

Positive selection on genes interacting with SARS-Cov2, comparison of different analysis

Marie Cariou

Janvier 2021

Contents

1	Files manipulations	2
1.1	Read Janet Young's table	2
1.2	Read DGINN Young table	2
1.3	Joining Young and DGINN Young table	2
1.4	Read DGINN Table	3
1.5	Join Table and DGINN table	6
1.6	Add DGINN results on bat dataset	7
1.7	Write the new table	11
2	Second Table	11
2.1	Primates	11
2.2	Bats	12
2.3	Merged table	14
3	Complete data	27

1 Files manipulations

1.1 Read Janet Young's table

```
workdir<-"/home/adminmarie/Documents/CIRI_BIBS_projects/2020_05_Etienne_covid/"

tab<-read.delim(paste0(workdir,
  "data/COVID_PAMLresults_332hits_plusBatScreens_2020_Apr14.csv"),
  fill=T, h=T, dec=",")
dim(tab)

## [1] 332 84
```

1.2 Read DGINN Young table

```
dginnY<-read.delim(paste0(workdir,
  "data/summary_primate_young.res"),
  fill=T, h=T)

dim(dginnY)

## [1] 1992 7
```

1.3 Joining Young and DGINN Young table

```
# correct gene names (MARC1)
val_remp=as.character(unique(dginnY$Gene)[(unique(dginnY$Gene) %in%
  tab$Gene.name)==F])

tab$Gene.name<-as.character(tab$Gene.name)
tab$Gene.name[158]<-val_remp
sum(unique(dginnY$Gene) %in% unique(tab$Gene.name))

## [1] 332
```

```

add_col<-function(method="PamLM1M2"){

tmp<-dginnY[dginnY$Method==method,
             c("Gene", "Omega", "PosSel", "PValue", "NbSites", "PSS")]

names(tmp)<-c("Gene.name", paste0("Omega_", method),
             paste0("PosSel_", method), paste0("PValue_", method),
             paste0("NbSites_", method), paste0("PSS_", method))

tab<-merge(tab, tmp, by="Gene.name")

return(tab)
}

tab<-add_col("PamLM1M2")
tab<-add_col("PamLM7M8")
tab<-add_col("BppM1M2")
tab<-add_col("BppM7M8")

# Manip pour la colonne BUSTED

tmp<-dginnY[dginnY$Method=="BUSTED",c("Gene", "Omega", "PosSel", "PValue")]
names(tmp)<-c("Gene.name", "Omega_BUSTED", "PosSel_BUSTED", "PValue_BUSTED")
tab<-merge(tab, tmp, by="Gene.name")

tmp<-dginnY[dginnY$Method=="MEME",c("Gene", "NbSites", "PSS")]
names(tmp)<-c("Gene.name", "NbSites_MEME", "PSS_MEME")
tab<-merge(tab, tmp, by="Gene.name")

```

1.4 Read DGINN Table

```

dginnT<-read.delim(paste0(workdir,
                           "data/DGINN_202005281649summary_cleaned.csv"),
                  fill=T, h=T, sep=",")

dim(dginnT)

```

```
## [1] 412 27

names(dginnT)

## [1] "File" "Name" "Gene"
## [4] "GeneSize" "NbSpecies" "omegaM0Bpp"
## [7] "omegaM0codeml" "BUSTED" "BUSTED.p.value"
## [10] "MEME.NbSites" "MEME.PSS" "BppM1M2"
## [13] "BppM1M2.p.value" "BppM1M2.NbSites" "BppM1M2.PSS"
## [16] "BppM7M8" "BppM7M8.p.value" "BppM7M8.NbSites"
## [19] "BppM7M8.PSS" "codemlM1M2" "codemlM1M2.p.value"
## [22] "codemlM1M2.NbSites" "codemlM1M2.PSS" "codemlM7M8"
## [25] "codemlM7M8.p.value" "codemlM7M8.NbSites" "codemlM7M8.PSS"

# Number of genes in dginn-primate output not present in the original table
dginnT[(dginnT$Gene %in% tab$Gene.name)==F,"Gene"]

## [1] ACE2 ADAM9[0-3120] ADAM9[3119-3927]
## [4] ATP5MGL C1H1ORF50 CEP135[0-3264]
## [7] CEP135[3263-3678] CEP43 COQ8B
## [10] COQ8A CSNK2A1 CSNK2B[0-609]
## [13] CSNK2B[608-2568] CYB5R1 DDX21[0-717]
## [16] DDX21[716-2538] DDX50 DNAJC15
## [19] DPH5[0-702] DPH5[701-1326] DPY19L2
## [22] ELOC ERO1B EXOSC3[0-1446]
## [25] EXOSC3[1445-1980] FBN3 GNB4
## [28] GNB2 GNB3 GOLGA7[0-312]
## [31] GOLGA7[311-549] GPX1[0-1218] GPX1[1217-2946]
## [34] HDAC1 HS6ST3 IMPDH1
## [37] ITGB1[0-2328] ITGB1[2327-2844] LMAN2L
## [40] MRPS5[0-1569] MRPS5[1568-3783] MARC2
## [43] MGRN1 NDFIP2[0-768] NDFIP2[767-1314]
## [46] NDUFAF2[0-258] NDUFAF2[257-744] NSD2
## [49] NUP58 NUP58[0-1824] NUP58[1823-2367]
## [52] PABPC3 POTPABPC1 PABPC4L
## [55] PABPC5 PCSK5 PRIM2[0-1071]
## [58] PRIM2[1070-1902] PRKACB PRKACG
## [61] PTGES2[0-1587] PTGES2[1586-2202] RAB8B
## [64] RAB13 RAB18[0-855] RAB18[854-1815]
## [67] RAB2B RAB5A RAB5B
```

```

## [70] RAB15          RALB          EZR
## [73] EZR[0-1458]      EZR[1457-3771] MSN
## [76] RETREG3         RHOB          RHOC
## [79] SLC44A2[0-2577]  SLC44A2[2576-3657] SPART
## [82] SRP72[0-2604]    SRP72[2603-3417]  STOM[0-1047]
## [85] STOM[1046-1800]  STOML3         TIMM29
## [88] TLE4            TLE2           TLE2[0-1302]
## [91] TLE2[1301-3987]  TMPRSS2        TOMM70
## [94] TOR1B           WASHC4         WFS1[0-2346]
## [97] WFS1[2345-3216]  YIF1B
## 411 Levels: AAR2 AASS AATF ABCC1 ACAD9 ACADM ACE2 ACSL3 ... ZYG11B

# This includes paralogs, recombinations found by DGINN and additionnal genes
# included on purpose

# Number of genes from the original list not present in DGINN output
tab[(tab$Gene.name %in% dginnT$Gene)==F,"Gene.name"]

## [1] "ADCK4"      "ARL6IP6"    "ATP5L"      "C19orf52"   "C1orf50"    "ERO1LB"
## [7] "FAM134C"    "FGFR10P"    "KIAA1033"   "MFGE8"      "NUPL1"      "SIGMAR1"
## [13] "SPG20"      "TCEB1"      "TCEB2"      "TOMM70A"    "USP13"      "VIMP"
## [19] "WHSC1"

names(dginnT)<-c("File", "Name", "Gene.name", "GeneSize",
  "dginn-primate_NbSpecies", "dginn-primate_omegaM0Bpp",
  "dginn-primate_omegaM0codeml", "dginn-primate_BUSTED",
  "dginn-primate_BUSTED.p.value", "dginn-primate_MEME.NbSites",
  "dginn-primate_MEME.PSS", "dginn-primate_BppM1M2",
  "dginn-primate_BppM1M2.p.value", "dginn-primate_BppM1M2.NbSites",
  "dginn-primate_BppM1M2.PSS", "dginn-primate_BppM7M8",
  "dginn-primate_BppM7M8.p.value", "dginn-primate_BppM7M8.NbSites",
  "dginn-primate_BppM7M8.PSS", "dginn-primate_codemlM1M2",
  "dginn-primate_codemlM1M2.p.value", "dginn-primate_codemlM1M2.NbSites",
  "dginn-primate_codemlM1M2.PSS", "dginn-primate_codemlM7M8",
  "dginn-primate_codemlM7M8.p.value", "dginn-primate_codemlM7M8.NbSites",
  "dginn-primate_codemlM7M8.PSS")

