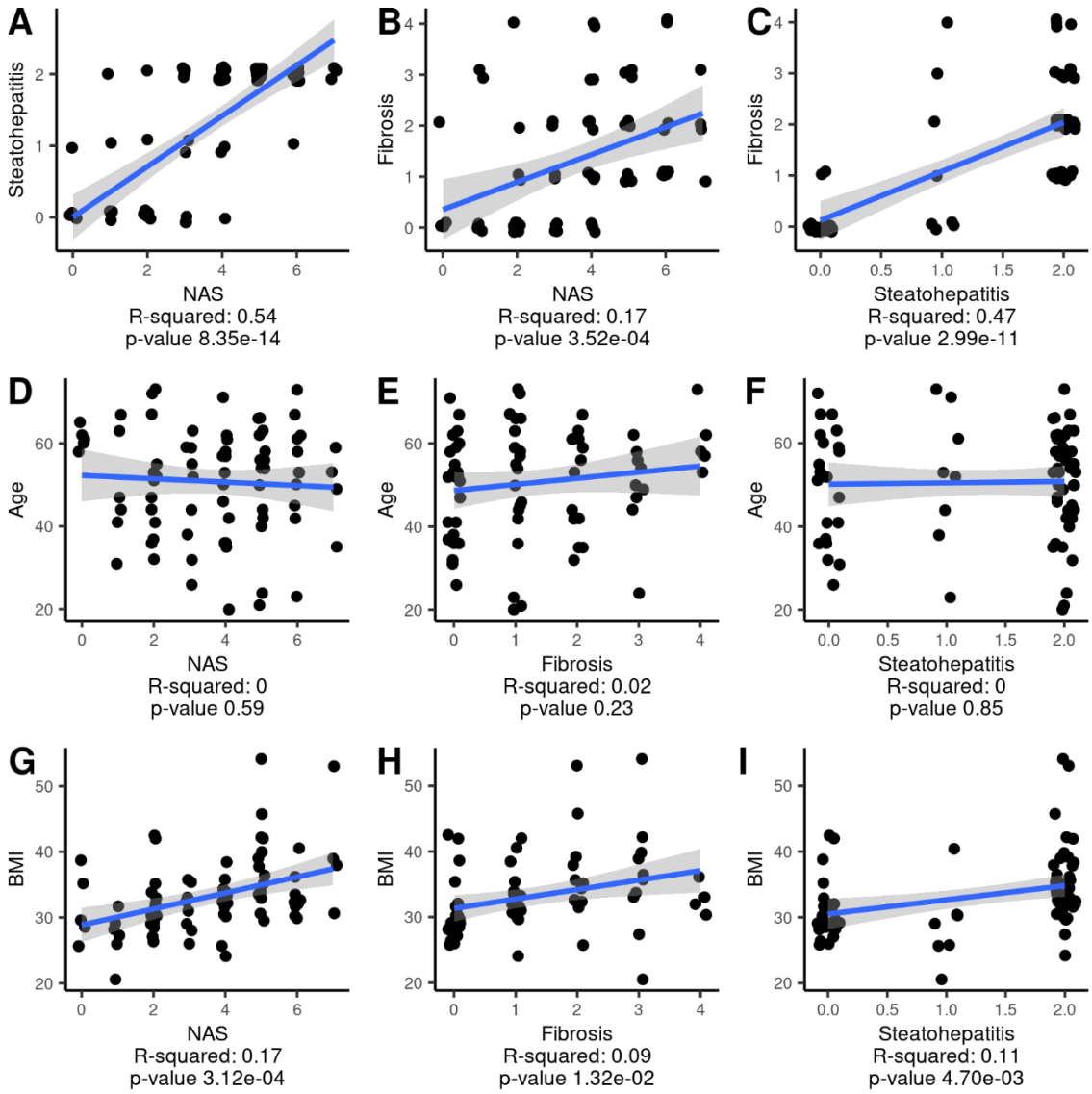
SUPPLEMENTAL FIGURE



SUPPLEMENTAL Figure S1: Correlation between the 3 types of liver disease (A-C), and regressions between disease phenotypes and age (D-F) and BMI (G-I).

CODE

```
#################### Initial Stuff ##################################################
library(readxl)
library(org.Hs.eg.db)
library(annotate)
library(ggplot2)
library(ggpubr)
library(grid)
library(gridExtra)
library(ggrepel)
library(dplyr)
```

```
#library(plotly)
#library(rgl)
#library(plot3D)
library(car)
library(e1071)
# library(grid)
# library(gridExtra)
# library(ggrepel)

#setwd("directory/NASH")

#NASH_RNAseq_data_gene_counts <- read.delim("Directory/NASH_RNAseq_data_gene_counts.txt")

#NASH_RNAseq_data_sample_list <- read.delim("Directory/NASH_RNAseq_data_sample_list.txt")
NASH_RNAseq_data_sample_list <- read.delim("directory/NASH_RNAseq_data_sample_list.txt")

#Lilly_Biopsy_Phenotype <- read_excel("Directory/Lilly Biopsy Phenotype.xlsx")
#phenotypes_pruned <- read.csv("Directory/phenotypes_pruned.csv")
phenotypes_pruned <- read.csv("directory/phenotypes_pruned.csv", stringsAsFactors=FALSE)

#`Transporters.and.Pharmacogenes.(Expanded)` <- read.csv("Directory/Transporters and Pharmacogenes
(Expanded).csv", header=FALSE)
#`Transporters.and.Pharmacogenes.(Core list)` <- read.csv("Directory/Transporters and Pharmacogenes (Core
list).csv", header=FALSE)
`Transporters.and.Pharmacogenes.(Expanded)` <- read.csv("directory/Transporters and Pharmacogenes
(Expanded).csv", header=FALSE)

############# Convert Entrez to Symbol & Match IDs
#####################################################################

a <- phenotypes_pruned
colnames(a)[1] <- "study_ID"
a <- merge(a,NASH_RNAseq_data_sample_list,by="study_ID")
rm(phenotypes_pruned,NASH_RNAseq_data_sample_list)
a$ID_in_data <- gsub("-","\\.",a$ID_in_data)
row.names(a) <- a$ID_in_data
a <- a[-c(1,12)]

# b <- NASH_RNAseq_data_gene_counts
# b$EntrezGene <- as.character(b$EntrezGene)
# b$EntrezGene <- getSYMBOL(b$EntrezGene,data='org.Hs.eg') #540 NA values created
# colnames(b)[1] <- "Gene"
# rm(NASH_RNAseq_data_gene_counts)
#
# b_raw <- b
# b_sum <- b[1,]
# b_sum[1,1] <- "total"
#
# for(n in 2:ncol(b_sum)){
#   b_sum[1,n] <- sum(b_raw[,n]) #adds up sum of gene reads per sample
# }
#
# for(n in 2:ncol(b)){
```

```
#   print(n)
#   for(i in 1:nrow(b)){
#     b[i,n] <- (b_raw[i,n]/b_sum[1,n])*1000000 #normalizes reads to sum of all reads per sample
#   }
# }
#
# b <- subset(b,b$Gene!="NA")
# row.names(b) <- b$Gene
# b <- b[-c(1)]
# b <- log(b) #log
# write.table(b,"NASH_RNAseq_data_logTPM_norm.txt")
#b <- read.csv("Directory/data frame ouputs/NASH_RNAseq_data_logTPM_norm.txt", sep="")
b <- read.csv("directory/NASH_RNAseq_data_logTPM_norm.txt", row.names=1, sep="", stringsAsFactors=FALSE)

############## Filter to pharmacogenes
#########################################################################################################
#####

c <- `Transporters.and.Pharmacogenes.(Expanded)`
rm(`Transporters.and.Pharmacogenes.(Expanded)`)
c <- c$V2
b1 <- subset(b,rownames(b) %in% c)

############## Prep data for analysis ################################################

b2 <- as.data.frame(t(b1))
#for values of zero, log(0) returns -Inf... these are still good data, so change to lowest value across all samples
#this is a conservative approach

#b2[b2=="-Inf"] <- NA

for(n in 1:ncol(b2)){
  x <- min(b2[,n][is.finite(b2[,n])])
  y <- "-Inf"
  for(i in 1:nrow(b2)){
    b2[i,n][b2[i,n] %in% y] <- x
  }
}

#write.csv(b2,file = "NASH_RNAseq_data_logTPM_adme_repInf.csv",row.names = T,quote = F)

d <- merge(a,b2,by=0)
p.gene.lm <- as.data.frame(row.names(b1))
colnames(p.gene.lm) <- "gene"
c <- row.names(b1)

################## Regressions for NAS ###################################
for(n in 1:nrow(p.gene.lm)){
  x <- c[n]
  #d1 <- d[c("NAS_score",x)]
  d1 <- d[c("nas",x)]
  colnames(d1) <- c("NAS","RNA")
  fit <- summary(lm(RNA~NAS,data=d1))
```

```
  p.gene.lm[n,2] <- x
  p.gene.lm[n,3] <- fit$coefficients[2]
  p.gene.lm[n,4] <- fit$coefficients[2,4]
  p.gene.lm[n,5] <- fit$r.squared
}
p.gene.lm <- p.gene.lm[-c(1)]
colnames(p.gene.lm) <- c("gene","slope","p_value","r_squared")
p.gene.lm$p_value_bonferonni <- p.gene.lm$p_value*(length(c))
p.gene.lm <- p.gene.lm[order(p.gene.lm$p_value_bonferonni),]
row.names(p.gene.lm) <- NULL

t1 <- subset(p.gene.lm, p.gene.lm[5]<=0.05)
t1$group <- "NAS Regression"
write.csv(p.gene.lm,"pharmacogene_NAS_regression_results_logTPM.csv")

############## Plot regressions ######################################################

regr <- function(n){
  x <- p.gene.lm$gene[n]
  d1 <- d[c("nas",x)]
  colnames(d1) <- c("NAS","RNA")
  fit <- summary(lm(RNA~NAS,data=d1))
  ggplot()+geom_point(data=d1,aes(x=NAS,y=RNA),size=1)+
    geom_smooth(data=d1,aes(x=NAS,y=RNA),method=lm)+
    ylab(paste0(x))+
    xlab(NULL)+
    #xlab("NAFLD Activity Score (NAS)")+
    theme(text = element_text(size=10),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          panel.background=element_blank(),
          axis.line=element_line(colour="black"))
}


nas <- ggarrange(regr(1),regr(2),regr(3),regr(4),regr(5),regr(6),regr(7),regr(8),regr(9),
          regr(10),regr(11),regr(12),regr(13),regr(14),
          nrow = 2,
          ncol = 7
          #labels = "AUTO"
          )
nas <- annotate_figure(nas, top = textGrob(" "), bottom = textGrob("NAFLD Activity Score (0-8)", gp = gpar(cex =
1)),fill_palette(NULL))
nas
#ggsave("regression_plots_logTPM_NAS.tiff",plot = last_plot(),width = 8,height = 8)
#write.csv()

############## S-plot ############################################
p.gene.lm$label <- p.gene.lm$gene
p.gene.lm$BH <- p.adjust(p.gene.lm$p_value,method = "BH")
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value_bonferonni)

nas_s <- ggplot()+geom_point(data=p.gene.lm,
```

```
            mapping=aes(y=p_value_bonferonni,x=reorder(gene,p_value_bonferonni),
                  color=slope),alpha=0.8,size=1.5)+
scale_color_gradientn(colours=c("firebrick","goldenrod","dodgerblue")
                #limits=c(0,0.05),
                #breaks=c(0,0.025,0.05),
                #na.value="grey"
                )+
geom_hline(yintercept = 0.05,
        linetype="dashed",
        color="black")+
geom_hline(yintercept = BHlim,
        linetype="dashed",
        color="grey50")+
geom_hline(yintercept = .05*length(c),
        linetype="dashed",
        color="grey80")+
theme(axis.text.x = element_blank(),
    panel.background = element_blank(),
    axis.line = element_line(size=0.5),
    axis.ticks.y = element_blank(),
    axis.ticks.x = element_blank(),
    legend.title = element_text(size=8),
    legend.text = element_text(size=8),
    axis.title = element_text(size=10),
    plot.title = element_text(hjust = 0.5)
    #legend.position = "top"
    )+
#scale_y_continuous(breaks = seq(-7,7,by=2))+
#ggtitle("NAFLD Activity Score (0-8)")+
xlab("Gene")+ylab("Bonferroni Corrected\n P-value")+
geom_label_repel(data=subset(p.gene.lm, p.gene.lm$p_value_bonferonni<0.05 |
                      p.gene.lm$gene=="CYP1A2" |
                      p.gene.lm$gene=="CYP3A4" |
                      p.gene.lm$gene=="CYP2E1" |
                      p.gene.lm$gene=="CYP2C19" |
                      p.gene.lm$gene=="CYP2D6" |
                      p.gene.lm$gene=="CYP2B6" |
