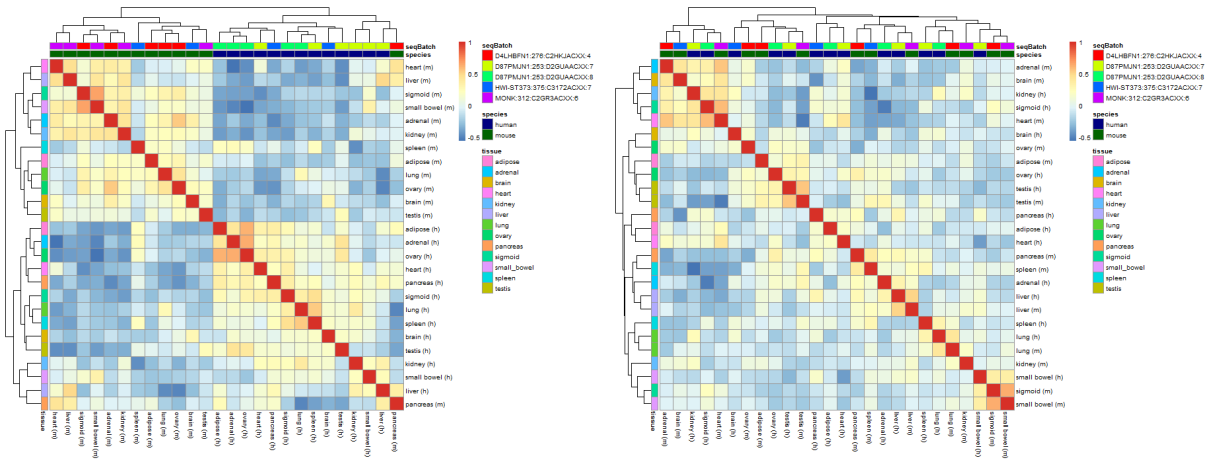


(a) Before removal of batch effect with ComBat

(b) After removal of batch effect by ComBat

Figure 9: Heatmaps clustered by complete linkage between FPKM profiles of biosamples,clustered by Euclidean distance

As pointed by Anshul in his comment on [2], Euclidean expression in higher dimension analysis could be problematic in that stronger signal will be overrepresented. PCA provide a simple solution to the dimension problem by reducing the dimensionality from 14744 ortholog pairs in a single mRNA profile, to 26 PC's as inherited from number of biosamples, that describes the same thing. Use of pearson correltaion between PC-vectors is also more justified since PC's are orthongonal by definition.. Nevertheless correlation between PC's require a more cautioned interpretation. Since PC's are constructed descriptors, their meaning is dependent on the compostion of factors. Use of PC heatmap sharpens the similarity in the network.



(a) Before removal of batch effect by ComBat, cell for correlation between orthologous tissue is bold-lined

(b) After removal of batch effect by ComBat

Figure 10: Heatmaps clustered by complete linkage between PC's of FPKM profiles of biosamples,clustered by Euclidean distance

The major clustering feature from figure 9 persists. After ComBat removal (figure 10b), there appears to be less tissue-dominated clustering compared to its equivalent of FPKM heatmap(figure 9). Use of ComBat possibly overcorrected the the model so that the meaning of PC's are overcomplicated to produce interpretable hierarchical clustering. However, if we were to correct for batch effect at such stage, it is unavoidable to remove effects of its confounders without enough technical repeats and thus DOF's. In figure 10a, there is more clustering by seqeunce batch compared with its equivalent FPKM heatmap. The discrepancy between species is reflected by cold colored patches at top-right and bottom left. On these patches, specific cells indicating orthologous correlation are hotter colored, usually the hottest in its row/column within the cold patch. Exceptions do exist:

- For mouse spleen, its correlation with particular human tissues is enhanced and with group of mouse tissues depressed, concording with its specific expression of some human-unspecific gene.
- For mouse sigmoid, its most correlated human tissue is small bowel. Several human tissues also correlate with it to a similar extent as human sigmoid is.
- Human ovary is most correlated to mouse spleen instead of mouse ovary. Within human, it is unusually strongly correlated with adipose and adrenal.
- Mouse pancrea is strongly correlated with human

These anomalies could possibly be explained by overrepresentation of tissue-specific genes.

Here and in [2] are automated with pheatmap package [pheatmap]. The quantitative side of this, however, is ambiguous. To deal with the whole dataset, we similarly have 2 ANOVA options as in data visualisation - one with PC's, one with fpkm matrix, in both of which we need to select a suitable stochastic structure so that we don't exhaust DOF .

## 2.5 Pairwise analysis

Pairwise correlation is the basis of hierarchial clustering. It seems difficult to incorporate ANOVA at this level. The notion behind an overall sample-sample correlation is "sample similarity". To understand patterns on the heatmap (figure 10a), We select 5 tissues, mouse sigmoid, mouse small bowel, mouse spleen, human sigmoid and human small bowel, to analyse closely.

Doing an overall comparision, we confirmed the correlation hierarchy as indicated on figure 10a. Between the strongly correlated tissue, more genes fall onto a region near the line of identity.

If we were to incorporate GO categories into the data, log-transformed fold-change would be a good responding variable to . Using model $\mu_i = \alpha_j + \epsilon_{ij}$(This model in fact does not measure anything but the random variation).