```

```

table(dginnT$`dginn-primate_BUSTED`)
table(dginnT$`dginn-primate_codemlM1M2`)
table(dginnT$`dginn-primate_codemlM7M8`)
table(dginnT$`dginn-primate_BppM1M2`)
table(dginnT$`dginn-primate_BppM7M8`)

table(dginnT$`dginn-primate_BUSTED`=="na",dginnT$`dginn-primate_codemlM1M2`=="na", dg
      dginnT$`dginn-primate_BppM1M2`=="na", dginnT$`dginn-primate_BppM7M8`=="na" )

```

1.5 Join Table and DGINN table

```

tab<-merge(tab,dginnT, by="Gene.name", all.x=T)

table(tab$`dginn-primate_BUSTED`)

##
##   N   na   Y
## 155  12 147

table(tab$`dginn-primate_codemlM1M2`)

##
##   N   na   Y
## 216  26  72

table(tab$`dginn-primate_codemlM7M8`)

##
##   N   na   Y
## 161  40 113

table(tab$`dginn-primate_BppM1M2`)

##
##   N   na   Y
## 252  21  41

table(tab$`dginn-primate_BppM7M8`)

```

```
##
##      N   na   Y
## 173  22 119

table(tab$`dginn-primate_BUSTED`=="na" | tab$`dginn-primate_codemlM1M2`=="na" | tab$`
      tab$`dginn-primate_BppM1M2`=="na" | tab$`dginn-primate_BppM7M8`=="na" )

##
## FALSE  TRUE
##   274    40
```

1.6 Add DGINN results on bat dataset

DGINN results from different analysis.

```
# original table
dginnbats<-read.delim(paste0(workdir,
                             "data/DGINN_202005281339summary_cleaned.tab"),
                      fill=T, h=T)

# rerun on corrected alignment
dginnbatsnew1<-read.delim(paste0(workdir,
                                  "data/DGINN_202011262248_summary.tab"),
                          fill=T, h=T)
dginnbatsnew2<-read.delim(paste0(workdir,
                                  "data/DGINN_202012192053_summary.tab"),
                          fill=T, h=T)

# colomne choice, BUSTED and Bppml form first file, codeml from the other one
dginnbatsnew<-dginnbatsnew1
dginnbatsnew$omegaM0codeml<-dginnbatsnew2$omegaM0codeml

dginnbatsnew$codemlM1M2<-dginnbatsnew2$codemlM1M2
dginnbatsnew$codemlM1M2_p.value<-dginnbatsnew2$codemlM1M2_p.value
dginnbatsnew$codemlM1M2_NbSites<-dginnbatsnew2$codemlM1M2_NbSites
dginnbatsnew$codemlM1M2_PSS<-dginnbatsnew2$codemlM1M2_PSS

dginnbatsnew$codemlM7M8<-dginnbatsnew2$codemlM7M8
dginnbatsnew$codemlM7M8_p.value<-dginnbatsnew2$codemlM7M8_p.value
```

```

dginnbatsnew$codemlM7M8_NbSites<-dginnbatsnew2$codemlM7M8_NbSites
dginnbatsnew$codemlM7M8_PSS<-dginnbatsnew2$codemlM7M8_PSS

####
## RIPK1 is actually a primate results
## 1. Take it and put it at the right place
ripk1<-as.vector(dginnbatsnew[dginnbatsnew$Gene=="RIPK1",])
tab$`dginn-primate_omegaM0Bpp`<-as.numeric(as.character(tab$`dginn-primate_omegaM0Bpp`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_BUSTED.p.value`<-as.numeric(as.character(tab$`dginn-primate_BUSTED.p.value`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_BppM1M2.p.value`<-as.numeric(as.character(tab$`dginn-primate_BppM1M2.p.value`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_BppM7M8.p.value`<-as.numeric(as.character(tab$`dginn-primate_BppM7M8.p.value`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_BppM7M8.PSS`<-as.numeric(as.character(tab$`dginn-primate_BppM7M8.PSS`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_codemlM1M2.p.value`<-as.numeric(as.character(tab$`dginn-primate_codemlM1M2.p.value`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_codemlM1M2.PSS`<-as.numeric(as.character(tab$`dginn-primate_codemlM1M2.PSS`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_codemlM7M8.p.value`<-as.numeric(as.character(tab$`dginn-primate_codemlM7M8.p.value`))

## Warning:  NAs introduits lors de la conversion automatique

tab$`dginn-primate_codemlM7M8.PSS`<-as.numeric(as.character(tab$`dginn-primate_codemlM7M8.PSS`))

## Warning:  NAs introduits lors de la conversion automatique

