

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

Marie Cariou

March 2021

Contents

1	Data	2
2	Comparison of dataset	2
2.1	Data	2
2.2	Omega plot	3
2.3	Mondrian	4
2.4	subsetR	7
3	Which are these genes?	10
3.1	Gene under positive selection in both bats and primates . . .	10
3.2	Gene under positive selection only in primates	12
3.3	Gene under positive selection only in bats	18
3.4	Figure tableau	20
4	Tanglegram	27

1 Data

Analysis were formatted by the script covid_comp_script0_table.Rnw.

```
home<-"/home/adminmarie/Documents/"
workdir<-paste0(home, "CIRI_BIBS_projects/2020_05_Etienne_covid/")

tab<-read.delim(paste0(workdir,
  "covid_comp/covid_comp_complete.txt"), h=T, sep="\t")
dim(tab)
```

```
home<-"/home/adminmarie/Documents/"
workdir<-paste0(home, "CIRI_BIBS_projects/2020_05_Etienne_covid/")

tab<-read.delim(paste0(workdir,
  "covid_comp/covid_comp_alldginn.txt"), h=T, sep="\t")
dim(tab)

## [1] 442 56
```

2 Comparison of dataset

2.1 Data

```
tmp<-na.omit(tab[,c("Gene.name", "bats_BUSTED", "bats_BppM1M2",
  "bats_BppM7M8", "bats_codemlM1M2", "bats_codemlM7M8",
  "dginn.primate_codemlM1M2", "dginn.primate_codemlM7M8",
  "dginn.primate_BppM1M2", "dginn.primate_BppM7M8",
  "dginn.primate_BUSTED")])
col<-c("Gene.name", "bats_BUSTED", "bats_BppM1M2",
  "bats_BppM7M8", "bats_codemlM1M2", "bats_codemlM7M8",
  "dginn.primate_codemlM1M2", "dginn.primate_codemlM7M8",
  "dginn.primate_BppM1M2", "dginn.primate_BppM7M8",
  "dginn.primate_BUSTED")
dim(tmp)

## [1] 324 11
```

2.2 Omega plot

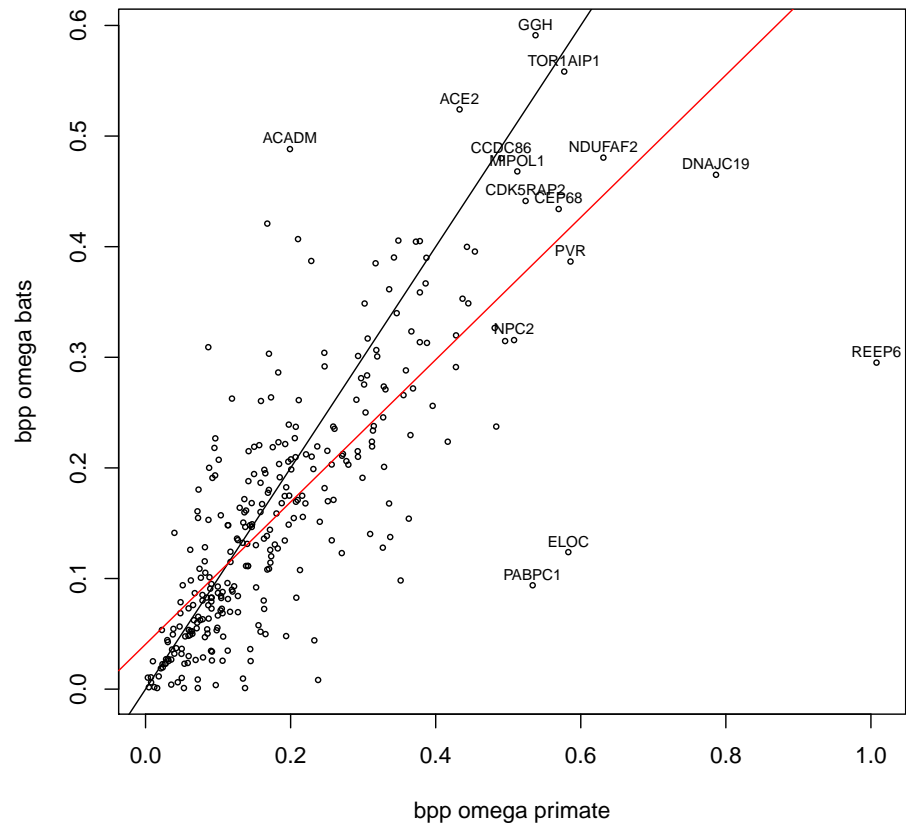
```
tab$dginn.primate_omegaM0Bpp[tab$dginn.primate_omegaM0Bpp=="na"]<-NA
x=as.numeric(as.character(
  tab$dginn.primate_omegaM0Bpp[tab$status=="shared"]))

tab$bats_omegaM0Bpp[tab$bats_omegaM0Bpp=="na"]<-NA
y=as.numeric(as.character(
  tab$bats_omegaM0Bpp[tab$status=="shared"]))

names(x)<-tab$Gene.name[tab$status=="shared"]

plot(x,y, xlab="bpp omega primate", ylab="bpp omega bats", cex=0.5)
abline(0,1)
abline(lm(y~x), col="red")

text(x[x>0.5 &y<0.4], (y[x>0.5 &y<0.4]+0.01),
      names(x)[x>0.5 &y<0.4], cex=0.7)
text(x[x<0.45 &y>0.45], (y[x<0.45 &y>0.45]+0.01),
      names(x)[x<0.45 &y>0.45], cex=0.7)
text(x[x>0.45 &y>0.4], (y[x>0.45 &y>0.4]+0.01),
      names(x)[x>0.45 &y>0.4], cex=0.7)
```



2.3 Mondrian

```
library(Mondrian)

monddata<-as.data.frame(tmp$Gene.name)

batstmp<-rowSums(cbind(tmp$bats_codemlM1M2=="Y",
                        tmp$bats_codemlM7M8=="Y",
                        tmp$bats_BppM1M2=="Y",
                        tmp$bats_BppM7M8=="Y",
                        tmp$bats_BUSTED=="Y"))
```

