# PopMLviz Software v1

User Manual

1/6/2022

# Contents

## 1. Quick start

PopMLvis is a population genetic analysis application. We provide two versions of PopMLvis; a web-based (online) and computer-based (offline) application. It provides a comprehensive interactive environment for scientists, bioinformaticians, and researchers to dig deeper in analyzing population genetic datasets. In order to understand the gene structure, our platform analysis includes dimensionality reduction algorithms, machine learning models, statistical measurements.

## 2. PopMLvis data types

PopMLViz supports different types of input datasets. This gives more flexibility to users on how this tool can be used for.

2.1. ***Genes population data***: This is the standard (raw) dataset that represents genes population. Each column represents a gene and each row represents an individual.

2.2. ***Correlation matrix data***: This dataset represents the correlation of each individual corresponding to the rest of all other individuals in the dataset.

2.3. ***Principal components data***: It represents the genes population dataset after applying principal component analysis (PCA) to the raw genes population dataset.

2.4. ***Genes structure dataset***: This is a customized dataset that is necessary to run genes structure algorithms. The dataset should include the kinship correlation matrix that specifies the family relations.

## 3. PopMLvis data format

PopMLViz accepts multiple file formats that represent gene population data.

3.1. ***Pickle file:*** it is a binary format that can be used to store gene population dataset, including metadata fields. Pickle is used internally by python to serialize objects and is a faster and

flexible format. However, it is not supported by many programs (applications/software).

3.2. ***Character Separated Value (CSV) file:*** it is the standard format that is widely used by many applications. It is easy to create a CSV file and to edit it. However, it is not efficient in terms of loading speed and disk usage. The format of the input CSV file could be tab, comma, or semicolon field separator with a single header raw that could include:

*id*: it represents the id for a single individual.
*Population*: it is an index that specifies to which population every individual belongs to.
*Gender*: a string that represents if the individual is male, or female.
*Age*: a numeric number that represents an individual's age.
*Phenotype*: an arbitrary string that describes the individual's phenotype.
*Metadata information*: these are extra columns that could be included in the dataset. It will be treated as meta information for each individual.

A typical PopMLvis CSV file will be like the following:
*id, population, gender, age, rs123, rs456, rs789*
xx, yy, zz, …

3.3. ***Gene structure files:*** it carries a binary format, and it is customized for running gene structure algorithms. The user must provide three binary files with extensions: .bed, .bim, and .fam. In addition, the user should specify the relationship between the individual's dataset through providing the kinship matrix in .txt, or .csv format.

## 4. Projections (dimensionality reduction algorithms)

PopMLvis supports multiple dimensionality reduction algorithms, which helps visualizing diversity in gene population dataset.

4.1. ***Principal components analysis (PCA):*** principal components analysis (PCA) is traditional, well-known, and the mostly used linear transformation technique to visualize the genetic diversity in a dataset. It focuses on capturing the direction of maximum variation in a dataset through these principal components.

4.2. ***Linear Discriminant analysis (LDA):*** It is a linear transformation technique, like PCA, with an aim of finding a linear combination of features that best explain the genotype dataset. Also, it could be categorized as a supervised dimensionality reduction technique, which could be exploited in classifying the dataset simultaneously.

4.3. ***t-Distributed Stochastic Neighbor Embedding (t-SNE):*** It is a non-linear transformation technique that is well-suited for embedding high-dimensional data for visualization in low dimensional space of two, or three dimensions. It tries to preserve the local structure (cluster) of genetic data and capture outliers simultaneously.

4.4. ***Principal component analysis in related samples (PC-Air):*** It is used to perform a principal components analysis using genome-wide SNP data for the detection of population structure in a sample. Unlike a standard PCA, PC-Air accounts for sample relatedness (*known* or *cryptic*) to provide accurate ancestry inference that is not confounded by family structure.