```



```

tab[tab$Gene.name=="RIPK1", "GeneSize"]<-ripk1$GeneSize
tab[tab$Gene.name=="RIPK1", "dginn-primate_NbSpecies"]<-ripk1$NbSpecies
tab[tab$Gene.name=="RIPK1", "dginn-primate_omegaM0Bpp"]<-ripk1$omegaM0Bpp
tab[tab$Gene.name=="RIPK1", "dginn-primate_omegaM0codeml"]<-ripk1$omegaM0codeml

tab[tab$Gene.name=="RIPK1", "dginn-primate_BUSTED"]<-ripk1$BUSTED
tab[tab$Gene.name=="RIPK1", "dginn-primate_BUSTED.p.value"]<-ripk1$BUSTED.p.value
tab[tab$Gene.name=="RIPK1", "dginn-primate_MEME.NbSites"]<-ripk1$MEME_NbSites
tab[tab$Gene.name=="RIPK1", "dginn-primate_MEME.PSS"]<-as.numeric(as.character(ripk1$M

## Warning:  NAs introduits lors de la conversion automatique

tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM1M2"]<-ripk1$BppM1M2
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM1M2.p.value"]<-ripk1$BppM1M2.p.value
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM1M2.NbSites"]<-ripk1$BppM1M2_NbSites
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM1M2.PSS"]<-ripk1$BppM1M2_PSS

tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM7M8"]<-ripk1$BppM7M8
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM7M8.p.value"]<-ripk1$BppM7M8.p.value
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM7M8.NbSites"]<-ripk1$BppM7M8_NbSites
tab[tab$Gene.name=="RIPK1", "dginn-primate_BppM7M8.PSS"]<-ripk1$BppM7M8_PSS

tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM1M2"]<-ripk1$codemlM1M2
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM1M2.p.value"]<-ripk1$codemlM1M2.p.va
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM1M2.NbSites"]<-ripk1$codemlM1M2_NbSi
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM1M2.PSS"]<-ripk1$codemlM1M2_PSS
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM7M8"]<-ripk1$codemlM7M8
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM7M8.p.value"]<-ripk1$codemlM7M8.p.va
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM7M8.NbSites"]<-ripk1$codemlM7M8_NbSi
tab[tab$Gene.name=="RIPK1", "dginn-primate_codemlM7M8.PSS"]<-ripk1$codemlM7M8_PSS

## 2. Remove it
dginnbatsnew<-dginnbatsnew[dginnbatsnew$Gene!="RIPK1",]

## suppress redundant lines
dginnbats<-dginnbats[(dginnbats$Gene %in% dginnbatsnew$Gene)==FALSE,]
names(dginnbatsnew)<-names(dginnbats)

#####
```

```

dginnbatsnew[,4]<-as.numeric(dginnbatsnew[,4])
dginnbats[,6]<-as.numeric(as.character(dginnbats[,6]))

## Warning:  NAs introduits lors de la conversion automatique

dginnbats[,8]<-as.character(dginnbats[,8])
dginnbats[,12]<-as.character(dginnbats[,12])
dginnbats[,13]<-as.numeric(as.character(dginnbats[,13]))

## Warning:  NAs introduits lors de la conversion automatique

dginnbats[,16]<-as.character(dginnbats[,16])
dginnbats[,17]<-as.numeric(as.character(dginnbats[,17]))

## Warning:  NAs introduits lors de la conversion automatique

## replace by new data
dginnbats<-rbind(dginnbats, dginnbatsnew)

names(dginnbats)<-c("File", "bats_Name", "cooper.batsGene", paste0("bats_",
  names(dginnbats)[-1:3]))
names(dginnbats)

##   [1] "File"                "bats_Name"
##   [3] "cooper.batsGene"     "bats_GeneSize"
##   [5] "bats_NbSpecies"      "bats_omegaM0Bpp"
##   [7] "bats_omegaM0codeml"  "bats_BUSTED"
##   [9] "bats_BUSTED.p.value" "bats_MEME.NbSites"
##  [11] "bats_MEME.PSS"       "bats_BppM1M2"
##  [13] "bats_BppM1M2.p.value" "bats_BppM1M2.NbSites"
##  [15] "bats_BppM1M2.PSS"    "bats_BppM7M8"
##  [17] "bats_BppM7M8.p.value" "bats_BppM7M8.NbSites"
##  [19] "bats_BppM7M8.PSS"    "bats_codemlM1M2"
##  [21] "bats_codemlM1M2.p.value" "bats_codemlM1M2.NbSites"
##  [23] "bats_codemlM1M2.PSS"  "bats_codemlM7M8"
##  [25] "bats_codemlM7M8.p.value" "bats_codemlM7M8.NbSites"
##  [27] "bats_codemlM7M8.PSS"

tab<-merge(tab,dginnbats, by="cooper.batsGene", all.x=T)

```

1.7 Write the new table

```
write.table(tab, "covid_comp_complete_old.txt", row.names=FALSE, quote=FALSE, sep="\t")
```

2 Second Table

Table containing the DGINN results for both Primates and bats. Conserve all genes.

2.1 Primates

```
dginnT<-read.delim(paste0(workdir,
  "data/DGINN_202005281649summary_cleaned.csv"),
  fill=T, h=T, sep=",")

dim(dginnT)

## [1] 412 27

names(dginnT)

## [1] "File" "Name" "Gene"
## [4] "GeneSize" "NbSpecies" "omegaM0Bpp"
## [7] "omegaM0codeml" "BUSTED" "BUSTED.p.value"
## [10] "MEME.NbSites" "MEME.PSS" "BppM1M2"
## [13] "BppM1M2.p.value" "BppM1M2.NbSites" "BppM1M2.PSS"
## [16] "BppM7M8" "BppM7M8.p.value" "BppM7M8.NbSites"
## [19] "BppM7M8.PSS" "codemlM1M2" "codemlM1M2.p.value"
## [22] "codemlM1M2.NbSites" "codemlM1M2.PSS" "codemlM7M8"
## [25] "codemlM7M8.p.value" "codemlM7M8.NbSites" "codemlM7M8.PSS"

# Rename the columns to include primate
names(dginnT)<-c("File", "Name", "Gene.name", "GeneSize",
  "dginn-primate_NbSpecies", "dginn-primate_omegaM0Bpp",
  "dginn-primate_omegaM0codeml", "dginn-primate_BUSTED",
  "dginn-primate_BUSTED.p.value", "dginn-primate_MEME.NbSites",
  "dginn-primate_MEME.PSS", "dginn-primate_BppM1M2",
  "dginn-primate_BppM1M2.p.value", "dginn-primate_BppM1M2.NbSites",
```

```
"dginn-primate_BppM1M2.PSS", "dginn-primate_BppM7M8",
"dginn-primate_BppM7M8.p.value", "dginn-primate_BppM7M8.NbSites",
"dginn-primate_BppM7M8.PSS", "dginn-primate_codemlM1M2",
"dginn-primate_codemlM1M2.p.value", "dginn-primate_codemlM1M2.NbSites",
"dginn-primate_codemlM1M2.PSS", "dginn-primate_codemlM7M8",
"dginn-primate_codemlM7M8.p.value", "dginn-primate_codemlM7M8.NbSites",
"dginn-primate_codemlM7M8.PSS")
```

2.2 Bats

```
# original table
dginnbats<-read.delim(paste0(workdir,
                             "data/DGINN_202005281339summary_cleaned-LE201108.txt"),
                      fill=T, h=T)

# rerun on corrected alignment
dginnbatsnew<-read.delim(paste0(workdir,
                                 "data/DGINN_202011262248_hyphybpp-202012192053_codeml-summary.txt"),
                         fill=T, h=T)
```

```
# Add both columns
dginnbatsnew$Lucie.s.comments<-" "
dginnbatsnew$Action.taken<-" "

# Homogenize column names
dginnbats$BUSTED_p.value<-dginnbats$BUSTED.p.value
dginnbats$MEME_NbSites<-dginnbats$MEME.NbSites
dginnbats$MEME_PSS<-dginnbats$MEME.PSS

dginnbats$BppM1M2_p.value<-dginnbats$BppM1M2.p.value
dginnbats$BppM1M2_NbSites<-dginnbats$BppM1M2.NbSites
dginnbats$BppM1M2_PSS<-dginnbats$BppM1M2.PSS

dginnbats$BppM7M8_p.value<-dginnbats$BppM7M8.p.value
dginnbats$BppM7M8_NbSites<-dginnbats$BppM7M8.NbSites
dginnbats$BppM7M8_PSS<-dginnbats$BppM7M8.PSS
```

```

dginnbats$codemlM1M2_p.value<-dginnbats$codemlM1M2.p.value
dginnbats$codemlM1M2_NbSites<-dginnbats$codemlM1M2.NbSites
dginnbats$codemlM1M2_PSS<-dginnbats$codemlM1M2.PSS