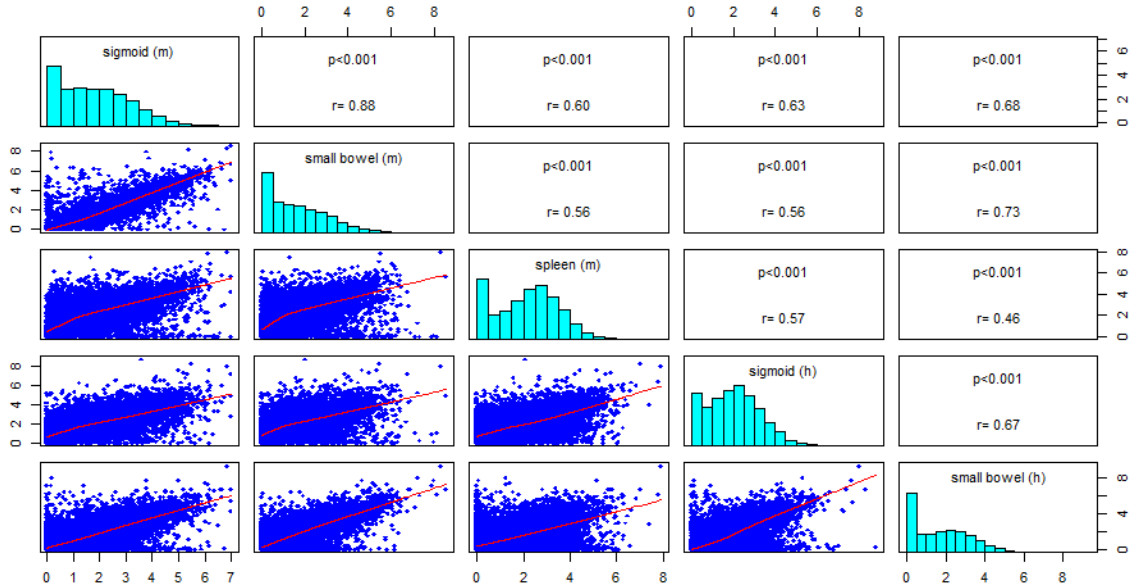
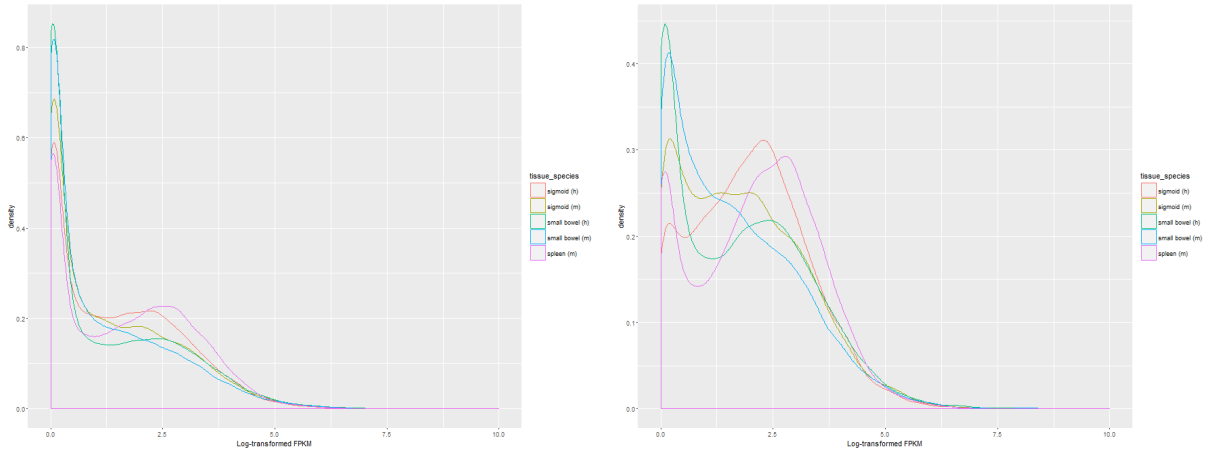



Figure 11: Comparision of mRNA profile between the 4 selected tissues. Scatter plot is smoothed with LOWESS, r is pearson correlation between each pair, p is significancy of the correlation.

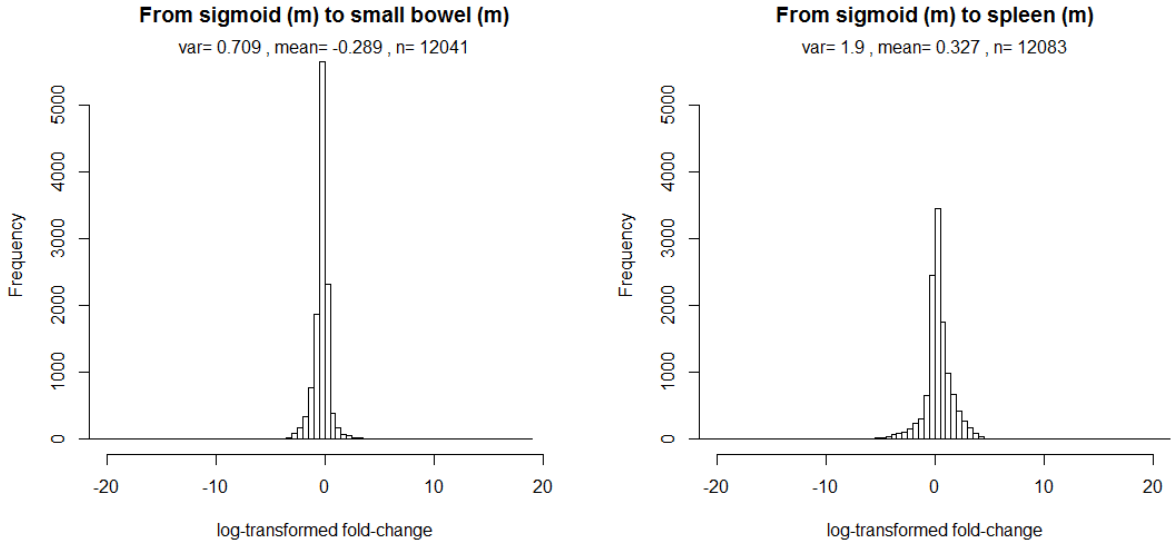
To get a flavour of variation in overall expression, density plots were prepared for the 5 tissues of concern above. Since the number of genes included are identical, use of density plots does not introduce distortion. The density plot confirmed that mouse spleen express genes at a generlly higher level than mouse sigmoid, with a similar numebr of expressed genes (though not necessarily the same gene).




