Preparing the data required for the model generation

For the model generation, it is necessary to have a .csv file containing the sample ID and the two HLA alleles from each HLA gene (see example below), and a plink file set with the variants genotypes from the same samples, which will be used as a reference.

1. HLA types

HLA_alleles.csv

	A	B	C	D	E	F	G	H	I
1	ID	HLA_A_1	HLA_A_2	HLA_B_1	HLA_B_2	HLA_C_1	HLA_C_2	HLA_DPB1_1	HLA_DPB1_2
2	Sample_01	02:01	03:01	41:02	15:16	17:01	16:01	105:01	105:01
3	Sample_02	30:01	33:01	42:01	45:01	17:01	16:01	01:01	17:01
4	Sample_03	03:01	02:02	45:01	14:02	08:02	06:02	02:01	01:01
5	Sample_04	23:01	01:01	44:03	08:01	07:01	04:01	01:01	04:02
6	Sample_05	03:01	03:01	14:02	07:02	07:02	08:02	04:01	02:01
7	Sample_06	03:01	68:02	53:01	49:01	07:01	04:01	11:01	01:01
8	Sample_07	74:01	33:03	35:01	53:01	07:18	04:01	11:01	105:01
9	Sample_08	03:01	02:02	44:03	15:16	14:03	14:02	01:01	19:01
10	Sample_09	23:01	29:02	42:01	50:01	17:01	06:02	105:01	02:01
11	Sample_10	11:01	02:02	81:01	14:01	18:01	08:02	04:01	105:01
12	Sample_11	11:01	33:01	41:02	35:01	17:01	04:01	04:01	04:01
13	Sample_12	30:02	03:01	49:01	58:02	07:01	06:02	18:01	17:01
14	Sample_13	66:02	02:01	44:03	56:01	04:01	01:02	01:01	04:01
15	Sample_14	23:01	02:01	14:02	39:10	08:02	12:03	11:01	17:01
16	Sample_15	68:02	74:01	51:01	18:01	08:02	02:02	01:01	03:01
17	Sample_16	23:01	74:01	53:01	07:02	04:01	07:02	03:01	02:01
18	Sample_17	23:01	68:02	45:01	53:01	04:01	16:01	01:01	105:01
19	Sample_18	02:05	23:01	49:01	14:02	07:01	08:02	104:01	11:01
20	Sample_19	02:01	33:01	45:01	45:01	16:01	16:01	01:01	27:01
21	Sample_20	02:01	02:02	42:01	58:01	17:01	07:18	131:01	01:01
22	Sample_21	23:01	02:01	15:03	44:02	07:04	02:10	105:01	03:01
23	Sample_22	30:02	68:01	18:01	58:02	05:01	06:02	01:01	18:01
24	Sample_23	03:01	23:01	58:01	49:01	07:18	07:01	01:01	01:01
25	Sample_24	23:01	74:01	81:01	53:01	04:01	08:02	01:01	01:01

2. Variant genotypes

To run plink you need to install the most appropriate version for your operating system (https://www.cog-genomics.org/plink/2.0/)

Converting data to the plink format. The code below will generate a set of files, the most important are:

.bed -> a binary genotype table / .bim -> each line contains information about a specific variant (Chr, rsID, genetic distance, position, ref allele, alt allele) / .fam -> Samples information - Family ID, Individual ID, Paternal ID, Maternal ID, Sex (1=male; 2=female; other=unknown), Phenotype

```
##from .vcf
./plink --vcf data.vcf --make-bed --out data
##or from map/pad data
./plink --file data.vcf --make-bed --out data
```

Restricting only the MHC region

```
./plink --bfile data --chr 6 --from-mb 29 --to-mb 34 --make-bed --out data_mhc-region
```

Applying Quality Control filters

```
./plink --bfile data_mhc-region --maf 0.01 --make-bed --out data_mhc-region_maf01 ##Filter all variants with allele frequency lower than 1%
./plink --bfile data_mhc-region_maf01 --geno 0.02 --make-bed --out data_mhc-region_maf01_geno02 ##Filter all variants with missing call rates higher than 2%
./plink --bfile data_mhc-region_maf01_geno02 --geno 0.02 --make-bed --out data_mhc-region_maf01_geno02_hwe1e-6 ## Filter all variants with a Hardy-Weinberg equilibrium exact test p-value lower than 1e-6)
```