```

```

dginnbats$codemlM7M8_p.value<-dginnbats$codemlM7M8.p.value
dginnbats$codemlM7M8_NbSites<-dginnbats$codemlM7M8.NbSites
dginnbats$codemlM7M8_PSS<-dginnbats$codemlM7M8.PSS

```

```

# Order columns in the same order in both tables
dginnbats<-dginnbats[,names(dginnbatsnew)]

```

```

names(dginnbatsnew) %in% names(dginnbats)

```

```

## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [14] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [27] TRUE TRUE TRUE

```

```

names(dginnbats)==names(dginnbatsnew)

```

```

## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [14] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [27] TRUE TRUE TRUE

```

```

# Put RIPK aside

```

```

ripk1<-dginnbatsnew[dginnbatsnew$Gene=="RIPK1",1:27]

```

```

# Add it to primate table

```

```

names(ripk1)<-names(dginnT)

```

```

ripk1$`dginn-primate_omegaM0Bpp`<-as.factor(ripk1$`dginn-primate_omegaM0Bpp`)
ripk1$`dginn-primate_BUSTED.p.value`<-as.factor(ripk1$`dginn-primate_BUSTED.p.value`)
ripk1$`dginn-primate_BppM1M2.p.value`<-as.factor(ripk1$`dginn-primate_BppM1M2.p.value`)
ripk1$`dginn-primate_BppM7M8.p.value`<-as.factor(ripk1$`dginn-primate_BppM7M8.p.value`)

```

```

dginnT<-rbind(dginnT, ripk1)

```

```

## Remove it Ripk1 from bats

```

```

dginnbatsnew<-dginnbatsnew[dginnbatsnew$Gene!="RIPK1",]

```

```

## suppress redundant lines
dginnbats<-dginnbats[(dginnbats$Gene %in% dginnbatsnew$Gene)==FALSE,]
names(dginnbatsnew)<-names(dginnbats)

## replace by new data
dginnbatsnew$omegaM0Bpp<-as.factor(dginnbatsnew$omegaM0Bpp)
dginnbatsnew$BppM1M2_p.value<-as.factor(dginnbatsnew$BppM1M2_p.value)
dginnbatsnew$BppM7M8_p.value<-as.factor(dginnbatsnew$BppM7M8_p.value)

dginnbats<-rbind(dginnbats, dginnbatsnew)

names(dginnbats)<-c("bats_File", "bats_Name", "Gene.name", paste0("bats_",
  names(dginnbats)[-(1:3)]))
names(dginnbats)

## [1] "bats_File" "bats_Name"
## [3] "Gene.name" "bats_GeneSize"
## [5] "bats_NbSpecies" "bats_omegaM0Bpp"
## [7] "bats_omegaM0codeml" "bats_BUSTED"
## [9] "bats_BUSTED_p.value" "bats_MEME_NbSites"
## [11] "bats_MEME_PSS" "bats_BppM1M2"
## [13] "bats_BppM1M2_p.value" "bats_BppM1M2_NbSites"
## [15] "bats_BppM1M2_PSS" "bats_BppM7M8"
## [17] "bats_BppM7M8_p.value" "bats_BppM7M8_NbSites"
## [19] "bats_BppM7M8_PSS" "bats_codemlM1M2"
## [21] "bats_codemlM1M2_p.value" "bats_codemlM1M2_NbSites"
## [23] "bats_codemlM1M2_PSS" "bats_codemlM7M8"
## [25] "bats_codemlM7M8_p.value" "bats_codemlM7M8_NbSites"
## [27] "bats_codemlM7M8_PSS" "bats_Lucie.s.comments"
## [29] "bats_Action.taken"

```

2.3 Merged table

```

#tidy.opts = list(width.cutoff = 60)
dim(dginnT)

## [1] 413 27

```

dginnT\$Gene.name

##	[1]	AAR2	AASS	AATF
##	[4]	ABCC1	ACAD9	ACADM
##	[7]	ACE2	ACSL3	ADAM9
##	[10]	ADAM9[0-3120]	ADAM9[3119-3927]	ADAMTS1
##	[13]	AGPS	AKAP8	AKAP8L
##	[16]	AKAP9	ALG11	ALG5
##	[19]	ALG8	ANO6	AP2A2
##	[22]	AP2M1	AP3B1	ARF6
##	[25]	ATE1	ATP13A3	ATP1B1
##	[28]	ATP5MGL	ATP6AP1	ATP6V1A
##	[31]	BAG5	BCKDK	BCS1L
##	[34]	BRD2	BRD4	BZW2
##	[37]	C1H10RF50	CCDC86	CDK5RAP2
##	[40]	CENPF	CEP112	CEP135
##	[43]	CEP135[0-3264]	CEP135[3263-3678]	CEP250
##	[46]	CEP350	CEP43	CEP68
##	[49]	CHMP2A	CHPF	CHPF2
##	[52]	CISD3	CIT	CLCC1
##	[55]	CLIP4	CNTRL	COL6A1
##	[58]	COLGALT1	COMT	COQ8B
##	[61]	COQ8A	CRTC3	CSDE1
##	[64]	CSNK2A1	CSNK2A2	CSNK2B
##	[67]	CSNK2B[0-609]	CSNK2B[608-2568]	CUL2
##	[70]	CWC27	CYB5B	CYB5R3
##	[73]	CYB5R1	DCAF7	DCAKD
##	[76]	DCTPP1	DDX10	DDX21
##	[79]	DDX21[0-717]	DDX21[716-2538]	DDX50
##	[82]	DNAJC11	DNAJC19	DNAJC15
##	[85]	DNMT1	DPH5	DPH5[0-702]
##	[88]	DPH5[701-1326]	DPY19L2	DPY19L1
##	[91]	ECSIT	EDEM3	EIF4E2
##	[94]	EIF4H	ELOC	EMC1
##	[97]	ERC1	ERGIC1	ERLEC1
##	[100]	ERMP1	ERO1B	ERP44
##	[103]	ETFA	EXOSC2	EXOSC3
##	[106]	EXOSC3[0-1446]	EXOSC3[1445-1980]	EXOSC5
##	[109]	EXOSC8	F2RL1	FAM162A