5. Clustering algorithms

5.1. ***K-means:*** It is one of the most popular clustering algorithms. It stores k-centroids, which is used to define the clusters. Then, each data point is assigned to the nearest cluster centroid. After that, it calculates the means (updated centroids) of data points in each cluster. This process is repeated until the assignments of data points no longer change.

5.2. ***Fuzzy c-means:*** It is similar to the K-means, but instead of assigning each data point to only one cluster, each data point can belong to many clusters with a weighting percentage. So,

as a data point close to the cluster centroid, as the weighting percentage increases and vice versa.

5.3. ***Hierarchical:*** The general strategy is to follow a bottom-up approach "*agglomerative*", where each data point starts in its own cluster and pairs of clusters are merged as one moves up the hierarchy. We end-up by having only one cluster for the whole genotype dataset. Then, based on the user's decision of how dissimilar clusters should be; a threshold value is applied. A dendrogram "tree-like" is the commonly used representation for hierarchical clustering.

## 6. Population structure and ancestry algorithms

6.1. ***Admixture:*** It is the mostly used algorithm to estimate ancestry in a model-based manner from large genotype datasets.

## 7. Detection and removal of outliers

It is important in the genetics field to detect outliers, as it could represent … . There are many outlier detection methods, PopMLvis supports some of them:

7.1. ***Statistical metrics:*** Assuming that the genotype dataset is normally distributed; we consider the data points which fall below m-3s or above m+3s are outliers. In addition, PopMLvis has an option to consider the data points that fall within either m±s or m±2s as outliers.

7.2. ***Isolation Forest:*** As the name indicates, it identifies anomalies by isolating outliers in the data. It is based on the decision-tree algorithm, where it recursively generates partitions on the dataset by randomly selecting a feature and then randomly selecting a split value for the feature.

7.3. ***Minimum Covariance Determinant:*** It estimates the mean and covariance matrix for each subset in the data. Then, it keeps the estimates for the subset whose covariance matrix has the smallest determinant (the most tightly distributed).

7.4. ***Local Outlier Factor:*** The anomaly score of each sample is called the Local Outlier Factor. It measures the local deviation of the density of a given sample with respect to its neighbors, where the locality is given by k-nearest neighbors, whose distance is used to estimate the local density.

7.5. ***OneClassSVM:*** It is a variation of the SVM classification algorithm. Here, the algorithm is modeled as one class, which permits the algorithm to capture the density of the majority class and classifies examples on the extremes of the density function as outliers.

## 8. Visualization

In this part, the user will visually experience how the PopMLvis looks like and learn how to use it.

8.1. ***Main dashboard:*** The main window of PopMLvis overviews all components of the application.


*Figure 1: The main window of PopMLvis*

As you can see in Figure (1), the primary PopMLvis dashboard can be categorized into five panels:

a. *Input panel:* This panel provides the user freedom to choose the dataset type he/she wants to upload based on his/her needs. In addition, we provide a sample dataset that can be used as well.


*Figure 2: Input panel of PopMLvis*

Once the user chooses a specific algorithm, PopMLvis will enable the corresponding button(s) to allow uploading the required file(s).

(a) Correlation Matrix


(b) PC-Air and Admixture


*Figure 3: Two different examples of Input algorithms*

As shown in Figure 3 (a), when the user chooses the raw correlation matrix; a choose file button is enabled. Also, a drop-down menu appears asking the user, which dimensionality reduction he/she wants to perform on the correlation matrix. Similarly, when the user chooses PC-Air and Admixture option; PopMLvis enables buttons for the two corresponding datasets to be uploaded (see Figure 3 (b)).

b. *Visualization panel:* This panel provides the user different options to choose from in terms of the number of dimensions (1D, 2D, or 3D) and which principal components to be viewed (see Figure 3).