                      p.gene.lm$gene=="CYP2C9" |
                      p.gene.lm$gene=="CYP3A5" |
                      p.gene.lm$gene=="CYP2C8"
                      ),
          aes(y=p_value_bonferonni,x=gene,label=gene),
          nudge_x = 50,
          size=3,
          nudge_y = -1.5,
          max.overlaps = 20,
          #box.padding = 0.3,
          #point.padding = 0.5,
          #force = 200
          min.segment.length = 0)+
scale_y_log10()+
geom_text(aes(200,0.03,
         label="BONFERRONI SIGNIFICANCE THRESHOLD"),
```

```
            size=2.5)+
  geom_text(aes(2,BHlim+2,
            label="BH THRESHOLD"),
            color="grey50",size=2.5,hjust=0)+
  geom_text(aes(2,(.05*length(c))+10,
            label="UNADJUSTED ALPHA"),
          color="grey80",size=2.5,hjust=0)

nas_s
#ggsave("p_value_s_plot_logTPM_NAS.tiff",plot = last_plot(),width = 8,height = 4)

plot <- ggarrange(nas,nas_s,
        nrow = 2,
        ncol = 1,
        labels = "AUTO"
        #heights = c(5, 6)
)
plot
ggsave("Figure1.tiff",plot = plot,width = 8,height = 6,bg = "white")

############## volcano ##############
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value)
nas_volc <- ggplot()+geom_point(data=p.gene.lm,
                      mapping=aes(y=-log10(p_value),x=slope,
                              color=-log10(p_value)),
                      alpha=0.8,
                      size=3)+
  scale_color_continuous(limits=c(-log10(BHlim),10))+
  geom_hline(yintercept = -log10(0.05),
           linetype="dashed",
           color="grey80")+
  geom_hline(yintercept = -log10(BHlim),
           linetype="dashed",
           color="grey50")+
  geom_hline(yintercept = -log10(.05/length(c)),
           linetype="dashed",
           color="black")+
  geom_text(aes(min(p.gene.lm$slope),-log10(.05/length(c))+.15,
            label="BONFERRONI SIGNIFICANCE THRESHOLD"),
          size=2.5,hjust=0)+
  geom_text(aes(min(p.gene.lm$slope),-log10(BHlim)+.15,
            label="BH THRESHOLD"),
          color="grey50",size=2.5,hjust=0)+
  geom_text(aes(min(p.gene.lm$slope),-log10(0.05)+.15,
            label="UNADJUSTED ALPHA"),
          color="grey80",size=2.5,hjust=0)+
  theme(panel.background = element_blank(),
      axis.line = element_line(size=0.5),
      legend.position = "none"
      )+
  scale_y_continuous(expand = c(.02,0),limits = c(0,10))+
  scale_x_continuous(expand = c(.02,0))+
  #ggtitle("NAFLD Activity Score")+
```

```
 xlab("Slope")+ylab("-Log10 P-value")+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope<0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = -.20,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = 10,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope>0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .20,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = 5,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>=.05 & BH<.05 &
slope<0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = -.15,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = 3.5,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>=.05 & BH<.05 &
slope>0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .20,
```

```
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = 1.5,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>=.05 & slope<0 & gene %in%
                                       c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                         "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                      ,gene,"")),
                     aes(y=-log10(p_value),x=slope,label=label),
                     #xlim = c(.2,.4),
                     #ylim = c(7,10),
                     nudge_x = -.20,
                     size=2.5,
                     #color="black",
                     segment.color = "grey80",
                     nudge_y = -1.2,
                     max.overlaps = 20,
                     #box.padding = 1,
                     #point.padding = 0.5,
                     #force_pull = 2,
                     #force = 200,
                     min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>=.05 & slope>0 & gene %in%
                                       c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                         "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                      ,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .25,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = -.75,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)

nas_volc

#ggsave("Supp_Figure1.tiff",plot = nas_volc,width = 8,height = 6,bg = "white")

plot1 <- ggarrange(nas,nas_volc,
           nrow = 2,
```

```
            ncol = 1,
            labels = "AUTO",
            heights = c(4, 6)
)
#plot1
ggsave("Figure1_alt.tiff",plot = plot1,width = 8,height = 8,bg = "white")

############## T-test ###########################################
d2 <- subset(d,d$nas>4)
d3 <- subset(d,d$nas<5)
tnas <- data.frame()
for(n in 1:length(c)){
  x <- c[[n]]
  xx <- t.test(d2[[x]],d3[[x]])
  tnas[n,1] <- x
  tnas[n,2] <- xx[["p.value"]]
}
tnas$V3 <- tnas$V2*nrow(tnas)
tnassig <- subset(tnas,tnas$V3<0.05)
tnassig <- tnassig[order(tnassig$V3),]
row.names(tnassig) <- tnassig$V1
tnassig <- tnassig[-c(1)]

d4 <- subset(d,select=c("nas",row.names(tnassig)))
d4$group <- d4$nas>4
d4$group <- gsub(TRUE,"NAS \u2265 5",d4$group)
d4$group <- gsub(FALSE,"NAS \u2264 4",d4$group)

d5 <- reshape2::melt(d4,c("nas","group"))
d5$variable <- factor(d5$variable)

t_nas <- ggplot()+
  geom_point(data=d5,aes(x=variable,y=value,fill=group),
          position = position_jitterdodge(jitter.width = 0.15,dodge.width = .75),size=0.3)+
  geom_boxplot(data=d5,aes(x=variable,y=value,fill=group),outlier.shape = NA,alpha=0.9)+
  theme(
    axis.text.x = element_text(angle = 90,size=10,vjust = .5),
    axis.title.y = element_text(size=10),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_blank(),
    legend.title = element_blank(),
    axis.line = element_line(),
    legend.position = "top")+
  ylab("Log Transcripts Per Million")
t_nas

tnassig$gene <- rownames(tnassig)
tnassig[4:6] <- NA
tnassig <- tnassig[c(3,4,1,5,2,6)]
colnames(tnassig) <- colnames(t1)
tnassig$group <- "NAS T-test"
```

```
table1 <- rbind(t1,tnassig)

ggsave("t_tests_NAS_logTPM.tiff",plot = tt,width = 4,height = 4)


################### Regressions for fibrosis ###################################
p.gene.lm <- as.data.frame(row.names(b1))
colnames(p.gene.lm) <- "gene"
c <- row.names(b1)

for(n in 1:nrow(p.gene.lm)){
  x <- c[n]
  d1 <- d[c("fibrosis",x)]
  colnames(d1) <- c("fibrosis","RNA")
  fit <- summary(lm(RNA~fibrosis,data=d1))
  p.gene.lm[n,2] <- x
  p.gene.lm[n,3] <- fit$coefficients[2]
  p.gene.lm[n,4] <- fit$coefficients[2,4]
  p.gene.lm[n,5] <- fit$r.squared
}
p.gene.lm <- p.gene.lm[-c(1)]
colnames(p.gene.lm) <- c("gene","slope","p_value","r_squared")
p.gene.lm$p_value_bonferonni <- p.gene.lm$p_value*(length(c))
p.gene.lm <- p.gene.lm[order(p.gene.lm$p_value_bonferonni),]
row.names(p.gene.lm) <- NULL

t2 <- subset(p.gene.lm, p.gene.lm[5]<=0.05)
t2$group <- "Fibrosis Regression"
table1 <- rbind(table1,t2)
write.csv(p.gene.lm,"pharmacogene_fibrosis_regression_results_logTPM.csv")

############## Plot regressions ############################################################
regr <- function(n){
  x <- p.gene.lm$gene[n]
  d1 <- d[c("fibrosis",x)]