(a) Before filtration

(b) After mitochondrial genes and lowly expressed genes are filtered

Figure 12: Comparision of FPKM distribution




(a) Between a strongly correlated pair

(b) Between a weakly correlated pair

Figure 13: Distribution of log-transformed fold-change of expression of shared genes between a tissue pair.

Fold-change is calculated between a strongly correlated tissue pair (mouse sigmoid and mouse small bowel, complete pearson correlation=0.884 , PCA pearson correlation=0.666 ),and betweenan typical weakly correlated pair ( mouse sigmoid and mouse spleen, complete pearson correlation=0.601, PCA perason correlation=0.093). Infigure 13, correlated tissue pair appear to have a less varied fold-change near 0. The avearage change in expression, however, is not zero.

# 3 Conclusion

We carried out extensive analysis on mRNA profiles using ANOVA, hierarchial clustering and pairwise comparison, during which we confirm that species and tissue are both strong predictors of gene expression, and did not find strong evidence of a sequence batch effect since its effect is masked by tissue-species interation. We do confirm the abnormal expression level of mitochondrial genes. Interestingly, it appears human orthologs tend to express with less specificities,while whether this is a sequencing artefact or reflect real biological difference remains a question.‘ Though I did not examine use of NACC in this manuscript, I believe it provide more detailed gene-wise view about conservation and divergenece between different biosamples at mRNA level, since use of it essentially synthesize related genes into metagenes, reducing the dimensionality as well to avoid complictaed issues. The original desgin by Shin et al. made it hard to examine interaction between species and tissue, and even harder for batch effect. That said, if paired tissues from different species are sequenced in the same batch, tissue would biosamples are to be sequenced withbe confounded with batch instead. Unless the batch could contain all the samples (which is not possible), confounding exists. We suggest to have more technical replication under each category and assess the quality of RNA-seq data in-depth before variance analysis to improve the robustness of any claim.

# References

[1] Joshua M Akey, Shameek Biswas, Jeffrey T Leek, and John D Storey. On the design and analysis of gene expression studies in human populations. *Nature genetics*, 39(7):807–8; author reply 808–9, jul 2007.

[2] Yoav Gilad and Orna Mizrahi-Man. A reanalysis of mouse ENCODE comparative gene expression data [v1; ref status: indexed, http://f1000r.es/5ez]. *F1000Research*, 121(4):1–32, 2015.

[3] Shin Lin, Yiing Lin, Joseph R. Nery, Mark A. Urich, Alessandra Breschi, Carrie A. Davis, Alexander Dobin, Christopher Zaleski, Michael A. Beer, William C. Chapman, Thomas R. Gingeras, Joseph R. Ecker, and Michael P. Snyder. Comparison of the transcriptional landscapes between human and mouse tissues. *Proceedings of the National Academy of Sciences*, 111(48):201413624, 2014.

[4] mouse ENCODE. mRNA-Seq files by Michale Synder as relevant.

[5] Jonathan Schug, Winfried-Paul Schuller, Claudia Kappen, J Michael Salbaum, Maja Bucan, and Christian J Stoeckert. Promoter features related to tissue specificity as measured by Shannon entropy. *Genome biology*, 6(4):R33, 2005.