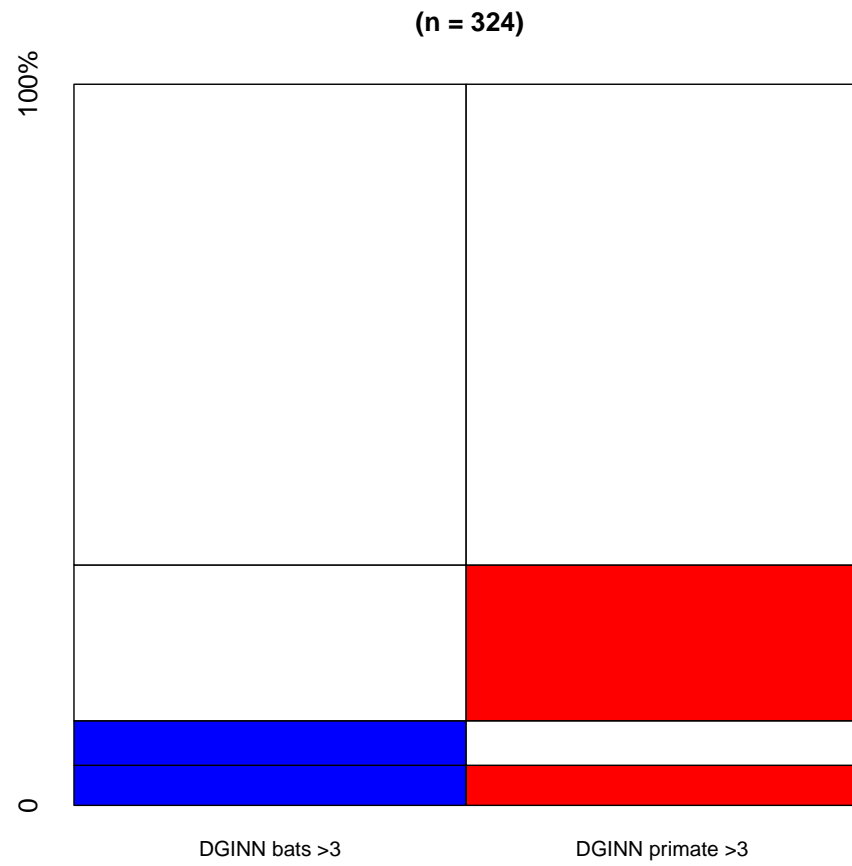
```

primate$tmp<-rowSums(cbind(tmp$"dginn.primate_codemlM1M2"=="Y",
                           tmp$"dginn.primate_codemlM7M8"=="Y",
                           tmp$"dginn.primate_BppM1M2"=="Y",
                           tmp$"dginn.primate_BppM7M8"=="Y",
                           tmp$"dginn.primate_BUSTED"=="Y"))

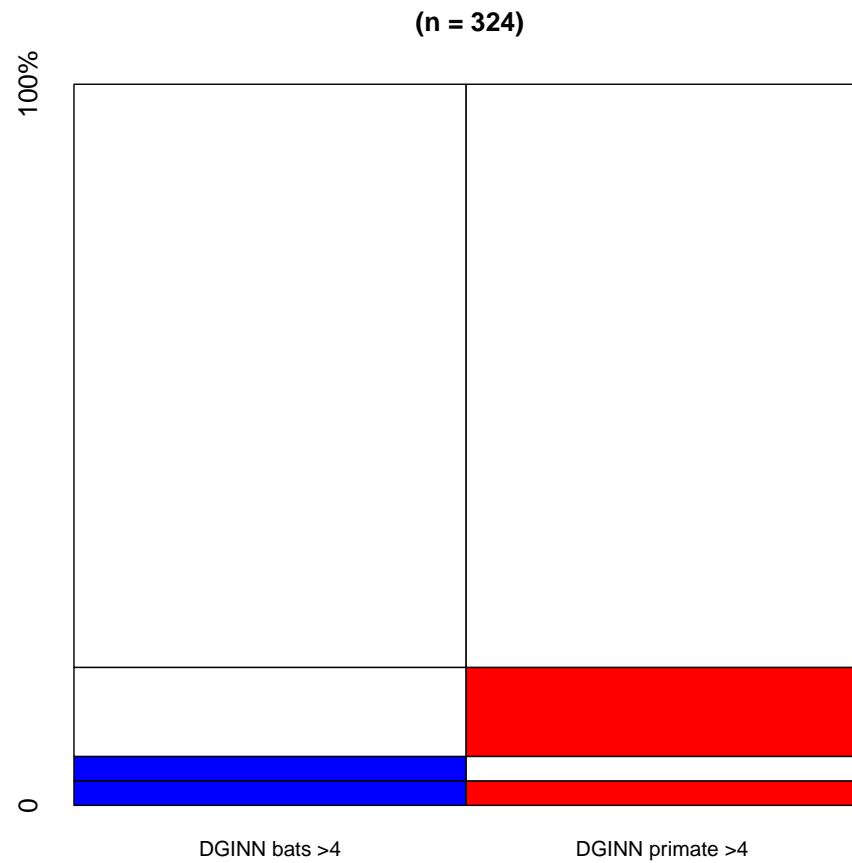
monddata$bats_dginn3<-ifelse(bat$tmp>=3, 1,0)
monddata$primate_dginn3<-ifelse(primate$tmp>=3, 1,0)
monddata$bats_dginn4<-ifelse(bat$tmp>=4, 1,0)
monddata$primate_dginn4<-ifelse(primate$tmp>=4, 1,0)

mondrian(monddata[,2:3],
         labels=c("DGINN bats >3", "DGINN primate >3"))