  colnames(d1) <- c("fibrosis","RNA")
  fit <- summary(lm(RNA~fibrosis,data=d1))
  ggplot()+geom_point(data=d1,aes(y=RNA,x=fibrosis))+
    geom_smooth(data=d1,aes(y=RNA,x=fibrosis),method=lm)+
    ylab(paste0(x))+
    xlab(NULL)+
    #xlab(paste0("Fibrosis (0-4)","\nR-squared: ",round(fit$r.squared,2)))+
    theme(text = element_text(size=10),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          panel.background=element_blank(),
          axis.line=element_line(colour="black"))
}

list <- list()
psig <- subset(p.gene.lm,p.gene.lm$p_value_bonferonni<0.05)
for(n in 1:nrow(psig)){
```

```
  A <- regr(n)
  list[[n]] <- A
}

#write.table(psig,"fibrosis_hits_regression.txt")

fib <- ggarrange(plotlist = list,ncol=6,nrow=4)
fib <- annotate_figure(fib, top = textGrob(" "), bottom = textGrob("Fibrosis Grade (0-4)", gp = gpar(cex =
1)),fill_palette(NULL))
fib

#ggsave("regression_plots_fibrosis_logTPM.tiff",plot = last_plot(),width = 8,height = 8)

############## S-plot ################################################
p.gene.lm$label <- p.gene.lm$gene
p.gene.lm$BH <- p.adjust(p.gene.lm$p_value,method = "BH")
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value_bonferonni)

s_fib <- ggplot()+geom_point(data=p.gene.lm,
              mapping=aes(y=p_value_bonferonni,x=reorder(gene,p_value_bonferonni),
                    color=slope),alpha=0.8,size=1.5)+
  scale_color_gradientn(colours=c("firebrick","goldenrod","dodgerblue")
                  #limits=c(0,0.05),
                  #breaks=c(0,0.025,0.05),
                  #na.value="grey"
  )+
  geom_hline(yintercept = 0.05,
          linetype="dashed",
          color="black")+
  geom_hline(yintercept = BHlim,
          linetype="dashed",
          color="grey50")+
  geom_hline(yintercept = .05*length(c),
          linetype="dashed",
          color="grey80")+
  theme(axis.text.x = element_blank(),
      panel.background = element_blank(),
      axis.line = element_line(size=0.5),
      axis.ticks.y = element_blank(),
      axis.ticks.x = element_blank(),
      legend.title = element_text(size=8),
      legend.text = element_text(size=8),
      axis.title = element_text(size=10))+
  #scale_y_continuous(breaks = seq(-7,7,by=2))+
  #ggtitle("Fibrosis")+
  xlab("Gene")+ylab("Bonferroni Corrected\nP-value")+
  geom_label_repel(data=subset(p.gene.lm, p.gene.lm$p_value_bonferonni<0.05 |
                         p.gene.lm$gene=="CYP1A2" |
                         p.gene.lm$gene=="CYP3A4" |
                         p.gene.lm$gene=="CYP2E1" |
                         p.gene.lm$gene=="CYP2C19" |
                         p.gene.lm$gene=="CYP2D6" |
                         p.gene.lm$gene=="CYP2B6" |
```

```
                    p.gene.lm$gene=="CYP2C9" |
                    p.gene.lm$gene=="CYP3A5" |
                    p.gene.lm$gene=="CYP2C8"
  ),
  aes(y=p_value_bonferonni,x=gene,label=gene),
  nudge_x = 50,
  size=3,
  nudge_y = -1.5,
  #box.padding = 0.3,
  #point.padding = 0.5,
  #force = 200
  min.segment.length = 0,
  max.overlaps = 20)+
  scale_y_log10()+
  geom_text(aes(200,0.03,
            label="BONFERRONI SIGNIFICANCE THRESHOLD"),
          size=2.5)+
  geom_text(aes(2,BHlim+2,
            label="BH THRESHOLD"),
          color="grey50",size=2.5,hjust=0)+
  geom_text(aes(2,(.05*length(c))+10,
            label="UNADJUSTED ALPHA"),
          color="grey80",size=2.5,hjust=0)

plot <- ggarrange(fib,s_fib,
              nrow = 2,
              ncol = 1,
              labels = "AUTO"
              #heights = c(5, 6)
)
plot
ggsave("Figure3.tiff",plot = plot,width = 8,height = 8,bg = "white")

############## volcano ##############
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value)
fib_volc <- ggplot()+geom_point(data=p.gene.lm,
                        mapping=aes(y=-log10(p_value),x=slope,
                                color=-log10(p_value)),
                        alpha=0.8,
                        size=3)+
  scale_color_continuous(limits=c(-log10(BHlim),10))+
  geom_hline(yintercept = -log10(0.05),
           linetype="dashed",
           color="grey80")+
  geom_hline(yintercept = -log10(BHlim),
           linetype="dashed",
           color="grey50")+
  geom_hline(yintercept = -log10(.05/length(c)),
           linetype="dashed",
           color="black")+
  geom_text(aes(min(p.gene.lm$slope),-log10(.05/length(c))+.15,
            label="BONFERRONI SIGNIFICANCE THRESHOLD"),
          size=2.5,hjust=0)+
```

```
 geom_text(aes(min(p.gene.lm$slope),-log10(BHlim)+.15,
           label="BH THRESHOLD"),
         color="grey50",size=2.5,hjust=0)+
 geom_text(aes(min(p.gene.lm$slope),-log10(0.05)+.15,
           label="UNADJUSTED ALPHA"),
         color="grey80",size=2.5,hjust=0)+
 theme(panel.background = element_blank(),
     axis.line = element_line(size=0.5),
     legend.position = "none"
 )+
 scale_y_continuous(expand = c(.02,0),limits = c(0,9))+
 scale_x_continuous(expand = c(.05,0))+
 #ggtitle("NAFLD Activity Score")+
 xlab("Slope")+ylab("-Log10 P-value")+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope<0,gene,"")),
              aes(y=-log10(p_value),x=slope,label=label),
              #xlim = c(.2,.4),
              #ylim = c(7,10),
              nudge_x = 0,
              size=2.5,
              #color="black",
              segment.color = "grey80",
              nudge_y = 5.5,
              max.overlaps = 20,
              box.padding = .4,
              #point.padding = 0.5,
              #force_pull = 2,
              #force = 200,
              min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope>0,gene,"")),
              aes(y=-log10(p_value),x=slope,label=label),
              xlim = c(0,.4),
              ylim = c(6,9),
              #nudge_x = .20,
              size=2.5,
              #color="black",
              segment.color = "grey80",
              nudge_y = 2,
              max.overlaps = 20,
              #box.padding = 1,
              #point.padding = 0.5,
              #force_pull = 2,
              #force = 200,
              min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>=.05 & BH<.015 &
slope<0,gene,"")),
              aes(y=-log10(p_value),x=slope,label=label),
              xlim = c(min(p.gene.lm$slope),0),
              ylim = c(4,7),
              nudge_x = -.15,
              size=2.5,
              #color="black",
              segment.color = "grey80",
```

```
            nudge_y = 2,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>=.015 & BH<=.05 & slope<0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            xlim = c(min(p.gene.lm$slope),-.1),
            ylim = c(2,3.7),
            #nudge_x = -.1,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = .2,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>=.05 & BH<.05 &
slope>0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .10,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            #nudge_y = 1,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
 geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>=.05 & slope<0 & gene %in%
                                      c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                        "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                     ,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            ylim = c(0,1.2),
