# PopMLvis Platform v1

User Manual

August 21, 2022

# Contents

## 1. Quick start

PopMLvis is a population genetic visualization application for genome-wide association study (GWAS) data. We provide two versions of PopMLvis; a web-based online version and a computer-based offline version/ application. PopMLvis provides a comprehensive interactive environment to explore population structure using genetic data, and explore/infer ancestry groups and clusters using various dimensionality reduction algorithms.

## 2. PopMLvis data types

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

2.1. ***Genome-wide association study data***: This is the standard dataset that is used to test the association between genetic variables and disease of interest. The data contains genotypes of subjects and is highly dimensional (thousands of SNPs and subjects).

2.2. ***Correlation/Kinship matrix data***: This dataset (N*N) contains the genetic correlation/kinship between all pairs of subjects.

2.3. ***Principal components data***: These are pre-computed principal components by the user using genotype data or other types of data.

2.4. ***Admixture dataset***: This dataset is the result of *Admixture* (or similar models), which contains the admixture fractions of subjects across a predefined number of clusters.

## 3. PopMLvis data format

PopMLVis accepts multiple file formats, which represent gene population data.

3.1. ***Comma Separated Value (CSV) file:*** common, space or tab-delimited input files are accepted. Headers are required and can include:

*IID*: it represents the id of a single individual.
*PC1:* the first Principal Component
*PC2:* the second Principal Component
…
*PCN*:the Nth Principal Component

*Metadata information*: these are extra columns that could be included in the dataset (e.g., Ancestry, Age, Sex, Phenotype status, SNPs, etc.).

3.2. ***GWAS data:*** this is the binary plink format. Three files are required: .bed, .bim, and .fam. In addition, if users want to run PC-air, they should provide a correlation/kinship matrix of all pairs of individuals (space- or comma-delimited). This can be computed by many tools such as plink, GCTA, KING, etc. If the kinship matrix is not provided, PopMLvis uses the identity matrix by default.

3.3. ***Pickle file:*** it is a binary format that can be used to store gene population datasets, including metadata fields. Pickle is used internally by python to serialize objects. It is a faster and more flexible format. However, it is not supported by many programs (applications/software).

Note that many kinship calculators provide outputs in a long format, e.g., in the case of GCTA, the output looks as follows:

IID1 IID1 Kinship1
IID1 IID2 Kinship2
IID1 IID3 Kinship3
etc.

To convert to our matrix format, users can use the following code in R:

```
#####
fam = read.table("data.fam")  # data.fam is the fam file used to compute kinship
grm = read.table("kinship.grm") # the long-formatted kinship
list_self = which(grm$V1 == grm$V2) # self-kinship
grm_noself= grm_noself[-c(list_self ),]
out = matrix(NA , nc=nrow(fam) , nr= nrow(fam))
diag(out) = grm[list_self,4]
out[upper.tri(out)] = (grm_noself$V4)
out[lower.tri(out)] = t(out)[lower.tri(out)]
write.table(out , "GCTA_matrix",quote=FALSE,row.names=FALSE,col.names=FALSE)
######
```

## 4. Projections (dimensionality reduction algorithms)

PopMLvis supports multiple dimensionality reduction algorithms, which help visualize the latent structure in GWAS dataset.

4.1. ***Principal components analysis (PCA):*** principal components analysis (PCA) is a traditional, well-known, and most 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. ***Principal component analysis accounting for relatedness between subjects (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.

4.3. ***Linear Discriminant analysis (LDA):*** It is a linear transformation technique, like PCA, to find a linear combination of features that best explain the GWAS dataset. It could be categorized as a supervised dimensionality reduction technique, which could be exploited in classifying the dataset simultaneously.

4.4. ***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 a low-dimensional space of two or three dimensions. It tries to preserve the local structure (cluster) of genetic data and capture outliers simultaneously.

5. Clustering algorithms

5.1. ***K-means:*** It is one of the most popular clustering algorithms. It stores k-centroids, which are used to define the clusters (ancestry groups). Then, each data point, which represents an individual, 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 assignment of data points no longer changes, which means that each subject is assigned to a given cluster (e.g., ancestry group).

5.2. ***Fuzzy c-means:*** It is similar to K-means, but instead of assigning each data point (i.e., individual) to only one cluster, each data point can belong to many clusters with a weighting percentage. The weighting percentage increases if data points are close to the cluster centroid and decrease if they are far from the centroids.

5.3. ***Hierarchical clustering:*** The general strategy is to follow a bottom-up approach "*agglomerative*", where each data point starts in its cluster and pairs of clusters are merged as one moves up the hierarchy. We end up 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. Admixture analysis

6.1. ***Admixture software:*** It is one of the widely used admixture algorithms to estimate ancestry fractions of each subject. This is a supervised approach where a predefined number of clusters should be selected by users before running *Admixture*.

7. Outlier detection

PopMLvis provides multiple outlier detection techniques to flag subjects that could be excluded from downstream analysis:

7.1. ***Statistical metrics:*** Using principal components (PCs), deviation from the mean is used to detect outliers ($\mu \pm 3\sigma$, $\mu \pm 2\sigma$, etc.). Users can define the list of PCs and the standard deviation threshold that can be used for outlier detection ($\mu \pm 3\sigma$ on PC1 *and* PC2; $\mu \pm 3\sigma$ on PC1 *or* PC2; etc.). **Figure A** shows Statistical metrics detection on a sample dataset.