```



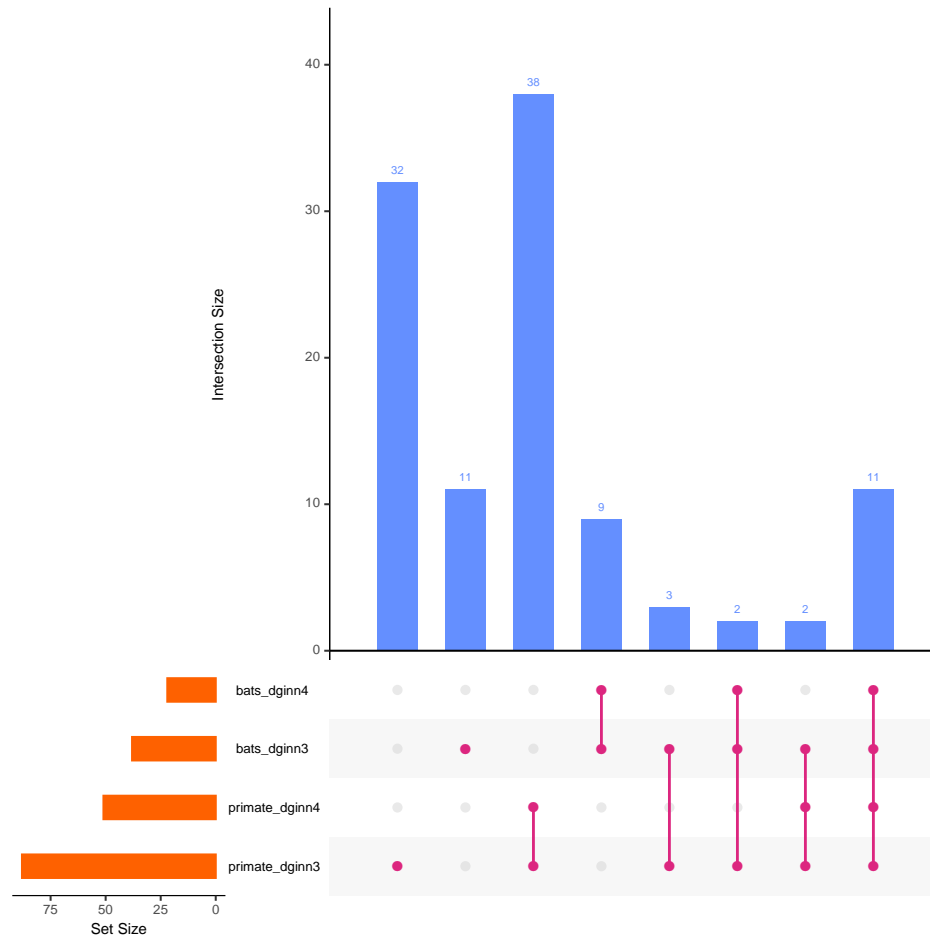
```
mondrian(monddata[,4:5],  
         labels=c("DGINN bats >4", "DGINN primate >4"))
```



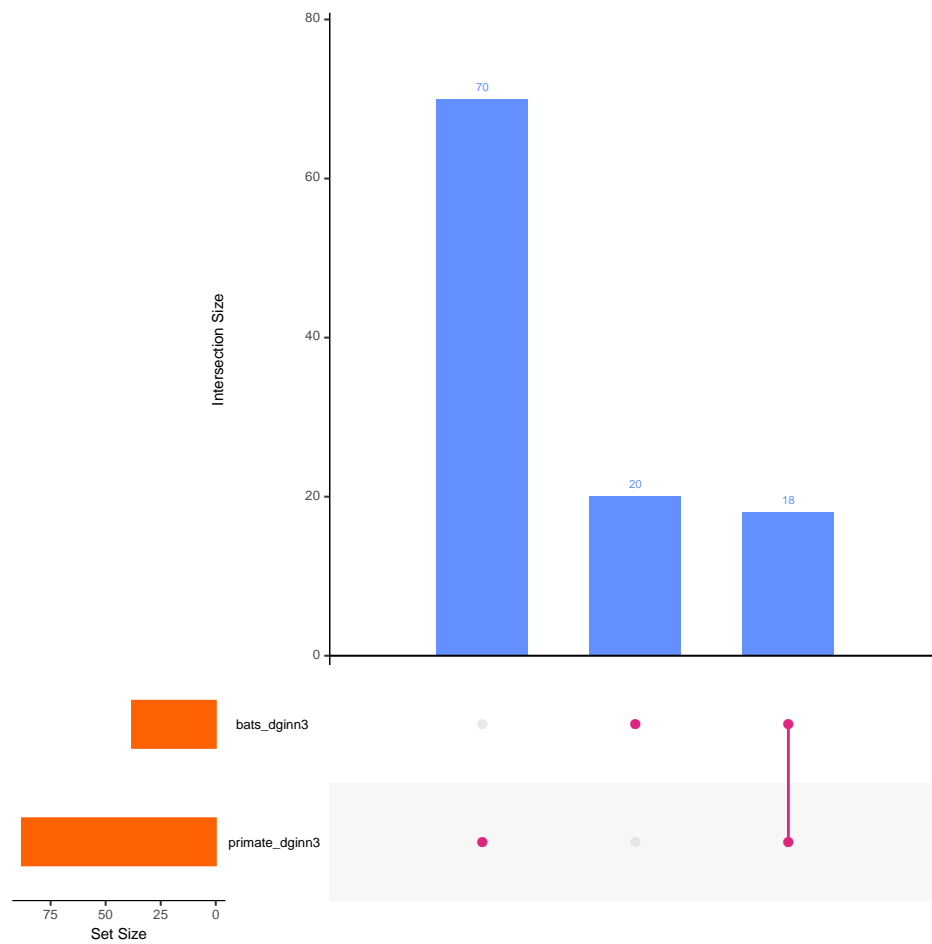
2.4 subsetR

```
library(UpSetR)

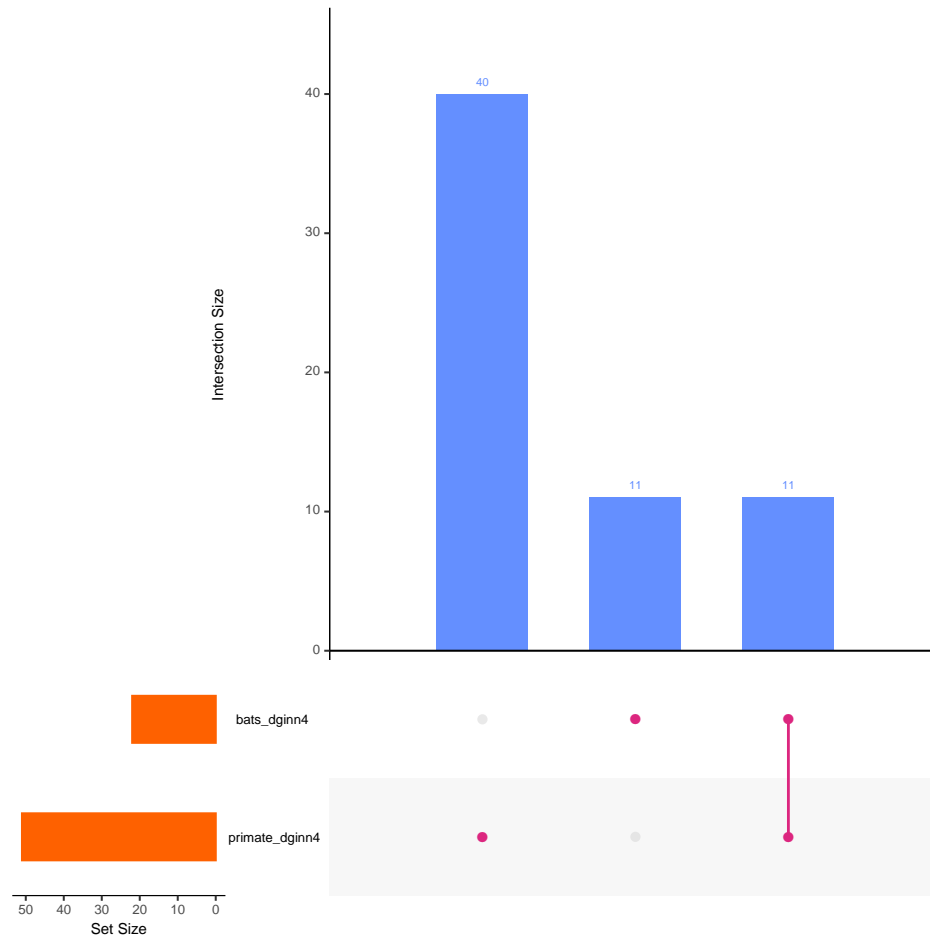
upset(monddata, nsets = 4, matrix.color = "#DC267F",
      main.bar.color = "#648FFF", sets.bar.color = "#FE6100")
```



```
upset(monddata[,1:3], nsets = 2, matrix.color = "#DC267F",
      main.bar.color = "#648FFF", sets.bar.color = "#FE6100")
```

```
upset(monddata[,c(1,4,5)], nsets = 2, matrix.color = "#DC267F",
      main.bar.color = "#648FFF", sets.bar.color = "#FE6100")
```