            nudge_x = -.20,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            #nudge_y = -1.2,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
```

```
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
  geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>=.05 & slope>0 & gene %in%
                                        c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                          "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                       ,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .25,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = -.75,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)

fib_volc

#ggsave("Supp_Figure2.tiff",plot = fib_volc,width = 8,height = 6,bg = "white")

plot1 <- ggarrange(fib,fib_volc,
            nrow = 2,
            ncol = 1,
            labels = "AUTO",
            heights = c(4, 6)
)
#plot1
ggsave("Figure3_alt.tiff",plot = plot1,width = 8,height = 8,bg = "white")

############## T-test 0 vs rest ##############################################
d2 <- subset(d,d$fibrosis==0)
d3 <- subset(d,d$fibrosis!=0)
tnas <- data.frame()
for(n in 1:length(c)){
  x <- c[[n]]
  xx <- t.test(d2[[x]],d3[[x]])
  tnas[n,1] <- x
  tnas[n,2] <- xx[["p.value"]]
}
tnas$V3 <- tnas$V2*nrow(tnas)
tnassig <- subset(tnas,tnas$V3<0.05)
tnassig <- tnassig[order(tnassig$V3),]
row.names(tnassig) <- tnassig$V1
tnassig <- tnassig[-c(1)]
d4 <- subset(d,select=c("fibrosis",row.names(tnassig)))
d4$group <- d4$fibrosis==0
d4$group <- gsub(TRUE,"Fibrosis = 0",d4$group)
```

```
d4$group <- gsub(FALSE,"Fibrosis \u2265 1",d4$group)

d5 <- reshape2::melt(d4,c("fibrosis","group"))
d5$variable <- factor(d5$variable)
d5 <- subset(d5,!is.na(d5$fibrosis))

bplot <- ggplot()+
  geom_point(data=d5,aes(x=variable,y=value,fill=group),
          position = position_jitterdodge(jitter.width = 0.15,dodge.width = 0.75),size=0.3)+
  geom_boxplot(data=d5,aes(x=variable,y=value,fill=group),outlier.shape = NA,alpha=0.9)+
  theme(
    axis.text.x = element_text(angle = 90,size=10,vjust = 0.5),
    axis.title.y = element_text(size=10),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank(),
    axis.line = element_line(),
    #panel.grid.minor = element_blank(),
    panel.background = element_blank(),
    legend.title = element_blank(),
    legend.position = "top")+
  ylab("Log Transcripts Per Million")
#ggsave("t_tests_fibrosis_0_rest_logTPM.tiff",plot = last_plot(),width = 4,height = 4)
row1 <- ggarrange(t_nas,bplot,labels = "AUTO",ncol=2,widths = c(15,8))

tnassig$gene <- rownames(tnassig)
tnassig[4:6] <- NA
tnassig <- tnassig[c(3,4,1,5,2,6)]
colnames(tnassig) <- colnames(t1)
tnassig$group <- "fibrosis 0 vs. rest T-test"

table1 <- rbind(table1,tnassig)

############## T-test 4 vs rest ##############################################
d2 <- subset(d,d$fibrosis==4)
d3 <- subset(d,d$fibrosis!=4)
tnas <- data.frame()
for(n in 1:length(c)){
  x <- c[[n]]
  xx <- t.test(d2[[x]],d3[[x]])
  tnas[n,1] <- x
  tnas[n,2] <- xx[["p.value"]]
}
tnas$V3 <- tnas$V2*nrow(tnas)
tnassig <- subset(tnas,tnas$V3<0.05)
tnassig <- tnassig[order(tnassig$V3),]
row.names(tnassig) <- tnassig$V1
tnassig <- tnassig[-c(1)]
d4 <- subset(d,select=c("fibrosis",row.names(tnassig)))
d4$group <- d4$fibrosis==4
d4$group <- gsub(TRUE,"Fibrosis = 4",d4$group)
d4$group <- gsub(FALSE,"Fibrosis \u2264 3",d4$group)

d5 <- reshape2::melt(d4,c("fibrosis","group"))
```

```
d5$variable <- factor(d5$variable)
d5$group <- factor(d5$group,levels=c("Fibrosis \u2264 3","Fibrosis = 4"))
d5 <- subset(d5,!is.na(d5$fibrosis))

cplot <- ggplot()+
  geom_point(data=d5,aes(x=variable,y=value,fill=group),
          position = position_jitterdodge(jitter.width = 0.15,dodge.width = 0.75),size=0.3)+
  geom_boxplot(data=d5,aes(x=variable,y=value,fill=group),outlier.shape = NA,alpha=0.9)+
  theme(
    axis.text.x = element_text(angle = 90,size=10,vjust = 0.5),
    axis.title.y = element_text(size=10),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank(),
    axis.line = element_line(),
    panel.grid.minor = element_blank(),
    panel.background = element_blank(),
    legend.title = element_blank(),
    legend.position = "top")+
  guides(fill=guide_legend(nrow = 2))+
  ylab("Log Transcripts Per Million")
cplot
#ggsave("t_tests_fibrosis_4_rest_logTPM.tiff",plot = last_plot(),width=8,height=4)

tnassig$gene <- rownames(tnassig)
tnassig[4:6] <- NA
tnassig <- tnassig[c(3,4,1,5,2,6)]
colnames(tnassig) <- colnames(t1)
tnassig$group <- "Fibrosis 4 vs. rest T-test"

table1 <- rbind(table1,tnassig)
############## T-test 0+1 vs rest ##############################################
d2 <- subset(d,d$fibrosis<2)
d3 <- subset(d,d$fibrosis>1 | is.na(d$fibrosis))
tnas <- data.frame()
for(n in 1:length(c)){
  x <- c[[n]]
  xx <- t.test(d2[[x]],d3[[x]])
  tnas[n,1] <- x
  tnas[n,2] <- xx[["p.value"]]
}
tnas$V3 <- tnas$V2*nrow(tnas)
tnassig <- subset(tnas,tnas$V3<0.05)
tnassig <- tnassig[order(tnassig$V3),]
row.names(tnassig) <- tnassig$V1
tnassig <- tnassig[-c(1)]
d4 <- subset(d,select=c("fibrosis",row.names(tnassig)))
d4$group <- d4$fibrosis<2
d4$group <- gsub(TRUE,"Fibrosis \u2264 1",d4$group)
d4$group <- gsub(FALSE,"Fibrosis \u2265 2",d4$group)

d5 <- reshape2::melt(d4,c("fibrosis","group"))
d5$variable <- factor(d5$variable)
d5 <- subset(d5,!is.na(d5$fibrosis))
```

```
dplot <- ggplot()+
  geom_point(data=d5,aes(x=variable,y=value,fill=group),
             position = position_jitterdodge(jitter.width = 0.15,dodge.width = 0.75),size=0.3)+
  geom_boxplot(data=d5,aes(x=variable,y=value,fill=group),outlier.shape = NA,alpha=0.9)+
  theme(
    axis.text.x = element_text(angle = 90,size=10,vjust = 0.5),
    axis.title.y = element_text(size=10),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_blank(),
    legend.title = element_blank(),
    legend.position = "top",
    axis.line = element_line())+
  ylab("Log Transcripts Per Million")
#ggsave("t_tests_fibrosis_0and1_rest_logTPM.tiff",plot = last_plot(),width=8,height=4)

tnassig$gene <- rownames(tnassig)
tnassig[4:6] <- NA
tnassig <- tnassig[c(3,4,1,5,2,6)]
colnames(tnassig) <- colnames(t1)
tnassig$group <- "Fibrosis 0+1 vs rest T-test"

table1 <- rbind(table1,tnassig)

################### Regressions for steatohepatitis ####################################
p.gene.lm <- as.data.frame(row.names(b1))
colnames(p.gene.lm) <- "gene"
c <- row.names(b1)
dd <- subset(d,!is.na(d$steatohep3))


for(n in 1:nrow(p.gene.lm)){
  x <- c[n]
  d1 <- dd[c("steatohep3",x)]
  colnames(d1) <- c("steatohep","RNA")
  fit <- summary(lm(RNA~steatohep,data=d1))
  p.gene.lm[n,2] <- x
  p.gene.lm[n,3] <- fit$coefficients[2]
  p.gene.lm[n,4] <- fit$coefficients[2,4]
  p.gene.lm[n,5] <- fit$r.squared
}
p.gene.lm <- p.gene.lm[-c(1)]
colnames(p.gene.lm) <- c("gene","slope","p_value","r_squared")
p.gene.lm$p_value_bonferonni <- p.gene.lm$p_value*(length(c))
p.gene.lm <- p.gene.lm[order(p.gene.lm$p_value_bonferonni),]
row.names(p.gene.lm) <- NULL

t3 <- subset(p.gene.lm, p.gene.lm[5]<=0.05)
t3$group <- "Steatohepatitis Regression"
table1 <- rbind(table1,t3)
#write.table(p.gene.lm,"pharmacogene_steatohep_regression_results_logTPM.txt",row.names=FALSE)
```

```
write.csv(p.gene.lm,"pharmacogene_steatohep_regression_results_logTPM.csv")

############## Plot regressions ########################################################
regr <- function(n){