Removing the AT/CG positions:

```
cat file.bim | grep -Pv "[AT]\s[AT]$" | grep -Pv "[GC]\s[GC]$" | cut -f 2 > toKeep.txt
./plink --bfile data_mhc-region_maf01_geno02_hwe1e-6 --extract toKeep.txt --make-bed --out data_mhc-region_maf01_geno02_hwe1e-6_noATGC
```

Since rsIDs can change between datasets, using Position to match the model and the dataset is recommended. Changing the rsIDs to "chr_position" may be convenient to avoid these incompatibilities. You can do this by adding "_old" to the bim file name, for further replacement of a new one

```
cat data_mhc-region_maf01_geno02_hwe1e-6_noATGC_old.bim | awk '{OFS="\t"} {$2=$1":"$4;} {print $0}' > data_mhc-region_maf01_geno02_hwe1e-6_noATGC.bim
```

Preparing the data that you want to predict

Variants genotypes

Converting data to the plink format

```
##from .vcf
./plink --vcf data_toPredict.vcf --make-bed --out data_toPredict
##or from map/pad data
./plink --file data_toPredict --make-bed --out data_toPredict
```

Restricting only the MHC region

```
./plink --bfile data_toPredict --chr 6 --from-mb 29 --to-mb 34 --make-bed --out data_toPredict _mhc-region
```

Applying Quality Control filters

```
./plink --bfile data_toPredict _mhc-region --maf 0.01 --make-bed --out data_toPredict _mhc-region_maf01
##Filter all variants with allele frequency lower than 1%
./plink --bfile data_toPredict _mhc-region_maf01--geno 0.02 --make-bed --out data_toPredict _mhc-region_maf01_geno02 ##Filter all variants with missing call rates higher than 2%
./plink --bfile data_toPredict _mhc-region_maf01_geno02 --geno 0.02 --make-bed --out data_toPredict _mhc-region_maf01_geno02_hwe1e-6 ## Filter all variants with a Hardy-Weinberg equilibrium exact test p-value lower than 1e-6)
```

Removing the AT/CG positions:

```
cat file.bim | grep -Pv "[AT]\s[AT]$" | grep -Pv "[GC]\s[GC]$" | cut -f 2 > toKeep.txt
./plink --bfile data_toPredict _mhc-region_maf01_geno02_hwe1e-6 --extract toKeep.txt --make-bed --out data_toPredict _mhc-region_maf01_geno02_hwe1e-6_noATGC
```

Changing the rsIDs to "chr_position" may be convenient to avoid incompatibilities between the model and dataset. You can do this by adding "_old" to the bim file name, for further replacement of a new one

```
cat data_toPredict_mhc-region_maf01_geno02_hwe1e-6_noATGC_old.bim | awk '{OFS="\t"} {$2=$1":"$4;} {print $0}' > data_toPredict _mhc-region_maf01_geno02_hwe1e-6_noATGC.bim
```

We recommend keeping only the variants that overlap between training and the test set

```
cat data_toPredict _mhc-region_maf01_geno02_hwe1e-6_noATGC.bim | cut -f 2 > data_toPredict _snp.txt
./plink --bfile data_mhc-region_maf01_geno02_hwe1e-6_noATGC --extract data_toPredict_snp.txt –make-bed --out data_QC_overlapped ##data to generate the model

cat data_QC_overlapped.bim | cut -f 2 > data_snp.txt
./plink --bfile data_toPredict_mhc-region_maf01_geno02_hwe1e-6_noATGC --extract data_snp.txt --make-bed --out data_toPredict _QC_overlapped ##data you want to perform HLA imputation
```

Installing packages in R

Installing HIBAG

From http://www.bioconductor.org/packages/devel/bioc/html/HIBAG.html

```
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")

BiocManager::install("HIBAG")
```

Note: The HIBAG.gpu installation is not necessary if there is no GPU (Graphics processing unit) in your server

Installing HIBAG.gpu

The devtools R package is needed to install HIBAG.gpu.

