

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

Marie Cariou

Janvier 2021

Contents

1	1st table	2
1.1	Primates	2
1.2	Bats	3
1.3	Merged table	6
2	Complete data	12
2.1	Read the original Young table	12
2.2	Read the gene names conversion table	12
2.3	Merge Young and DGINN table	13

1 1st table

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

1.1 Primates

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

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

dim(dginnT)

## [1] 412 27

#names(dginnT)

# Rename the columns to include primate
names(dginnT)<-c("File", "Name", "Gene.name", "GeneSize",
                 "dginn-primate_NbSpecies", "dginn-primate_omegaMOBpp",
                 "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")
```

Add SELENOS

```
selenos<-read.delim(paste0(workdir,
                           "data/resSELENOS.tab"))
```

```

# liste of colonne

colonnes<-c("File", "Name", "Gene", "GeneSize",
  "NbSpecies", "omegaM0Bpp", "omegaM0codeml", "BUSTED",
  "BUSTED_p.value", "MEME_NbSites", "MEME_PSS", "BppM1M2",
  "BppM1M2_p.value", "BppM1M2_NbSites", "BppM1M2_PSS", "BppM7M8",
  "BppM7M8_p.value", "BppM7M8_NbSites", "BppM7M8_PSS", "codemlM1M2",
  "codemlM1M2_p.value", "codemlM1M2_NbSites", "codemlM1M2_PSS", "codemlM7M8",
  "codemlM7M8_p.value", "codemlM7M8_NbSites", "codemlM7M8_PSS")

selenos<-selenos[,colonnes]

```

```

names(selenos)<-names(dginnT)
selenos[,6]<-as.factor(selenos[,6])
selenos[,9]<-as.factor(selenos[,9])
selenos[,11]<-as.factor(selenos[,11])

selenos[,13]<-as.factor(selenos[,13])
selenos[,17]<-as.factor(selenos[,17])
selenos[,21]<-as.factor(selenos[,21])
selenos[,25]<-as.factor(selenos[,25])

## convertir les pvalues
dginnT<-rbind(dginnT, selenos)

```

1.2 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 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 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"           "Gene.name"
##   [6] "bats_omegaM0Bpp"     "bats_omegaM0codeml"  "bats_BUSTED"
##  [11] "bats_MEME_PSS"       "bats_BppM1M2"        "bats_BppM1M2_p.value"
##  [16] "bats_BppM7M8"        "bats_BppM7M8_p.value" "bats_BppM7M8_NbSites"
##  [21] "bats_codemlM1M2_p.value" "bats_codemlM1M2_NbSites" "bats_codemlM1M2_PSS"
##  [26] "bats_codemlM7M8_NbSites" "bats_codemlM7M8_PSS" "bats_Lucie.s.comments"

```

1.3 Merged table

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

## [1] 414 27

#dginnT$Gene.name
dim(dginnbats)

## [1] 353 29

#dginnbats$Gene.name
```

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 0.99999906049202           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] 413 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"	"PIG0"	"PIGS"	"PITRM1"
##	[201]	"PKP2"	"PLAT"	"PLD3"	"PLEKHA5"	"PLEKHF2"
##	[206]	"PLOD2"	"PMPCA"	"PMPCB"	"POFUT1"	"KDEL1C1"
##	[211]	"KDEL1C2"	"POLA1"	"POLA2"	"POR"	"PPIL3"
##	[216]	"PPT1"	"PRIM1"	"PRIM2"	"PRKACA"	"PRKAR2A"
##	[221]	"PRKAR2B"	"PRRC2B"	"PSMD8"	"PTBP2"	"PTGES2"
##	[226]	"PUSL1"	"PVR"	"QSOX2"	"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"	"SELENOS"	"RIPK1"	

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

```
## [1] 324

# 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] " "NDUFAF2[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" "CUNH10RF50"
## [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" "TMPRSS2_cut"

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

## [1] 29

```

```

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
##           29           89          324

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

```

2 Complete data

Merge the previous tab with J Young's original table.

2.1 Read the original Young table

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

## [1] 332 84

young$PreyGene<-as.character(young$PreyGene)
young$PreyGene[young$PreyGene=="MTARC1"]<-"MARC1"
```

2.2 Read the gene names conversion table

```
usthem<-read.delim(paste0(workdir,
  "/data/table_gene_name_correspondence.csv"),
  h=T, sep=";")

young[young$PreyGene %in% usthem$Us, c("PreyGene", "Gene.name")]

##      PreyGene  Gene.name
## 57    TIMM29    C19orf52
## 107   ER01B     ER01LB
## 111   NUP58     NUPL1
## 115   COQ8B     ADCK4
## 118   SPART     SPG20
## 131   NSD2      WHSC1
## 149  RETREG3    FAM134C
## 158   MARC1    01/03/2020
## 197   ELOC      TCEB1
## 268  TOMM70     TOMM70A
## 269  WASHC4     KIAA1033

usthem[order(usthem$Us),]
```

```
##           Us      Else
## 1      COQ8B      ADCK4
## 2        ELOC      TCEB1
## 3      ER01B      ER01LB
## 4      MARC1      MTARC1
## 5        NSD2      WHSC1
## 6      NUP58      NUPL1
## 7      PCSK5
## 8 RETREG3      FAM134C
## 9      SPART      SPG20
## 10 TIMM29      C19orf52
## 11 TOMM70      TOMM70A
## 12 WASHC4      KIAA1033
```

2.3 Merge Young and DGINN table

Based on which column?