3 Which are these genes?

3.1 Gene under positive selection in both bats and primates

4 methods:

```
monddata[monddata$bats_dginn4==1 & monddata$primate_dginn4==1,]

##      tmp$Gene.name bats_dginn3 primate_dginn3 bats_dginn4
## 6          ACADM          1          1          1
## 7           ACE2          1          1          1
## 109          GGH          1          1          1
```

##	117	GOLGA7	1	1	1
##	134	IDE	1	1	1
##	139	ITGB1	1	1	1
##	146	LMAN2	1	1	1
##	212	POLA1	1	1	1
##	264	SLC27A2	1	1	1
##	302	TOR1AIP1	1	1	1
##	315	VPS39	1	1	1
##		primate_dginn4			
##	6		1		
##	7		1		
##	109		1		
##	117		1		
##	134		1		
##	139		1		
##	146		1		
##	212		1		
##	264		1		
##	302		1		
##	315		1		

3 methods:

```
monddata[monddata$bats_dginn3==1 & monddata$primate_dginn3==1,]
```

##	tmp\$Gene.name	bats_dginn3	primate_dginn3	bats_dginn4
##	6	ACADM	1	1
##	7	ACE2	1	1
##	9	ADAM9	1	0
##	34	CDK5RAP2	1	0
##	71	EDEM3	1	1
##	109	GGH	1	1
##	117	GOLGA7	1	1
##	134	IDE	1	1
##	139	ITGB1	1	1
##	146	LMAN2	1	1
##	157	MIPOL1	1	0
##	159	MOV10	1	0
##	212	POLA1	1	1
##	239	RAP1GDS1	1	1

##	257	SCCPDH	1	1	0
##	264	SLC27A2	1	1	1
##	302	TOR1AIP1	1	1	1
##	315	VPS39	1	1	1
##		primate_dginn4			
##	6		1		
##	7		1		
##	9		0		
##	34		1		
##	71		0		
##	109		1		
##	117		1		
##	134		1		
##	139		1		
##	146		1		
##	157		1		
##	159		0		
##	212		1		
##	239		0		
##	257		0		
##	264		1		
##	302		1		
##	315		1		

3.2 Gene under positive selection only in primates

4 methods:

```
monddata[monddata$bats_dginn4==0 & monddata$primate_dginn4==1,]
```

##	tmp\$Gene.name	bats_dginn3	primate_dginn3	bats_dginn4	
##	31	BRD4	0	1	0
##	34	CDK5RAP2	1	1	0
##	37	CEP135	0	1	0
##	40	CEP68	0	1	0
##	47	CLIP4	0	1	0
##	67	DNMT1	0	1	0
##	68	DPH5	0	1	0
##	75	EMC1	0	1	0

## 80	ER01B	0	1	0
## 101	FYC01	0	1	0
## 105	GCC2	0	1	0
## 110	GHITM	0	1	0
## 111	GIGYF2	0	1	0
## 112	GLA	0	1	0
## 127	HECTD1	0	1	0
## 143	LARP1	0	1	0
## 144	LARP4B	0	1	0
## 150	MARK1	0	1	0
## 157	MIPOL1	1	1	0
## 160	MPHOSPH10	0	1	0
## 166	MYCBP2	0	1	0
## 171	NDUFAF2	0	1	0
## 172	NDUFB9	0	1	0
## 187	NUP58	0	1	0
## 195	PCNT	0	1	0
## 218	PRIM2	0	1	0
## 220	PRKAR2A	0	1	0
## 227	PVR	0	1	0
## 245	REEP6	0	1	0
## 248	RIPK1	0	1	0
## 253	SAAL1	0	1	0
## 260	SEPSECS	0	1	0
## 262	SIRT5	0	1	0
## 263	SLC25A21	0	1	0
## 297	TMEM39B	0	1	0
## 299	TMPRSS2	0	1	0
## 305	TUBGCP2	0	1	0
## 308	UBAP2	0	1	0
## 311	UGGT2	0	1	0
## 322	ZNF318	0	1	0
##	primate_dginn4			
## 31	1			
## 34	1			
## 37	1			
## 40	1			
## 47	1			
## 67	1			

```
## 68      1
## 75      1
## 80      1
## 101     1
## 105     1
## 110     1
## 111     1
## 112     1
## 127     1
## 143     1
## 144     1
## 150     1
## 157     1
## 160     1
## 166     1
## 171     1
## 172     1
## 187     1
## 195     1
## 218     1
## 220     1
## 227     1
## 245     1
## 248     1
## 253     1
## 260     1
## 262     1
## 263     1
## 297     1
## 299     1
## 305     1
## 308     1
## 311     1
## 322     1
```