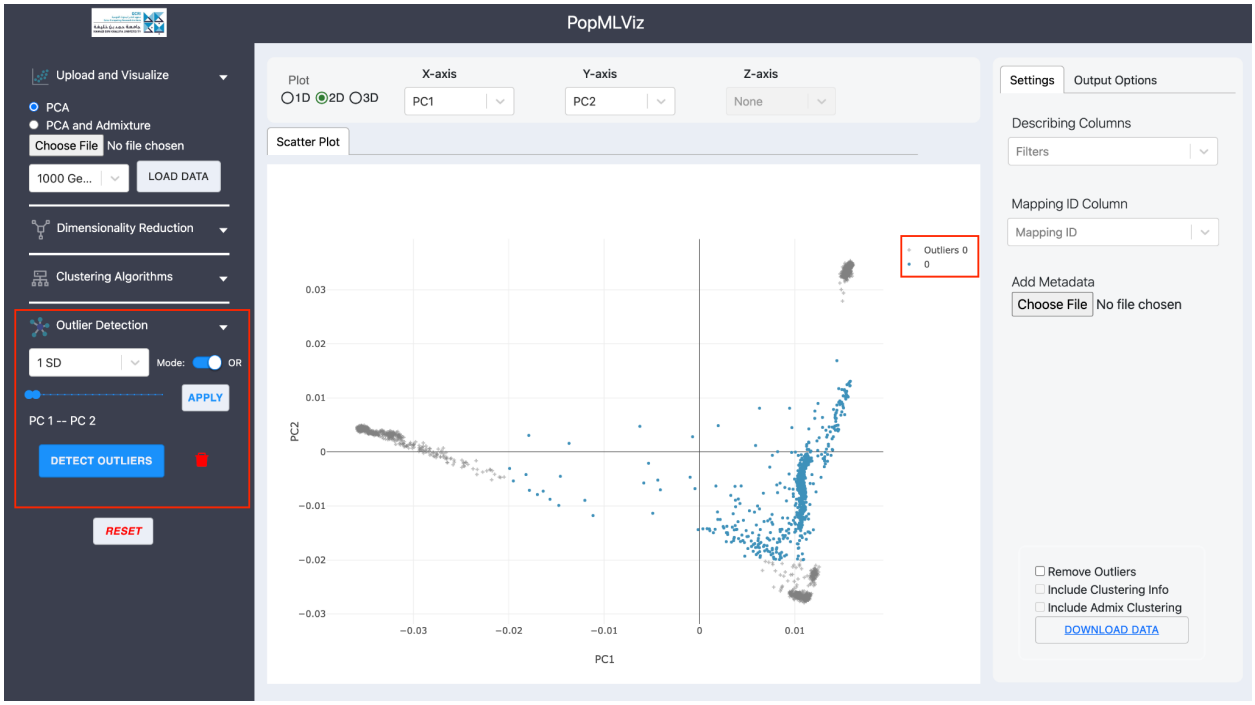


*Figure A: Outlier detection using statistical metrics ($\mu \pm \sigma$)*

7.2. ***Isolation Forest:*** This method identifies anomalies by isolating outliers in the data. It is based on a 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 (e.g., PC). **Figure B** shows Isolation Forest Outlier detection on a sample dataset.

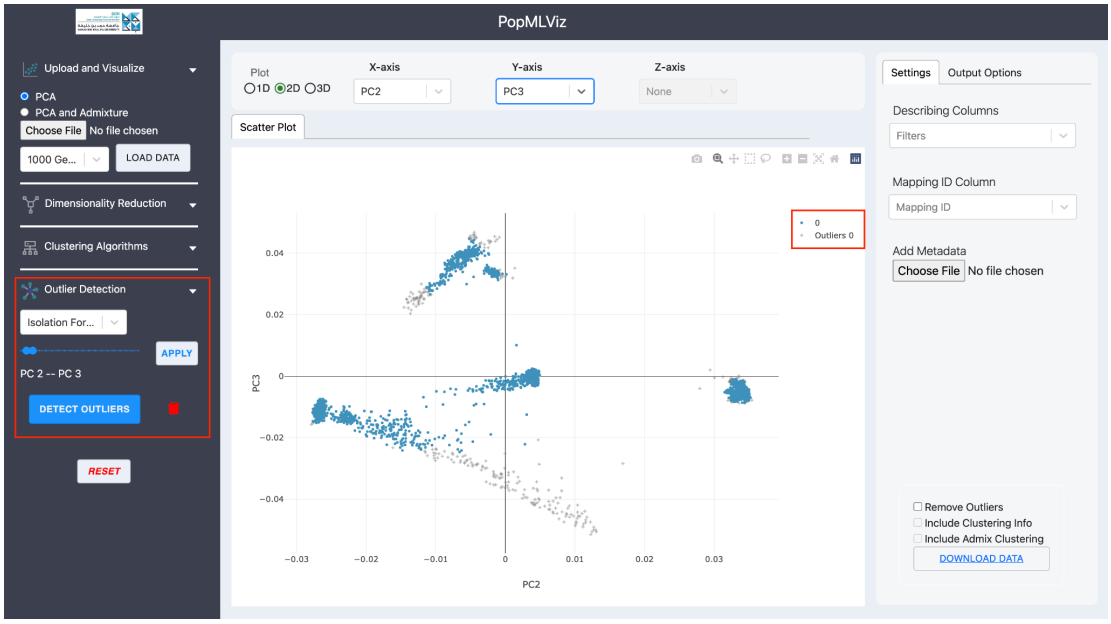


*Figure B: Isolation Forest outlier detection algorithm*

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). **Figure C** shows minimum covariance determinant detection on a sample dataset.