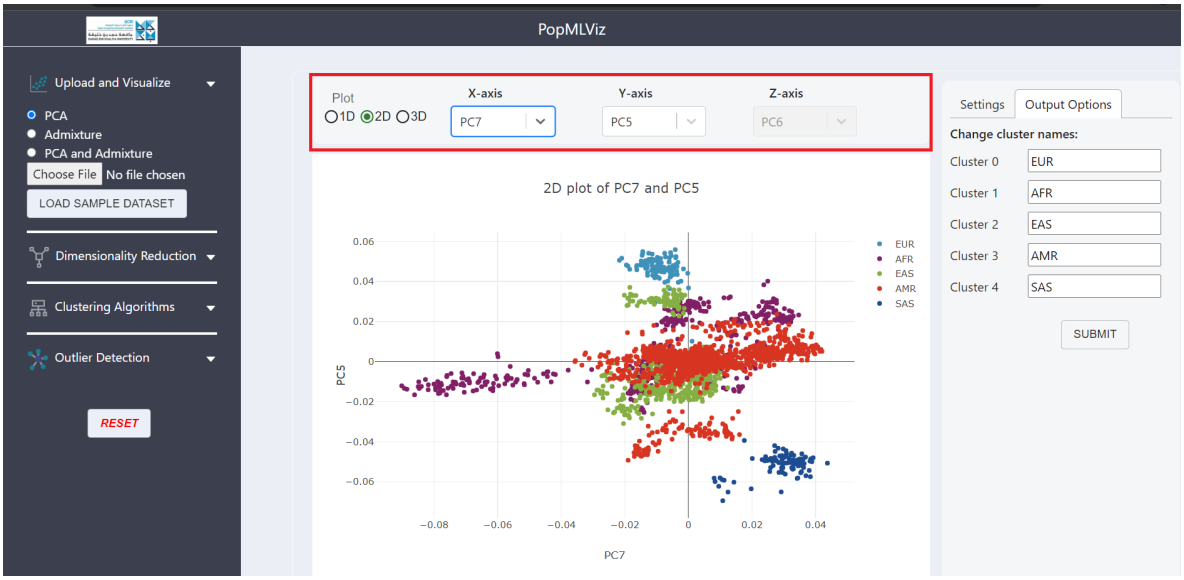


*Figure 3: Viewing the principal components*

c. *Clustering panel:* This panel allows the user to apply variety of clustering algorithms to the uploaded dataset and visualize the results spontaneously. For each algorithm, the user can setup the parameters such as the number of clusters.


*Figure 4: Clustering algorithms*

d. *Outlier detection panel:* The user can specify which principal component he/she wants to remove outliers from, and it is up to the user to choose more than one principal component (see

Figure 5). Also, the user can select if he/she wants to do AND or OR operation when there is more than one principal component. Moreover, the user has to decide the range of samples that should be considered as outlier, for example, one standard-deviation, two standard-deviations, etc.