  x <- p.gene.lm$gene[n]
  d1 <- dd[c("steatohep3",x)]
  colnames(d1) <- c("steatohep","RNA")
  fit <- summary(lm(RNA~steatohep,data=d1))
  ggplot()+geom_point(data=d1,aes(y=RNA,x=steatohep))+
    geom_smooth(data=d1,aes(y=RNA,x=steatohep),method=lm)+
    ylab(paste0(x))+
    #xlab(paste0("steatohep","\nR-squared: ",round(fit$r.squared,2)))+
    xlab(NULL)+
    theme(text = element_text(size=10),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(),
          panel.background=element_blank(),
          axis.line=element_line(colour="black")
          )+scale_x_continuous(breaks=c(0,1,2))
}

list <- list()
psig <- subset(p.gene.lm,p.gene.lm$p_value_bonferonni<0.05)
for(n in 1:nrow(psig)){
  A <- regr(n)
  list[[n]] <- A
}

write.csv(psig,"steatohep_hits_regression.csv")

row1 <- ggarrange(plotlist = list,nrow=1)
row1 <- annotate_figure(row1, top = textGrob(" "), bottom = textGrob("Steatohepatitis (0-2)", gp = gpar(cex =
1)),fill_palette(NULL))

#ggsave("regression_plots_steatohep_logTPM.tiff",plot = last_plot(),width = 6,height=6)

############## S-plot ##############################################
p.gene.lm$label <- p.gene.lm$gene
p.gene.lm$BH <- p.adjust(p.gene.lm$p_value,method = "BH")
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value_bonferonni)

row2 <- ggplot()+geom_point(data=p.gene.lm,
                mapping=aes(y=p_value_bonferonni,x=reorder(gene,p_value_bonferonni),
                        color=slope),alpha=0.8,size=1.5)+
  scale_color_gradientn(colours=c("firebrick","goldenrod","dodgerblue")
                    #limits=c(0,0.05),
                    #breaks=c(0,0.025,0.05),
                    #na.value="grey"
  )+
  geom_hline(yintercept = 0.05,
             linetype="dashed",
             color="black")+
  geom_hline(yintercept = BHlim,
```

```
        linetype="dashed",
        color="grey50")+
 geom_hline(yintercept = .05*length(c),
        linetype="dashed",
        color="grey80")+
 theme(axis.text.x = element_blank(),
     panel.background = element_blank(),
     axis.line = element_line(size=0.5),
     axis.ticks.y = element_blank(),
     axis.ticks.x = element_blank(),
     legend.title = element_text(size=8),
     legend.text = element_text(size=8),
     axis.title = element_text(size=10))+
 #scale_y_continuous(breaks = seq(-7,7,by=2))+
 #ggtitle("Steatohepatitis")+
 xlab("Gene")+ylab("Bonferroni Corrected\nP-value")+
 geom_label_repel(data=subset(p.gene.lm, p.gene.lm$p_value_bonferonni<0.05 |
                       p.gene.lm$gene=="CYP1A2" |
                       p.gene.lm$gene=="CYP3A4" |
                       p.gene.lm$gene=="CYP2E1" |
                       p.gene.lm$gene=="CYP2C19" |
                       p.gene.lm$gene=="CYP2D6" |
                       p.gene.lm$gene=="CYP2B6" |
                       p.gene.lm$gene=="CYP2C9" |
                       p.gene.lm$gene=="CYP3A5" |
                       p.gene.lm$gene=="CYP2C8"
 ),
 aes(y=p_value_bonferonni,x=gene,label=gene),
 nudge_x = 50,
 size=3,
 nudge_y = -1.5,
 max.overlaps = 20,
 #box.padding = 0.3,
 #point.padding = 0.5,
 #force = 200
 min.segment.length = 0)+
 scale_y_log10()+
 geom_text(aes(200,0.03,
          label="BONFERRONI SIGNIFICANCE THRESHOLD"),
       size=2.5)+
 geom_text(aes(2,BHlim+2,
          label="BH THRESHOLD"),
       color="grey50",size=2.5,hjust=0)+
 geom_text(aes(2,(.05*length(c))+10,
          label="UNADJUSTED ALPHA"),
       color="grey80",size=2.5,hjust=0)

fig4 <- ggarrange(row1,row2,nrow=2,labels = "AUTO",heights = c(3,5))
ggsave("Figure4.tiff",plot = fig4,width = 8,height = 4,bg="white")

############## volcano ##############
BHlim <- max(subset(p.gene.lm,p.gene.lm$BH<.05)$p_value)
sth_volc <- ggplot()+geom_point(data=p.gene.lm,
```

```
                    mapping=aes(y=-log10(p_value),x=slope,
                            color=-log10(p_value)),
                    alpha=0.8,
                    size=3)+
scale_color_continuous(limits=c(-log10(BHlim),10))+
geom_hline(yintercept = -log10(0.05),
        linetype="dashed",
        color="grey80")+
geom_hline(yintercept = -log10(BHlim),
        linetype="dashed",
        color="grey50")+
geom_hline(yintercept = -log10(.05/length(c)),
        linetype="dashed",
        color="black")+
geom_text(aes(min(p.gene.lm$slope),-log10(.05/length(c))+.15,
          label="BONFERRONI SIGNIFICANCE THRESHOLD"),
       size=2.5,hjust=0)+
geom_text(aes(min(p.gene.lm$slope),-log10(BHlim)+.15,
          label="BH THRESHOLD"),
       color="grey50",size=2.5,hjust=0)+
geom_text(aes(min(p.gene.lm$slope),-log10(0.05)+.15,
          label="UNADJUSTED ALPHA"),
       color="grey80",size=2.5,hjust=0)+
theme(panel.background = element_blank(),
    axis.line = element_line(size=0.5),
    legend.position = "none"
)+
scale_y_continuous(expand = c(.02,0),limits = c(0,9))+
scale_x_continuous(expand = c(.05,0))+
#ggtitle("NAFLD Activity Score")+
xlab("Slope")+ylab("-Log10 P-value")+
geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope<0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = -.1,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = 2,
            max.overlaps = 20,
            box.padding = .4,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni<.05 & slope>0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(0,.4),
            #ylim = c(6,9),
            nudge_x = .10,
            size=2.5,
            #color="black",
```

```
            segment.color = "grey80",
            nudge_y = 2,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
  geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>.05 & BH<.05 &
slope<0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(min(p.gene.lm$slope),0),
            #ylim = c(4,7),
            nudge_x = -.15,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = -.5,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
  geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(p_value_bonferonni>.05 & BH<.05 &
slope>0,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .15,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            #nudge_y = 1,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
  geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>.05 & slope<0 & gene %in%
                                        c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                          "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                       ,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            ylim = c(0,1.2),
            nudge_x = -.20,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            #nudge_y = -1.2,
            max.overlaps = 20,
```

```
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)+
  geom_label_repel(data= p.gene.lm %>% mutate(label = ifelse(BH>.05 & slope>0 & gene %in%
                                        c("CYP1A2","CYP3A4","CYP2E1","CYP2C19",
                                          "CYP2D6","CYP2B6","CYP2C9","CYP3A5","CYP2C8")
                                       ,gene,"")),
            aes(y=-log10(p_value),x=slope,label=label),
            #xlim = c(.2,.4),
            #ylim = c(7,10),
            nudge_x = .25,
            size=2.5,
            #color="black",
            segment.color = "grey80",
            nudge_y = -.75,
            max.overlaps = 20,
            #box.padding = 1,
            #point.padding = 0.5,