3 methods:

```
monddata[monddata$bats_dginn3==0 & monddata$primate_dginn3==1,]

##      tmp$Gene.name bats_dginn3 primate_dginn3 bats_dginn4
```

## 19	AP2A2	0	1	0
## 23	ATE1	0	1	0
## 31	BRD4	0	1	0
## 32	BZW2	0	1	0
## 37	CEP135	0	1	0
## 40	CEP68	0	1	0
## 47	CLIP4	0	1	0
## 48	CNTRL	0	1	0
## 67	DNMT1	0	1	0
## 68	DPH5	0	1	0
## 72	EIF4E2	0	1	0
## 75	EMC1	0	1	0
## 80	ER01B	0	1	0
## 83	EXOSC2	0	1	0
## 101	FYC01	0	1	0
## 105	GCC2	0	1	0
## 110	GHITM	0	1	0
## 111	GIGYF2	0	1	0
## 112	GLA	0	1	0
## 118	GOLGB1	0	1	0
## 119	GORASP1	0	1	0
## 125	HDAC2	0	1	0
## 127	HECTD1	0	1	0
## 131	HS6ST2	0	1	0
## 143	LARP1	0	1	0
## 144	LARP4B	0	1	0
## 145	LARP7	0	1	0
## 150	MARK1	0	1	0
## 154	MDN1	0	1	0
## 160	MPHOSPH10	0	1	0
## 164	MRPS5	0	1	0
## 166	MYCBP2	0	1	0
## 168	NAT14	0	1	0
## 171	NDUFAF2	0	1	0
## 172	NDUFB9	0	1	0
## 176	NGLY1	0	1	0
## 181	NPC2	0	1	0
## 187	NUP58	0	1	0
## 195	PCNT	0	1	0

##	202	PITRM1	0	1	0
##	204	PLAT	0	1	0
##	208	PLOD2	0	1	0
##	210	PMPCB	0	1	0
##	214	POR	0	1	0
##	218	PRIM2	0	1	0
##	220	PRKAR2A	0	1	0
##	224	PTBP2	0	1	0
##	227	PVR	0	1	0
##	230	RAB14	0	1	0
##	232	RAB1A	0	1	0
##	233	RAB2A	0	1	0
##	242	RBX1	0	1	0
##	245	REEP6	0	1	0
##	248	RIPK1	0	1	0
##	250	RPL36	0	1	0
##	253	SAAL1	0	1	0
##	260	SEPSECS	0	1	0
##	262	SIRT5	0	1	0
##	263	SLC25A21	0	1	0
##	278	STOM	0	1	0
##	291	TIMM8B	0	1	0
##	297	TMEM39B	0	1	0
##	299	TMPRSS2	0	1	0
##	303	TRIM59	0	1	0
##	304	TRMT1	0	1	0
##	305	TUBGCP2	0	1	0
##	308	UBAP2	0	1	0
##	311	UGGT2	0	1	0
##	313	USP54	0	1	0
##	322	ZNF318	0	1	0
##		primate_dginn4			
##	19		0		
##	23		0		
##	31		1		
##	32		0		
##	37		1		
##	40		1		
##	47		1		

## 48	0
## 67	1
## 68	1
## 72	0
## 75	1
## 80	1
## 83	0
## 101	1
## 105	1
## 110	1
## 111	1
## 112	1
## 118	0
## 119	0
## 125	0
## 127	1
## 131	0
## 143	1
## 144	1
## 145	0
## 150	1
## 154	0
## 160	1
## 164	0
## 166	1
## 168	0
## 171	1
## 172	1
## 176	0
## 181	0
## 187	1
## 195	1
## 202	0
## 204	0
## 208	0
## 210	0
## 214	0
## 218	1
## 220	1

```
## 224      0
## 227      1
## 230      0
## 232      0
## 233      0
## 242      0
## 245      1
## 248      1
## 250      0
## 253      1
## 260      1
## 262      1
## 263      1
## 278      0
## 291      0
## 297      1
## 299      1
## 303      0
## 304      0
## 305      1
## 308      1
## 311      1
## 313      0
## 322      1
```

3.3 Gene under positive selection only in bats

4 methods:

```
monddata[monddata$bats_dginn4==1 & monddata$primate_dginn4==0,]

##      tmp$Gene.name bats_dginn3 primate_dginn3 bats_dginn4
## 14      AKAP9          1          0          1
## 26      ATP6AP1        1          0          1
## 44      CISD3          1          0          1
## 71      EDEM3          1          1          1
## 77      ERGIC1         1          0          1
## 136     IMPDH2         1          0          1
## 137     INHBE          1          0          1
```

```
## 231      RAB18      1      0      1
## 239      RAP1GDS1    1      1      1
## 268      SLC44A2    1      0      1
## 284      TBK1      1      0      1
##      primate_dginn4
## 14      0
## 26      0
## 44      0
## 71      0
## 77      0
## 136     0
## 137     0
## 231     0
## 239     0
## 268     0
## 284     0
```

3 methods:

```
monddata[monddata$bats_dginn3==1 & monddata$primate_dginn3==0,]
```

```
##      tmp$Gene.name bats_dginn3 primate_dginn3 bats_dginn4
## 5      ACAD9      1      0      0
## 11     AGPS      1      0      0
## 14     AKAP9      1      0      1
## 26     ATP6AP1    1      0      1
## 44     CISD3      1      0      1
## 49     COL6A1     1      0      0
## 77     ERGIC1     1      0      1
## 122    GRIPAP1    1      0      0
## 123    GRPEL1     1      0      0
## 136    IMPDH2     1      0      1
## 137    INHBE      1      0      1
## 151    MARK2      1      0      0
## 185    NUP214     1      0      0
## 217    PRIM1      1      0      0
## 226    PUSL1      1      0      0
## 231    RAB18      1      0      1
## 267    SLC30A9    1      0      0
## 268    SLC44A2    1      0      1
```

## 269	SLC9A3R1	1	0	0
## 284	TBK1	1	0	1
##	primate_dginn4			
## 5	0			
## 11	0			
## 14	0			
## 26	0			
## 44	0			
## 49	0			
## 77	0			
## 122	0			
## 123	0			
## 136	0			
## 137	0			
## 151	0			
## 185	0			
## 217	0			
## 226	0			
## 231	0			
## 267	0			
## 268	0			
## 269	0			
## 284	0			