## [112]	FAM8A1	FAM98A	FAR2
## [115]	FASTKD5	FBLN5	FBN1
## [118]	FBN3	FBN2	FBXL12
## [121]	FKBP10	FKBP15	FKBP7
## [124]	FOXRED2	FYC01	G3BP1
## [127]	G3BP2	GCC1	GCC2
## [130]	GDF15	GFER	GGCX
## [133]	GGH	GHITM	GIGYF2
## [136]	GLA	GNB4	GNB2
## [139]	GNB1	GNB3	GNG5
## [142]	GNG5	GOLGA2	GOLGA3
## [145]	GOLGA7	GOLGA7 [0-312]	GOLGA7 [311-549]
## [148]	GOLGB1	GORASP1	GPAA1
## [151]	GPX1	GPX1 [0-1218]	GPX1 [1217-2946]
## [154]	GRIPAP1	GRPEL1	GTF2F2
## [157]	HDAC2	HDAC1	HEATR3
## [160]	HECTD1	HMOX1	HOOK1
## [163]	HS2ST1	HS6ST2	HS6ST3
## [166]	HSBP1	HYOU1	IDE
## [169]	IL17RA	IMPDH1	IMPDH2
## [172]	INHBE	INTS4	ITGB1
## [175]	ITGB1 [0-2328]	ITGB1 [2327-2844]	JAKMIP1
## [178]	LARP1	LARP4B	LARP7
## [181]	LMAN2	LMAN2L	LOX
## [184]	MAP7D1	MARK1	MARK2
## [187]	MARK3	MAT2B	MDN1
## [190]	MEPCE	MIB1	MIPOL1
## [193]	MOGS	MOV10	MPHOSPH10
## [196]	MRPS2	MRPS25	MRPS27
## [199]	MRPS5	MRPS5 [0-1569]	MRPS5 [1568-3783]
## [202]	MARC1	MARC2	MTCH1
## [205]	MYCBP2	MGRN1	NARS2
## [208]	NAT14	NDFIP2	NDFIP2 [0-768]
## [211]	NDFIP2 [767-1314]	NDUFAF1	NDUFAF2
## [214]	NDUFAF2 [0-258]	NDUFAF2 [257-744]	NDUFB9
## [217]	NEK9	NEU1	NGDN
## [220]	NGLY1	NIN	NINL
## [223]	NLRX1	NOL10	NPC2
## [226]	NPTX1	NSD2	NUP210

## [229]	NUP214	NUP54	NUP58
## [232]	NUP58[0-1824]	NUP58[1823-2367]	NUP62
## [235]	NUP88	NUP98	NUTF2
## [238]	OS9	PABPC3	POTPABPC1
## [241]	PABPC1	PABPC4	PABPC4L
## [244]	PABPC5	PCNT	PCSK6
## [247]	PCSK5	PDE4DIP	PDZD11
## [250]	PIGO	PIGS	PITRM1
## [253]	PKP2	PLAT	PLD3
## [256]	PLEKHA5	PLEKHF2	PLOD2
## [259]	PMPCA	PMPCB	POFUT1
## [262]	KDELC1	KDELC2	POLA1
## [265]	POLA2	POR	PPIL3
## [268]	PPT1	PRIM1	PRIM2
## [271]	PRIM2[0-1071]	PRIM2[1070-1902]	PRKACB
## [274]	PRKACG	PRKACA	PRKAR2A
## [277]	PRKAR2B	PRRC2B	PSMD8
## [280]	PTBP2	PTGES2	PTGES2[0-1587]
## [283]	PTGES2[1586-2202]	PUSL1	PVR
## [286]	QSOX2	RAB10	RAB8B
## [289]	RAB13	RAB14	RAB18
## [292]	RAB18[0-855]	RAB18[854-1815]	RAB1A
## [295]	RAB2B	RAB2A	RAB5C
## [298]	RAB5A	RAB5B	RAB7A
## [301]	RAB15	RAB8A	RAE1
## [304]	RALB	RALA	RAP1GDS1
## [307]	RBM28	RBM41	RBX1
## [310]	EZR	EZR[0-1458]	EZR[1457-3771]
## [313]	RDX	MSN	REEP5
## [316]	REEP6	RETREG3	RHOB
## [319]	RHOC	RHOA	RIPK1
## [322]	RNF41	RPL36	RRP9
## [325]	RTN4	SAAL1	SBN01
## [328]	SCAP	SCARB1	SCCPDH
## [331]	SDF2	SEPSECS	SIL1
## [334]	SIRT5	SLC25A21	SLC27A2
## [337]	SLC30A6	SLC30A7	SLC30A9
## [340]	SLC44A2	SLC44A2[0-2577]	SLC44A2[2576-3657]
## [343]	SLC9A3R1	SLU7	SMOC1

```

## [346] SNIP1          SPART          SRP19
## [349] SRP54             SRP72          SRP72[0-2604]
## [352] SRP72[2603-3417] STC2           STOM
## [355] STOM[0-1047]      STOM[1046-1800] STOML3
## [358] STOML2            SUN2           TAPT1
## [361] TARS2             TBCA           TBK1
## [364] TBKBP1            TCF12          THTPA
## [367] TIMM10            TIMM10B        TIMM29
## [370] TIMM8B            TIMM9           TLE1
## [373] TLE3              TLE4           TLE2
## [376] TLE2[0-1302]     TLE2[1301-3987] AES
## [379] TM2D3             TMED5          MEM39B
## [382] TMEM97            TMPRSS2        TOMM70
## [385] TOR1A             TOR1B          TOR1AIP1
## [388] TRIM59            TRMT1          TUBGCP2
## [391] TUBGCP3           TYSND1         UBAP2
## [394] UBAP2L            UBXN8          UGGT2
## [397] UPF1              USP54          VPS11
## [400] VPS39             WASHC4         WFS1
## [403] WFS1[0-2346]     WFS1[2345-3216] YIF1A
## [406] YIF1B             ZC3H18         ZC3H7A
## [409] ZDHHC5            ZNF318         ZNF503
## [412] ZYG11B            RIPK1
## 416 Levels: AAR2 AASS AATF ABCC1 ACAD9 ACADM ACE2 ACSL3 ... SELENOS

dim(dginnbats)

## [1] 353 29

dginnbats$Gene.name

## [1] AAR2          AASS          AATF
## [4] ABCC1         ACAD9         ACADM
## [7] ACE2          ACSL3         ADAM9
## [10] ADAM9[0-2769] ADAM9[2768-3030] ADAMTS1
## [13] AGPS          AKAP8         AKAP8L
## [16] AKAP9         ALG11         ALG5
## [19] ALG8          ANO6          AP2A2
## [22] AP2M1         AP3B1         ARF6
## [25] ARL6IP6       ATP13A3       ATP1B1

```

##	[28]	ATP5MG	ATP6AP1	ATP6V1A
##	[31]	BAG5	BCKDK	BCS1
##	[34]	BRD2	BRD4	BZW2
##	[37]	CUNH1ORF50	CCDC86	CDK5RAP2
##	[40]	CENPF	CEP112	CEP135
##	[43]	CEP250	CEP350	CEP68
##	[46]	CHMP2A	CHPF	CHPF2
##	[49]	CISD3	CIT	CLCC1
##	[52]	CLIP4	CNTRL	COLGALT1
##	[55]	COMT	CRTC3	CSDE1
##	[58]	CSNK2A2	CSNK2B	CUL2
##	[61]	CWC27	CYB5BR3	DCAF7
##	[64]	DCAKD	DCTPP1	DDX10
##	[67]	DNAJC11	DNAJC19	DNMT1
##	[70]	DPH5	DPY19L1	ECSIT
##	[73]	EDEM3	EIF4E2	EIF4H
##	[76]	ELOC	EMC1	ERC1
##	[79]	ERGIC1	ERLEC1	ERMP1
##	[82]	ERP44	EXOSC2	EXOSC3
##	[85]	EXOSC5	EXOSC8	F2RL1
##	[88]	FAM162A	FAM8A1	FAM98A
##	[91]	FAR2	FASTKD5	FBLN5
##	[94]	FBN1	FBN2	FBXL12
##	[97]	FKBP10	FKBP15	FKBP7
##	[100]	FOXRED2	FYC01	G3BP1
##	[103]	G3BP2	GCC1	GCC2
##	[106]	GDF15	GFER	GGCX
##	[109]	GGH	GHITM	GIGYF2
##	[112]	GLA	GNG5	GOLGA2
##	[115]	GOLGA3	GOLGB1	GORASP1
##	[118]	GPAA1	GPX1	GRIPAP1
##	[121]	GRPEL1	GTF2F2	HDAC2
##	[124]	HEATR3	HECTD1	HMOX1
##	[127]	HOOK1	HS2ST1	HS6ST2
##	[130]	HYOU1	IDE	IDE[0-2343]
##	[133]	IDE[2342-3240]	IDE[3239-4911]	IL17RA
##	[136]	IMPDH2	INHBE	ITGB1
##	[139]	JAKMIP1	LARP1	LARP4B
##	[142]	LARP7	LMAN2	LOX