*Figure C: Minimum Covariance Determinant outlier detection algorithm*

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 for 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. **Figure D** shows Local Outlier Factor detection on a sample dataset.

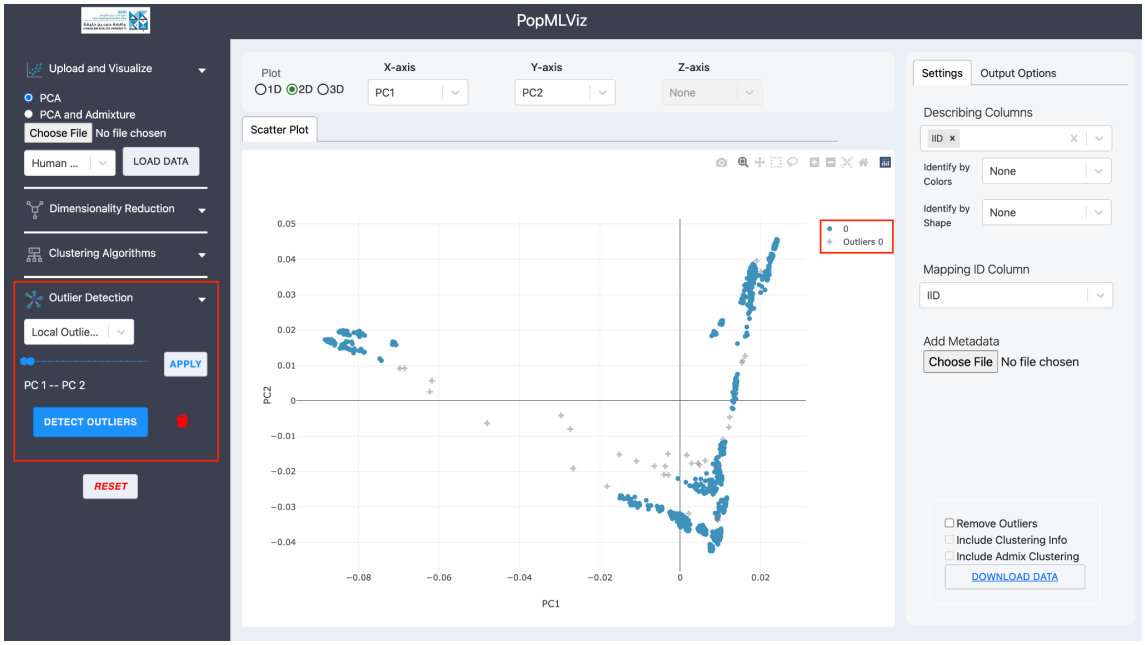


*Figure D: Local Outlier Factor outlier detection algorithm*

7.5. ***OneClassSVM:*** It is a variation of the SVM classification algorithm. 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. **Figure E** shows OneClassSVM detection on a sample dataset.