            #force_pull = 2,
            #force = 200,
            min.segment.length = 0)

sth_volc

ggsave("Supp_Figure3.tiff",plot = sth_volc,width = 8,height = 6,bg = "white")

plot1 <- ggarrange(row1,sth_volc,
            nrow = 2,
            ncol = 1,
            labels = "AUTO",
            heights = c(2, 6)
)
#plot1
ggsave("Figure4_alt.tiff",plot = plot1,width = 8,height = 6,bg = "white")

############## T-test 2 vs rest ##############################################
d2 <- subset(d,d$steatohep3==2)
d3 <- subset(d,d$steatohep3!=2)
tnas <- data.frame()
for(n in 1:length(c)){
  x <- c[[n]]
  xx <- t.test(d2[[x]],d3[[x]])
  tnas[n,1] <- x
  tnas[n,2] <- xx[["p.value"]]
}
tnas$V3 <- tnas$V2*nrow(tnas)
tnassig <- subset(tnas,tnas$V3<0.05)
tnassig <- tnassig[order(tnassig$V3),]
row.names(tnassig) <- tnassig$V1
tnassig <- tnassig[-c(1)]
d4 <- subset(d,select=c("steatohep3",row.names(tnassig)))
```

```
d4$group <- d4$steatohep==2
d4$group <- gsub(TRUE,"Steatohep = 2",d4$group)
d4$group <- gsub(FALSE,"Steatohep \u2264 1",d4$group)

d5 <- reshape2::melt(d4,c("steatohep3","group"))
d5$variable <- factor(d5$variable)
d5$group <- factor(d5$group,levels=c("Steatohep \u2264 1","Steatohep = 2"))
d5 <- subset(d5,!is.na(d5$steatohep3))

eplot <- ggplot()+
  geom_point(data=d5,aes(x=variable,y=value,fill=group),
          position = position_jitterdodge(jitter.width = 0.15,dodge.width = 0.75),size=0.3)+
  geom_boxplot(data=d5,aes(x=variable,y=value,fill=group),outlier.shape = NA,alpha=0.9)+
  theme(
    axis.text.x = element_text(angle = 90,size=10,vjust=0.5),
    axis.title.y = element_text(size=10),
    axis.title.x = element_blank(),
    axis.line = element_line(),
    legend.position = "top",
    axis.ticks.x = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_blank(),
    legend.title = element_blank())+
  ylab("Log Transcripts Per Million")

row2 <- ggarrange(cplot,dplot,eplot,labels = c("C","D","E"),ncol=3,widths = c(2,3,2.5))
ggarrange(row1,row2,nrow=2)

ggsave("Figure2.tiff",plot = last_plot(),width = 8,height=8)

tnassig$gene <- rownames(tnassig)
tnassig[4:6] <- NA
tnassig <- tnassig[c(3,4,1,5,2,6)]
colnames(tnassig) <- colnames(t1)
tnassig$group <- "Steatohepatitis 2 vs rest T-test"

table1 <- rbind(table1,tnassig)
write.csv(table1,file="table1.csv")
############# between variables ######################
c1 <- read.csv("directory/Nick's Analyses/c.csv", row.names=1)
c1$steatohep[c1$steatohep==1] <- 0
c1$steatohep[c1$steatohep==2] <- 1
c1$steatohep[c1$steatohep==3] <- 1
c1$steatohep[c1$steatohep==4] <- 2

fit1 <- summary(lm(fibrosis~nas,data = c1))
fit2 <- summary(lm(fibrosis~steatohep,data = c1))
fit3 <- summary(lm(steatohep~nas,data = c1))

A <- ggplot()+geom_point(data=c1,aes(y=steatohep,x=nas),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=steatohep,x=nas),method=lm)+
  ylab("Steatohepatitis")+
```

```
  xlab(paste0("NAS","\nR-squared: ",round(fit3$r.squared,2),"\np-value
",formatC(fit3$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

B <- ggplot()+geom_point(data=c1,aes(y=fibrosis,x=nas),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=fibrosis,x=nas),method=lm)+
  ylab("Fibrosis")+
  xlab(paste0("NAS","\nR-squared: ",round(fit1$r.squared,2),"\np-value
",formatC(fit1$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

C <- ggplot()+geom_point(data=c1,aes(y=fibrosis,x=steatohep),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=fibrosis,x=steatohep),method=lm)+
  ylab("Fibrosis")+
  xlab(paste0("Steatohepatitis","\nR-squared: ",round(fit2$r.squared,2),"\np-value
",formatC(fit2$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

fit4 <- summary(lm(age~nas,data = c1))
fit5 <- summary(lm(age~fibrosis,data = c1))
fit6 <- summary(lm(age~steatohep,data = c1))

D <- ggplot()+geom_point(data=c1,aes(y=age,x=nas),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=age,x=nas),method=lm)+
  ylab("Age")+
  xlab(paste0("NAS","\nR-squared: ",round(fit4$r.squared,2),"\np-value ",round(fit4$coefficients[2,4],2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

E <- ggplot()+geom_point(data=c1,aes(y=age,x=fibrosis),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=age,x=fibrosis),method=lm)+
  ylab("Age")+
  xlab(paste0("Fibrosis","\nR-squared: ",round(fit5$r.squared,2),"\np-value ",round(fit5$coefficients[2,4],2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))
```

```
F <- ggplot()+geom_point(data=c1,aes(y=age,x=steatohep),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=age,x=steatohep),method=lm)+
  ylab("Age")+
  xlab(paste0("Steatohepatitis","\nR-squared: ",round(fit6$r.squared,2),"\np-value ",round(fit6$coefficients[2,4],2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

fit7 <- summary(lm(bmi~nas,data = c1))
fit8 <- summary(lm(bmi~fibrosis,data = c1))
fit9 <- summary(lm(bmi~steatohep,data = c1))

G <- ggplot()+geom_point(data=c1,aes(y=bmi,x=nas),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=bmi,x=nas),method=lm)+
  ylab("BMI")+
  xlab(paste0("NAS","\nR-squared: ",round(fit7$r.squared,2),"\np-value
",formatC(fit7$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

H <- ggplot()+geom_point(data=c1,aes(y=bmi,x=fibrosis),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=bmi,x=fibrosis),method=lm)+
  ylab("BMI")+
  xlab(paste0("Fibrosis","\nR-squared: ",round(fit8$r.squared,2),"\np-value
",formatC(fit8$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

I <- ggplot()+geom_point(data=c1,aes(y=bmi,x=steatohep),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=c1,aes(y=bmi,x=steatohep),method=lm)+
  ylab("BMI")+
  xlab(paste0("Steatohepatitis","\nR-squared: ",round(fit9$r.squared,2),"\np-value
",formatC(fit9$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

ggarrange(A,B,C,D,E,F,G,H,I,labels = "AUTO")
ggsave("disease_comparisons_wcov.tiff",plot = last_plot(),width = 6,height = 6)

  ########## meta analysis #################
meta_analysis <- read_excel("meta-analysis.xlsx")
```

```
meta_analysis[,4:5] <- str_split_fixed(meta_analysis$gene," ",2)
vec <- unique(meta_analysis$V1)
vec_df <- as.data.frame(vec)
for(n in 1:nrow(vec_df)){
  x <- subset(meta_analysis,meta_analysis$V1==vec_df[n,1])
  y <- unique(x$gene)
  xx <- as.data.frame(table(x$V2))[order(as.data.frame(table(x$V2))$Freq,decreasing = T),]
  if(length(y)==1){
    vec_df[n,2] <- nrow(x)
    vec_df[n,3] <- x[1,1]
  }
  if(length(y)==2){
    vec_df[n,2] <- nrow(x)-2
    vec_df[n,3] <- paste0(x[1,1]," ",xx[1,4])
  }
}
vec_df <- subset(vec_df,vec_df$V2>0)
write.csv(vec_df,"comparison_stats_together.csv")
#ggplot(data=subset(vec_df,vec_df$V2>2),aes(x=reorder(gene,-V2),y=V2))+geom_col()

### compare with our data
meta_analysis <- read_excel("meta-analysis.xlsx")
meta_analysis[,4:5] <- str_split_fixed(meta_analysis$gene," ",2)
meta_analysis_sub <- subset(meta_analysis,meta_analysis$`agree with us`=="yes" | meta_analysis$`agree with
us`=="no")
meta_analysis_sub <- meta_analysis_sub[c(1:3)]
meta_analysis_sub <- distinct(meta_analysis_sub)
meta_analysis_sub <- meta_analysis_sub[order(meta_analysis_sub$`number of studies`,decreasing = T),]
write.csv(meta_analysis_sub,"comparison_stats.csv")


#################### skewness ##############################

bskew <- data.frame(gene = colnames(b), skew = NA)
for(n in 1:nrow(bskew)){
  bskew[n,2] <- skewness(b[,bskew[n,1]])
}

bskew1 <- merge(g,bskew,by="gene",all = T)