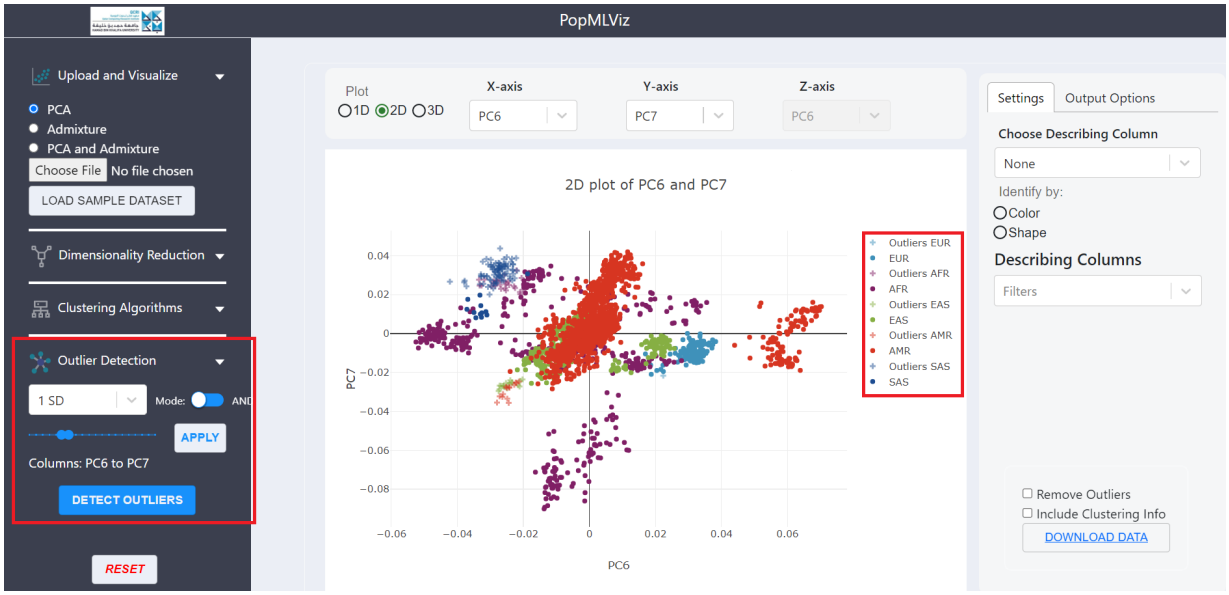


*Figure 5: Outlier detection panel*

## 9. Exporting outputs

After performing the required operations, the user can export the output in a csv file. As you can see in Figure (6), the user can download the data with the following options

### a.) Removing Outliers

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 | PC9 | PC10 | PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 | PC18 | PC19 | PC20 | IID | ancestry | cluster |
| 2 | 0.01125 | -0.02719 | -0.01224 | 0.017589 | -0.00123 | 0.006608 | 0.001872 | -0.01893 | 0.010684 | 0.014557 | -0.0036 | -0.02652 | 0.001872 | 0.014295 | 0.026714 | -0.01264 | 0.000767 | -0.00563 | 0.013243 | -0.00162 | HG00096 | EUR | AMR |
| 3 | 0.011029 | -0.02677 | -0.01082 | 0.016826 | 0.000965 | 0.007595 | 0.007671 | -0.03098 | 0.009948 | 0.005814 | 0.004331 | 0.001444 | 0.008292 | 0.017359 | 0.019141 | -0.00817 | 0.001275 | 0.002693 | 0.010754 | 0.005611 | HG00097 | EUR | AMR |
| 4 | 0.01131 | -0.02687 | -0.01176 | 0.015817 | 0.001032 | 0.00609 | 0.005654 | -0.03414 | 0.01149 | 0.012079 | 0.007453 | 0.000689 | 0.003353 | 0.020264 | 0.021398 | -0.01321 | 0.0078 | 0.002397 | -0.00273 | 0.003889 | HG00099 | EUR | AMR |
| 5 | 0.010922 | -0.02702 | -0.01218 | 0.018376 | -0.00071 | 0.0043 | -0.00307 | -0.0024 | 0.001356 | 0.01578 | -0.00669 | 0.040794 | -0.01158 | 0.017124 | 0.019595 | -0.00603 | 0.008113 | 0.005191 | 0.000113 | 0.009873 | HG00100 | EUR | AMR |
| 6 | 0.01119 | -0.0268 | -0.01239 | 0.016547 | 0.001938 | 0.005802 | 0.00268 | -0.02904 | 0.008261 | 0.010632 | 0.008104 | 0.007729 | 0.007059 | 0.019129 | 0.039442 | -0.01268 | 0.007005 | -0.00064 | 0.002292 | 0.005298 | HG00101 | EUR | AMR |

