Open Access

OBM Genetics

Original Research

Integrated Web Application (Snips2HLA-HsG) Development for Sample Preparation and Model Creation for HLA Allele Prediction with the SNP Data Using HIBAG Package of Bioconductor and R Programming

Balamurugan Sivaprakasam *, Prasanna Sadagopan

Department of Computer Science, Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai - 117, India; E-Mails: sivabala76@gmail.com; prasanna.scs@velsuniv.ac.in; ORCID: 0000-0002-4766-837X

* **Correspondence:** Balamurugan Sivaprakasam; E-Mail: sivabala76@gmail.com; ORCID: 0000-0002-4766-837X

Academic Editor: Ivan Y Iourov

OBM Genetics
2024, volume 8, issue 2
doi:10.21926/obm.genet.2402243

Received: December 24, 2023
Accepted: May 22, 2024
Published: June 14, 2024

Abstract

The present study introduces Snips2HLA-HsG, an integrated application designed for SNP genotype analysis and HLA allele type prediction. Leveraging attribute bagging, a powerful ensemble classifier technique from the Bioconductor HIBAG package, Snips2HLA-HsG offers a comprehensive response for genetic analysis. Accessible via https://snips2hla.shinyapps.io/hla_home/, the application distinguishes itself by prioritizing user-friendliness and integrating all-purpose functionalities, including sample preparation, model generation, HLA prediction, and accuracy assessment. In contrast to the fragmented landscape of existing HLA imputation software, this study addresses the need for an integrated, user-centric platform. By streamlining processes and enhancing accessibility, Snips2HLA-HsG ensures usability, even for biologists with limited computer proficiency. Future updates will address the choice between one or ten classifiers, aiming to optimize server utility and meet research needs effectively by adding more classifiers to utilize multiple cores for faster calculations. Looking ahead, Snips2HLA-HsG will undergo regular updates and maintenance to ensure continued effectiveness and relevance in genetic research. Maintenance efforts will focus on resolving issues or bugs and providing ongoing user support.

Keywords
Human leukocyte antigen; HIBAG; Bioconductor; ensemble methods; R

1. Introduction

In humans, the extended major histocompatibility complex (MHC) region, also known as the human leukocyte antigen (HLA), is present in the short arm of chromosome 6p21.3, covering over 7 Mb and contains many genes. The exact number of genes within the extended MHC region is challenging to determine precisely due to ongoing research and the discovery of new genes. However, estimates suggest approximately 200 to 300 genes in this region. This region includes classical and non-classical genes crucial in immune responses and transplantation [1, 2]. These particular genes are recognized as the most complex and diverse in the human genome (among the individuals in a population), recently referred to as hyper polymorphic rather than simply polymorphic. These genes encode protein products that serve as mediators, facilitating interactions with pathogen and cellular peptides. So far, the classical HLA genes, Class I, such as HLA-A, HLA-B, HLA-C, and Class II, such as HLA-DPA1, HLA-DPB1, HLA-DQA1, HLA-DQB1, HLA-DRA, and HLA-DRB1 are studied intensively [3, 4].

Direct typing of HLA alleles for extensive studies is expensive, and HLA typing through computational prediction is inexpensive [5]. HLA imputation is the method of predicting a person's HLA genotype using the data about a person’s SNP genotype at locations flanking the classical HLA sites. The reason SNPs in the MHC region are informative in predicting the HLA genotype is linkage disequilibrium [6]. Not a single variant of SNP can predict the HLA alleles, but using many SNP in the surrounding area of HLA loci can produce accurate inferences. Therefore, the SNP data are of primary importance in the imputation process. Also, the HLA imputation needs previous knowledge regarding which SNP variants are linked with which HLA alleles in previously genotyped samples called reference panels. Various factors like the density of SNP data in the HLA area, the size of the reference panel, the closeness amongst the reference panel and the samples being imputed, the degree of the polymorphism at the locus being imputed, the frequency of the HLA allele as well as the computational tool used influence the accuracy of HLA imputation [7].

Many methods are available for doing HLA imputation from HLA and SNP genotype data using different algorithms [8-15]. This article says that Leslie et al. (2008) [5] developed a methodology for predicting HLA alleles from a database of SNP haplotypes carrying known HLA alleles, using an identity-by-descent (IBD) model which helped to create LDMhc algorithm. Dilthey et al. (2011) subsequently developed an integrated software, HLA*IMP, for imputing classical HLA alleles from SNP genotypes based on LDMhc with a modified SNP selection function [8]. BEAGLE, an alternative imputation method to the approximate coalescent models, allows prediction multiallelic loci [9]. SNP2HLA imputes amino acids and HLA alleles in the MHC region from SNP genotype data [10]. Subsequently, the HIBAG method was developed for the researchers with published population-based models instead of requiring access to extensive training sample datasets. It combines the concepts of Attribute Bagging, an ensemble classifier method, and haplotype inference for SNPs and HLA types [11]. HLA-check and HLA*IMP performance varies with the reference panel provided [12]. The web interface, HLA-IMPUTER, was developed using the bagging algorithm imputation framework [13], where the user can input their SNP genotype data and impute HLA alleles using the reference panels. Another method, DEEP*HLA, a multi-task convolution deep learning method, was developed to accurately impute genotypes of HLA genes from Single Nucleotide Variation (SNV) level data [14]. Likewise, CookHLA is another method that performed well when the reference panel was small or ethnically unmatched and was accurate in imputing rare alleles [15].

In comparing HIBAG with other related applications, Boegel (2012) emphasizes the suitability of HIBAG for HLA typing when SNP data is available, particularly in scenarios such as genome-wide association studies (GWAS) where direct HLA typing data is inaccessible [16]. Additionally, HIBAG, HLA*IMP, Minimac, and SNP2HLA are discussed by Dilthey et al., 2012 [17] and Sakaue et al., 2023 [18] as tools employed for HLA imputation from SNP data, each presenting distinctive levels of accessibility and functionality. While HIBAG is constrained by limited public accessibility on the web, HLA*IMP, Minimac, and SNP2HLA are readily available to researchers. Despite their accessibility differences, each tool operates on its own statistical methodology and approach to HLA imputation. The choice between HIBAG or HLA*IMP:03 or, Minimac or other tools for HLA imputation depends on the nature of available genetic data and the specific objectives of the research or academic study.

Therefore, in developing the HLA imputation web application, this study opted for HIBAG due to its good accuracy, particularly when dealing with complex datasets [7, 19, 20]. Additionally, HIBAG incorporates attribute bagging, a technique that enhances accuracy and stability by training each classifier on random subsets of the data. By capitalizing on these strengths, the present study developed the web application Snips2HLA-HsG. This user-friendly interface allows researchers and academics to generate models based on the SNP genotypes sample dataset, predict HLA allele types, and evaluate accuracy. Snips2HLA-HsG is now active on the Shiny web server and can be accessed via https://snips2hla.shinyapps.io/hla_home/. It offers two functionalities: "Model Generation" and "HLA Type Prediction".