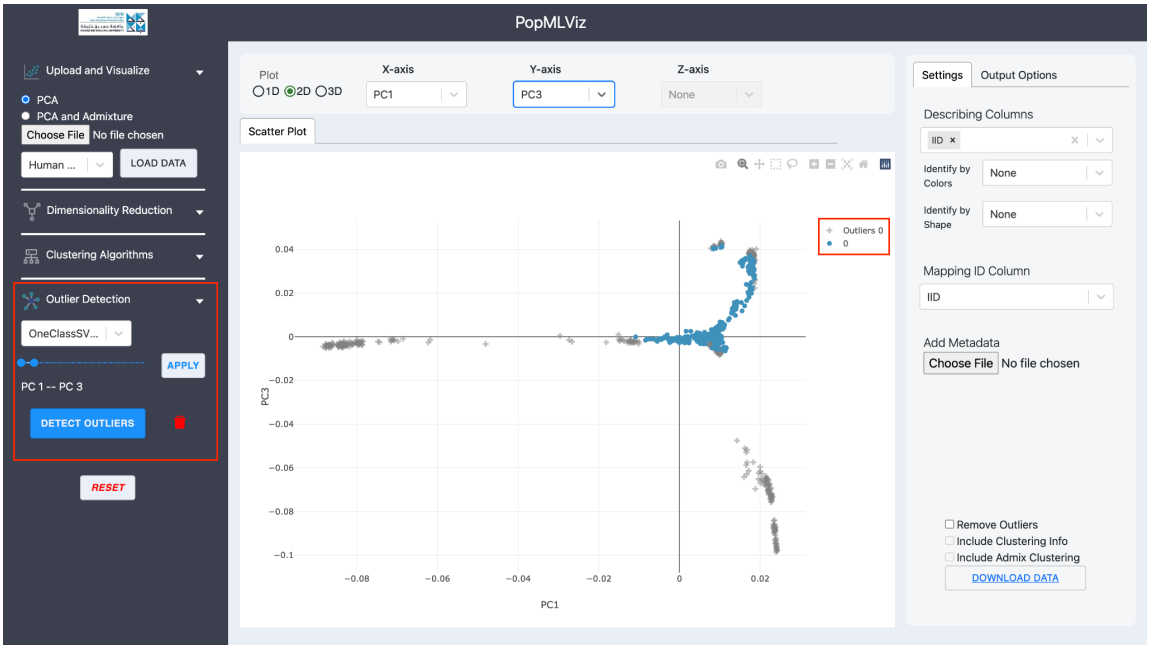


*Figure E: OneClassSVM outlier detection algorithm*

# 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.

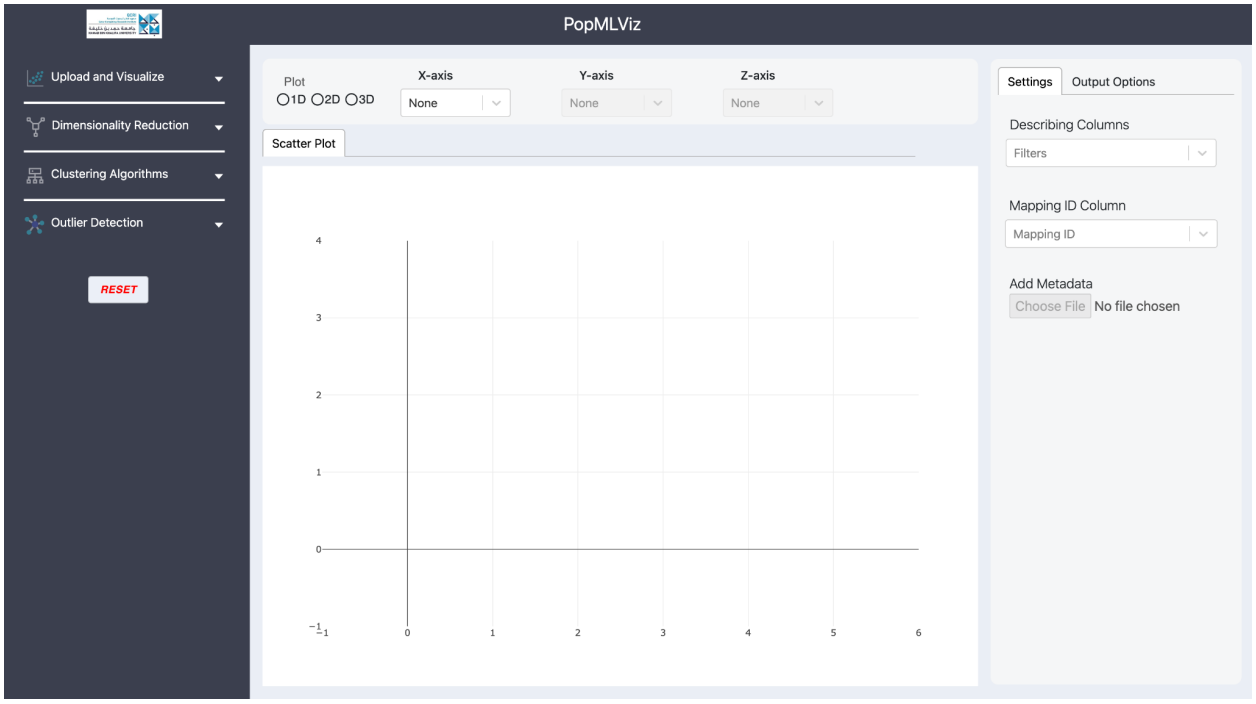


*Figure 1: The main window of PopMLvis*

As can be seen in Figure 1, the primary PopMLvis dashboard can be categorized into five panels:

a. *<u>Upload and Visualize:</u>* This panel provides the user freedom to choose the dataset type they want, between the PCA data and the Admixture results.
We also provide two sample datasets (1000 Genomes Project (1KG) and Human *Human Genome Diversity Project* (*HGDP*)) that can be used as well. There is a separate visualization for PCA and Admixture components.