3.4 Figure tableau

```

tablo<-as.data.frame(tmp$Gene.name)
tablo$nbats<-batstmp
tablo$nprimates<-primatetmp

plot(NULL, xlim=c(-0.5,5.5), ylim=c(-3,5.5),
      xlab="bats", ylab="primates",
      main="Genes supported by x,y methods in bats and primates",
      bty="n",
      xaxt="n", yaxt="n")

text(x=rep(-0.6, 6), y=0:5, 0:5)
text(y=rep(-0.65, 6), x=0:5, 0:5)

```

```

sapply(seq(from=-0.5, to=5.5, by=1), function(x){
  segments(x0=x, x1=x, y0=-0.5, y1=5.5)
})

## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL

sapply(seq(from=-0.5, to=5.5, by=1), function(x){
  segments(x0=-0.5, x1=5.5, y0=x, y1=x)
})

## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL

```

```

##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL

for (p in 0:5){
  for (b in 0:5){
    tmp<-tablo$tmp$Gene.name`[tablo$nbats==b & tablo$nprimates==p]
    if(length(tmp)>0 & length(tmp)<=8){
      text(b,seq(from=(p-0.4), to=(p+0.4), length.out = length(tmp)),
            tmp, cex=0.4)
    }else if (length(tmp)>8 & length(tmp)<=16){
      print(c(p, b))
      text((b-0.3),seq(from=(p-0.4), to=(p+0.4), length.out = 8),
            tmp[1:8], cex=0.4)
      text((b+0.3),seq(from=(p-0.4), to=(p+0.4), length.out = (length(tmp)-8)),
            tmp[9:length(tmp)], cex=0.4)
    }else if (length(tmp)>16){
      text(b,p, paste0(length(tmp), " values"))
    }
  }
}

## [1] 1 2
## [1] 2 0
## [1] 2 1
## [1] 2 2
## [1] 3 0
## [1] 3 1
## [1] 4 0
## [1] 4 1

tmp<-tablo$tmp$Gene.name`[tablo$nbats==0 & tablo$nprimates==1]
text(-0.4,-1.2, "p=1/n=0", cex=0.6)
text(seq(from=0.1, to=5.5, length.out = 19),

```

```

-1.1,
tmp[1:19],
cex=0.4)
text(seq(from=0.1, to=5.5, length.out = length(tmp)-19),
-1.3,
tmp[20:length(tmp)],
cex=0.4)

tmp<-tablo$tmp$Gene.name`[tablo$nbats==1 & tablo$nprimates==1]
text(-0.4,-1.7, "p=1/n=1", cex=0.6)
text(seq(from=0.1, to=5.5, length.out = 18),
-1.6,
tmp[1:18],
cex=0.4)
text(seq(from=0.1, to=4.5, length.out = length(tmp)-18),
-1.8,
tmp[19:length(tmp)],
cex=0.4)

tmp<-tablo$tmp$Gene.name`[tablo$nbats==0 & tablo$nprimates==0]
text(-0.4,-2.3, "p=0/n=0", cex=0.6)
text(seq(from=0.1, to=5.5, length.out = 17),-2.1, tmp[1:17], cex=0.4)
text(seq(from=0.1, to=5.5, length.out = 17),-2.3, tmp[18:34], cex=0.4)
text(seq(from=0.1, to=5.5, length.out = length(tmp)-34),-2.5, tmp[35:length(tmp)], cex=0.4)

tmp<-tablo$tmp$Gene.name`[tablo$nbats==2 & tablo$nprimates==0]
text(-0.4,-2.9, "p=0/n=2", cex=0.6)
text(seq(from=0.1, to=5.5, length.out = 18),-2.8, tmp[1:18], cex=0.4)
text(seq(from=0.1, to=1, length.out = length(tmp)-18),-3.0, tmp[19:length(tmp)], cex=0.4)

```

Genes supported by x,y methods in bats and primates

primates	5	ZNF318 REEP6 PRIM2 NDUFAF2	NUP58 DNMT1 CEP68	SEPSECS PVR MYCBP2 HECTD1 FYCO1 EMC1		ITGB1 ACADM	VPS39 TOR1AIP1 GOLGA7 GGH ACE2
	4	SLC25A21 NUPFB9 LARP4B GHTM GCC2 ERO1B DPH5 CEP135	UBAP2 TUBGCP2 TMPPRS2	SIRT5 SAA1 PRKAR2A PHOSPH10 MARK1 LARP1 GLA GIGYF2	UGGT2	TMEM39B RIPK1 PCNT CLIP4 BRD4	POLA1 LMAN2 MIPOL1 IDE SLC27A2
	3	NAT14 LARP7 HS6ST2 HMC2 GORASP1 EXOSC2 SCV ATE1	TRIM59 TIMM8B RBX1 RAB2A RAB1A RAB14 RTBP2 NPC2	RPL36 PLOC2 PLAT PITRM1 NGUY1 EIF4E2 NTRNL AP2A2	USP54 TRMT1	POR PMPCB MRPS5 MDN1 GOLGB1	SCCPDH MOV10 ADAM9 EDEM3
	2	NDUFAF1 FKBP1 FBXL12 ERLEC1 CHPF CEP150 ATP13A3 ALG5	ZYG11B TBKBP1 SMOC1 RBM41 RAB5C PMPCA NINL	NLRX1 MRPS27 CWC27 CRT23 CIT CEP150 CENPF ABCC1	ZDHHC5 TAP11 SLU7 SCAP RTN4 PRKAR2B PABPC4	SCARB1 SRNO1 NUP210 GOLGA2 FBN2 FBN1 FAR2 ANC8 WASHC4	WFS1 PRIM1 NUP214 COL6A1 AKAP9 TBK1
	1	37 values	27 values	PSMD8 POFUT1 NSD2 MARK3 HYOU1 GTF15 COLGALT1 GIMP2A	TUBGCP3 TLE1 TBCA RRP9 PTGES2	PUSL1 GRIPAP1 AGPS ACAD9	IMPDH2 CISD3 ATP6AP1 RAB18 SLC44A2
	0	49 values	28 values	22 values	SLC9A3R1 SLC30A9 MARK2 GRPEL1	ERGIC1	INHBE
		0	1	2	3	4	5