2. Materials and Methods

The statistical programming language R, Bioconductor package, HIBAG, and its supplementary software tools are freely available for the research community, which helps develop user-friendly applications for HLA allele prediction. R programming environment (base R) is the core software that provides the fundamental components and functionality for executing R code. On the other hand, RStudio, an integrated development environment (IDE), offers a user-friendly graphical interface by providing a more convenient and feature-rich environment for working with R code. In the present study, version 4.3.3 of the R programming environment and the release of version - 2023.12.1 with build number 402 of RStudio were used from http://www.r-project.org and https://www.rstudio.com, respectively. Bioconductor is a collection of packages and tools specifically designed to analyze and manipulate biological data in the R programming language. These packages cover a wide range of bioinformatics and computational biology tasks. The latest version 3.18 was used in the application development. HIBAG has been widely used in HLA imputation studies and is available as an R/Bioconductor package. Its latest version (1.38.2) was obtained from http://www.bioconductor.org/packages/HIBAG and executed on the Windows operating system, but it can also be installed on other operating systems such as Linux and macOS. Using the above-mentioned R functionalities and with the appropriate data resources, the present study developed a user-friendly integrated web application for sample preparation, model generation, HLA allele typing prediction, and accuracy evaluation, which are discussed as follows.

2.1 Development of Snips2HLA-HsG Dashboard Application for HLA Allele Type Prediction

Snips2HLA-HsG is an integrated dashboard web application written in R programming language, and RStudio is used as an interface. The R packages on CRAN and Bioconductor, such as rJava, xlsx, shiny, tools, HIBAG, Tidyquery, Dplyr, RSQLite, Shinydashboard, shinyCSSLoaders, shiny widgets, DT and Shiny are used.

The Shiny apps have two components: a user-interface (ui) script and a server script. The ui script controls the layout and appearance of an application. In ui.R, under the shiny package, the functions like column(), Selectinput(), SubmitButton(), DTOutput(), Shinyuioutput(), downloadBttn(), and shiny::plotOutput() are used. Under the shinydashboard package, the functions like DashboardPage(), Dashboardheader(), Dashboardsidebar(), Sidebarmenu(), Menuitem(), Tabitem(), Box() are used. Under DT package, functions like dataTableOutput(), DTOutput() are used. Under the shinyCSSLoader package, Withspinner() is used. Under the Shinywidget package, updateprogressBar(), get() and progressbar() are used for laying out the user interface. The server script contains the computer's instructions to build an application. In server.R, the functions like downloadhandler(), updateprogressbar(), renderUI(), renderDataTable(), IncludeHTML() and renderPlot() are used for application development. Similarly, in the HIBAG package, the functions like hlaBED2Geno(), hlaModelFromObj(), hlaPredict(), hlaAllele(), hlaCompareAllele(), hlaReport() and hlaReportPlot() are used.

Snips2HLA-HsG is a multi-platform application that can be initiated locally from any computer. The App has been tested on Chrome, Internet Explorer, Firefox, and Opera browsers. Additionally, the Snips2HLA-HsG app has been hosted on the cloud by the shinyapps.io server and is accessible at https://snips2hla.shinyapps.io/hla_home/.

The entire methodology of doing HLA imputation is shown in the workflow (Figure 1).

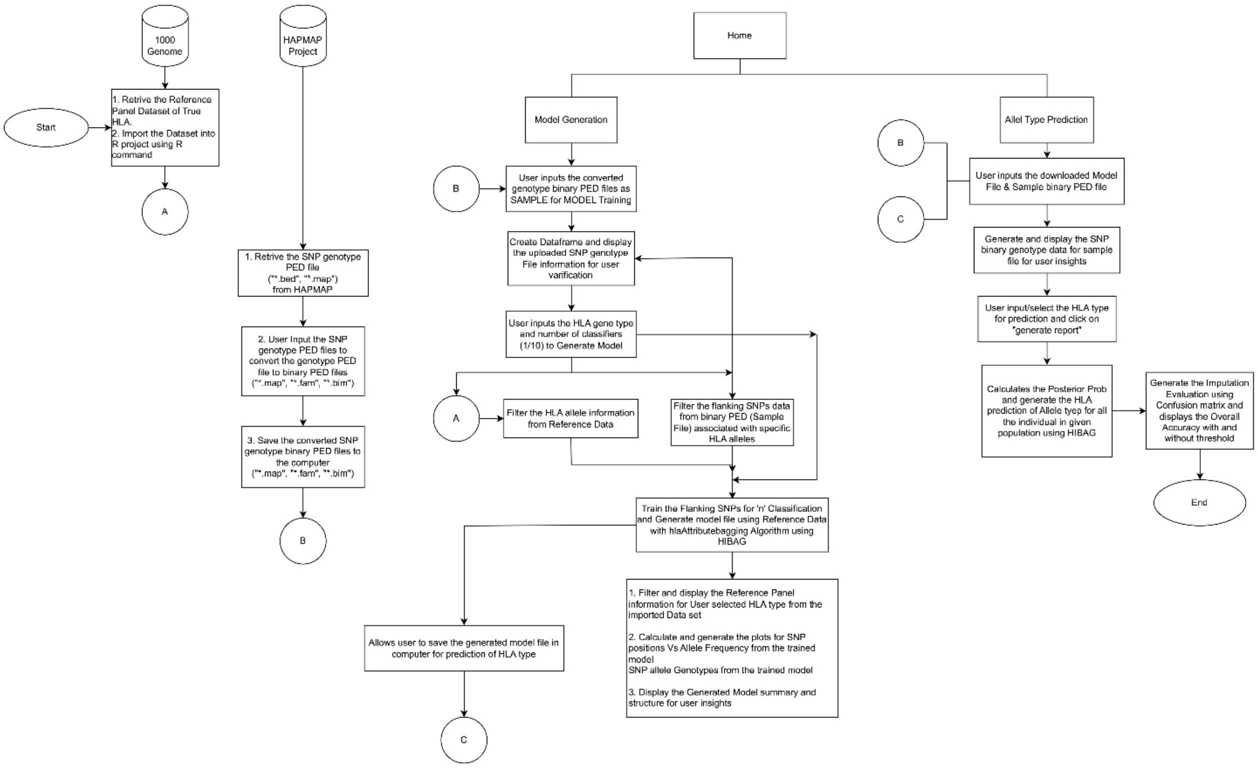

1000 Genome
HAPMAP Project
Start
1. Retrive the Reference Panel Dataset of True HLA.
2. Import the Dataset into R project using R command
A
1. Retrive the SNP genotype PED file ("*.bed", "*.map") from HAPMAP
2. User Input the SNP genotype PED files to convert the genotype PED file to binary PED files ("*.map", "*.fam", "*.bim")
3. Save the converted SNP genotype binary PED files to the computer ("*.map", "*.fam", "*.bim")
B
Home
Model Generation
B
User inputs the converted genotype binary PED files as SAMPLE for MODEL Training
Create Dataframe and display the uploaded SNP genotype File information for user varification
User inputs the HLA gene type and number of classifiers (1/10) to Generate Model
A
Filter the HLA allele information from Reference Data
Filter the flanking SNPs data from binary PED (Sample File) associated with specific HLA alleles
Train the Flanking SNPs for 'n' Classification and Generate model file using Reference Data with hlaAttributebagging Algorithm using HIBAG
Allows user to save the generated model file in computer for prediction of HLA type
C
1. Filter and display the Reference Panel information for User selected HLA type from the imported Data set
2. Calculate and generate the plots for SNP positions Vs Allele Frequency from the trained model
SNP allele Genotypes from the trained model
3. Display the Generated Model summary and structure for user insights
Allel Type Prediction
B
C
User inputs the downloaded Model File & Sample binary PED file
Generate and display the SNP binary genotype data for sample file for user insights
User input/select the HLA type for prediction and click on "generate report"
Calculates the Posterior Prob and generate the HLA prediction of Allele tyep for all the individual in given population using HIBAG
Generate the Imputation Evaluation using Confusion matrix and displays the Overall Accuracy with and without threshold
End

Figure 1 The workflow of preparing sample, generation of the model, HLA type prediction and model validation in the development of an integrated web application Snips2HLA-HsG.

