BIC metabolomics data analysis

In this document we present the joint analysis of the different metabolomics datasets submitted to BIC (currently, as of June 2019). We rely on the following resources for metabolomics-specific issues (in addition to the MOP): Gorrochategui et al. (https://www.sciencedirect.com/science/article/pii/S0165993616300425), section 3.2.5 on data intensity normalization. Li et al. (https://academic.oup.com/nar/article/45/W1/W162/3835313) for description of normalization tools.

Briefly, metabolic data are senitive to drift and batch effects, especially in long-term studies. To cope with these issues, studies usually use quality control samples throughout the project for assessing the batches, and internal standards and/or quality control metabolites for normalization. Notable methods here are RUV and CCMN. Downstream normalization is used to either deal with the heteroscedasticity of the metabolite-level data (e.g., Pareto normalization) or to account for sample-to-sample variation (MSTUS, Quantile, Median, etc.).

Load the parsed meta-data (from the cloud), required for all analyses presented here.

```
system(paste("~/google-cloud-sdk/bin/gsutil",
             "cp gs://bic_data_analysis/pass1a/pheno_dmaqc/merged_dmaqc_data.RData",
             "."))
load("merged_dmaqc_data.RData")
system("rm merged_dmaqc_data.RData")

# load required libraries
library(ggplot2);library(reshape2);library(gridExtra)

# load our helper functions
source("https://raw.githubusercontent.com/david-dd-amar/motrpac/master/tools/preprocessing_helper_functi
```

1 Untrageted data from Broad

Unfortunately, as of June 2019, we do not have batch or qc metrics info with this submission. Load the data:

```
broad_dir = "/Users/David/Desktop/MoTrPAC/data/pass_1a/metabolomics/broad_untargeted/"
data_matrix_file = "broad_pass1a_combined_wide.txt"
raw_data_broad = read.delim(paste(broad_dir,data_matrix_file,sep=""),check.names = F,
                            stringsAsFactors = F)
sample_info_file = "broad_pass1a_sampleType.txt"
sample_info = read.delim(paste(broad_dir,sample_info_file,sep=""),check.names = F,
                            stringsAsFactors = F)
# get the samples data using the vial ids:
broad_meta = merged_dmaqc_data[is.element(
  set=colnames(raw_data_broad),merged_dmaqc_data$viallabel),]
rownames(broad_meta) = broad_meta$viallabel
print("Broad untargeted data loaded, the represented samples are:")
```

```
## [1] "Broad untargeted data loaded, the represented samples are:"
```