p=1/n=0 AATF ACSL3 AKAPBL AP2M1 BCKDK CLCC1 CSDE1 CSNK2B DCAF7 GFER GNB1 GNG5 GOLGA3 GPX1 HSBP1 MAP7D1 MRPS2 NDFIP2 NUP98 NUTF2 PDE4DIPDZ11 PIGO PKP2 PLD3 PPL13 RAB7ARETREGSELENOS SRP19 TARS2 TCF12 TLE3 TMEM97TOMM70 UBXN8 ZNF503

p=1/n=1 ALG8 AP3B1 ATP6V1ACEP112 CHPF2 DDX21 ERC1 FAM8A1 FBLN5 G3BP1 HMOX1 KDELC1 LOX NARS2 NOL10 NUP62 QSOX2 RAE1 SLC30A7 SRP72 TIMM29 TM2D3 TOR1A UBAP2L VPS11 ZC3H18 ZC3H7A

p=0/n=0 AAR2 ADAMTS1 AKAP8 ARF6 BAG5 CCDC86CSNK2A2 DCAKD DCTPP1DNAJC19DPY19L1 ECSIT EIF4H ETFA EXOSC5 EXOSC8FAM162A FAM98A FASTKD5 FKBP15 GTF2F2 HOOK1 KDELC2 MARC1 MAT2B MOGS NGDN NUP54 NUP88 OS9 PABPC1 PIGS PLEKH45 RAB10 RALA REEP5 RHOA RNFA1 SDF2 SLC30A6 SNIP1 SRP54 STC2 STOML2 THTPA TIMM10B TIMM9 UPF1 YIF1A

p=0/n=2 AASS ATP1B1 COMT CUL2 DDX10 ELOC ERMP1FOXRED2GPA1 HS2ST1JAKMIP1MEPC6 MIB1 NEK9 NEU1 PCSK6 PLEKH2F2 RBM28 RDX SUN2 TMED5TYSND1

bats

```
write.csv(tablo[tablo$nbats>=3,"tmp$Gene.name"], "batssup3.csv",
          row.names=FALSE,
          quote=FALSE)

write.csv(tablo[tablo$nprimates>=3,"tmp$Gene.name"], "primatessup3.csv",
          row.names=FALSE,
          quote=FALSE)

write.csv(tablo, "primatesVbats.csv",
          row.names=FALSE,
          quote=FALSE)
```

Restreindre ce tableau aux gènes présent dans l'analyse de Krogan.


```

# Reading the Krogan table
tab<-read.delim(paste0(workdir,
  "covid_comp/covid_comp_complete.txt"),
  fill=T, h=T, dec=",")
dim(tab)

## [1] 332 141

#Adding ACE2 and TMPRSS2
krogan<-c(as.character(tab$merge.Gene), "ACE2", "TMPRSS2")

# The list
length(krogan)

## [1] 334

krogan

## [1] "AAR2" "AASS" "AATF" "ABCC1" "ACAD9"
## [6] "ACADM" "ACSL3" "ADAM9" "ADAMTS1" "AGPS"
## [11] "AKAP8" "AKAP8L" "AKAP9" "ALG11" "ALG5"
## [16] "ALG8" "ANO6" "AP2A2" "AP2M1" "AP3B1"
## [21] "ARF6" "ARL6IP6" "ATE1" "ATP13A3" "ATP1B1"
## [26] "ATP5MG" "ATP6AP1" "ATP6V1A" "BAG5" "BCKDK"
## [31] "BCS1L" "BRD2" "BRD4" "BZW2" "C1orf50"
## [36] "CCDC86" "CDK5RAP2" "CENPF" "CEP112" "CEP135"
## [41] "CEP250" "CEP350" "CEP43" "CEP68" "CHMP2A"
## [46] "CHPF" "CHPF2" "CISD3" "CIT" "CLCC1"
## [51] "CLIP4" "CNTRL" "COL6A1" "COLGALT1" "COMT"
## [56] "COQ8B" "CRTC3" "CSDE1" "CSNK2A2" "CSNK2B"
## [61] "CUL2" "CWC27" "CYB5B" "CYB5R3" "DCAF7"
## [66] "DCAKD" "DCTPP1" "DDX10" "DDX21" "DNAJC11"
## [71] "DNAJC19" "DNMT1" "DPH5" "DPY19L1" "ECSIT"
## [76] "EDEM3" "EIF4E2" "EIF4H" "ELOB" "ELOC"
## [81] "EMC1" "ERC1" "ERGIC1" "ERLEC1" "ERMP1"
## [86] "ER01B" "ERP44" "ETFA" "EXOSC2" "EXOSC3"
## [91] "EXOSC5" "EXOSC8" "F2RL1" "FAM162A" "FAM8A1"
## [96] "FAM98A" "FAR2" "FASTKD5" "FBLN5" "FBN1"
## [101] "FBN2" "FBXL12" "FKBP10" "FKBP15" "FKBP7"
## [106] "FOXRED2" "FYCO1" "G3BP1" "G3BP2" "GCC1"