2.2 Data Resources

HapMap Phase III was completed before the 1000 Genomes project, providing valuable insights into human genome variation. The HapMap project, which concluded its data collection and updates in 2010 with the release of HapMap Phase III data, has not received further official updates since then. Currently, the HapMap project offers genome-wide SNP data in phases I-II with approximately 3 million SNP markers in 270 individuals of four populations (YRI, CHB, JPT, and CEU) and in phase III with around 1.5 million markers of 150 individuals from the populations, LWK, MKK, TSI, GIH, CHD, and ASW. Recent initiatives and resources, such as the 1000 Genomes Project (KG: http://www.1000genomes.org/) and Human Genome Diversity Project (HGDP: https://www.internationalgenome.org/data-portal/data-collection/hgdp), have provided comprehensive catalogs of genetic variation and population data. Researchers now depend on these newer resources for up-to-date data in their studies.

2.3 Nature and the Availability of the Sample SNP Genotype Data

Using binary files of SNP genotype data with PLINK is recommended because they are more efficient than reading large text files. This speeds up data processing, essential for handling extensive genetic data [21]. Generally, PLINK data consists of two files: one containing information on the individuals and their genotypes (*.ped) and the other containing information on the genetic markers (*.map) [21]. In contrast, binary PLINK data consists of three files: a binary file containing individual identifiers (IDs) and genotypes (*.bed), and two text files containing information on the individuals (*.fam) and on the genetic markers (*.bim).

To create the *.ped file for PLINK using genotype data from different populations, specific mandatory columns are required, including (i) Family ID (population name), (ii) Individual ID (individual name), (iii) Paternal ID (father's ID or "0" if unknown), (iv) Maternal ID (mother's ID or "0" if unknown), (v) Sex (1 = male, 2 = female, 0 = unknown), (vi) Phenotype (0 = unaffected, 1 = affected, or 0 = unknown), and (vii) SNP representation (e.g., AA or AG) encoded as ACGT or 1-4 (where: 1 = A, 2 = C, 3 = G, 4 = T). A tool like IGG3 serves to facilitate the genotype integration process, and its primary function is to take SNP genotype data from a given dataset and format it in a way that is compatible with popular genotype imputation tools such as PLINK, BEAGLE, and HIBAG [9, 22]. This allows researchers to seamlessly incorporate their SNP genotypes into these imputation tools for further processing.

To avoid the need to access these large SNP genotype datasets, databases such as the 1000 Gen omes Project, HGDP, and HapMap project maintain a few SNP genotype sample files in PLINK form at. These files can be found at the following URLs: https://ftp.ncbi.nlm.nih.gov/hapmap/genotypes/2010-05_phaseIII/plink_format/ and https://ftp.ncbi.nlm.nih.gov/hapmap/genotypes/hapmap3/plink_format/draft_1/.

To make it even easier, the SNP genotype sample data files are also available on the HIBAG website (https://hibag.s3.amazonaws.com/index.html) in the “ImmPuteDataPackage.zip” for the research community. The data in this package generally includes SNP markers selected within the MHC of chromosome 6, ranging from 25759242 to 33534827, with a missing call rate threshold of 10%. There are 6241, 17160, and 16896 SNP markers in the CEU, YRI, and CHB + JPT populations [23].

2.3.1 Integration of PLINK in the Application, Snips2HLA-HsG

The general procedure for converting the genotype PED file to the binary PED (*.bed) format involves using PLINK [11]. It's important to note that this file conversion method requires users to memorize specific commands and syntax but is not so efficient for beginners and less technically-inclined users. To simplify it, the present study has integrated an in-built PLINK command into the application for SNP genotype data file conversion. Therefore, one can submit *.ped and *.map files to obtain binary PED (*.bed, *.fam, *.bim) files. By default, Shiny limits file uploads to 5 MB per file. However, this limit can be modified using the shiny.maxRequestSize option. In the present application development, the size limit was increased using the option "shiny " to accommodate larger input files as maxRequestSize = 1000 x 1024^2. This allows for handling larger input files of about 1000 MB in size.

In the Snips2HLA-HsG application home page, the link “Model Generation” in the side bar menu is designed to submit the sample files (*.bed, *.fam, and *.bim) in the corresponding input boxes. After submitting the input files, the user-defined reactive function is executed, and the three input files are passed as arguments to the HIBAG function ‘hlaBED2geno()’, and retrieves the class of input SNP genotype data and stores it as the data frame or R object. Then, it displays data in table format using the shiny functions dataTableOutput() and renderDataTable().

2.4 Generation of Population-Specific Models through the Integrated Web Application

The availability of pre-fit classifiers (published population-based models) for the research community can also be used to avoid accessing large training datasets. These can be downloaded from the webpage of HIBAG platform-specific parameter estimates (http://zhengxwen.github.io/HIBAG/platforms.html). However, it's important to note that these models will not provide accurate results for all the samples.

Keeping this in mind, the present study developed the application to create a specific model for the selected sample SNP genotype data. The available SNP and HLA data from 2693 samples in the 1000 Genomes dataset (IDAWG_2018.xlsx) include allele genotype details for HLA-A, HLA-B, HLA-C, HLA-DQB1, and DRB1 [24, 25]. This data was retrieved using the read_excel() function, and each column contains the data for each HLA type, including Region, Population, Sample_ID, Allele 1, and Allele 2. These data are saved separately as data frames. The renderDataTable() function displays customizable data tables in Shiny applications. It requires a data frame object as input and generates the data in a tabular format within the Shiny UI.

With the availability of these data, the current study has developed a GUI application that allows the end-users to create a population-specific model by submitting sample SNP genotype data. The current training dataset includes SNP and HLA data from the 1000 Genomes project, and it will be regularly updated as new samples are added to the databases [23]. This process also involves identifying the surrounding SNPs in the sample SNP genotype data that will be used for training and to ensure accurate results, it is crucial to specify the human genome reference as "hg19" along with the SNP data. This specification is necessary to precisely locate the specified HLA gene, enabling the identification of SNPs within its neighboring region. It's advisable to consider a flanking region of 500 kb on both sides for this purpose. To simplify this process, one can utilize the hlaAttrBagging() function, which provides access to the training algorithm. The setting generates individual classifiers to construct an ensemble model using the HIBAG function hlaModelToObj() is possible. In the current study, building one or ten classifiers (nclassifier = 1.10) is given as the application is for academic purposes. However, for accurate data, it is recommended that end-users generate 100 individual classifiers to obtain accurate results. Once the model generation is complete, end-users can save the model as an R object file (*.RData) for future use.