[6] Feng Yue, Yong Cheng, Alessandra Breschi, Jeff Vierstra, Weisheng Wu, Tyrone Ryba, Richard Sandstrom, Zhihai Ma, Carrie Davis, Benjamin D Pope, Yin Shen, Dmitri D Pervouchine, Sarah Djebali, Robert E Thurman, Rajinder Kaul, Eric Rynes, Anthony Kirilusha, Georgi K Marinov, Brian A Williams, Diane Trout, Henry Amrhein, Katherine Fisher-Aylor, Igor Antoshechkin, Gilberto DeSalvo, Lei-Hoon See, Meagan Fastuca, Jorg Drenkow, Chris Zaleski, Alex Dobin, Pablo Prieto, Julien Lagarde, Giovanni Bussotti, Andrea Tanzer, Olgert Denas, Kanwei Li, M A Bender, Miaohua Zhang, Rachel Byron, Mark T Groudine, David McCleary, Long Pham, Zhen Ye, Samantha Kuan, Lee Edsall, Yi-Chieh Wu, Matthew D Rasmussen, Mukul S Bansal, Manolis Kellis, Cheryl A Keller, Christapher S Morrissey, Tejaswini Mishra, Deepti Jain, Nergiz Dogan, Robert S Harris, Philip Cayting, Trupti Kawli, Alan P Boyle, Ghia Euskirchen, Anshul Kundaje, Shin Lin, Yiing Lin, Camden Jansen, Venkat S Malladi, Melissa S Cline, Drew T Erickson, Vanessa M Kirkup, Katrina Learned, Cricket A Sloan, Kate R Rosenbloom, Beatriz Lacerda de Sousa, Kathryn Beal, Miguel Pignatelli, Paul Flicek, Jin Lian, Tamer Kahveci, Dongwon Lee, W. James Kent, Miguel Ramalho Santos, Javier Herrero, Cedric Notredame, Audra Johnson, Shinny Vong, Kristen Lee, Daniel Bates, Fidencio Neri, Morgan Diegel, Theresa Canfield, Peter J Sabo, Matthew S Wilken, Thomas A Reh, Erika Giste, Anthony Shafer, Tanya Kutyavin, Eric Haugen, Douglas Dunn, Alex P Reynolds, Shane Neph, Richard Humbert, R. Scott Hansen, Marella De Bruijn, Licia Selleri, Alexander Rudensky, Steven Josefowicz, Robert Samstein, Evan E Eichler, Stuart H Orkin, Dana Levasseur, Thalia Papayannopoulou, Kai-Hsin Chang, Arthur Skoultchi, Srikanta Gosh, Christine Disteche, Piper Treuting, Yanli Wang, Mitchell J Weiss, Gerd A Blobel, Xiaoyi Cao, Sheng Zhong, Ting Wang, Peter J Good, Rebecca F Lowdon, Leslie B Adams, Xiao-Qiao Zhou, Michael J Pazin, Elise A Feingold, Barbara Wold, James Taylor, Ali Mortazavi, Sherman M Weissman, John A Stamatoyannopoulos, Michael P Snyder, Roderic Guigo, Thomas R Gingeras, David M Gilbert, Ross C Hardison, Michael A Beer, and Bing Ren. A comparative encyclopedia of DNA elements in the mouse genome. *Nature*, 515(7527):355–364, nov 2014.

# Appendix:Associated R-code

## main.R:

#Attempt to load extra signal data from Babara Wold Lab at UCLA. # dat1<-read.delim('ENCFF215BWH.tsv',header=
# summary(dat1) # plot(log(dat1$FPKM+1)) # plot(logTransformed_fpkmMat[,1]) # plot((logTransformed_fpkmMat[,1
# as.data.frame(scan('ENCFF215BWH.tsv')) # install.packages("pheatmap") citation('pheatmap')
# GeneCode<-read.delim("dataAndConfigurationFiles/modencode.common.orth_human_mouse_one2one.txt",header
= F) # write.csv(file='mouseGeneCode.csv',GeneCode[,5]) # write.table(file='mouseGeneCode.txt',GeneCode[,5])
# head(GeneCode) # summary(GeneCode) # perform log transformation
library(ggplot2) library(lattice) library(rgl) library(lme4) library(reshape) source('my_functions.r')
#include my.melt(), sgnGene.linePlot(), panel.cor(), etc. See the end of codes
setwd("C:/Users/Feng/Google Drive/Of interests/BIOL7015_RData/TissueORSpecies") op<-par()
# Data Loading name<-read.delim( "Stanford_datasets.txt" ,header = F) fpkmMat<-read.delim("Stanford_datasets_f
,header=F) names(name)<-c('tissueNsp','seqBatch','species','tissue') datasets = as.data.frame(scan("Stanford_datasets.t
rawCounts <- as.matrix(read.table('Stanford_datasets_rawCountsMat.txt',header=FALSE,sep='\t')) geneDe-
tails <- as.data.frame(scan("ortholog_GC_table.txt",skip=1,list(mouse_name="",mouse_GC = 0.0,hu-
man_name = "",human_GC=0.0))) colnames(rawCounts) <- datasets$setname rownames(rawCounts) <-
geneDetails$human_name rownames(datasets) <- datasets$setname rownames(geneDetails) <- geneDe-
tails$human_name fpkmMat_unfiltered<-fpkmMat logTransformed_fpkmMat = log(fpkmMat+1) colnames(logTransforme
name$tissueNsp rownames(logTransformed_fpkmMat)<-geneDetails$human_name

#Initial processing range(logTransformed_fpkmMat) which(logTransformed_fpkmMat[,1]>10) # Before filtration 817 1745 11298 12259 12833 index<-c(sample(1:nrow(logTransformed_fpkmMat),100), 817,1745,11298,12259,12833) sgnGene.linePlot(index = index)

#Filter out genes with the lowest 30% expression or in mitochondria rowSums = apply(rawCounts,1,function(x) sum(x)) quantile(rowSums,probs = 0.3) # result is 2947.9 filter <- apply(rawCounts,1,function(x) sum(x)>2947.9 ) mt <- grep("mt-",geneDetails$mouse_name) filteredNames <- setdiff(rownames(rawCounts[filter,]),rownames(rawCounts filteredIndex<- setdiff( which(filter),mt) logTransformed_fpkmMat<-logTransformed_fpkmMat[filteredIndex,]

#visuallise after processing range(logTransformed_fpkmMat) #[1] 0.00000 10.45385 which(logTransformed_fpkmMat[,1 #nothing left x<-which(logTransformed_fpkmMat[,1]>7.5)#[1] 1665 2916 3773 4955 8670 index<-c(sample(1:nrow(logTra sgnGene.linePlot(index = index)