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

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

##
## FALSE  TRUE
##      3   329

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

## [1] "POGLUT3" "POGLUT2" "C1orf50"

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

## [1] "ACE2" "ADAM9 [0-2769] " "ADAM9 [0-3120] "
## [4] "ADAM9 [2768-3030] " "ADAM9 [3119-3927] " "AES"
## [7] "ATP5MGL" "BCS1" "C1H10RF50"
## [10] "CEP135 [0-3264] " "CEP135 [3263-3678] " "COQ8A"
## [13] "CSNK2A1" "CSNK2B [0-609] " "CSNK2B [608-2568] "
## [16] "CUNH10RF50" "CYB5BR3" "CYB5R1"
## [19] "DDX21 [0-717] " "DDX21 [716-2538] " "DDX50"
## [22] "DNAJC15" "DPH5 [0-702] " "DPH5 [701-1326] "
```

##	[25]	"DPY19L2"	"EXOSC3[0-1446] "	"EXOSC3[1445-1980] "
##	[28]	"EZR"	"EZR[0-1458] "	"EZR[1457-3771] "
##	[31]	"FBN3"	"FGFR10P"	"GNB2"
##	[34]	"GNB3"	"GNB4"	"GNG5_like"
##	[37]	"GOLGA7[0-312] "	"GOLGA7[311-549] "	"GPX1[0-1218] "
##	[40]	"GPX1[1217-2946] "	"HDAC1"	"HS6ST3"
##	[43]	"IDE[0-2343] "	"IDE[2342-3240] "	"IDE[3239-4911] "
##	[46]	"IMPDH1"	"ITGB1[0-2328] "	"ITGB1[2327-2844] "
##	[49]	"KDELC1"	"KDELC2"	"LMAN2L"
##	[52]	"MARC2"	"MGRN1"	"MRPS5[0-1569] "
##	[55]	"MRPS5[1568-3783] "	"MSN"	"NDFIP2[0-768] "
##	[58]	"NDFIP2[767-1314] "	"NDUFAF2[0-258] "	"NDUFAF2[257-744] "
##	[61]	"NUP58[0-1824] "	"NUP58[1823-2367] "	"PABPC3"
##	[64]	"PABPC4L"	"PABPC5"	"PCSK5"
##	[67]	"POTPABPC1"	"PRIM2[0-1071] "	"PRIM2[1070-1902] "
##	[70]	"PRKACB"	"PRKACG"	"PTGES2[0-1587] "
##	[73]	"PTGES2[0-513] "	"PTGES2[1586-2202] "	"PTGES2[512-2070] "
##	[76]	"RAB13"	"RAB15"	"RAB18[0-855] "
##	[79]	"RAB18[854-1815] "	"RAB2B"	"RAB5A"
##	[82]	"RAB5B"	"RAB8B"	"RALB"
##	[85]	"REEP6_like"	"REEP6_old"	"RHOB"
##	[88]	"RHOC"	"SCARB1[0-2004] "	"SCARB1[2003-2289] "
##	[91]	"SELENOS[0-927] "	"SELENOS[926-1137] "	"SLC44A2[0-2577] "
##	[94]	"SLC44A2[0-2820] "	"SLC44A2[2576-3657] "	"SLC44A2[2819-3792] "
##	[97]	"SRP72[0-2604] "	"SRP72[2603-3417] "	"STOM[0-1047] "
##	[100]	"STOM[1046-1800] "	"STOML3"	"TLE2"
##	[103]	"TLE2[0-1302] "	"TLE2[1301-3987] "	"TLE4"
##	[106]	"TMPRSS2"	"TMPRSS2_cut"	"TOR1B"
##	[109]	"WFS1[0-2346] "	"WFS1[2345-3216] "	"YIF1B"
##	[112]	"ZC3H18[0-1101] "	"ZC3H18[1100-3678] "	

Merge them and keep only the krogan genes

```
# creation of a dedicated column
young$merge.Gene<-young$PreyGene
tab$merge.Gene<-tab$Gene.name
tablo<-merge(young, tab, by="merge.Gene", all.x=TRUE)

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