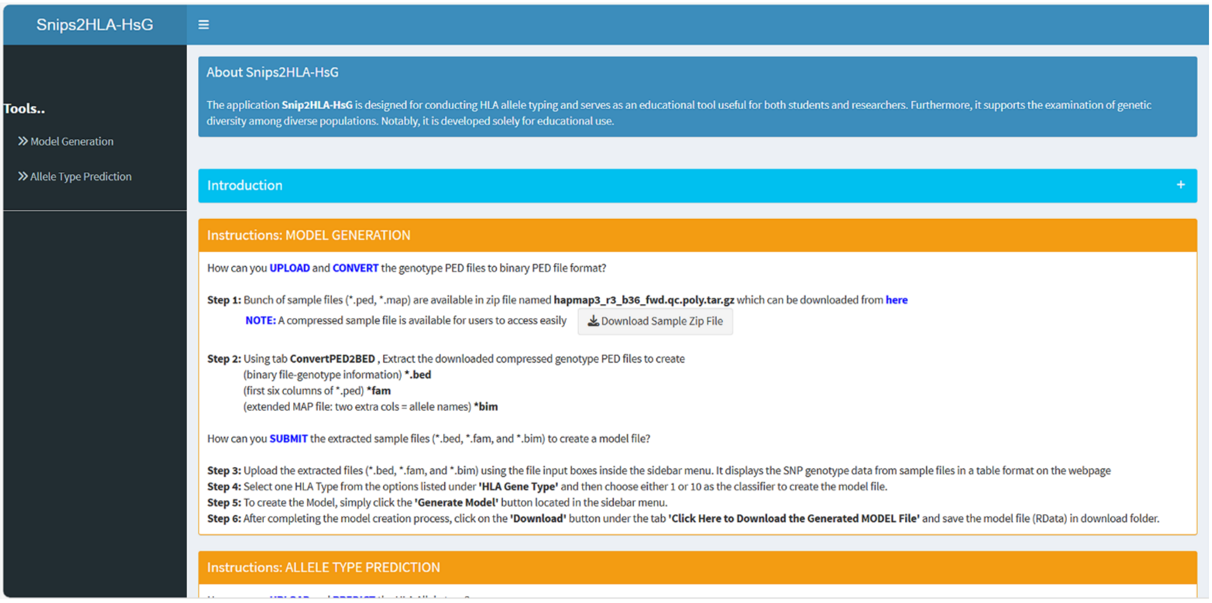
2.5 HLA Allele Type Prediction and Accuracy Evaluation

After the sample and model files are prepared, they are imported in the R session. When the user defined R script is executed, it does get input data as R object and models for the given HLA type (HLA-A, HLA-B, HLA-C, HLA-DQB1 and HLA-DRBI) as arguments. To predict the HLA type based on the generated model, the HIBAG function 'hlaPredict()' is used. As a result, this program writes the imputation results to a text file, which means the predicted results are displayed in a table using the R function 'write.table()' to export the resultant data from data frame to a file called 'result.txt' located on the working directory. The consequent file has the columns such as "sample ID", "allele 1", "allele 2", "probability," and "matching."

Evaluating HLA-type prediction involves two main approaches, without and with a threshold. The former provides a broad evaluation of all imputed HLA types, while the latter focuses on more reliable predictions by applying a confidence threshold of 50%. Cross-validation is conducted to assess performance using the "IDAWG_2018" dataset, and confusion matrices are generated to analyze sensitivity, specificity, and predictive values for each allele.

3. Results and Discussion

In this study, an integrated web application, Snips2HLA-HsG, was developed. This application allows for the preparation of SNP genotype sample data, generation of models, and the prediction of HLA allele types, along with accuracy evaluation. It is deployed at the URL https://snips2hla.shinyapps.io/hla_home/, with two available links: “Model Generation” (https://snips2hla.shinyapps.io/hla_mod/) and “HLA Type Prediction” (https://snips2hla.shinyapps.io/hla_pred/). The homepage is shown in Figure 2 and the results obtained from its links are discussed as follows.



About Snips2HLA-HsG

The application **Snip2HLA-HsG** is designed for conducting HLA allele typing and serves as an educational tool useful for both students and researchers. Furthermore, it supports the examination of genetic diversity among diverse populations. Notably, it is developed solely for educational use.

Instructions: MODEL GENERATION

How can you **UPLOAD** and **CONVERT** the genotype PED files to binary PED file format?

Step 1: Bunch of sample files (*.ped, *.map) are available in zip file named **hapmap3_r3_b36_fwd.qc.poly.tar.gz** which can be downloaded from **here**

NOTE: A compressed sample file is available for users to access easily

Step 2: Using tab **ConvertPED2BED** , Extract the downloaded compressed genotype PED files to create
(binary file-genotype information) ***.bed**
(first six columns of *.ped) ***fam**
(extended MAP file: two extra cols = allele names) ***bim**

How can you **SUBMIT** the extracted sample files (*.bed, *.fam, and *.bim) to create a model file?

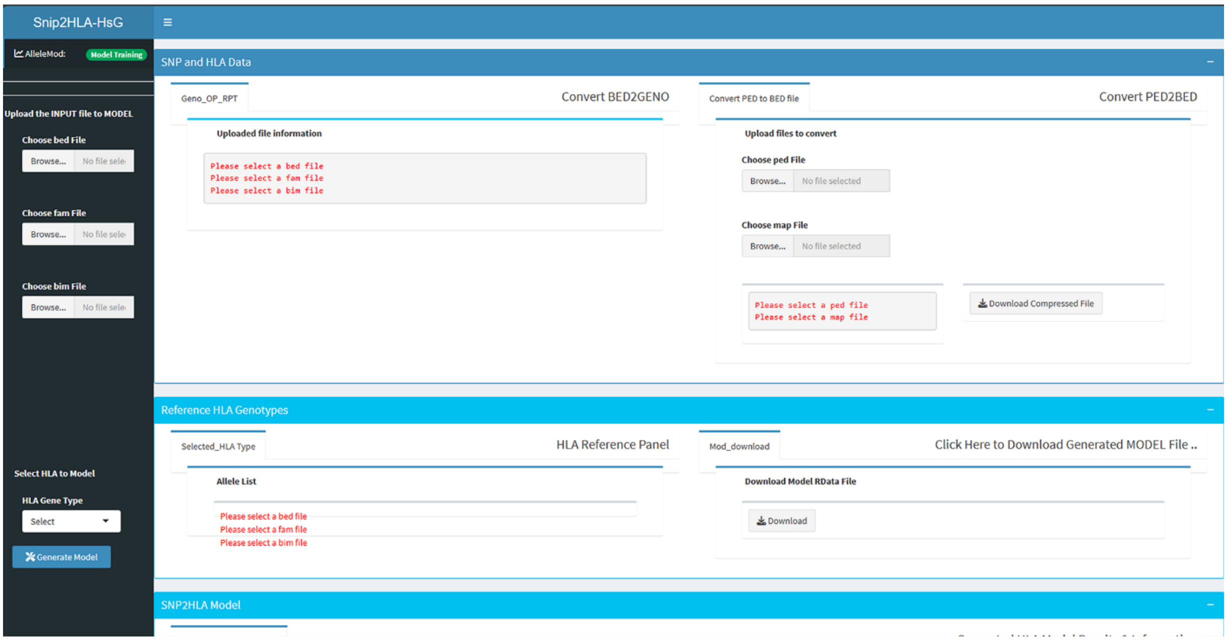
Step 3: Upload the extracted files (*.bed, *.fam, and *.bim) using the file input boxes inside the sidebar menu. It displays the SNP genotype data from sample files in a table format on the webpage
Step 4: Select one HLA Type from the options listed under **'HLA Gene Type'** and then choose either 1 or 10 as the classifier to create the model file.
Step 5: To create the Model, simply click the **'Generate Model'** button located in the sidebar menu.
Step 6: After completing the model creation process, click on the **'Download'** button under the tab **'Click Here to Download the Generated MODEL File'** and save the model file (RData) in download folder.