From https://stackoverflow.com/questions/61046116/issue-installing-devtools-on-r-and-dependency-on-usethis-and-fs

```
install.packages("Rcpp", dependencies = TRUE)
install.packages("fs")
```

Then from https://github.com/zhengxwen/HIBAG.gpu

```
library("devtools")
```

```
install_github("zhengxwen/HIBAG.gpu")
```

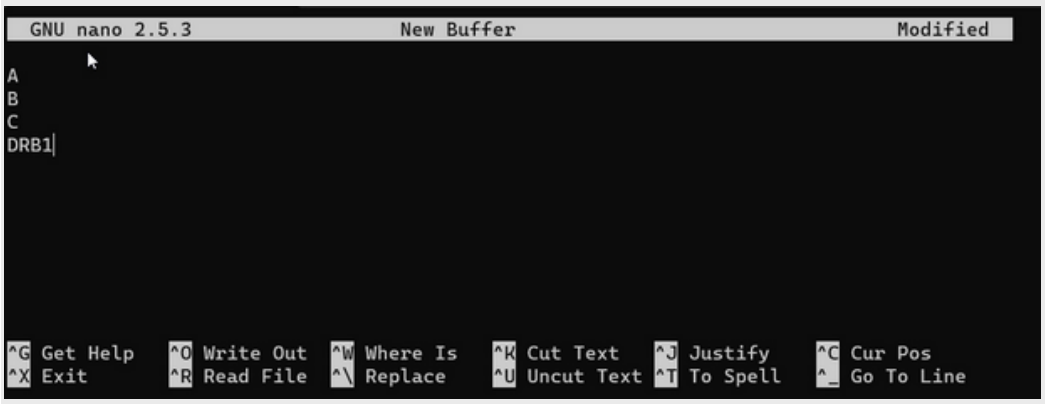
Model building (using the script: *modelGenerator.r*).

Note the scripts below are set up to hg38 data, if necessary it should be changed

It is necessary to have a text file with each gene from each you want to create a model

HLA_list.txt

nano



*You can create it using *nano* or in a text editor file

Building a HIBAG model via parallel computation:

```
Rscript modelGenerator.r -f data_QC_overlapped -H HLA_list.txt -t HLA_alleles.csv -b 100 -O output_folder/ --CPU
```

*However, if you are using GPU, you should remove the --CPU option

Predictions (using the script: validation_gpu.r)

The script below should be launched for each model

```
Rscript validation_gpu.r -f data_toPredict_QC_overlapped -m model_data_QC_overlapped_A.Rdata -o predictions_model_A --c --CPU
```

* If you are using GPU, you should remove the --CPU option

*Alternatively, you can use the script ***model_prediction_HIBAG.r*** for simple usage*

Results

It will be generated a .csv file with the predicted HLA alleles for each sample, followed by their posterior probability values

	A	B	C	D
1	ID	HLA_A_1	HLA_A_2	postProb
2	SAMPLE_01	01:01	02:01	0,617078861797461
3	SAMPLE_02	01:01	30:01	0,478012043769948
4	SAMPLE_03	01:01	31:01	0,124019505471573
5	SAMPLE_04	02:01	02:01	0,535062641873922
6	SAMPLE_05	01:01	29:02	0,14627683946046
7	SAMPLE_06	03:01	29:02	0,16282772127571
8	SAMPLE_07	02:01	02:01	0,638922649480318
9	SAMPLE_08	01:01	24:02	0,486881610815595
10	SAMPLE_09	03:01	11:01	0,597342125551997
11	SAMPLE_10	01:01	23:01	0,422464690315759
12	SAMPLE_11	68:01	68:02	0,32369229596327
13	SAMPLE_12	03:01	31:01	0,252490170510584
14	SAMPLE_13	01:01	02:01	0,571596199714553
15	SAMPLE_14	11:01	68:01	0,396103610841756
16	SAMPLE_15	30:01	33:03	0,492218875084921
17	SAMPLE_16	02:01	11:01	0,389785843387968
18	SAMPLE_17	02:01	02:05	0,188107191871708
19	SAMPLE_18	24:02	26:01	0,148860090728983
20	SAMPLE_19	02:01	30:01	0,316192227983402
21	SAMPLE_20	03:01	03:01	0,742764600398815
22	SAMPLE_21	32:01	33:03	0,274316905726737
23	SAMPLE_22	31:01	33:03	0,223052948689337
24	SAMPLE_23	23:01	68:01	0,395571594970403
25	SAMPLE_24	02:01	03:01	0,442007752673461