## [145]	MAP7D1	MARK1	MARK2
## [148]	MARK3	MAT2B	MDN1
## [151]	MEPCE	MFGE8	MIB1
## [154]	MIPOL1	MOGS	MPHOSPH10
## [157]	MRPS2	MRPS25	MRPS27
## [160]	MRPS5	MTCH1	MYCBP2
## [163]	NARS2	NAT14	NDFIP2
## [166]	NDUFAF1	NDUFAF2	NDUFB9
## [169]	NEK9	NEU1	NGDN
## [172]	NGLY1	NIN	NINL
## [175]	NLRX1	NOL10	NPC2
## [178]	NPTX1	NSD2	NUP210
## [181]	NUP214	NUP54	NUP58
## [184]	NUP62	NUP88	NUP98
## [187]	NUTF2	OS9	PABPC4
## [190]	PCNT	PCSK5	PDZD11
## [193]	PIGO	PIGS	PITRM1
## [196]	PKP2	PLAT	PLD3
## [199]	PLEKHA5	PLEKHF2	PLOD2
## [202]	PMPCA	PMPCB	POFUT1
## [205]	KDELC1	KDELC2	POLA1
## [208]	POLA2	POR	PPIL3
## [211]	PPT1	PRIM1	PRIM2
## [214]	PRKACA	PRKAR2A	PRKAR2B
## [217]	PRRC2B	PSMD8	PTBP2
## [220]	PTGES2	PTGES2 [0-513]	PTGES2 [512-2070]
## [223]	PUSL1	PVR	QSOX2
## [226]	RAB10	RAB14	RAB18
## [229]	RAB1A	RAB2A	RAB5C
## [232]	RAB7A	RAB8A	RAE1
## [235]	RALA	RAP1GDS1	RBM28
## [238]	RBM41	RBX1	REEP5
## [241]	REEP6	RETREG3	RHOA
## [244]	RIPK1	RNF41	RPL36
## [247]	RRP9	RTN4	SAAL1
## [250]	SBN01	SCAP	SCARB1
## [253]	SCARB1 [0-2004]	SCARB1 [2003-2289]	SCCPDH
## [256]	SELENOS [0-927]	SELENOS [926-1137]	SEPSECS
## [259]	SIGMAR1	SIL1	SIRT5

```

## [262] SLC25A21          SLC27A2          SLC30A6
## [265] SLC30A7             SLC30A9          SLC44A2
## [268] SLC44A2[0-2820]     SLC44A2[2819-3792] SLC9A3R1
## [271] SLU7                SMOC1            SNIP1
## [274] SPART              SRP19            SRP54
## [277] SRP72              STC2             STOM
## [280] STOML2             SUN2             TAPT1
## [283] TBK1               TBKBP1           TCF12
## [286] THTPA              TIMM10           TIMM10B
## [289] TIMM29             TIMM8B           TIMM9
## [292] TLE1               TLE3             TLE5
## [295] TM2D3              TMED5            MEM97
## [298] TOMM70             TOR1A            TOR1AIP1
## [301] TRIM59             TRMT1            TUBGCP2
## [304] TUBGCP3            UBAP2            UBAP2L
## [307] UBXN8              UGGT2            UPF1
## [310] USP13              USP54            VPS11
## [313] VPS39              WASHC4           WFS1
## [316] YIF1A              ZC3H18           ZC3H18[0-1101]
## [319] ZC3H18[1100-3678] ZC3H7A           ZDHHC5
## [322] ZNF318             ZNF503           ZYG11B
## [325] ATE1               FGFR10P          COL6A1
## [328] COQ8B              CYB5B            DDX21
## [331] ELOB               ERO1B            ETFA
## [334] GNB1               GOLGA7           HSBP1
## [337] INTS4              MOV10            MARC1
## [340] PABPC1             PCSK6            PDE4DIP
## [343] RDX                REEP6-A          REEP6-B
## [346] SDF2               SELENOS          TARS2
## [349] TBCA               TMEM39B          TMPRSS2
## [352] TMPRSS2            TYSND1
## 352 Levels: AAR2 AASS AATF ABCC1 ACAD9 ACADM ACE2 ACSL3 ... REEP6-B

```

Manual corrections:

TMPRSS2 in bats

```

dginnbats[dginnbats$Gene.name=="TMPRSS2",]

##                                bats_File bats_Name Gene.name
## 2810          TMPRSS2_bat_same_mafft_prank    TMPRSS2    TMPRSS2

```

```

## 2910 TMPRSS2_bat_select_cut_mafft_prank    TMPRSS2    TMPRSS2
##      bats_GeneSize bats_NbSpecies    bats_omegaM0Bpp
## 2810          1174          12 0.140290584008726
## 2910          574          12 0.129489038364869
##      bats_omegaM0codeml bats_BUSTED bats_BUSTED_p.value
## 2810          0.145          N          0.9333
## 2910          0.127          N          0.9358
##      bats_MEME_NbSites
## 2810          12
## 2910          19
##
## 2810          630, 644, 649, 688, 775, 888, 921, 1003, 1051, 105
## 2910 59, 73, 78, 108, 115, 117, 121, 133, 144, 241, 259, 288, 321, 403, 421, 451,
##      bats_BppM1M2 bats_BppM1M2_p.value bats_BppM1M2_NbSites
## 2810          N    0.999999010422051          0
## 2910          N    0.999999906049202          0
##      bats_BppM1M2_PSS bats_BppM7M8 bats_BppM7M8_p.value
## 2810          na          N    0.621882294670985
## 2910          na          N    0.334893426994811
##      bats_BppM7M8_NbSites bats_BppM7M8_PSS bats_codemlM1M2
## 2810          0          na          N
## 2910          0          na          N
##      bats_codemlM1M2_p.value bats_codemlM1M2_NbSites
## 2810          1.0          0
## 2910          1.0          0
##      bats_codemlM1M2_PSS bats_codemlM7M8 bats_codemlM7M8_p.value
## 2810          na          N    0.788991288016829
## 2910          na          N    0.4210515526274131
##      bats_codemlM7M8_NbSites bats_codemlM7M8_PSS
## 2810          0          na
## 2910          0          na
##      bats_Lucie.s.comments bats_Action.taken
## 2810
## 2910

# keeping the uncut one
# renaming the other one TMPRSS2_cut
dginnbats$Gene.name<-as.character(dginnbats$Gene.name)
dginnbats[dginnbats$bats_File=="TMPRSS2_bat_select_cut_mafft_prank", "Gene.name"]<-"TM