Instructions: ALLELE TYPE PREDICTION

Figure 2 Home page of the integrated web application, Snips2HLA-HsG.

3.1 Results Obtained through SNP Genotype Samples Preparation

The conversion of SNP genotype data into the *.ped and *.map file formats is a standard practice in population genetics and genetic analysis. These file formats are widely associated with the PLINK software package, which is a versatile tool for various genetic analyses, including the prediction of HLA allele types. Adopting the *.ped and *.map formats enhances data sharing and collaboration among researchers due to the standardized representation of genetic information, which is particularly valuable in HLA allele-type prediction research. For the benefit of users, sample data from the HapMap project for the Mexican population is provided on the homepage. The data includes MEX.ped (476 MB) and MEX.map (33 MB) files. The conversion from the PED (MEX.ped and MEX.map) file format to binary PED (MEX.bed, MEX.bim, and MEX.fam) will take a few minutes and be stored in the working directory. The SNP genotype data constitutes the primary sample for this study. The utilized application, Snips2HLA-HsG, the URL, https://snips2hla.shinyapps.io/hla_mod/ incorporates three designated fields to accommodate the input of genotype sample files, namely, the *.bed file, *.bim file, and *.fam file. This conversion offers more compact storage, improved computational efficiency, and compatibility with PLINK. These advantages contribute to streamlined genetic data analysis, enabling researchers to work more efficiently when dealing with large datasets. These selectable choices hold a prominent position on the application's main interface, as vividly portrayed in Figure 3.



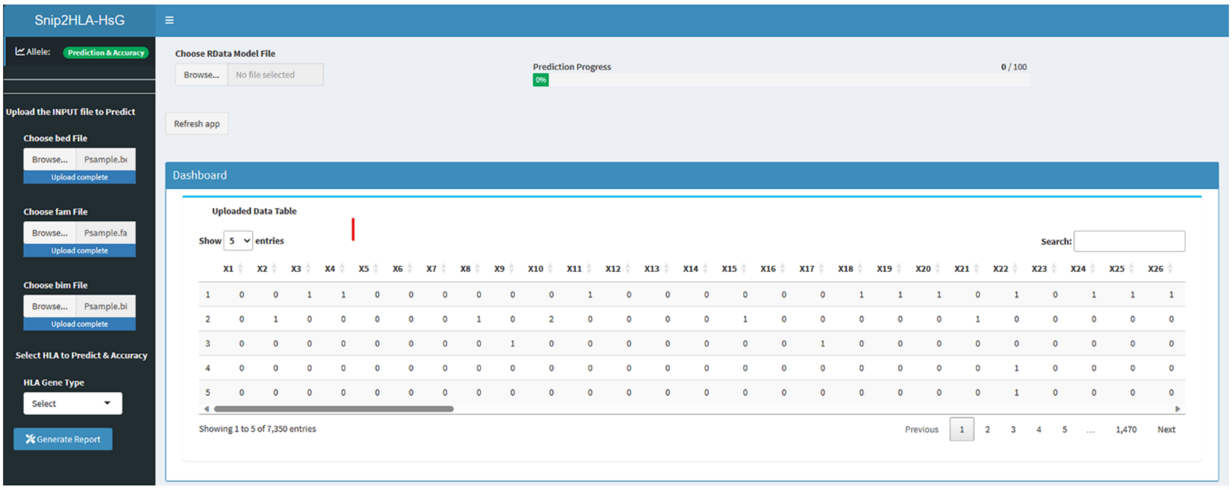
```
Please select a bed file
Please select a fam file
Please select a bim file
```

```
Please select a ped file
Please select a map file
```

Please select a bed file
Please select a fam file
Please select a bim file

Figure 3 Webpage where the input boxes for the generation of sample and Model.

As delineated in the methodology section, the approach involves a sequence of transformations. Specifically, initially in the *.ped and *.map file formats, the provided SNP genotype data undergoes conversion to yield *.bed, *.bim, and *.fam files. The outcome of this conversion materializes in the configuration of a numeric matrix format, wherein the BB genotype (signifying the absence of A allele) is represented by 0, the AB genotype (indicating the presence of one A allele) is represented by 1, and the AA genotype (indicating the presence of two A alleles) is denoted by 2. Instances of missing genotypes find representation as NA. Figure 4 visually explains these details, making it easier to understand and interpret.



Uploaded Data Table

Show 5 entries

Search:

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22	X23	X24	X25	X26
1	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	1	0	1	1	1
2	0	1	0	0	0	0	0	1	0	2	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Showing 1 to 5 of 7,350 entries

Previous 1 2 3 4 5 … 1,470 Next

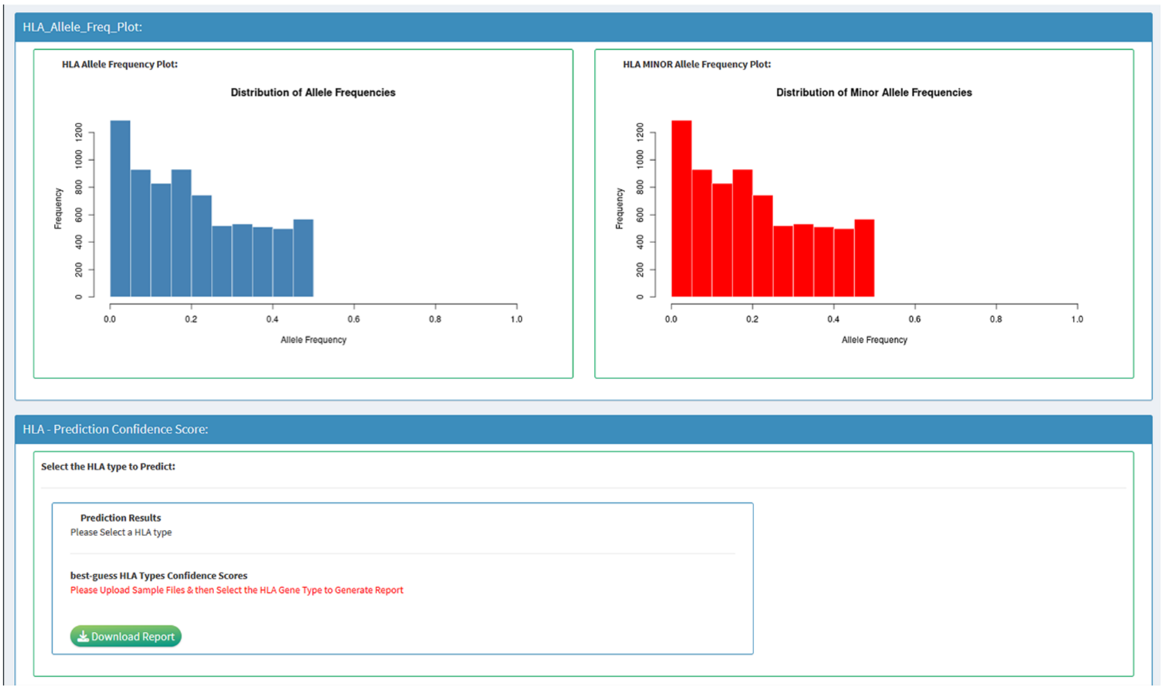
Figure 4 Displays the binary BED in numeric matrix format to represent the genotype.