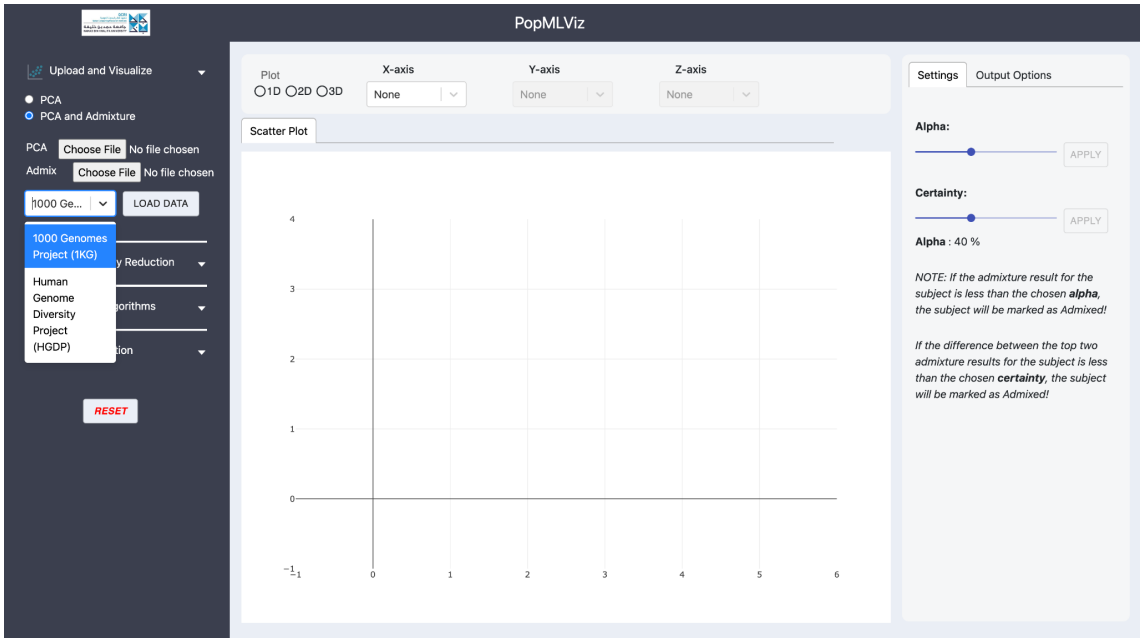


*Figure 2: Input panel of PopMLvis*

- PCA and Admixture - It is the combination of the PCA data and the Admixture results. Admixture will try to cluster the data based on their ancestry, and the output will be a set of probabilities $p_1, p_2, .. p_n$ where $n$ is the number of clusters.
    - We say subject $s_i$ belongs to cluster $k$, if
        - $p_k = max\ (p_1, p_2, \ldots p_k, \ldots, p_n)$ and $p_k >$ ***alpha***; ***or***
        - $p_k = max\ (p_1, p_2, \ldots p_k, \ldots, p_n)$, and $p_k - p_j >$ ***certainty***, where $p_j = max\ (p_1, p_2, \ldots p_{k-1}, p_{k+1}, \ldots, p_n)$ i.e the second largest probability
    - The visualization result is a bar plot and a scatter plot.
      In the bar plot, each bar of height 100% corresponds to one subject, where the stacked colors describe the genetic component proportions of the subject. The greyed out subjects are admixed. Figure 3(b) and 4(b)
      In the scatter plot, each subject is colored by the dominant cluster assigned by admixture, based on alpha/certainty. The greyed out subjects are admixed. Figure 3(a) and 4(a)

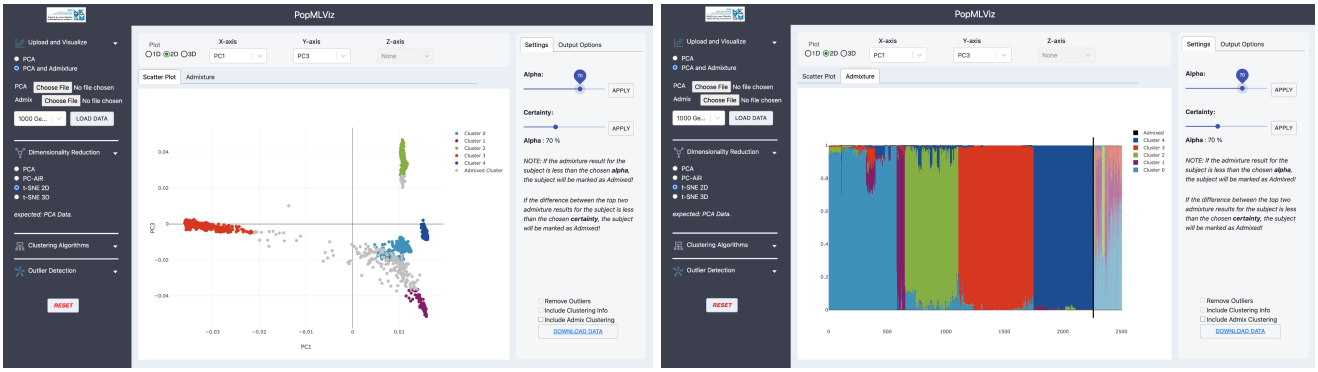


*Figure 3: (a) The effect of alpha in PCA scatter plot*
*Figure 3: (b) The effect of alpha in Admixture bar plot*

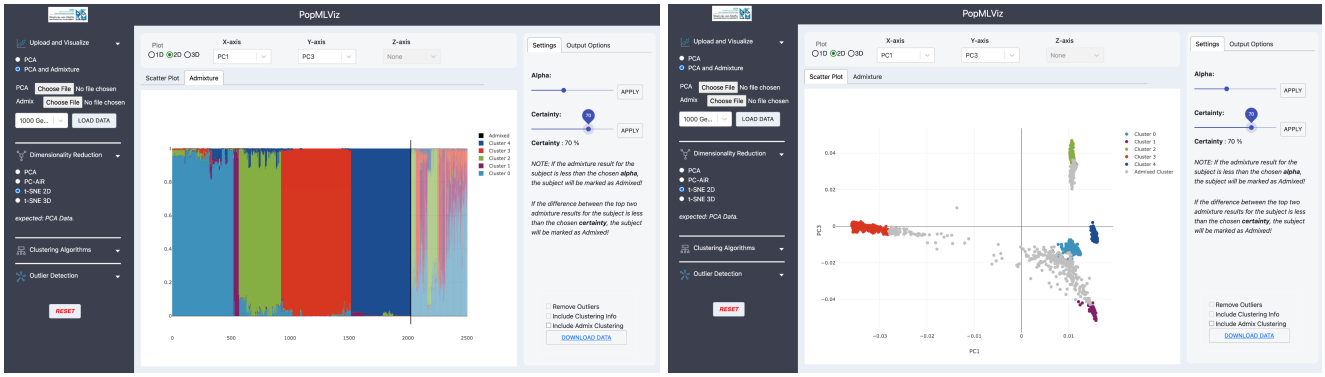


*Figure 4: (a) The effect of certainty in PCA scatter plot*
*Figure 4: (b) The effect of certainty in Admixture bar plot*