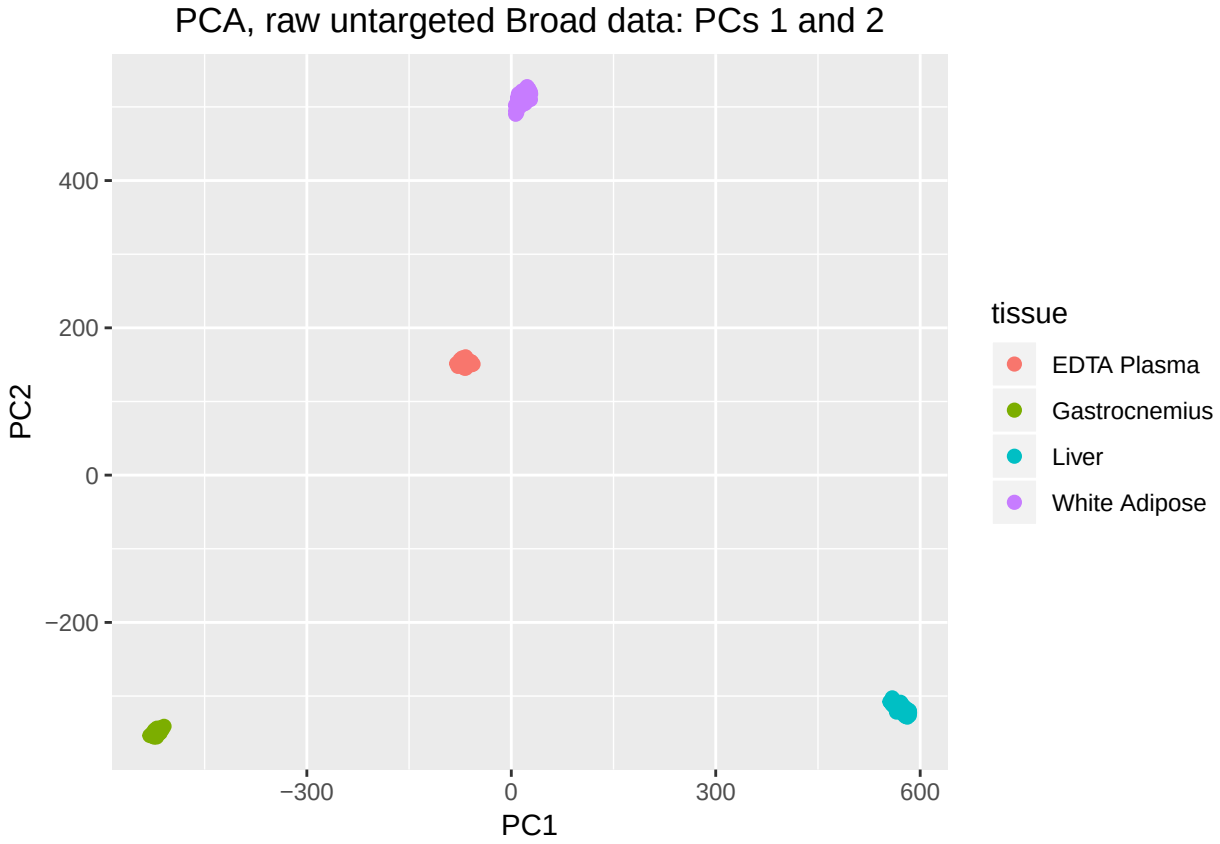
```
print(table(broad_meta$specimen.processing.sampletypedescription))
```

```
##
##   EDTA Plasma Gastrocnemius         Liver White Adipose
##            78            78            78            78
```

1.1 PCA plots

We start with the most basic analysis: a simple PCA plot, colored by tissue.

```
raw_data_for_pca = raw_data_broad[,rownames(broad_meta)]
raw_data_for_pca = log(raw_data_for_pca+1,base=2)
raw_data_for_pca[is.na(raw_data_for_pca)] = 0
raw_broad_data_pca = prcomp(t(raw_data_for_pca))
raw_broad_data_pca = raw_broad_data_pca$x
df = data.frame(raw_broad_data_pca[,1:10],
                tissue = broad_meta[,"specimen.processing.sampletypedescription"])
ggplot(df,aes(x=PC1, y=PC2, color=tissue)) +
  geom_point(size=2) + ggtitle("PCA, raw untargeted Broad data: PCs 1 and 2") +
  theme(plot.title = element_text(hjust = 0.5))
```



1.2 Sanity check: abundance data distribution

Here we go over the data from each tissue, ignoring the reference/control samples for now (as we do not have the batch data). We treat NAs as unmeasured metabolites (a value of 0), and remove rows with zero variance.

```
par(mfrow=c(1,2))
tissue2filtered_data = list()
# bxplots = list()
for (tissue in unique(broad_meta$sampletypedescription)){
  print(paste("--- Analyzing data of tissue:",tissue))
  curr_vialids = as.character(
    broad_meta$viallabel[broad_meta$sampletypedescription==tissue])
```

```
  tissue_data = raw_data_broad[,curr_vialids]
  print(paste("numnber of NAs (zeroed)",sum(is.na(tissue_data))))
  tissue_data[is.na(tissue_data)] = 0
  tissue_data = log(tissue_data+1,base=2)
  print(paste("excluded rows with zero variance",sum(apply(tissue_data,1,sd)==0)))
  tissue_data = tissue_data[!apply(tissue_data==0,1,all),]

  boxplot(tissue_data[,1:10],main=tissue,names=NULL,labels=NULL)
  tissue_data = run_quantile_normalization(tissue_data)
  tissue2filtered_data[[tissue]] = tissue_data

}
```

1.3 Correlations with meta/pheno data

1.4 Differential analysis