a <- ggplot()+geom_density(data = bskew1,aes(x=skew,fill=group),alpha=0.5)+ggtitle("Gene Expression Skewness")+
  scale_x_continuous(limits = c(-3,2))+guides(fill=guide_legend("Associated \nWith"))
a

askew <- data.frame(gene = colnames(a2), skew = NA)
for(n in 1:nrow(askew)){
  askew[n,2] <- skewness(a2[,askew[n,1]])
}
colnames(askew) <- c("mir","skew")

jmir <- as.data.frame(unique(data.frame(j$mir,j$gene)))
colnames(jmir) <- c("mir","gene")
```

```
askew1 <- merge(askew,jmir,by="mir",all=T)

b <- ggplot()+geom_density(data = askew1,aes(x=skew,fill=gene),alpha=0.5)+ggtitle("MicroRNA Expression
Skewness")+
  scale_x_continuous(limits = c(-2,2))+guides(fill=guide_legend("Associated \nWith"))
b

phenotypes_pruned <- read.csv("directory/phenotypes_pruned.csv")
phenotypes_pruned <- phenotypes_pruned[c(5,6,11)]
colnames(phenotypes_pruned) <- c("NAS","Fibrosis","Steatohep")
melt <- reshape2::melt(phenotypes_pruned)

c <- ggplot()+geom_density(data = melt,aes(x=value,fill=variable),alpha=0.5)+ggtitle("Phenotype Distribution")+
  scale_x_continuous(limits = c(-1,10))+guides(fill=guide_legend("Phenotype"))
c

ggarrange(a,b,c,labels="AUTO")

ggsave(plot = last_plot(),filename = "distributions.tiff",device = "tiff",width = 10,height = 10)

########## absolute CYP2c19 expression change ##############
reads <- read.csv("directory/Nick's Analyses/data frame ouputs/NASH_RNAseq_data_TPM_norm.txt", sep="",
stringsAsFactors=FALSE)
reads1 <- subset(reads, reads$Gene=="CYP2C19")
row.names(reads1) <- reads1$Gene
reads1 <- reads1[-c(1)]
reads1 <- as.data.frame(t(reads1))

phen <- d[c(1,5,6,11)]
row.names(phen) <- phen$Row.names
phen <- phen[-c(1)]

q <- merge(phen,reads1,by=0)
row.names(q) <- q$Row.names
q <- q[-c(1)]

fit1 <- summary(lm(nas~CYP2C19,data = q))
fit2 <- summary(lm(fibrosis~CYP2C19,data = q))
fit3 <- summary(lm(steatohep3~CYP2C19,data = q))

A <- ggplot()+geom_point(data=q,aes(y=CYP2C19,x=nas),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=q,aes(y=CYP2C19,x=nas),method=lm)+
  ylab("CYP2C19")+
  xlab(paste0("NAS","\nR-squared: ",round(fit1$r.squared,2),"\np-value
",formatC(fit1$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

B <- ggplot()+geom_point(data=q,aes(y=CYP2C19,x=fibrosis),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=q,aes(y=CYP2C19,x=fibrosis),method=lm)+
```

```
  ylab("CYP2C19")+
  xlab(paste0("Fibrosis","\nR-squared: ",round(fit2$r.squared,2),"\np-value
",formatC(fit2$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

C <- ggplot()+geom_point(data=q,aes(y=CYP2C19,x=steatohep3),position = position_jitter(width=0.1,height = 0.1))+
  geom_smooth(data=q,aes(y=CYP2C19,x=steatohep3),method=lm)+
  ylab("CYP2C19")+
  xlab(paste0("Steatohepatitis","\nR-squared: ",round(fit3$r.squared,2),"\np-value
",formatC(fit3$coefficients[2,4],format="e",digits=2)))+
  theme(text = element_text(size=8),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank(),
        panel.background=element_blank(),
        axis.line=element_line(colour="black"))

ggarrange(A,B,C,labels = "AUTO")
ggsave(plot = last_plot(),filename = "CYP2C19_absolute.tiff",device = "tiff",width = 5,height = 5)


############## venn diagram #################
library(UpSetR)
z1 <- read.csv("directory/pharmacogene_fibrosis_regression_results_logTPM.csv", row.names=1)
z2 <- read.csv("directory/pharmacogene_NAS_regression_results_logTPM.csv", row.names=1)
z3 <- read.csv("directory/pharmacogene_steatohep_regression_results_logTPM.csv", row.names=1)
z1$group <- "fibrosis"
z1$fibBH <- p.adjust(z1$p_value,method = "BH")
z1sub <- subset(z1,z1$fibBH<0.05)
colnames(z1sub)[2] <- "FIBslope"
z2$group <- "NAS"
z2$nasBH <- p.adjust(z2$p_value,method = "BH")
z2sub <- subset(z2,z2$nasBH<0.05)
colnames(z2sub)[2] <- "NASslope"
z3$group <- "steatohep"
z3$steatBH <- p.adjust(z3$p_value,method = "BH")
z3sub <- subset(z3,z3$steatBH<0.05)
colnames(z3sub)[2] <- "STHslope"

zmerge <- merge(z1sub[c(1,7)],z2sub[c(1,7)],by="gene",all=T)
zmerge <- merge(zmerge,z3sub[c(1,7)],by="gene",all=T)

zmerge[is.na(zmerge)] <- 0
zmerge$fibBH[zmerge$fibBH>0] <- 1
zmerge$nasBH[zmerge$nasBH>0] <- 1
zmerge$steatBH[zmerge$steatBH>0] <- 1

row.names(zmerge) <- zmerge$gene
zmerge <- zmerge[-c(1)]
```

```
colnames(zmerge) <- c("FIB","NAS","STH")
pdf(file="upset.pdf", width = 5,height = 5)
upset(zmerge,
    order.by = "degree",
    nintersects=4,
    sets.x.label = "Pharmacogenes",
    mainbar.y.label = "Pharmacogene\nIntersections",
    mb.ratio = c(0.7,0.3),
    text.scale = c(2, 1.5, 1.5, 1.5, 2, 2),
    line.size = 1.2,
    point.size = 5)
dev.off()

zmerge1 <- merge(z1sub[c(1,2,7)],z2sub[c(1,2,7)],by="gene",all=T)
zmerge1 <- merge(zmerge1,z3sub[c(1,2,7)],by="gene",all=T)
zmerge1 <- na.omit(zmerge1)

for(n in 1:nrow(zmerge1)){
 zmerge1[n,8] <- mean(c(zmerge1[n,3],zmerge1[n,5],zmerge1[n,7]))
}
colnames(zmerge1)[8] <- "pAVG"
zmerge1$FIB_FC <- exp(1)^zmerge1$FIBslope
zmerge1$NAS_FC <- exp(1)^zmerge1$NASslope
zmerge1$STH_FC <- exp(1)^zmerge1$STHslope

zmerge2 <- zmerge1[c(1,9,10,11)]
colnames(zmerge2) <- c("gene","FIB","NAS","STH")
df <- reshape2::melt(zmerge2)
colnames(df) <- c("gene","Group","fold_change")

zmerge3 <- zmerge1[c(1,3,5,7)]
colnames(zmerge3) <- c("gene","FIB","NAS","STH")
df2 <- reshape2::melt(zmerge3)
colnames(df2) <- c("gene","Group","BH p-value")
df3 <- merge(df,df2,by=c("gene","Group"))
df3$`p-value (BH)` <- log10(1/df3$`BH p-value`)
df3$Group <- as.factor(df3$Group)
df3$Group <- factor(df3$Group,levels=c("NAS","FIB","STH"))
df3 <- df3[order(df3$`BH p-value`,decreasing = F),]
row.names(df3) <- NULL

ggplot()+geom_point(data = df3,aes(x=reorder(gene,fold_change),y=fold_change,shape=Group,size=`p-value
(BH)`,fill=Group))+
 theme_minimal()+
 theme(axis.text = element_text(size=10),
     axis.text.x = element_text(angle=90,vjust=0.5,face = "bold",color = "black"),
     legend.key.size = unit(4,"mm"),
     axis.line = element_line()
     )+
 ylab("mRNA Percent Change per Disease Unit")+
 xlab(NULL)+
 scale_shape_manual(values=c(21,22,23))+
 scale_fill_manual(values=c("firebrick","darkslategrey","goldenrod4"))+
```

```
  scale_size_continuous(breaks=c(2,3,4,5),labels = c(">0.01","0.001","0.0001","<0.00001"),range = c(0,7))+
  guides(shape = guide_legend(override.aes = list(size = 5)))

ggsave("BH percent change.tiff",plot=last_plot(),height=4,width=5,bg="white")
write.csv(zmerge1,"BH_table.csv")

####### FOREST PLOTS and META Anlaysis ###############
library(meta)
library(metafor)
library(tidyverse)
library(ggplot2)
library(dmetar)
setwd("directory/meta")
forest <- read.csv("directory/meta/forest2.csv", stringsAsFactors=FALSE)

# bmj "how to convert pvalue to CI"
forest$z <- (-.862)+sqrt(.743-(2.404*log(forest$P.val)))
forest$SE <- abs(forest$Log_FC/forest$z)
forest$lowerCI <- forest$Log_FC-(1.96*forest$SE)
forest$upperCI <- forest$Log_FC+(1.96*forest$SE)
forest$Comparison <- gsub("Severe NAFLD vs.","Severe NAFLD vs.\n",forest$Comparison)
forest$Comparison <- gsub("High steatosis vs.","High steatosis vs.\n",forest$Comparison)
forest$lab <- paste0(forest$Author,"    N=",forest$N)
forest$Comparison <- gsub("F3-4 vs. F0-1","Fibrosis 3-4 vs. 0-1",forest$Comparison)
forest$Comparison <- gsub("steatosis vs. control","Steatosis vs. control",forest$Comparison)
forest$Comparison <- gsub("NAS 5-8 vs. NAS 0-4","NAFLD Activity Score\n5-8 vs. 0-4",forest$Comparison)