3.2 Results Obtained through Model Generation

On the application webpage, located at https://snips2hla.shinyapps.io/hla_mod/, users encounter three input boxes designed for the upload of sample SNP genotype files in *.bed, *.bim, and *.fam formats. Users can select their own files or opt for those stored in the working directory. Under the outlined methodology in section 2.4, population-specific models take shape through the utilization of reference data for the true HLA allele "IDAWG_2018." Following the importation of SNP genotype sample files, users are empowered to specify the HLA gene type (e.g., DQB1) to create a gene and population-specific model.

The runtime for generating models from the sample binary PED files of the Mexican population (MEX.bed, MEX.bim, and MEX.fam) will take a couple of minutes and the resultant model file will be stored in the working directory as MEX.RData. Since the KG files contain a mixture of populations, it takes 15-20 minutes with 10 classifiers. If the number of classifiers increases, so does the processing time. This is why the application offers the choice of using either 1 or 10 classifiers.

The construction of plots showing the distribution of HLA allele frequencies, along with minor allele frequencies, is depicted in Figure 5. This visualization aids in understanding genetic patterns in population data. The model can be visualized in a scatterplot that shows the relationship between the frequency of SNP usage in an individual classifier and the genome coordinates, along with the model summary (Figure 6). Generating *.RData models for HLA imputation from binary bed files, using "IDAWG_2018" data, is crucial. Genetic Diversity, HLA Loci Complexity, and linkage disequilibrium patterns influence imputation accuracy. Population-specific models capture allele frequencies, manage loci complexity, and account for unique LD profiles, enhancing accuracy. Computational efficiency is vital for large-scale data, addressing admixture patterns. It optimizes algorithms and data processing for accuracy, efficiency, and relevance to the population studied.



HLA_Allele_Freq_Plot:

HLA Allele Frequency Plot:

HLA MINOR Allele Frequency Plot:

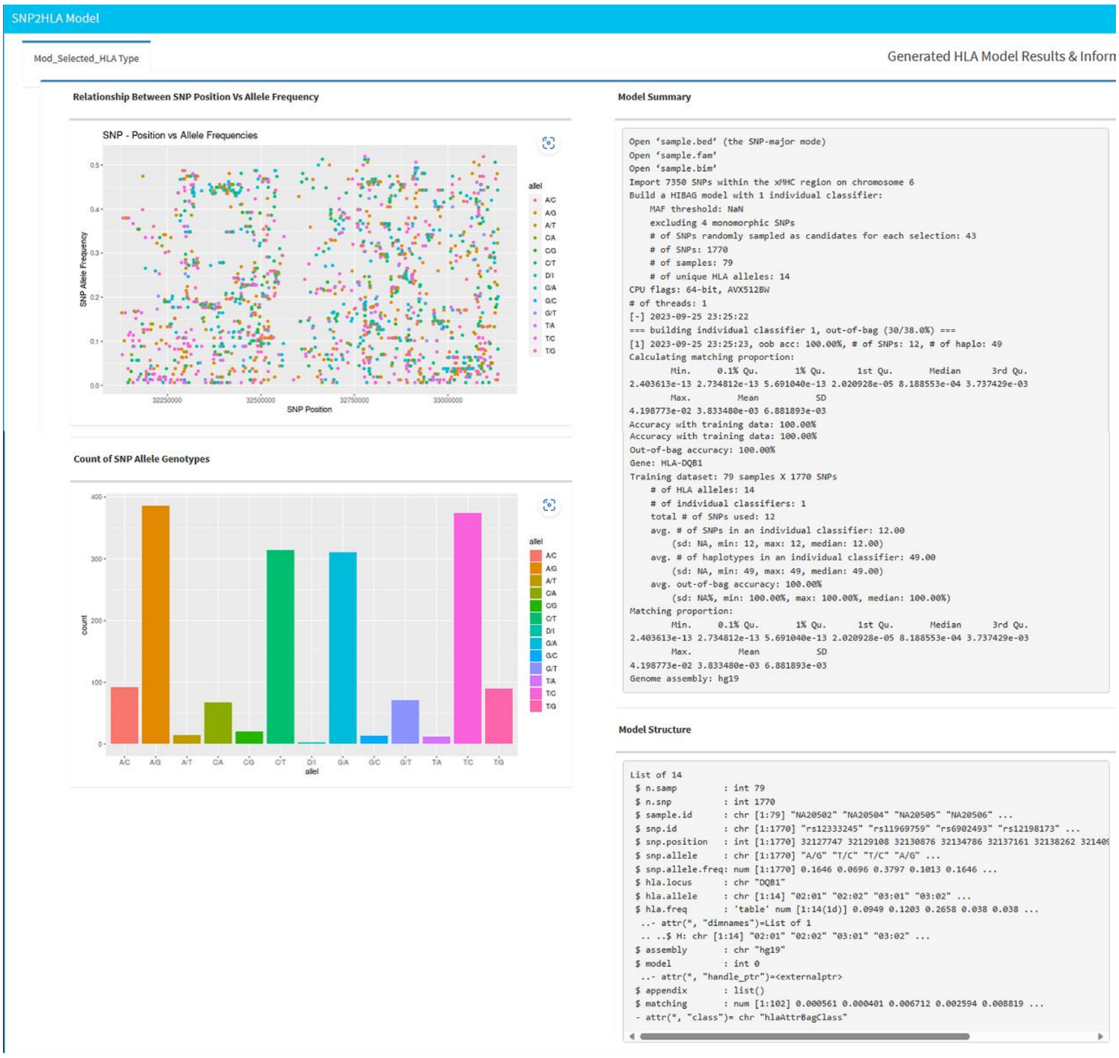
HLA - Prediction Confidence Score:

Select the HLA type to Predict:

Prediction Results
Please Select a HLA type

best-guess HLA Types Confidence Scores
Please Upload Sample Files & then Select the HLA Gene Type to Generate Report

Figure 5 Plots showing the distribution of HLA allele Frequency long with minor allele.