```

RIPK1: ANcestral version kept, suppress it "RIPK1_sequences_filtered_longestORFs_mafft_mincov_p

```
dginnT<-dginnT[dginnT$File!="RIPK1_sequences_filtered_longestORFs_mafft_mincov_prank"]
```

REEP6 eA et B

```
dginnbats$Gene.name<-as.character(dginnbats$Gene.name)
dginnbats[dginnbats$bats_File=="REEP6_sequences_filtered_longestORFs_D210gp1_prank",
dginnbats[dginnbats$bats_File=="REEP6_LA_bat_select_mafft_prank", "Gene.name"]<-"REEP
dginnbats[dginnbats$bats_File=="REEP6_LB_bat_select_mafft_prank", "Gene.name"]<-"REEP
```

GNG5

```
dginnT$Gene.name<-as.character(dginnT$Gene.name)
dginnT[dginnT$File=="GNG5_sequences_filtered_longestORFs_D189gp2_prank", "Gene.name"]
```

```
dim(dginnbats)

## [1] 353 29

dim(dginnT)

## [1] 412 27

# genes in common
common<-dginnT$Gene.name[dginnT$Gene.name %in% dginnbats$Gene.name]
common

## [1] "AAR2" "AASS" "AATF" "ABCC1" "ACAD9"
## [6] "ACADM" "ACE2" "ACSL3" "ADAM9" "ADAMTS1"
## [11] "AGPS" "AKAP8" "AKAP8L" "AKAP9" "ALG11"
## [16] "ALG5" "ALG8" "ANO6" "AP2A2" "AP2M1"
## [21] "AP3B1" "ARF6" "ATE1" "ATP13A3" "ATP1B1"
## [26] "ATP6AP1" "ATP6V1A" "BAG5" "BCKDK" "BRD2"
## [31] "BRD4" "BZW2" "CCDC86" "CDK5RAP2" "CENPF"
## [36] "CEP112" "CEP135" "CEP250" "CEP350" "CEP68"
## [41] "CHMP2A" "CHPF" "CHPF2" "CISD3" "CIT"
## [46] "CLCC1" "CLIP4" "CNTRL" "COL6A1" "COLGALT1"
```

##	[51]	"COMT"	"COQ8B"	"CRTC3"	"CSDE1"	"CSNK2A2"
##	[56]	"CSNK2B"	"CUL2"	"CWC27"	"CYB5B"	"DCAF7"
##	[61]	"DCAKD"	"DCTPP1"	"DDX10"	"DDX21"	"DNAJC11"
##	[66]	"DNAJC19"	"DNMT1"	"DPH5"	"DPY19L1"	"ECSIT"
##	[71]	"EDEM3"	"EIF4E2"	"EIF4H"	"ELOC"	"EMC1"
##	[76]	"ERC1"	"ERGIC1"	"ERLEC1"	"ERMP1"	"ER01B"
##	[81]	"ERP44"	"ETFA"	"EXOSC2"	"EXOSC3"	"EXOSC5"
##	[86]	"EXOSC8"	"F2RL1"	"FAM162A"	"FAM8A1"	"FAM98A"
##	[91]	"FAR2"	"FASTKD5"	"FBLN5"	"FBN1"	"FBN2"
##	[96]	"FBXL12"	"FKBP10"	"FKBP15"	"FKBP7"	"FOXRED2"
##	[101]	"FYCO1"	"G3BP1"	"G3BP2"	"GCC1"	"GCC2"
##	[106]	"GDF15"	"GFER"	"GGCX"	"GGH"	"GHITM"
##	[111]	"GIGYF2"	"GLA"	"GNB1"	"GNG5"	"GOLGA2"
##	[116]	"GOLGA3"	"GOLGA7"	"GOLGB1"	"GORASP1"	"GPAA1"
##	[121]	"GPX1"	"GRIPAP1"	"GRPEL1"	"GTF2F2"	"HDAC2"
##	[126]	"HEATR3"	"HECTD1"	"HMOX1"	"HOOK1"	"HS2ST1"
##	[131]	"HS6ST2"	"HSBP1"	"HYOU1"	"IDE"	"IL17RA"
##	[136]	"IMPDH2"	"INHBE"	"INTS4"	"ITGB1"	"JAKMIP1"
##	[141]	"LARP1"	"LARP4B"	"LARP7"	"LMAN2"	"LOX"
##	[146]	"MAP7D1"	"MARK1"	"MARK2"	"MARK3"	"MAT2B"
##	[151]	"MDN1"	"MEPCE"	"MIB1"	"MIPOL1"	"MOGS"
##	[156]	"MOV10"	"MPHOSPH10"	"MRPS2"	"MRPS25"	"MRPS27"
##	[161]	"MRPS5"	"MARC1"	"MTCH1"	"MYCBP2"	"NARS2"
##	[166]	"NAT14"	"NDFIP2"	"NDUFAF1"	"NDUFAF2"	"NDUFB9"
##	[171]	"NEK9"	"NEU1"	"NGDN"	"NGLY1"	"NIN"
##	[176]	"NINL"	"NLRX1"	"NOL10"	"NPC2"	"NPTX1"
##	[181]	"NSD2"	"NUP210"	"NUP214"	"NUP54"	"NUP58"
##	[186]	"NUP62"	"NUP88"	"NUP98"	"NUTF2"	"OS9"
##	[191]	"PABPC1"	"PABPC4"	"PCNT"	"PCSK6"	"PCSK5"
##	[196]	"PDE4DIP"	"PDZD11"	"PIGO"	"PIGS"	"PITRM1"
##	[201]	"PKP2"	"PLAT"	"PLD3"	"PLEKHA5"	"PLEKHF2"
##	[206]	"PLOD2"	"PMPCA"	"PMPCB"	"POFUT1"	"KDELCL1"
##	[211]	"KDELCL2"	"POLA1"	"POLA2"	"POR"	"PPIL3"
##	[216]	"PPT1"	"PRIM1"	"PRIM2"	"PRKACA"	"PRKAR2A"
##	[221]	"PRKAR2B"	"PRRC2B"	"PSMD8"	"PTBP2"	"PTGES2"
##	[226]	"PUSL1"	"PVR"	"QS0X2"	"RAB10"	"RAB14"
##	[231]	"RAB18"	"RAB1A"	"RAB2A"	"RAB5C"	"RAB7A"
##	[236]	"RAB8A"	"RAE1"	"RALA"	"RAP1GDS1"	"RBM28"
##	[241]	"RBM41"	"RBX1"	"RDX"	"REEP5"	"REEP6"


```

## [246] "RETREG3" "RHOA" "RNF41" "RPL36" "RRP9"
## [251] "RTN4" "SAAL1" "SBN01" "SCAP" "SCARB1"
## [256] "SCCPDH" "SDF2" "SEPSECS" "SIL1" "SIRT5"
## [261] "SLC25A21" "SLC27A2" "SLC30A6" "SLC30A7" "SLC30A9"
## [266] "SLC44A2" "SLC9A3R1" "SLU7" "SMOC1" "SNIP1"
## [271] "SPART" "SRP19" "SRP54" "SRP72" "STC2"
## [276] "STOM" "STOML2" "SUN2" "TAPT1" "TARS2"
## [281] "TBCA" "TBK1" "TBKBP1" "TCF12" "THTPA"
## [286] "TIMM10" "TIMM10B" "TIMM29" "TIMM8B" "TIMM9"
## [291] "TLE1" "TLE3" "TM2D3" "TMED5" "TMEM39B"
## [296] "TMEM97" "TMPRSS2" "TOMM70" "TOR1A" "TOR1AIP1"
## [301] "TRIM59" "TRMT1" "TUBGCP2" "TUBGCP3" "TYSND1"
## [306] "UBAP2" "UBAP2L" "UBXN8" "UGGT2" "UPF1"
## [311] "USP54" "VPS11" "VPS39" "WASHC4" "WFS1"
## [316] "YIF1A" "ZC3H18" "ZC3H7A" "ZDHHC5" "ZNF318"
## [321] "ZNF503" "ZYG11B" "RIPK1"

length(dginnT$Gene.name[dginnT$Gene.name %in% dginnbats$Gene.name])

## [1] 323

# genes only in primates
onlyprimates<-dginnT$Gene.name[(dginnT$Gene.name %in% dginnbats$Gene.name)==FALSE]
onlyprimates

## [1] "ADAM9[0-3120] " "ADAM9[3119-3927] " "ATP5MGL"
## [4] "BCS1L" "C1H10RF50" "CEP135[0-3264] "
## [7] "CEP135[3263-3678] " "CEP43" "COQ8A"
## [10] "CSNK2A1" "CSNK2B[0-609] " "CSNK2B[608-2568] "
## [13] "CYB5R3" "CYB5R1" "DDX21[0-717] "
## [16] "DDX21[716-2538] " "DDX50" "DNAJC15"
## [19] "DPH5[0-702] " "DPH5[701-1326] " "DPY19L2"
## [22] "EXOSC3[0-1446] " "EXOSC3[1445-1980] " "FBN3"
## [25] "GNB4" "GNB2" "GNB3"
## [28] "GNG5_like" "GOLGA7[0-312] " "GOLGA7[311-549] "
## [31] "GPX1[0-1218] " "GPX1[1217-2946] " "HDAC1"
## [34] "HS6ST3" "IMPDH1" "ITGB1[0-2328] "
## [37] "ITGB1[2327-2844] " "LMAN2L" "MRPS5[0-1569] "
## [40] "MRPS5[1568-3783] " "MARC2" "MGRN1"
## [43] "NDFIP2[0-768] " "NDFIP2[767-1314] " "NDUFAB2[0-258] "