b. *Dimensionality Reduction:* If the input data consists of a large number of features, PopMLvis is compatible with performing Dimensionality Reduction algorithms.

As shown in Figure 5, PopMLvis supports 4 dimensionality reduction algorithms to make it possible for the user to analyze high dimensional data more efficiently.

All the options have an expected data type input.

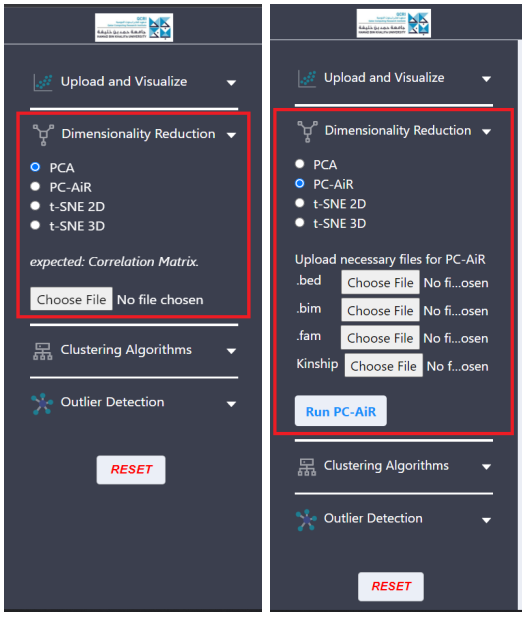


*Figure 5: Two different examples of Dimensionality Reduction algorithms*
*(a) PCA (expected input: Correlation Matrix/Genetic Relationship Matrix)*
*(b) PC-Air (expected input: PLINK files, optional: Kinship)*

1. PCA:
    a. **Expected input:** Correlation Matrix (N*N)
    b. **Output:** Low Dimensional Data (N*k, where k is defined by the user)
2. PC-AiR:
    a. **Expected input:**
        i. .bed (PLINK binary biallelic genotype table)
        ii. .bim (PLINK extended MAP file)
        iii. .fam (PLINK sample information file)
        iv. Kinship**(optional)** : A symmetric matrix of pairwise kinship coefficients for every pair of individuals in the sample.
        *If the kinship matrix is not provided, the result will be a usual PCA.*
    b. **Output:** Low Dimensional Data (N*k, where k is defined by the user)

3. T-SNE 2D:
    a. **Expected input:** PCA data  or Correlation Matrix/GRM
    b. **Output:** 2D data, see Figure 6.


*Figure 6: 2D dimensionality reduction of 1000 Genomes Project*

4. T-SNE 3D:
    a. **Expected input:** PCA data or Correlation Matrix/GRM
    b. **Output:** 3D data, see Figure 7.

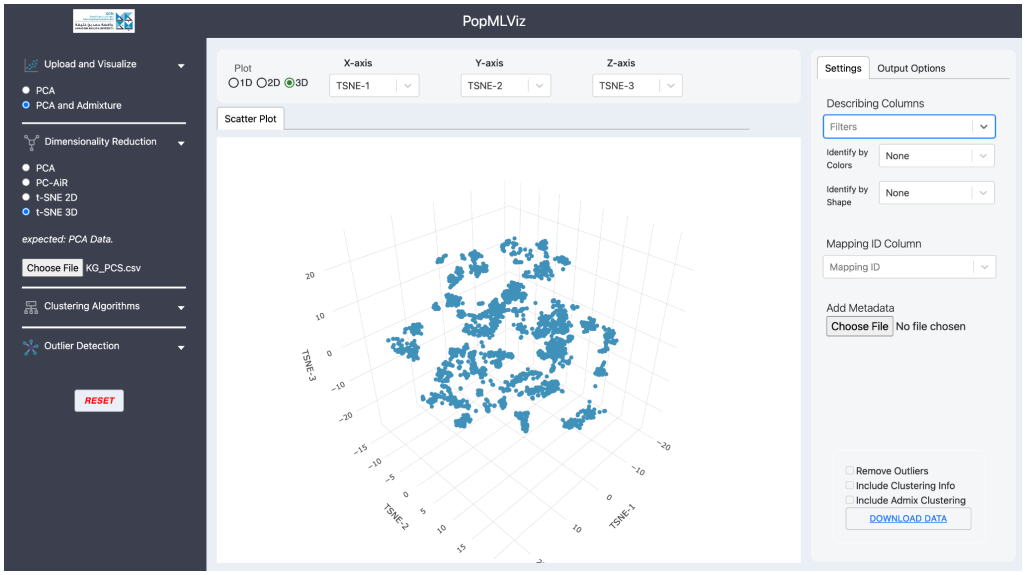


*Figure 7: 3D dimensionality reduction of 1000 Genomes Project*

b. *Visualization panel:* This panel provides the user with 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 8).

