

```
#Run RENUM:
```

```
#!!! Include the generation into the renumbered data file!!
```

```
# and the TBV as well for the validation later
```

```
ulimit -s unlimited
```

```
# 3) run renumf90
```

```
echo renum.par | renumf90 | tee renum.log
```

```
# 4) Identify genotyped animals:
```

```
awk '{if ($6>=10) print $0}' renadd02.ped | sort +0 -1 > gen.ped
```

```
wc -l gen.ped
```

```
wc -l snp3.2k
```

```
head snp3.2k_XrefID
```

```
# 5
```

```
#Run preGSf90
```

```
cp renf90.par pregs1.par
```

```
echo "OPTION saveCleanSNPs" >> pregs1.par
```

```
echo pregs1.par | preGSf90 | tee pregs1.log
```

```
#Add header to the map file
```

```
sed '1 i\SNPID CHR POS' mrkmap.txt > newmap.txt
```

```
sed -i 's:chrinfo mrkmap.txt:map_info newmap.txt:g' pregs1.par
```

```
echo pregs1.par | preGSf90 | tee pregs1.log
```

```
grep 'Number of SNPs:' pregs1.log
```

```
grep 'Number of effective SNPs' pregs1.log
```

```
grep -A 5 "Possible genotype duplicates" pregs1.log
```

```
grep -A 15 "Correlation of Genomic Inbreeding" pregs1.log
```

```
grep -A 3 "Statistic of Rel. Matrix A22" pregs1.log
```

```
grep -A 3 "Statistic of Genomic Matrix" pregs1.log
```

```
# 6
```

```
#Run PREGS 2nd time, using clean file!
```

```
cp renf90.par pregs2.par
```

```
sed -i 's:snp3.2k:snp3.2k_clean:g' pregs2.par
```

```
sed -i 's:chrinfo mrkmap.txt:map_info newmap.txt_clean:g' pregs2.par
```

```
echo "OPTION no_quality_control" >> pregs2.par
```

```
echo "OPTION plotpca" >> pregs2.par
```

```
echo pregs2.par | preGSf90 | tee pregs2.log
```

7

#Run BLUP with full data:

```
mkdir blup ; cd blup
cp ../renf90.par par.temp
grep -v OPTION par.temp > blup.par
sed -i 's:renf90.dat:../renf90.dat:g' blup.par
sed -i 's:renadd02.ped:../renadd02.ped:g' blup.par
echo 'OPTION use_yams' >> blup.par
time echo blup.par | blupf90 | tee blup.log
cp solutions blup_solutions
```

cd ..

#Run ssGBLUP with full data:

```
mkdir ssgblup; cd ssgblup
cp ../renf90.par par.temp
grep -v OPTION par.temp > ssgblup.par
sed -i 's:renf90.dat:../renf90.dat:g' ssgblup.par
sed -i 's:renadd02.ped:../renadd02.ped:g' ssgblup.par
echo "OPTION SNP_file ../snp3.2k_clean" >> ssgblup.par
echo "OPTION map_info ../newmap.txt_clean" >> ssgblup.par
echo "OPTION no_quality_control" >> ssgblup.par
echo 'OPTION use_yams' >> ssgblup.par
time echo ssgblup.par | blupf90 | tee ssgblup.log
cp solutions ssgblup_solutions
```

8)

#Run Validation:

```
cd ..
mkdir validation; cd validation
```

#Create reduced dataset:

```
awk '$4<5' ../renf90.dat > renf90_red.dat
wc -l renf90_red.dat
```

#1.Run BLUP with reduced:

```
cp ../renf90.par par.temp
grep -v OPTION par.temp > blup.par
sed -i 's:renf90.dat:renf90_red.dat:g' blup.par
sed -i 's:renadd02.ped:../renadd02.ped:g' blup.par
```

```
echo 'OPTION use_yams' >> blup.par
echo blup.par | blupf90 | tee blup.log
cp solutions blup_r_solutions
```

#2.Run ssGBLUP with reduced:

```
cp ../renf90.par par.temp
grep -v OPTION par.temp > ssgblup.par
sed -i 's:renf90.dat:renf90_red.dat:g' ssgblup.par
sed -i 's:renadd02.ped:../renadd02.ped:g' ssgblup.par
echo "OPTION SNP_file ../snp3.2k_clean" >> ssgblup.par
echo "OPTION map_info ../newmap.txt_clean" >> ssgblup.par
echo "OPTION no_quality_control" >> ssgblup.par
echo 'OPTION use_yams' >> ssgblup.par
echo ssgblup.par | blupf90 | tee ssgblup.log
cp solutions ssgblup_r_solutions
```

#Prepare validation files:

#Validation is on 2000 animals from generation 5 with genotypes and phenotypes!