#Calculate Shannon Entropy calc.entropy<-function(dat=logTransformed_fpkmMat){ dat<-as.matrix(dat) rowsum<-apply(dat,MARGIN=1,sum) p<-dat/rowsum entropy<-apply(X=p,MARGIN=1,FUN=function(x){ sum(-x * log2(x),na.rm=T)}) return(entropy) } entropy_human<-calc.entropy(dat=logTransformed_fpkmMat[,1:13]) entropy_mouse<-calc.entropy(dat=logTransformed_fpkmMat[,14:26]) entropy<-data.frame(entropy_human,entropy_mous

spec_in_human<-(entropy_human < 2) spec_in_mouse<-(entropy_mouse < 2) spec_in_both=spec_in_human&spec_in_m spec_in_either=spec_in_human|spec_in_mouse

spec_in_mouse_only=(spec_in_mouse&(!spec_in_human)) spec_in_human_only=(spec_in_human&(!spec_in_mouse)) spec_change_mouse<-(entropy_human>3&spec_in_mouse) spec_change_human<- entropy_mouse>3&spec_in_human

plotData<-rbind(data.frame(entropy=entropy_human,Species='human'),data.frame(entropy=entropy_mouse,Species= ggplot(plotData,aes(x=entropy,fill=Species))+geom_histogram(alpha=1,position='dodge',width=0.5) cor(entropy_huma

plot(entropy_human~entropy_mouse,bg='lightgrey',pch=20,col=as.factor(spec_change_mouse)) rect(par("usr")[1],par( = rgb(0,0,1,0.2)) grid(col='black') mod<- lm(entropy_human~entropy_mouse) summary(mod) abline(mod,col=2) abline(a=0,b=1) cf<-round(coef(mod),3) eq <- paste0("y = ", cf[1],ifelse(sign(cf[2])==1, " + ", " - "), abs(cf[2]), " x ") mtext(eq, 3, line=1) mtext(paste('r=',round(summary(mod)$r.squared^0.5,3)))

plot(entropy_human_filtered~entropy_mouse_filtered,bg='lightgrey',pch=20,col=rgb(0,0,0,0.2)) rect(par("usr")[1],par( = rgb(0,0,1,0.2))+grid(col='black') mod<- lm(entropy_human_filtered~entropy_mouse_filtered) summary(mod) abline(mod,col=2)+abline(a=0,b=1) cf<-round(coef(mod),3) eq <- paste0("y = ", cf[1],ifelse(sign(cf[2])==1, " + ", " - "), abs(cf[2]), " x ") mtext(eq, 3, line=1)+mtext(paste('r=',round(summary(mod)$r.squared^0.5,3)))

## sign check to avoid having plus followed by minus for negative coefficients theme_set(theme_gray(base_size = 18)) e1<-entropy_human[spec_in_mouse_only] e2<-entropy_mouse[spec_in_human_only] plotData<-rbind(data.frame(en in human of mouse-only-tissue-specific genes') ,data.frame(entropy=e2,annotation='Entropy in mouse of human-only-tissue-specific genes') ) ggplot(plotData,aes(x=entropy,fill=annotation))+geom_histogram(alpha=1,position=

spec_change_human_max<-apply(MARGIN=1,logTransformed_fpkmMat[spec_change_human,],FUN=which.max) spec_change_mouse_max<-apply(MARGIN=1,logTransformed_fpkmMat[spec_change_mouse,],FUN=which.max) tisspec_of_mouse<-13+apply(X=logTransformed_fpkmMat[spec_in_mouse,14:26], MARGIN = 1, FUN=which.max) tisspec_of_human<-apply(X=logTransformed_fpkmMat[spec_in_human,1:13], MARGIN = 1, FUN=which.max) plotData<-melt(list(c(tisspec_of_human,tisspec_of_mouse),c(spec_change_human_max,spec_change_mouse_max))) if(!is.factor(plotData$category)){ colnames(plotData)<-c('biosample','category') plotData$category<-as.factor(plotData levels(plotData$category)<-c('all tissue-specific genes','genes with a loss of specificiy')}

ggplot(plotData,aes(x=biosample,fill=category))+geom_bar(position = 'identity')+scale_x_discrete(limits=name$tissu hjust=1,vjust=.5))+ coord_cartesian(ylim=c(0,100)) levels(plotData$category)<-c('all tissue-specific genes','genes that lose their specificiy')

#Singe Gene Anova

#A fixed effect model SUMsq<-apply(MARGIN=1,X=logTransformed_fpkmMat,var)*26 hist(SUMsq,breaks=100) nGene<-nrow(logTransformed_fpkmMat) vartab1=data.frame(gene=character(),species=numeric(),tissue=numeric(),resi ptab=data.frame(species=logical(),tissue=logical()) for(i in 1:nGene) { #i=1 lmData<-name lmData$expr<- as.numeric(logTransformed_fpkmMat[i,]) my_lm_fix.effect<-lm(expr~species+tissue,data=lmData) aov<- anova(my_lm_fix.effect) var<-c(aov$'Sum Sq') ptab[i,]<-aov$'Pr(>F)'<0.05 varPercent<-vapply(var,FUN=function(x){x = 1) varPercent<-c(rownames(logTransformed_fpkmMat)[i],varPercent) vartab1[i,]<-varPercent }

ptab # tabulate the resultant variance fractions vartab1.ratio<-data.matrix(vartab1) ptab$none<- (!ptab$species)&(!ptab$tissue) ptab$both<-ptab$species&ptab$tissue ptab$tissue_only<-ptab$tissue&!ptab$both ptab$species_only<-ptab$species&!ptab$both col=rainbow(4,alpha=0.7) plot(species~tissue,col=col[1],data = vartab1.ratio[ptab$species,],pch='.') points(species~tissue,col=col[2],data = vartab1.ratio[ptab$tissue,],pch='.') points(species~tissue,col=col[3],data = vartab1.ratio[ptab$both,],pch='.') points(species~tissue,col=col[4],data = vartab1.ratio[ptab$none,],pch='.') legend(0.5,0.9,legend=c('Species effect significant n=2959','Tissue effect significant n=2318','Both effects significant n=2226','No effect significant n=2806'),fill=col,cex=0.7) apply(ptab, 2, sum)