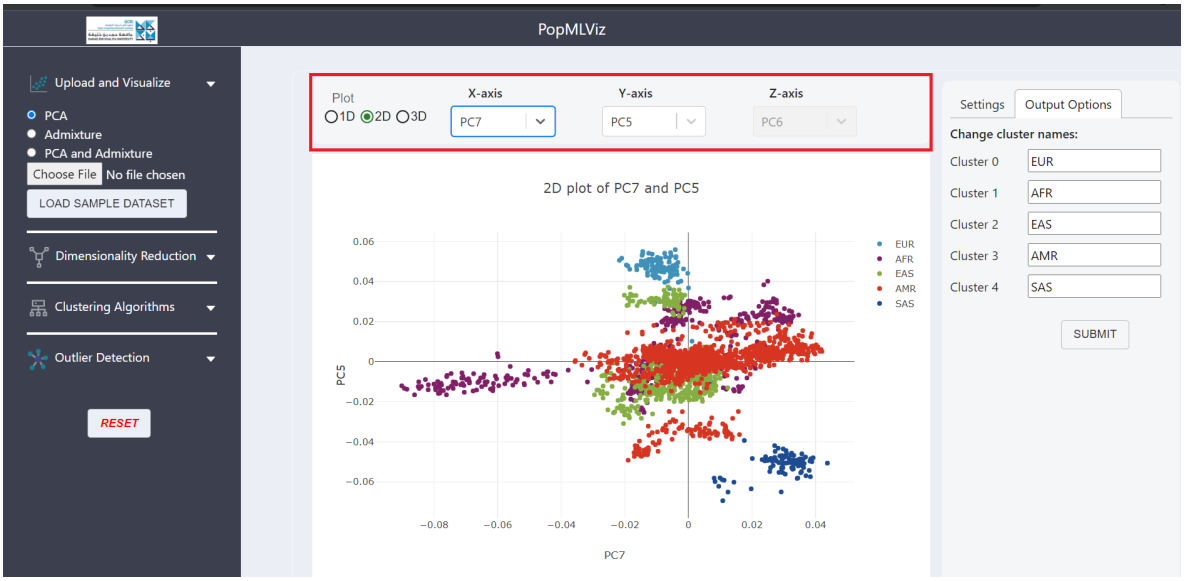


*Figure 8: Viewing the principal components*

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


*Figure 9: Clustering algorithms*

d. *Outlier detection panel:* The user can specify which principal component they want to remove outliers from, and it is up to the user to choose more than one principal component (see Figure 10). Also, the user can select if they want to do "AND" or "OR" operations when there is more than one principal component. Moreover, the user has to decide the deviation from the mean to be flagged as an outlier (e.g., 1SD, 2SD, etc.).

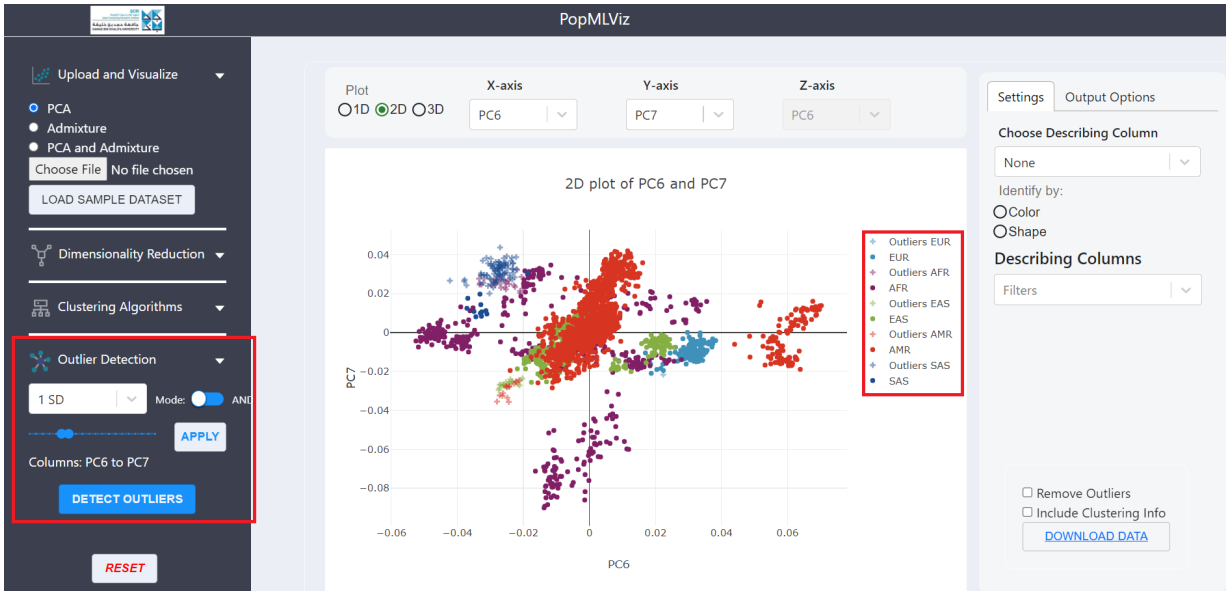


*Figure 10: Outlier detection panel*

## 9. Exporting outputs

After performing the required operations, the user can export the output in a csv file. As you can see in Figure (11), 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 11: (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 11: (b) Including Cluster Information*