```

```

## [46] "NDUFAF2[257-744] " "NUP58[0-1824] " "NUP58[1823-2367] "
## [49] "PABPC3" "POTPABPC1" "PABPC4L"
## [52] "PABPC5" "PRIM2[0-1071] " "PRIM2[1070-1902] "
## [55] "PRKACB" "PRKACG" "PTGES2[0-1587] "
## [58] "PTGES2[1586-2202] " "RAB8B" "RAB13"
## [61] "RAB18[0-855] " "RAB18[854-1815] " "RAB2B"
## [64] "RAB5A" "RAB5B" "RAB15"
## [67] "RALB" "EZR" "EZR[0-1458] "
## [70] "EZR[1457-3771] " "MSN" "RHOB"
## [73] "RHOC" "SLC44A2[0-2577] " "SLC44A2[2576-3657] "
## [76] "SRP72[0-2604] " "SRP72[2603-3417] " "STOM[0-1047] "
## [79] "STOM[1046-1800] " "STOML3" "TLE4"
## [82] "TLE2" "TLE2[0-1302] " "TLE2[1301-3987] "
## [85] "AES" "TOR1B" "WFS1[0-2346] "
## [88] "WFS1[2345-3216] " "YIF1B"

length(dginnT$Gene.name[(dginnT$Gene.name %in% dginnbats$Gene.name)==FALSE])

## [1] 89

# genes only in bats
onlybats<-dginnbats$Gene.name[(dginnbats$Gene.name %in% dginnT$Gene.name)==FALSE]
onlybats

## [1] "ADAM9[0-2769] " "ADAM9[2768-3030] " "ARL6IP6"
## [4] "ATP5MG" "BCS1" "CUNH1ORF50"
## [7] "CYB5BR3" "IDE[0-2343] " "IDE[2342-3240] "
## [10] "IDE[3239-4911] " "MFGE8" "PTGES2[0-513] "
## [13] "PTGES2[512-2070] " "REEP6_old" "SCARB1[0-2004] "
## [16] "SCARB1[2003-2289] " "SELENOS[0-927] " "SELENOS[926-1137] "
## [19] "SIGMAR1" "SLC44A2[0-2820] " "SLC44A2[2819-3792] "
## [22] "TLE5" "USP13" "ZC3H18[0-1101] "
## [25] "ZC3H18[1100-3678] " "FGFR10P" "ELOB"
## [28] "REEP6_like" "SELENOS" "TMPRSS2_cut"

length(dginnbats$Gene.name[(dginnbats$Gene.name %in% dginnT$Gene.name)==FALSE])

## [1] 30

```

```

tab<-merge(dginnT, dginnbats, by="Gene.name", all.x=T, all.y=T)
dim(tab)

## [1] 442 55

# add column "shared"/"only bats"/"only primates"
tab$status<-""
tab$status[tab$Gene.name %in% common]<-"shared"
tab$status[tab$Gene.name %in% onlyprimates]<-"onlyprimates"
tab$status[tab$Gene.name %in% onlybats]<-"onlybats"
table(tab$status)

##
##      onlybats onlyprimates      shared
##           30           89          323

write.table(tab, "covid_comp_alldginn.txt", sep="\t")

```

3 Complete data

Merge the previous tab with J Young's original table. **Will replace the 1st part of this script**

```

young<-read.delim(paste0(workdir,
  "data/COVID_PAMLresults_332hits_plusBatScreens_2020_Apr14.csv"),
  fill=T, h=T, dec=",")
dim(young)

## [1] 332 84

```

How many genes in the Young table are not in the DGINN table. And who are they?

```

table(young$Gene.name %in% tab$Gene.name)

##
## FALSE  TRUE
##    15   317

```

```

young[(young$Gene.name %in% tab$Gene.name)==FALSE, "Gene.name"]

## [1] C19orf52 ERO1LB ATP5L NUPL1 ADCK4 SPG20
## [7] WHSC1 FAM134C 01/03/2020 VIMP TCEB1 C1orf50
## [13] TOMM70A KIAA1033 TCEB2
## 332 Levels: 01/03/2020 AAR2 AASS AATF ABCC1 ACAD9 ACADM ... ZYG11B

```

Merge them and keep only the krogan genes

```

tablo<-merge(young, tab, by="Gene.name", all.x=TRUE)

write.table(tablo, "covid_comp_complete.txt", row.names=FALSE, quote=TRUE, sep="\t")

```