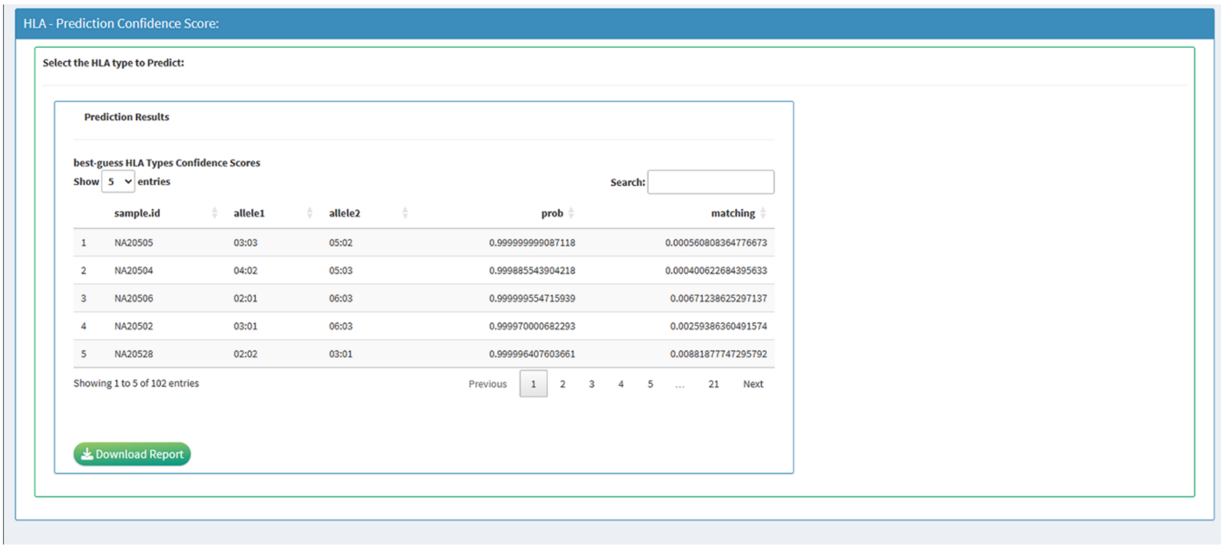
```
Open 'sample.bed' (the SNP-major mode)
Open 'sample.fam'
Open 'sample.bim'
Import 7350 SNPs within the xMHC region on chromosome 6
Build a HIBAG model with 1 individual classifier:
    MAF threshold: NaN
    excluding 4 monomorphic SNPs
    # of SNPs randomly sampled as candidates for each selection: 43
    # of SNPs: 1770
    # of samples: 79
    # of unique HLA alleles: 14
CPU flags: 64-bit, AVX512BW
# of threads: 1
[-] 2023-09-25 23:25:22
=== building individual classifier 1, out-of-bag (30/38.0%) ===
[1] 2023-09-25 23:25:23, oob acc: 100.00%, # of SNPs: 12, # of haplo: 49
Calculating matching proportion:
        Min.     0.1% Qu.       1% Qu.      1st Qu.       Median      3rd Qu.
2.403613e-13 2.734812e-13 5.691040e-13 2.020928e-05 8.188553e-04 3.737429e-03
        Max.         Mean           SD
4.198773e-02 3.833480e-03 6.881893e-03
Accuracy with training data: 100.00%
Accuracy with training data: 100.00%
Out-of-bag accuracy: 100.00%
Gene: HLA-DQB1
Training dataset: 79 samples X 1770 SNPs
    # of HLA alleles: 14
    # of individual classifiers: 1
    total # of SNPs used: 12
    avg. # of SNPs in an individual classifier: 12.00
        (sd: NA, min: 12, max: 12, median: 12.00)
    avg. # of haplotypes in an individual classifier: 49.00
        (sd: NA, min: 49, max: 49, median: 49.00)
    avg. out-of-bag accuracy: 100.00%
        (sd: NA%, min: 100.00%, max: 100.00%, median: 100.00%)
Matching proportion:
        Min.     0.1% Qu.       1% Qu.      1st Qu.       Median      3rd Qu.
2.403613e-13 2.734812e-13 5.691040e-13 2.020928e-05 8.188553e-04 3.737429e-03
        Max.         Mean           SD
4.198773e-02 3.833480e-03 6.881893e-03
Genome assembly: hg19
```

```
List of 14
 $ n.samp         : int 79
 $ n.snp          : int 1770
 $ sample.id      : chr [1:79] "NA20502" "NA20504" "NA20505" "NA20506" ...
 $ snp.id         : chr [1:1770] "rs12333245" "rs11969759" "rs6902493" "rs12198173" ...
 $ snp.position   : int [1:1770] 32127747 32129108 32130876 32134786 32137161 32138262 32140
 $ snp.allele     : chr [1:1770] "A/G" "T/C" "T/C" "A/G" ...
 $ snp.allele.freq: num [1:1770] 0.1646 0.0696 0.3797 0.1013 0.1646 ...
 $ hla.locus      : chr "DQB1"
 $ hla.allele     : chr [1:14] "02:01" "02:02" "03:01" "03:02" ...
 $ hla.freq       : 'table' num [1:14(1d)] 0.0949 0.1203 0.2658 0.038 0.038 ...
  ..- attr(*, "dimnames")=List of 1
  .. ..$ H: chr [1:14] "02:01" "02:02" "03:01" "03:02" ...
 $ assembly       : chr "hg19"
 $ model          : int 0
  ..- attr(*, "handle_ptr")=<externalptr>
 $ appendix       : list()
 $ matching       : num [1:102] 0.000561 0.000401 0.006712 0.002594 0.008819 ...
 - attr(*, "class")= chr "hlaAttrBagClass"
```

Figure 6 Depiction of the relationship between SNP positions vs. Allele Frequency and count of SNP allele genotypes.

3.3 HLA Type Prediction Results Using the Generated Model and Accuracy Evaluation

One can use Snips2HLA-HsG at https://snips2hla.shinyapps.io/hla_pred/ to predict HLA allele types by uploading SNP genotype files (*.bed, *.bim, *.fam) and their corresponding model files (*.RData). Choose the HLA gene type for computational prediction based on population-specific models from "IDAWG_2018" data in *.Rdata format, ensuring reliability in the genetics process. Prediction results, including probability and matching scores, sample ID, allele 1, and allele 2, are displayed in Figure 7.



HLA - Prediction Confidence Score:

Select the HLA type to Predict:

Prediction Results

best-guess HLA Types Confidence Scores

Show 5 entries

Search:

	sample.id	allele1	allele2	prob	matching
1	NA20505	03:03	05:02	0.999999999087118	0.000560808364776673
2	NA20504	04:02	05:03	0.999885543904218	0.000400622684395633
3	NA20506	02:01	06:03	0.999999554715939	0.00671238625297137
4	NA20502	03:01	06:03	0.999970000682293	0.00259386360491574
5	NA20528	02:02	03:01	0.999996407603661	0.00881877747295792

Showing 1 to 5 of 102 entries

Figure 7 Screenshot showing the result of HLA allele type prediction.

The runtime for HLA type prediction and accuracy evaluation using the sample binary PED files of the Mexican population (MEX.bed, MEX.bim, and MEX.fam) against MEX.RData will take a few minutes, and the results will be displayed. The KG files take a few minutes longer. Therefore, the processing time depends on the nature of the sample and model files.

Population-specific models, derived and stored from "IDAWG_2018" data in *.Rdata format, align HLA type predictions with the target population's genetic characteristics. These models capture allele frequencies, linkage disequilibrium patterns, and genetic diversity, enhancing prediction reliability. The "Allele1" and "Allele2" data provide predicted HLA alleles, considering population-specific allele frequency distributions. The "Probability" data indicates confidence levels, with higher scores indicating more reliable predictions. The “Matching" data assesses how closely predicted alleles match known ones from "IDAWG_2018, " contributing to prediction accuracy. Calibration and training using actual HLA allele data optimize model parameters for accurate predictions. Confusion matrices offer valuable information on sensitivity, specificity, and predictive values for each allele, guiding further refinement of prediction algorithms, Following these steps, executing prediction analysis for HLA-A, B, -C, -DRB1, and -DQB1, utilizing the Snips2HLA-HsG application with a user-friendly interface for input data files. The markdown format exportation of findings ensures accessibility for end-users, fostering transparency and trust in HLA-type prediction outcomes.