```

## [111]	"GCC2"	"GDF15"	"GFER"	"GGCX"	"GGH"
## [116]	"GHITM"	"GIGYF2"	"GLA"	"GNB1"	"GNG5"
## [121]	"GOLGA2"	"GOLGA3"	"GOLGA7"	"GOLGB1"	"GORASP1"
## [126]	"GPAA1"	"GPX1"	"GRIPAP1"	"GRPEL1"	"GTF2F2"
## [131]	"HDAC2"	"HEATR3"	"HECTD1"	"HMOX1"	"HOOK1"
## [136]	"HS2ST1"	"HS6ST2"	"HSBP1"	"HYOU1"	"IDE"
## [141]	"IL17RA"	"IMPDH2"	"INHBE"	"INTS4"	"ITGB1"
## [146]	"JAKMIP1"	"LARP1"	"LARP4B"	"LARP7"	"LMAN2"
## [151]	"LOX"	"MAP7D1"	"MARC1"	"MARK1"	"MARK2"
## [156]	"MARK3"	"MAT2B"	"MDN1"	"MEPCE"	"MFGE8"
## [161]	"MIB1"	"MIPOL1"	"MOGS"	"MOV10"	"MPHOSPH10"
## [166]	"MRPS2"	"MRPS25"	"MRPS27"	"MRPS5"	"MTCH1"
## [171]	"MYCBP2"	"NARS2"	"NAT14"	"NDFIP2"	"NDUFAF1"
## [176]	"NDUFAF2"	"NDUFB9"	"NEK9"	"NEU1"	"NGDN"
## [181]	"NGLY1"	"NIN"	"NINL"	"NLRX1"	"NOL10"
## [186]	"NPC2"	"NPTX1"	"NSD2"	"NUP210"	"NUP214"
## [191]	"NUP54"	"NUP58"	"NUP62"	"NUP88"	"NUP98"
## [196]	"NUTF2"	"OS9"	"PABPC1"	"PABPC4"	"PCNT"
## [201]	"PCSK6"	"PDE4DIP"	"PDZD11"	"PIGO"	"PIGS"
## [206]	"PITRM1"	"PKP2"	"PLAT"	"PLD3"	"PLEKHA5"
## [211]	"PLEKHF2"	"PLOD2"	"PMPCA"	"PMPCB"	"POFUT1"
## [216]	"POGLUT2"	"POGLUT3"	"POLA1"	"POLA2"	"POR"
## [221]	"PPIL3"	"PPT1"	"PRIM1"	"PRIM2"	"PRKACA"
## [226]	"PRKAR2A"	"PRKAR2B"	"PRRC2B"	"PSMD8"	"PTBP2"
## [231]	"PTGES2"	"PUSL1"	"PVR"	"QSOX2"	"RAB10"
## [236]	"RAB14"	"RAB18"	"RAB1A"	"RAB2A"	"RAB5C"
## [241]	"RAB7A"	"RAB8A"	"RAE1"	"RALA"	"RAP1GDS1"
## [246]	"RBM28"	"RBM41"	"RBX1"	"RDX"	"REEP5"
## [251]	"REEP6"	"RETREG3"	"RHOA"	"RIPK1"	"RNF41"
## [256]	"RPL36"	"RRP9"	"RTN4"	"SAAL1"	"SBN01"
## [261]	"SCAP"	"SCARB1"	"SCCPDH"	"SDF2"	"SELENOS"
## [266]	"SEPSECS"	"SIGMAR1"	"SIL1"	"SIRT5"	"SLC25A21"
## [271]	"SLC27A2"	"SLC30A6"	"SLC30A7"	"SLC30A9"	"SLC44A2"
## [276]	"SLC9A3R1"	"SLU7"	"SMOC1"	"SNIP1"	"SPART"
## [281]	"SRP19"	"SRP54"	"SRP72"	"STC2"	"STOM"
## [286]	"STOML2"	"SUN2"	"TAPT1"	"TARS2"	"TBCA"
## [291]	"TBK1"	"TBKBP1"	"TCF12"	"THTPA"	"TIMM10"
## [296]	"TIMM10B"	"TIMM29"	"TIMM8B"	"TIMM9"	"TLE1"
## [301]	"TLE3"	"TLE5"	"TM2D3"	"TMED5"	"TMEM39B"

```
## [306] "TMEM97"      "TOMM70"      "TOR1A"       "TOR1AIP1"    "TRIM59"
## [311] "TRMT1"       "TUBGCP2"     "TUBGCP3"     "TYSND1"      "UBAP2"
## [316] "UBAP2L"      "UBXN8"       "UGGT2"       "UPF1"        "USP13"
## [321] "USP54"       "VPS11"       "VPS39"       "WASHC4"      "WFS1"
## [326] "YIF1A"       "ZC3H18"      "ZC3H7A"      "ZDHHC5"      "ZNF318"
## [331] "ZNF503"      "ZYG11B"      "ACE2"        "TMPRSS2"

#In the table, I select line that match the krogan gene name liste
tabloK<-tablo[tablo$tmp$Gene.name` %in% krogan,]

# How many gene lost?
dim(tablo)

## [1] 324 3

dim(tabloK)

## [1] 321 3

# Les gènes perdus (dans le tableau mais pas dans la liste de Krogan)
sort(tablo$tmp$Gene.name`[tablo$tmp$Gene.name` %in% krogan==F])

## [1] KDELC1 KDELC2 PCSK5
## 442 Levels: AAR2 AASS AATF ABCC1 ACAD9 ACADM ACE2 ACSL3 ... ZYG11B

# Les gènes de Krogan non présent dans cette liste
sort(krogan[krogan %in% tablo$tmp$Gene.name`==F])

## [1] "ARL6IP6" "ATP5MG" "BCS1L" "C1orf50" "CEP43" "CYB5R3"
## [7] "ELOB" "MFGE8" "POGLUT2" "POGLUT3" "SIGMAR1" "TLE5"
## [13] "USP13"

write.csv(tabloK, "primatesVbats_onlykrogan.csv", row.names=FALSE, quote=FALSE)
```

4 Tanglegram

```
#install.packages('dendextend') # stable CRAN version
library(dendextend) # load the package
```

```

#install.packages("phytools") # stable CRAN version
library(phytools) # load the package
library(ggraph)
library(igraph)
library(tidyverse)

##
tmp<-tablo[(tablo$nbats!=0 | tablo$nprimates!=0),]
tmp<-head(tmp, 20)
#tmp<-rbind(as.matrix(tmp), c("outgroup", 50, 50))
tmp<-as.data.frame(tmp)
matbats<-hclust(dist(tmp$nbats))
matpri<-hclust(dist(tmp$nprimates))

tmp[order(tmp$nbats),]

dendpri<-as.dendrogram(matpri)
dendbats<-as.dendrogram(matbats)

labels(dendpri)<-as.character(tmp$tmp$Gene.name[labels(dendpri)])
labels(dendbats)<-as.character(tmp$tmp$Gene.name[labels(dendbats)])

tmp[order(tmp$nprimates, decreasing=FALSE),]$'tmp$Gene.name'> order
dendpri<-dendextend::rotate(dendpri, order=order)

tmp[order(tmp$nbats, decreasing=FALSE),]$'tmp$Gene.name'> order
dendbats<-dendextend::rotate(dendbats, order=order)

#### Il faut swapper certains neud de l'arbrese

class(labels(dendpri))

dend12 <- dendlist(dendbats, dendpri)

?png
png("tanglegramm.png", width = 1800, height = 3000)

```

```

tanglegram(dend12, columns_width=c(3, 3,3), axes=FALSE,
            edge.lwd=0, margin_inner=6,
            margin_top=2,
            main_left="                bats",
            main_right = "primates                ",
            lwd=0.5,
            cex_main=1,
            lab.cex=1,
            k_labels=6)

dev.off()

tmp
?tanglegram

```