Pong User Pipeline and Tutorial

Genelle F Harrison

3/31/2021

KIRpong

Author: Laura A Leaton, Genelle F Harrison, Paul Norman Maintainer: Genelle F. Harrison GenelleFH@gmail.com Description: PONG is a modification of HIBAG and was designed to impute KIR3DL1/S1 KIR3DL1/S1 alleles from chromosome 19 WG-SNP data. PONG uses the R statistical programming language. We provide models built from the Global populations availible in the 1000 Genomes Project for HG19 and HG38. PONG leverages the functions built in HIBAG using attribute bagging and an ensemble classifier method with haplotype inference for SNPs and KIR3DL1/S1 types. Please use with an R version < 4.0.

Installing KIRpong

The best way to install KIRpong is through the command line. R CMD INSTALL KIRpong/

Models for KIR3DL1/S1 imputation

KIRpong relies on a model built from a training data set. We have created a model that uses data from the 1000 Genomes project. To build this model we randomly samples 50% of individuals from each of the 26 Global populaitons. We built this model both for HG19 and HG38. Users can also build a model with their own training sets using the following pipeline.

```
library(KIRpong)
```

```
## PONG (KIR3DL1/S1 Genotype Imputation with Attribute Bagging): v1.0.0
```

```
## Supported by Streaming SIMD Extensions 2 (SSE2)
```

Building your own KIRpong Model

To build a model you will need WG-SNP data in Plink binary format (.bed, .bim, .fam) coupled with KIR3DL1/S1 alleles. Duplicated positions and rsIDs should be removed and SNPs with a count of two or less should be removed. This tutorial provides an example of building a model built from individuals in the European Super population. The Plink files can be found in this folder: KIRpong/inst/extdata/Chr19.hg38_TGP_EUR_Populations_Model_Input.

Import Plink BED files

To run this tutorial, set the working directory to the directory where the KIRpong package is. Here is an example.If you are building your own model then specify the file path to your Plink binary files.Be sure the set the assembly to "hg19" or "hg38" depending on what you are using.

```
setwd("/Users/genelleharrison/Dropbox/UC_Denver/Projects/KIRpong/Scripts/Versions")
path_inputs=getwd()

geno = hlaBED2Geno(bed.fn= paste0(path_inputs,"/KIRpong/inst/extdata/Chr19.hg38_TGP_EUR_Populations_Mod
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/
## Import 63669 SNPs within the KIR gene cluster on chromosome 19.
```

Import KIR3DL1/S1 allele data

If you are running with your own KIR3DL1/S1 the first column will be the sampleID and columns 2 and 3 will be the KIR3DL1/S1 alleles. Here is an example:

```
SampleID Allele1 Allele2
NA06989 *00501 *01301
NA06994 *00402 *00501
NA07051 *00101 *01301
NA07347 *00101 *00700
NA10847 *01301 *01301
NA11892 *00501 *01502
NA11894 *00101 *00501
NA11918 *00200 *00401
NA11919 *00101 *00401
```

```
D <- read.table(paste0(path_inputs,"/KIRpong/inst/extdata/EUR_KIR_Typing_Alleles_For_Model.txt"), heade
```

Building the model

The functions shown herein are from the program HIBAG, which is why the functions say "hla". The fourth arguement in the "snpid" object is where the flanking regions are set. Currently we will sample 10bp upstream and downstream of the KIR gene cluster.

```
train.HLA <- hlaAllele(D$SampleID, H1=D$Allele1, H2=D$Allele2, locus="KIR3DLS1", assembly="hg38")
summary(train.HLA)
snpid <- hlaFlankingSNP(geno$snp.id, geno$snp.position, "KIR3DLS1", 10*10, assembly="hg38")
train.geno <- hlaGenoSubset(geno, snp.sel=match(snpid, geno$snp.id))

set.seed(1000)
model <- hlaAttrBagging(train.HLA, train.geno, nclassifier=100, verbose.detail=TRUE)

summary(model)
model.obj <- hlaModelToObj(model)
save(model.obj, file=paste0(path_inputs,"/KIRpong/data/Tutorial_Model_EUR.RData"))
```

Imputing KIR3DL1/S1 alleles using KIRpong:

Imputation of KIR3DL1/S1 alleles can be done using one of our pre-built models or with a model you built using the pipeline above. In this tutorial we will use the model built above from European data in the 1000 Genomes Project. The input data is again Plink binary files (.bed, .bim, .fam) as well as the model (model_file.RData).

```
library(KIRpong)
setwd("/Users/genelleharrison/Dropbox/UC_Denver/Projects/KIRpong/Scripts/Versions")
path_inputs=getwd()

bim_file=paste0(path_inputs,"/KIRpong/inst/extdata/Chr19.hg38_TGP_EUR_Populations_Test_Input.bim")
fam_file=paste0(path_inputs,"/KIRpong/inst/extdata/Chr19.hg38_TGP_EUR_Populations_Test_Input.fam")
bed_file=paste0(path_inputs,"/KIRpong/inst/extdata/Chr19.hg38_TGP_EUR_Populations_Test_Input.bed")
model_file=paste0(path_inputs,"/KIRpong/data/Tutorial_Model_EUR.RData")
```

Imputing KIR3DL1/S1 alleles using KIRpong:

Next we can run a PONG imputation of KIR3DL1S1 alleles. Be sure to make sure you have specified the genome assembly and that it matches the model.

```
path_to_file=paste0(path_inputs,"/KIRpong/inst/extdata/Results")
HLA_allele="KIR3DLS1"
setwd(path_to_file)

model.list=get(load(model_file))
hla.id=HLA_allele
yourgeno=hlaBED2Geno(bed.fn=bed_file, fam.fn=fam_file, bim.fn=bim_file, assembly="hg38")
```

```
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/i
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/i
## Open "/Users/genelleharrison/Dropbox (Personal)/UC_Denver/Projects/KIRpong/Scripts/Versions/KIRpong/i
## Import 63829 SNPs within the KIR gene cluster on chromosome 19.
```

```
print(hla.id)
```

```
## [1] "KIR3DLS1"
```

```
model=hlaModelFromObj(model.list)
pred.guess=predict(model, yourgeno, type="response+prob", match.type = "Position")
```

```
## KIRpong model: 100 individual classifiers, 1043 SNPs, 14 unique KIR3DL1/S1 alleles.
## Predicting based on the averaged posterior probabilities from all individual classifiers.
## Model assembly: hg38, SNP assembly: hg38
## There are 47 missing SNPs (4.5%).
## There are 14 variants in total with switched allelic strand orders.
## Due to stand ambiguity (such like C/G), the allelic strand orders of 246 variants are determined by a
## The number of samples: 177.
## Predicting:  Tue Nov 30 13:01:20 2021    0%
## Predicting:  Tue Nov 30 13:01:20 2021    100%
```

```
pred.guess_value = as.data.frame(pred.guess$value)
pred.guess_postprob = as.data.frame(pred.guess$postprob)

write.table(pred.guess_value, paste0(path_inputs,"/KIRpong/inst/extdata/Results/Chr19.hg38_TGP_EUR_Popul

write.table(pred.guess_postprob, paste0(path_inputs,"/KIRpong/inst/extdata/Results/Chr19.hg38_TGP_EUR_Po
```