4. Conclusions

The primary objective of the present study is to develop an integrated application, Snips2HLA-HsG, for SNP genotype analysis and HLA allele type prediction, including accuracy evaluation. This is achieved by utilizing attribute bagging, an ensemble classifier technique in the Bioconductor HIBAG package. The application can be accessed via the URL: https://snips2hla.shinyapps.io/hla_home/. While numerous software tools exist for HLA imputation, a significant portion of them lacks user-friendliness, integration of all-purpose including sample preparation, model generation and HLA prediction, and accuracy evaluation. In contrast, the application developed in this study prioritizes user-friendliness, ensuring accessibility even for biologists with limited computer knowledge.

The application utilizes SNP and HLA allele genotype data from trusted primary biological resources like HapMap, 1000 genomes, IMGT/HLA, HGDP, and the Allele Frequency Database. Notably, the application simplifies the sample and model preparation process, catering to the end users. Furthermore, users can visualize the SNP genotype data patterns in the generated models. Another advantage is providing real-time results to users as they input their data, enhancing the user experience and eliminating the wait for results in emails. Overall, it is an educational tool for students and researchers to understand the preparation of samples and models, as well as HLA allele prediction and accuracy checking.

The application offers one or ten classifiers to accommodate data types and processing times. Using fewer classifiers (1 or 10) may limit accuracy. Therefore, the study plans to address this limitation in future updates by adding more classifiers to optimize server utility. This enhancement aims to utilize multiple cores for faster calculations and to meet research needs effectively. Looking ahead, the application, Snips2HLA-HsG, will undergo regular updates and maintenance to ensure its continued effectiveness and relevance in genetic research. Maintenance efforts will focus on resolving any issues or bugs that may arise and providing ongoing support to users.

Author Contributions

Dr. Balamurugan Sivaprakasam was responsible for designing methodology, data curation, writing and reviewing. Dr. Prasanna Sadagopan involved in supervision.

Competing Interests

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Naito T, Okada Y. HLA imputation and its application to genetic and molecular fine-mapping of the MHC region in autoimmune diseases. Semin Immunopathol. 2022; 44: 15-28.
2. Sanchez-Mazas A. A review of HLA allele and SNP associations with highly prevalent infectious diseases in human populations. Swiss Med Wkly. 2020; 150: w20214.
3. Gonzalez-Galarza FF, McCabe A, Dos Santos EJ, Jones AR, Middleton D. A snapshot of human leukocyte antigen (HLA) diversity using data from the allele frequency net database. Hum Immunol. 2021; 82: 496-504.
4. Thorisson GA, Smith AV, Krishnan L, Stein LD. The international HapMap project web site. Genome Res. 2005; 15: 1592-1593.
5. Leslie S, Donnelly P, McVean G. A statistical method for predicting classical HLA alleles from SNP data. Am J Hum Genet. 2008; 82: 48-56.
6. Gao J, Zhu C, Zhu Z, Tang L, Liu L, Wen L, et al. The human leukocyte antigen and genetic susceptibility in human diseases. J BioX Res. 2019; 2: 112-120.
7. Douillard V, Castelli EC, Mack SJ, Hollenbach JA, Gourraud PA, Vince N, et al. Approaching genetics through the MHC lens: Tools and methods for HLA research. Front Genet. 2021; 12: 774916.

8. Dilthey AT, Moutsianas L, Leslie S, McVean G. HLA* IMP-an integrated framework for imputing classical HLA alleles from SNP genotypes. Bioinformatics. 2011; 27: 968-972.
9. Browning SR, Browning BL. Haplotype phasing: Existing methods and new developments. Nat Rev Genet. 2011; 12: 703-714.
10. Jia X, Han B, Onengut-Gumuscu S, Chen WM, Concannon PJ, Rich SS, et al. Imputing amino acid polymorphisms in human leukocyte antigens. PLoS One. 2013; 8: e64683.
11. Zheng X, Shen J, Cox C, Wakefield JC, Ehm MG, Nelson MR, et al. HIBAG-HLA genotype imputation with attribute bagging. Pharmacogenomics J. 2014; 14: 192-200.
12. Jeanmougin M, Noirel J, Coulonges C, Zagury JF. HLA-check: Evaluating HLA data from SNP information. BMC Bioinformatics. 2017; 18: 334.
13. Shen JJ, Yang C, Wang YF, Wang TY, Guo M, Lau YL, et al. HLA-IMPUTER: An easy to use web application for HLA imputation and association analysis using population-specific reference panels. Bioinformatics. 2019; 35: 1244-1246.
14. Naito T, Suzuki K, Hirata J, Kamatani Y, Matsuda K, Toda T, et al. A deep learning method for HLA imputation and trans-ethnic MHC fine-mapping of type 1 diabetes. Nat Commun. 2021; 12: 1639.
15. Cook S, Choi W, Lim H, Luo Y, Kim K, Jia X, et al. Accurate imputation of human leukocyte antigens with CookHLA. Nat Commun. 2021; 12: 1264.
16. Boegel S, Löwer M, Schäfer M, Bukur T, De Graaf J, Boisguérin V, et al. HLA typing from RNA-seq sequence reads. Genome Med. 2013; 4: 102.
17. Dilthey A, Leslie S, Moutsianas L, Shen J, Cox C, Nelson MR, et al. Multi-population classical HLA type imputation. PLoS Comput Biol. 2013; 9: e1002877.
18. Sakaue S, Gurajala S, Curtis M, Luo Y, Choi W, Ishigaki K, et al. Tutorial: A statistical genetics guide to identifying HLA alleles driving complex disease. Nat Protoc. 2023; 18: 2625-2641.
19. Sivaprakasam B, Sadagopan P. HLA allele type prediction: A review on concepts, methods and algorithms. Asian J Biol Life Sci. 2023; 12: 206-215.
20. Nanjala R, Mbiyavanga M, Hashim S, de Villiers S, Mulder N. Assessing HLA imputation accuracy in a west African population. bioRxiv. 2023. doi: 10.1101/2023.01.23.525129.
21. Chang CC. Data management and summary statistics with PLINK. In: Statistical population genomics. Methods in Molecular Biology. New York, NY: Humana Press; 2020. pp. 57-73.
22. Li MX, Jiang L, Kao PY, Sham PC, Song YQ. IGG3: A tool to rapidly integrate large genotype datasets for whole-genome imputation and individual-level meta-analysis. Bioinformatics. 2009; 25: 1449-1450.
23. Zheng X. Imputation-based HLA typing with SNPs in GWAS studies. In: HLA typing. Methods in Molecular Biology. New York, NY: Humana Press; 2018. pp. 163-176.
24. Zheng-Bradley X, Flicek P. Applications of the 1000 genomes project resources. Brief Funct Genomics. 2017; 16: 163-170.
25. Belsare S, Levy-Sakin M, Mostovoy Y, Durinck S, Chaudhuri S, Xiao M, et al. Evaluating the quality of the 1000 genomes project data. BMC Genomics. 2019; 20: 620.