*Figure 6: (a) Data excluding outliers*

### b.) Including Clustering Information

| | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 | PC9 | PC10 | PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 | PC18 | PC19 | PC20 | IID | ancestry | cluster | outlier |
| 2 | -0.02719 | -0.01224 | 0.017589 | -0.00123 | 0.006608 | 0.001872 | -0.01893 | 0.010684 | 0.014557 | -0.0036 | -0.02652 | 0.001872 | 0.014295 | 0.026714 | -0.01264 | 0.000767 | -0.00563 | 0.013243 | -0.00162 | HG00096 | EUR | AMR | 0 |
| 3 | -0.02677 | -0.01082 | 0.016826 | 0.000965 | 0.007595 | 0.007671 | -0.03098 | 0.009948 | 0.005814 | 0.004331 | 0.001444 | 0.008292 | 0.017359 | 0.019141 | -0.00817 | 0.001275 | 0.002693 | 0.010754 | 0.005611 | HG00097 | EUR | AMR | 0 |
| 4 | -0.02687 | -0.01176 | 0.015817 | 0.001032 | 0.00609 | 0.005654 | -0.03414 | 0.01149 | 0.012079 | 0.007453 | 0.000689 | 0.003353 | 0.020264 | 0.021398 | -0.01321 | 0.0078 | 0.002397 | -0.00273 | 0.003889 | HG00099 | EUR | AMR | 0 |
| 5 | -0.02702 | -0.01218 | 0.018376 | -0.00071 | 0.0043 | -0.00307 | -0.0024 | 0.001356 | 0.01578 | -0.00669 | 0.040794 | -0.01158 | 0.017124 | 0.019595 | -0.00603 | 0.008113 | 0.005191 | 0.000113 | 0.009873 | HG00100 | EUR | AMR | 0 |
| 6 | -0.0268 | -0.01239 | 0.016547 | 0.001938 | 0.005802 | 0.00268 | -0.02904 | 0.008261 | 0.010632 | 0.008104 | 0.007729 | 0.007059 | 0.019129 | 0.039442 | -0.01268 | 0.007005 | -0.00064 | 0.002292 | 0.005298 | HG00101 | EUR | AMR | 0 |
| 7 | -0.02681 | -0.01142 | 0.017157 | 0.00037 | 0.007465 | 0.001083 | -0.02782 | 0.005286 | 0.008128 | 0.00306 | 0.007449 | 0.003999 | 0.016458 | 0.032868 | -0.01325 | 0.006091 | 0.000315 | 0.006644 | 0.012942 | HG00102 | EUR | AMR | 0 |
| 8 | -0.02661 | -0.01177 | 0.017444 | -0.00124 | 0.004727 | 0.00506 | -0.02233 | 0.010386 | 0.015461 | 0.008195 | -0.02155 | 0.004586 | 0.007559 | 0.02404 | -0.01152 | -0.00152 | -0.00195 | 0.012246 | 0.001235 | HG00103 | EUR | AMR | 0 |
| 9 | -0.02652 | -0.01137 | 0.017209 | 0.001566 | 0.006101 | 0.007439 | -0.02789 | 0.008366 | 0.011046 | 0.002586 | 0.006362 | 0.012514 | 0.018737 | 0.027375 | -0.00675 | 0.006601 | 0.006262 | -0.00193 | -0.00247 | HG00105 | EUR | AMR | 0 |