# Mixed model library(lme4) dat=as.matrix(logTransformed_fpkmMat) random.models<-vector('list',nGene)

```
for (i in 1:nGene){ random.models[[i]]<-lmer(dat[i,]~(1|name$tissue)+(1|name$species)) } rmMod.vartab<-
matrix(rep(0,nGene*3),ncol=nGene,nrow = 3) rownames(rmMod.vartab)<-x$grp for(i in 1:nGene){ rmMod.vartab[,i]<-
as.data.frame(summary(random.models[[i]])$varcor)$vcov } (rmMod.vartab.ratio[1:3,1:4]) rmMod.vartab.ratio<-
t(t(rmMod.vartab)/apply(MARGIN=2,rmMod.vartab,sum)) plot(rmMod.vartab.ratio[2,]~rmMod.vartab.ratio[1,],ylab='
plot(species~tissue,data=vartab1.ratio) gNo<-as.data.frame(table(c('tissue','species','residual')[apply(MARGIN=2,rmMo
colnames(gNo)<-c('Factor','Frequency') library(gridExtra) tt <- ttheme_default(colhead=list(fg_params
= list(parse=TRUE))) tbl <- tableGrob(gNo, rows=NULL, theme=tt) grid.arrange(tbl)
    #Overall ANOVA # simple regression including seqBatch lmDat<-my.melt(logTransformed_fpkmMat[index,])
mod1<-lm(expr~(tissue)*human_name,data=lmDat) mod2<-lm(expr~(tissue+seqBatch)*human_name,data=lmDat)
mod3<-lm(expr~(seqBatch+tissue+species)*human_name,data=lmDat) mod4<-lm(expr~(seqBatch+species+tissue)*hu
sum(spec_change_mouse|spec_change_human) anova(mod1,mod2,mod3,mod4)
    # FDR test nGene<-dim(logTransformed_fpkmMat)[1] pVec<-numeric() for(i in 1:500){ index<-
sample(1:nGene,176) lmDat<-my.melt(logTransformed_fpkmMat[index,]) mod5<-lm(expr~(species+tissue)*human_name
mod6<-lm(expr~(species+seqBatch+tissue)*human_name,data=lmDat) x<-anova(mod5,mod6) pVec[i]<-
x$`Pr(>F)`[2] } write.table(file='pVec.txt',pVec) x<-log(pVec) hist(log(pVec),breaks=100) plot(density(log(pVec)),main=
mtext(side=1,line=2,'log(likelihood of hypothised model)') mtext(at=c(-5),paste('log(p)=',round(v,3)))
v=log(anova(fxmod1,fxmod2,fxmod4)$`Pr(>F)`[3]) abline(v=v,col=2)
    # Pairwise Analysis fiveTissue<-c('sigmoid (m)','small bowel (m)','spleen (m)','sigmoid (h)','small
bowel (h)') sub<-subset(tmp,tmp$tissue_species %in% fiveTissue) plotData<-as.data.frame(sub) ggplot(plotData,aes(exp
color = tissue_species)) + geom_density(alpha = 1)+xlab('Log-transformed FPKM')+xlim(c(0,10))
    sgnGene.linePlot(index=which(spec_change_mouse))
    colnames(fpkmMat)<-name$tissueNsp fpkmMat_nozero<-fpkmMat[-which(apply(fpkmMat,MARGIN=1,function(x){
plotData<-logTransformed_fpkmMat[,fiveTissue] pairs(plotData, lower.panel=panel.smooth1, upper.panel=panel.cor,diag
abline(h=1) index=c(grep('sigmoid \\(m)',name$tissueNsp),grep('small bowel \\(m)',name$tissueNsp))
FC<-as.data.frame(logTransformed_fpkmMat[,index]) FC$FC<-FC[,2]/FC[,1] FC$logFC<-log(FC$FC)
hist(FC$logFC,breaks=50,xlim=c(-20,20),ylim=c(0,5500),xlab = 'log-transformed fold-change',main =
'From sigmoid (m) to small bowel (m)' ) var<-var(logFC_finite<-FC$logFC[is.finite(FC$logFC)]) mean<-
mean(logFC_finite) mtext(side=3,paste('var=',signif(var,3),',','mean=',signif(mean,3),',','n=',length(logFC_finite)))
    mod<-lm(logFC_finite~1,na.action = na.omit) plot(mod,which=2) logFC_finite[7230] sgnGene.linePlot(index.backsear
logTransformed_fpkmMat[7707,index] sgnGene.linePlot(index=vapply(c(7230), index.backsearch, FUN.VALUE
= 1))
    index.backsearch<-function(index){ old_index<-which(is.finite(FC$logFC))[index] return(old_index)
}
    head(logTransformed_fpkmMat) index=c(grep('sigmoid \\(m)',name$tissueNsp),grep('spleen \\(m)',name$tissueNsp)
FC<-as.data.frame(logTransformed_fpkmMat[,index]) FC$FC<-FC[,2]/FC[,1] FC$logFC<-log(FC$FC)
hist(FC$logFC,breaks=100,xlim=c(-20,20),ylim=c(0,5500),xlab = 'log-transformed fold-change',main =
'From sigmoid (m) to spleen (m)' ) var<-var(logFC_finite<-FC$logFC[is.finite(FC$logFC)]) mean<-
mean(logFC_finite) mtext(side=3,paste('var=',signif(var,3),',','mean=',signif(mean,3),',','n=',length(logFC_finite)))
    mod<-lm(logFC_finite~1,na.action = na.omit) plot(mod,which=2) logFC_finite[7230] index.backsearch(7230)
logTransformed_fpkmMat[7707,index] sgnGene.linePlot(index=vapply(c(7230), index.backsearch, FUN.VALUE
= 1))
    source('Gilad_ComBat.R') #Normalising with ComBat dat_gilad<-logTransformed_fpkmMat[,]
    ##8. PCA by Shin and by Gilad (after ComBat) colnames(dat_gilad) <-paste(datasets$setname,datasets$seqBatch)
colnames(dat_gilad) <-paste(datasets$seqBatch)
    # perform prcomp on the transposed matrix from which columns (genes) of zero variance have
been removed dat_gilad<-logTransformed_fpkmMat[,] transposedat_gilad = t(dat_gilad) pca_proc_shin<-
pca_proc <- prcomp(transposedat_gilad[,apply(transposedat_gilad, 2, var, na.rm=TRUE) != 0],scale=TRUE,center=TRU
colnames(combat)<-paste(datasets$setname) transposed_combat <- t(combat) # perform prcomp on the
transposed matrix from which columns (genes) of zero variance have been removed pca_proc_combat<-
pca_proc <- prcomp(transposed_combat[,apply(transposed_combat, 2, var, na.rm=TRUE) != 0],scale=TRUE,center=TRU
rownames(pca_proc_shin$x)<-name$tissueNsp row.names(pca_proc_combat$x)<-name$tissueNsp colnames(combat)<-
name$tissueNsp rownames(pca_proc_shin$x)<-name$tissueNsp row.names(pca_proc_combat$x)<-name$tissueNsp
    #Check relation between native genes number and pca pca_proc<-pca_proc_shin plotData = datasets[,c("setname","sp
plotData$PC1 <- pca_proc$x[,1] plotData$PC2 <- pca_proc$x[,2] plotData$PC3 <- pca_proc$x[,3] plot-
Data$PC4 <- pca_proc$x[,4] plotData$PC5 <- pca_proc$x[,5] plotData$PC6 <- pca_proc$x[,6] plotData$nativeNo<-
as.numeric(table(factor(spec_change_mouse_max,levels=1:26)))
    plotData<-plotData[-26,] # 2D plots of pairs of principal components qplot(PC1,nativeNo,data=plotData,color=specie
(21% variability)",ylab="PC2 (12% variability)")+scale_shape_manual(values=c("adipose"=0,"adrenal"=1,"brain"=2,"s
```