forest$Comparison <- gsub("control","Control",forest$Comparison)
forest$Comparison <- gsub("mild-no","Mild-No",forest$Comparison)
forest$Comparison <- gsub("steatosis","Steatosis",forest$Comparison)
forest$Comparison <- gsub("low-no","Low-No",forest$Comparison)

############## ggplot ###################

forest2 <- subset(forest,forest$Comparison=="Fibrosis 3-4 vs. 0-1" |
            forest$Comparison=="High Steatosis vs.\n Low-No Steatosis" |
            forest$Comparison=="NAFLD Activity Score\n5-8 vs. 0-4" |
            forest$Comparison=="NASH vs. Control" |
            forest$Comparison=="Severe NAFLD vs.\n Mild-No NAFLD" |
            forest$Comparison=="Steatosis vs. Control" |
            forest$Comparison=="NASH vs. Steatosis" |
            forest$Comparison=="NAFLD vs. Control" |
            forest$Comparison=="NASH vs. Control-Steatosis"|
            forest$Comparison=="NASH vs. NAFL")

forest2$Comparison <- gsub("NASH vs. NAFL","NASH vs. Steatosis",forest2$Comparison)

forest2$Comparison <- as.factor(forest2$Comparison)
forest2$Comparison <- ordered(forest2$Comparison,levels=c("Fibrosis 3-4 vs. 0-1",
                      "Severe NAFLD vs.\n Mild-No NAFLD",
                      "NAFLD vs. Control",
                      "NAFLD Activity Score\n5-8 vs. 0-4",
                      "NASH vs. Control",
                      "NASH vs. Control-Steatosis",
```

```
                    "NASH vs. Steatosis",
                    "Steatosis vs. Control",
                    "High Steatosis vs.\n Low-No Steatosis"))

ggplot()+
 geom_point(data=forest2,mapping=aes(x=Log_FC,y=lab))+
 geom_errorbar(data=forest2,mapping=aes(y=lab,x=Log_FC,xmin=lowerCI,xmax=upperCI))+
 theme(
  panel.grid.major=element_blank(),
     panel.grid.minor=element_blank(),
    #  panel.background=element_blank(),
     # axis.line=element_line(colour="black"),
      #axis.text.x=element_blank(),
      #axis.ticks.x=element_blank(),
      axis.title=element_text(size=12),
      axis.text.x=element_text(size=12),
      axis.text.y=element_text(size=8),
      strip.text.y.right=element_text(angle=0,size=10))+
 ylab(NULL)+
 xlab("Log2 Fold Change")+
 ggtitle("CYP2C19 Differential Expression in NAFLD and NASH")+
 geom_vline(xintercept = 0,linetype="dashed",color="firebrick",size=.6,alpha=0.85)+
 scale_y_discrete(expand = c(0,1))+
 scale_x_continuous(breaks=seq(from=-5,to=5,by=1),limits = c(-5,5),expand = c(0,0))+
 facet_grid(Comparison~.,scales="free_y",space="free_y")

ggsave(plot = last_plot(),filename = "Supp_meta.tiff",width = 7,height = 7)

################ using the meta package ######################

##Fibrosis 3-4 vs. 0-1
forest1 <- subset(forest,forest$Comparison=="Fibrosis 3-4 vs. 0-1")
forest11 <- forest1[c(1,2,3,6,9,10)]
forest1$N_weight <- (forest1$N)*forest1$Log_FC
n_weight <- sum(forest1$N_weight)/sum(forest1$N)
m.gen <- metagen(TE=Log_FC,
            #seTE=SE,
            pval = P.val,
            studlab=Author,
            data=forest1,
            sm="SMD",
            #method.sd = "Wan",
            #n.e = N,
            #w.random=weights,
            method.tau = "REML",
            detail.tau = TRUE,
            fixed = FALSE,
            random = TRUE,
            #hakn = FALSE,
            prediction = FALSE,
            title = "CYP2C19 DIfferential Expression in NAFLD and NASH")

summary(m.gen)
```

```
pdf(file="fibrosis_forest.pdf", width = 8,height = 3.9)
forest.meta(m.gen,
        sortvar = TE,
        predict = FALSE,
        print.tau2 = TRUE,
        leftcols = c("studlab","Year","Tech","N"),
        leftlabs = c("Author","Year","Tech","N"),
        rightcols = c("effect", "ci","w.random"),
        rightlabs = c("Value","95% CI","Weight"),
        xlab = "Log2 Fold Change",
        smlab="Fibrosis 3-4 vs. 0-1\nCYP2C19 mRNA Expression",
        hetstat = FALSE,
        colgap="5mm",
        colgap.studlab="-22mm",
        text.random = "Random Effects Model Estimate",
        #comb.random=TRUE,
        calcwidth.random=F,
        just="left",
        just.addcols="left",
        #label.right = "increased",
        #backtransf = TRUE,
        plotwidth="55mm",
        text.addline1 = paste0("Sample size-weighted effect = ",round(n_weight,digits = 2))
        #calcwidth.addline=F
        )
dev.off()

### NASH vs. Control
forest2 <- subset(forest,forest$Comparison=="NASH vs. Control")
forest3 <- forest[52,]
forest3$Author <- paste0(forest3$Author,"*")
forest2 <- rbind(forest3,forest2)
forest2$N_weight <- (forest2$N)*forest2$Log_FC
n_weight <- sum(forest2$N_weight)/sum(forest2$N)
m.gen2 <- metagen(TE=Log_FC,
            #seTE=SE,
            pval = P.val,
            studlab=Author,
            data=forest2,
            sm="SMD",
            method.tau = "REML",
            fixed = FALSE,
            hakn = TRUE,
            prediction = FALSE,
            title = "CYP2C19 DIfferential Expression in NAFLD and NASH")


pdf(file="NASH_forest.pdf", width = 8,height = 4.7)
forest.meta(m.gen2,
        sortvar = TE,
        predict = FALSE,
        print.tau2 = FALSE,
```

```
        leftcols = c("studlab","Year","Tech","N"),
        leftlabs = c("Author","Year","Tech","N"),
        rightcols = c("effect", "ci","w.random"),
        rightlabs = c("Value","95% CI","Weight"),
        xlab = "Log2 Fold Change",
        smlab="NASH vs. Control\nCYP2C19 mRNA Expression",
        hetstat = FALSE,
        colgap="5mm",
        colgap.studlab="-22mm",
        text.random = "Random Effects Model Estimate",
        #comb.random=TRUE,
        calcwidth.random=F,
        just="left",
        just.addcols="left",
        #label.right = "increased",
        #backtransf = TRUE
        plotwidth="55mm",
        text.addline1 = paste0("Sample size-weighted effect = ",round(n_weight,digits = 2))
)
dev.off()

### NAFLD vs. control
forest4 <- subset(forest,forest$Comparison==forest[4,6])
forest3 <- forest[27,]
forest5 <- forest[c(42,32,24),]
forest3 <- rbind(forest3,forest5)
forest3$Author <- paste0(forest3$Author,"*")
forest5 <- rbind(forest4,forest3)

forest5$N_weight <- (forest5$N)*forest5$Log_FC
n_weight <- sum(forest5$N_weight)/sum(forest5$N)

m.gen3 <- metagen(TE=Log_FC,
            #seTE=SE,
            pval = P.val,
            studlab=Author,
            data=forest5,
            sm="SMD",
            method.tau = "REML",
            fixed = FALSE,
            hakn = TRUE,
            prediction = FALSE,
            title = "CYP2C19 DIfferential Expression in NAFLD and NASH")

pdf(file="NAFLD_forest.pdf", width = 8,height = 4.5)
forest.meta(m.gen3,
        sortvar = TE,
        predict = FALSE,
        print.tau2 = FALSE,
        leftcols = c("studlab","Year","Tech","N"),
        leftlabs = c("Author","Year","Tech","N"),
        rightcols = c("effect", "ci","w.random"),
        rightlabs = c("Value","95% CI","Weight"),
```

```
        xlab = "Log2 Fold Change",
        smlab="NAFLD Activity Score 5-8 vs. 0-4\nCYP2C19 mRNA Expression",
        hetstat = FALSE,
        colgap="5mm",
        colgap.studlab="-22mm",
        text.random = "Random Effects Model Estimate",
        #comb.random=TRUE,
        calcwidth.random=F,
        just="left",
        just.addcols="left",
        #label.right = "increased",
        #backtransf = TRUE
        plotwidth="55mm",
        text.addline1 = paste0("Sample size-weighted effect = ",round(n_weight,digits = 2))
)
dev.off()
help(metagen)
```