#1004 - 3 = 1001

```
#awk '{print $1}' ../snp3.2k_clean_XrefID | sort +0 -1 > gen.ped
```

```
#join -1 +1 -2 +3 gen.ped young.dat | awk '!a[$1]++' > gen_rec
```

```
awk '$4==5' ../renf90.dat | sort +2 -3 > young.dat
```

```
sort +0 -1 ../snp3.2k_clean_XrefID > gen.id
```

```
join -1 +1 -2 +3 gen.id young.dat > gen_rec
```

```
awk '{print $1,$2,$6}' gen_rec | sort +0 -1 > yid.tmp
```

#ID, ORIG_ID, TBV

```
awk '{if ($1==1 && $2==2) print $3,$4}' blup_r_solutions | sort +0 -1 > blup.tmp
```

```
awk '{if ($1==1 && $2==2) print $3,$4}' ssgblup_r_solutions | sort +0 -1 > ssgblup.tmp
```

```
join -1 +1 -2 +1 blup.tmp ssgblup.tmp > gb.tmp
```

```
join -1 +1 -2 +1 yid.tmp gb.tmp > solutions_all
```

ID, ORIG_ID, TBV, EBV, GEBV

#Start R for calculations:

R

```
a=read.table(file="solutions_all")
```

```
head(a)
```

```
colnames(a)<- c("id", "orig_id", "tbv", "ebv", "gebv")
```

```
head(a)
```

```

# BLUP acc
round(cor(a$tbv,a$ebv,use="pairwise.complete.obs"),2)
round(cor(a$tbv,a$gebv,use="pairwise.complete.obs"),2)

#BLUP: TBV~EBV
reg=lm(a$tbv~a$ebv)
summary(reg)
#b0=0.53
#b1=0.83
#R2=0.24

#Check bias and inflation (b0 & b1)
#ssGBLUP: TBV~GEBV
reg2=lm(a$tbv~a$gebv)
summary(reg2)
#b0=0.51
#b1=0.84
#R2=0.41

q()
n

# 9)
#Predictivity / predictive ability:
#Go to the existing BLUP folder:
cd ../blup

cp blup.par pred.par
#random effects
echo 'OPTION include_effects 2' >> pred.par
echo pred.par | predictf90 | tee pred.log

# format: new_id y1 y_hat1 e1
awk '{print $1,$2}' yhat_residual | sort +0 -1 > yxb.tmp
join -1 +1 -2 +1 ../validation/solutions_all yxb.tmp > solutions_all2
# ID, ORIG_ID, TBV, EBV, GEBV, Yhat

#Start R for calculations:
R
a=read.table(file="solutions_all2")
colnames(a)<- c("id", "orig_id", "tbv", "ebv", "gebv", "yhat")
head(a)
round(cor(a$yhat,a$ebv,use="pairwise.complete.obs"),2)
round(cor(a$yhat,a$gebv,use="pairwise.complete.obs"),2)

```

```
#Predictiviy BLUP: 0.27
#Predictiviy ssGBLUP: 0.37
```

```
# OPTIONAL EXERCISES
```

```
## 10) Run GBLUP
```

```
cd ../
mkdir gblup; cd gblup
awk '{print $1,0,0}' ../snp3.2k_clean > ped.gen
awk '{print $1}' ../snp3.2k_clean | sort +0 -1 > sgen.temp
sort +0 -1 ../data3.txt > sphen.temp
join -1 +1 -2 +1 sgen.temp sphen.temp > phen.gen
```

```
#run renumf90 first
cp ../snp3.2k_clean .
cp ../newmap.txt_clean .
cp ../renum.par .
sed -i 's:data3.txt:phen.gen:g' renum.par
sed -i 's:ped3.txt:ped.gen:g' renum.par
sed -i 's:snp3.2k:snp3.2k_clean:g' renum.par
sed -i 's:chrinfo mrkmap.txt:map_info newmap.txt_clean:g' renum.par
echo 'OPTION no_quality_control' >> renum.par
sed -i '32s/0/1/' renum.par
echo renum.par | renumf90 | tee renum.log
```

```
#run gblup
cp renf90.par gblup.par
echo 'OPTION tunedG 0' >> gblup.par
echo 'OPTION AlphaBeta 1 0' >> gblup.par
echo gblup.par | blupf90 | tee gblup.log
```

```
sed -i '30s/1 0/0.99 0.01/' gblup.par
echo gblup.par | blupf90 | tee gblup.log
```