qplot(PC2,nativeNo,data=plotData,color=species,shape=tissue,xlab="PC1 (21% variability)",ylab="PC2 (12% variability)")+scale_shape_manual(values=c("adipose"=0,"adrenal"=1,"brain"=2,"sigmoid"=3,"heart"=4,"kidney mod<-lm(PC1~nativeNo,data=plotData) mod<-lm(PC2~nativeNo,data=plotData) mod<-lm(PC3~nativeNo,data=plotD mod<-lm(PC4~nativeNo,data=plotData) mod<-lm(PC5~nativeNo,data=plotData) mod<-lm(PC6~nativeNo,data=plotD anova(mod) summary(mod) qplot(PC1,nativeNo,data=plotData,color=species,shape=tissue,xlab="PC1 (21% variability)",ylab="PC2 (12% variability)")+scale_shape_manual(values=c("adipose"=0,"adrenal"=1,"brain"=2,"s qplot(PC2,nativeNo,data=plotData,color=species,shape=tissue,xlab="PC1 (21% variability)",ylab="PC2 (12% variability)")+scale_shape_manual(values=c("adipose"=0,"adrenal"=1,"brain"=2,"sigmoid"=3,"heart"=4,"kidney

# heatmap of Pearson correlations # Heatmaps using euclidean distances between library(pheatmap)

#Annotating the pheatmaps annotation<- name rownames(annotation) <- annotation$tissueNsp # check out the row names of annotation_row species <- c("navy", "darkgreen") names(species) <- c("human", "mouse") seqBatch <-rainbow(5) names(seqBatch)<- levels(name$seqBatch) anno_colors <- list(species=species,seqBatch=seqBatch)

name$tissueNsp->colnames(dat_gilad) pheatmap(cor(dat_gilad) ,annotation_col=annotation[,3:2,drop=F] ,annotation_row=annotation[,4,drop=F] ,annotation_colors = anno_colors ,legend = T ,annotation_legend = T ) # complete linkage; euclidean distance

pheatmap(cor(combat),show_rownames = T,annotation_col=annotation[,3:2,drop=F],annotation_row=annotation[,4,c = anno_colors,legend = T) pheatmap(cor(t(pca_proc_shin$x)),show_rownames = T,annotation_col=annotation[,3:2,drop= = anno_colors,legend = T) pheatmap(cor(t(pca_proc_combat$x)),show_rownames = T,annotation_col=annotation[,3:2,dro = anno_colors,legend = T) pheatmap(cor(t(pca_proc_shin2$x)),show_rownames = T,annotation_col=annotation[,3:2,drop= = anno_colors,legend = T) pheatmap(cor(dat_gilad)) #pearson correlation; complete linkage pheatmap(cor(dat_gilad),clus #pearson correlation; average linkage pheatmap(cor(dat_gilad),clustering_method="single") #pearson correlation; single linkage plot(tmp$expr~tmp$tissue_species,las=2)