*Figure 6: (b) Including Cluster Information*

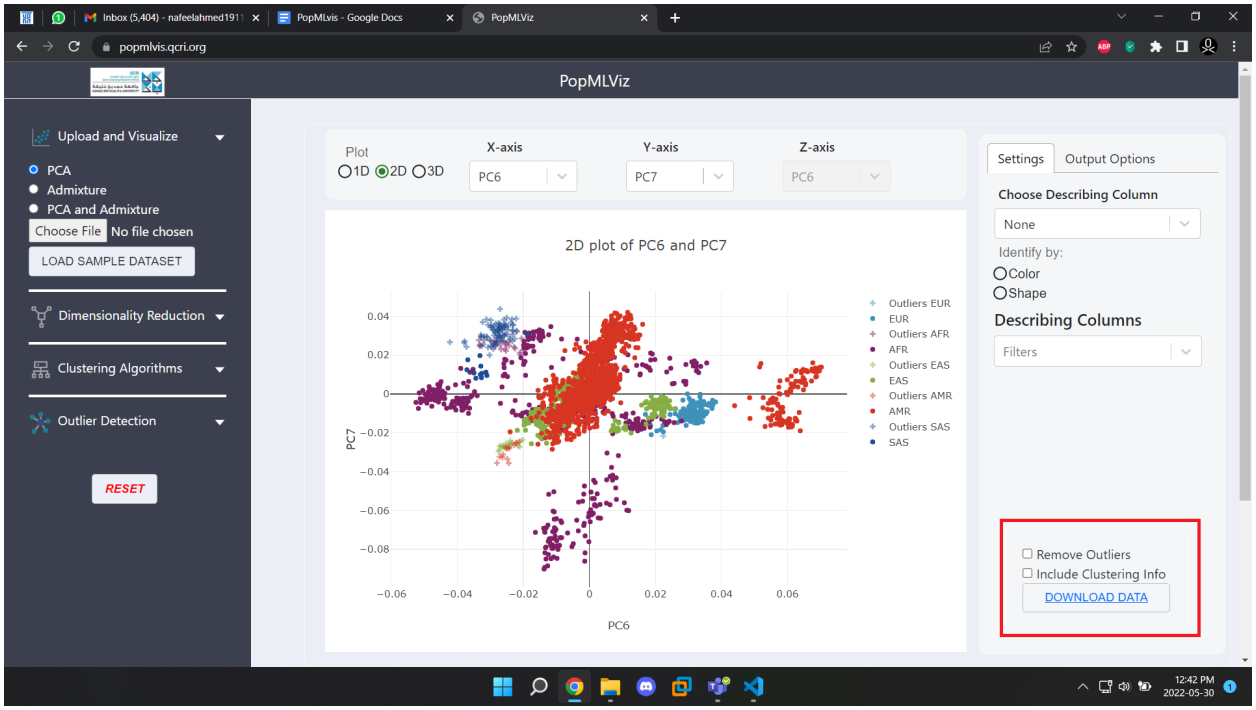


*Figure 6: (c) Download Data panel*

## 10. How to install PopMLvis on your machine?

In case the user use's confidential data, PopMLvis provides an offline version, where the user can install it on his/her own machine. PopMLvis supports windows, macOS, and linux operating systems.

The github repository provides an installation script which can be used to install the software and run it locally at the defined port.

Usage: *foo@bar: ~$ ./installationScript.sh*

Alternatively, you can manually install the software using the following commands

Usage:

1.) Cloning the Github Repository

*foo@bar: ~$ git clone https://github.com/qcri/QCAI-PopMLVis.git*

*foo@bar: ~$ cd PopMLViz*

2.) Setting up a flask environment and installing the requirements

*foo@bar: ~$ mkdir backend/data*

*foo@bar: ~$ cd backend/data*

*foo@bar: ~$ source flaskenv/bin/activate*

*foo@bar: ~$ pip install -r flask_req.txt*

*foo@bar: ~$ ./run*

3.) Setting up parameters for the frontpage and starting the software

*foo@bar: ~$ cd frontend/*
*foo@bar: ~$ echo "REACT_APP_DOMAIN=localhost" > .env*
*foo@bar: ~$ echo "REACT_APP_PROTOCOL=http" >> .env*
*foo@bar: ~$ echo "REACT_APP_PORT=:5000" >> .env*
*foo@bar: ~$ npm install*
*foo@bar: ~$ npm start*

## 11. Citation and further information

If you find our platform useful, please cite our paper:

## 12. Technical support

In case the user has any question, or suggestion regarding the platform, he/she can send us an enquiry at popmlvissupport@QCRI.org

## 13. Future work