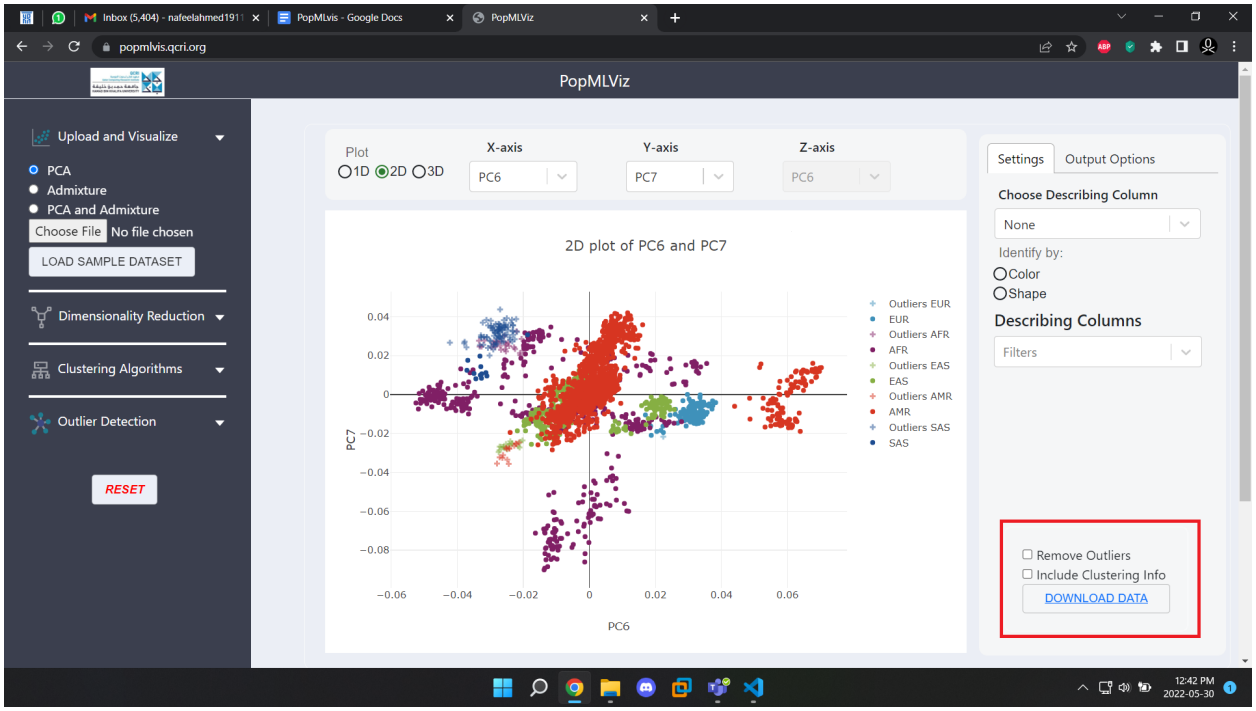


*Figure 11: (c) Download Data panel*

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

In case the user uses confidential data, PopMLvis provides an offline version, where the user can install it on their machine. PopMLvis is supported on Windows, macOS, and Linux operating systems.
The GitHub repository provides an installation script that can be used to install and run the software locally.

**Getting Started**
These instructions will cover usage information for the docker container.

**Prerequisites**
In order to run this container you'll need docker installed.
- Windows https://docs.docker.com/windows/started/
- OS X https://docs.docker.com/mac/started/
- Linux https://docs.docker.com/linux/started/

If cloning the repository, install .
- git-lfs. https://github.com/git-lfs/git-lfs/blob/main/README.md

**Usage**
Download the source code by either:
- Cloning the repository from command line
  - $ git clone https://github.com/qcri/QCAI-PopMLVis.git
- Or downloading ZIP, see Figure 12.
  NOTE: this might take a while because it will download a few big genotype files

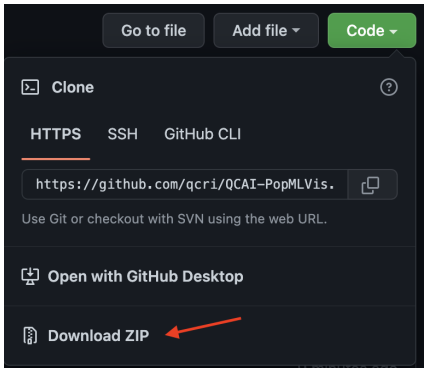


*Figure 11*

There are two configuration files, *backend/gunicorn_local.conf*, and *frontend/envfile*. You can choose to modify them if you want to change configs such as the exposed ports, the location of the log outputs, or multiprocessing power.

After that, in your command line, navigate to the folder you downloaded/cloned and run:

```
$ docker-compose up
```

Docker will start fetching the required images. *NOTE: this might take a while*. After the initialization is done, open your web-browser and load http://localhost:3000

## 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 inquiry at popmlvissupport@QCRI.org

## 13. Future work