## my.functions.R:

my.melt<-function(dat=logTransformed_fpkmMat){ human_name<-rownames(dat) tmp=cbind(data.frame(expr=dat[,1] for(i in 2:dim(dat)[2]){tmp=rbind(tmp,cbind(data.frame(expr=dat[,i]),name[i,],gene=human_name))} colnames(tmp)<- c('expr','tissue_species','seqBatch','species','tissue','human_name') return(tmp) } sgnGene.linePlot<-function(index,dat= op<-par() colnames(dat)<-name$tissueNsp yrange<-range(dat) # xrange<-range(x<-as.factor(colnames(dat))) # Highly expressed gene # dat<-dat_raw index %in% mt nindex<-length(index) linecolors<-rainbow(nindex) if(legend){op<-par() par(xpd=T,mar=c(5.1, 4.1, 4.1, 8.1))} if(!add) { plot(xrange<-c(1,26), yrange,xaxt='n', xlab=", ylab="Log(FPKM+1)") } axis(1,main='Biosample', at=1:26, cex.axis=0.75,labels=name$tissueNsp,las=2) for(i in 1:length(index) ){ k<-index[i] ?lines lines(x=1:26,y=dat[k,],col=linecolors[i] ) } geneDetails$human_name[1745] if(legend){ legend(legend = index,fill = linecolors,"topright", inset=c(-0.25,0)) par(op$mar) } mtext("Biosample", side=1, line=5) }

#for pairwise plot,The following panel functions are quoted from http://stackoverflow.com/questions/15271103/how-to-modify-this-correlation-matrix-plot/15271627#15271627

panel.cor <- function(x, y, digits=2, cex.cor) { usr <- par("usr"); on.exit(par(usr)) par(usr = c(0, 1, 0, 1)) r <- abs(cor(x, y)) txt <- format(c(r, 0.123456789), digits=digits)[1] test <- cor.test(x,y) Signif <- ifelse(round(test$p.value,3)<0.001,"p<0.001",paste("p=",round(test$p.value,3))) text(0.5, 0.25, paste("r=",txt)) text(.5, .75, Signif) }

panel.smooth1<-function (x, y, col = rgb(0,0,1,0.5), bg = NA, pch = 20, cex = 0.8, col.smooth = "red", span = 2/3, iter = 3, ...) { points(x, y, pch = pch, col = col, bg = bg, cex = cex) ok <- is.finite(x) & is.finite(y) if (any(ok)) lines(stats::lowess(x[ok], y[ok], f = span, iter = iter), col = col.smooth, ...) # mtext('hi',at=c(5,5)) } panel.smooth2<-function (x, y, col = "blue", bg = NA, pch = 18, cex = 0.8, col.smooth = "red", span = 2/3, iter = 3, ...) { points(x, y, pch = pch, col = col, bg = bg, cex = cex) ok <- is.finite(x) & is.finite(y) if (any(ok)) abline(lm(y[ok]~x[ok]), col = col.smooth, ...) }

panel.hist <- function(x, ...) { usr <- par("usr"); on.exit(par(usr)) par(usr = c(usr[1:2], 0, 1.5) ) h <- hist(x, plot = FALSE) breaks <- h$breaks; nB <- length(breaks) y <- h$counts; y <- y/max(y) rect(breaks[-nB], 0, breaks[-1], y, col="cyan", ...) }

## gilad_ComBat.R:

#3. Use EDASeq to normalize data within lanes, accounting for GC content source("https://bioconductor.org/biocLite.R" # biocLite("EDASeq") filteredRawCounts <- rawCounts[filteredNames,] library(EDASeq) GCnorm-Counts <- filteredRawCounts GCnormCounts[,1:13] <- withinLaneNormalization(filteredRawCounts[,1:13],geneDetails[fil GCnormCounts[,14:26] <- withinLaneNormalization(filteredRawCounts[,14:26],geneDetails[filteredNames,"mouse_GC"],w

#4. depth normalize,using TMM scaling factors - divide by sum, then multiply by mean of sums # biocLite("edgeR") library(edgeR) origColSums <- apply(rawCounts,2,function(x) sum(x)) normFactors <- calcNormFactors(GCnormCounts,method='TMM') colSums = apply(GCnormCounts,2,function(x) sum(x)) normalizedColSums <- origColSums i <- 1 while (i<length(colSums)){ normalizedColSums[i] <- origColSums[i]* normFactors[i] i <- i+1 } meanDepth <- mean(normalizedColSums) filteredDepthNormCounts <- GCnormCounts i <- 1 while (i<ncol(filteredDepthNormCounts)){ filteredDepthNormCounts[,i] <- (GCnormCounts[,i]/normalizedColSums[i])*meanDepth i <- i+1 }

#5. log transformation logTransformedDepthNormCounts <- log2(filteredDepthNormCounts+1)

#6. use combat on log2 values to remove batch effects # install.packages('nlme') # source("https://bioconductor.org/b # biocLite("sva") library(sva) meta <- data.frame(seqBatch = datasets$seqBatch,tissue=datasets$tissue,species=datasets design <- model.matrix(~1,data=meta) combat <- ComBat(dat= logTransformedDepthNormCounts,batch=meta$seqBat # numCovs=NULL, par.prior=TRUE)

library(sva) meta <- data.frame(seqBatch = datasets$seqBatch,tissue=datasets$tissue,species=datasets$species) design <- model.matrix(~1,data=meta) combat <- ComBat(dat= logTransformedDepthNormCounts,batch=meta$seqBat # numCovs=NULL, par.prior=TRUE)