Explore and Play with SNPs Data for Fun

CUI Yiran
Student ID. 20789781
ycuiat@connect.ust.hk
Room 3588
Department of Civil and Environmental Engineering, HKUST

Abstract

The article explores the application of PCA, kernel-PCA, MDS, and random projection with both PCA and MDS by SNPs data. The SNPs data is from Quanhua MU and Yoonhee Nam. The dataset consists of 488,919 SNPs (Single Nucleid Polymorphism) from 1064 individuals from all over the world, and each element is of three choices, 0 (for 'AA'), 1 (for 'AC'), 2 (for 'CC'). PCA and MDS show good performances on the dataset, and by comparing results from random projection and normal PCA, the blessing of high dimensionality of data is obvious and the computation time reduces significantly. Besides, Results show the SNPs inherit a linear relationship with people's mitigation history.

1 Introduction

Data reduction and visualizations are popular topics with current AI technology developments. Principle component analysis (PCA), multidimensional scaling (MDS) and related techniques, like kernel-PCA and random projections, are well-known tools in that field. Therefore, this article shows the results of applying both PCA, MDS, kernel-PCA and random projection techniques with Single Nucleid Polymorphism (SNPs) data to explore their performance.

The SNPs data is originally from [1], but their dataset contains empty values (indicated by 9). Therefore, a cleaned dataset from Quanhua MU and Yoonhee Nam is selected. Data is available from `https://drive.google.com/file/d/1a9I8_akfCMHBRrPMdnWkjyL9fKcQbJJq/view` and `https://drive.google.com/file/d/11Q-8B57WDQnrIV92b-h_WLqDGviiYsm2/view`. The dataset contains 488,919 rows and 1064 columns responding to SNPs data and individuals, respectively. Therefore, The dataset is $\mathbb{R}^{488,919\times 1064}$ dimensions. The dataset is centered before processing and in the following article, the dataset is always assumed centered without loss of any generality.

A direct PCA is performed on the dataset and all eigenvalues and eigenvectors are calculated. An MDS is also performed on the dataset but only the first two eigenvalues and eigenvectors are calculated. Their results show good indicators of human mitigation history but this implies the linearity in the data. However, due to curiosity, a kernel PCA is performed with a radial basis function kernel and various sigma (variance). The default sigma is set to be the inverse of the number of features $488,919^{-1}$, then sigma is changed to values that $10^{-3}, 10^{-2}, \ldots, 10^{6}$ times the defaults values to evaluate the performance. After that, random projections are performed and accompanied by both PCA and MDS. The dimension of the random projection matrix (P) is set to $100, 500, 1000, 2000, 4000, 6000, 8000, 10000$, and 12000. From Johnson-Lindenstrauss lemma, the required P is $11,228$ with requiring $\epsilon = 0.1$, so values below the required P are tested and the consumed CPU time is recorded. Results show a significant reduction in CPU times.

Submitted to CSIC-5011 (2022) Mini Project. Do not distribute.

Lastly, all codes, data and preliminary computation results are published and can be found here: https://drive.google.com/drive/folders/1-qGwQTkoglsmlxm0ZLAP8zYI2IBNIOH5?usp=share_link.

2 Theoretical background

Both PCA, MDS, kernel-PCA, and random projections are applied in this article. In this section, their theoretical backgrounds are briefly introduced. Note that the dataset is assumed to be centered without loss of generality, and more details of them can be found in [2].

2.1 Principle component analysis (PCA)

The principal component analysis is a popular method for dimensionality reduction with high dimensional Euclidean data. It looks the best Affine Approximation of data and the best approximation in terms of Euclidean distance is given by:

$$\min_{\beta,\mu,U} I := \sum_{i=1}^{n} \|x_i - (\mu + U\beta_i)\|^2$$

where $U \in \mathbb{R}^{p\times k}$, $U^T U = I_k$ and $\sum_{i=1}^{n} \beta_i = 0$. By linear algebra skills, the problem is transferred to eigenvector decomposition of the empirical matrix $\hat{\Sigma} = \hat{U}\hat{\Lambda}\hat{U}^T$ and evaluate the projection of centered data points on top k eigenvectors as the principle components. A clear example can be found from statistical views is that $\alpha\hat{\Sigma}\alpha^T$ will give the variances of data set $\alpha^T x_1, \alpha^T x_2, \ldots, \alpha^T x_n$.

2.2 Multidimensional scaling (MDS)

Multidimensional Scaling aims to recover Euclidean coordinates given pairwise distance metrics or dissimilarities, and it is equivalent to PCA when pairwise distances are Euclidean. The core of MDS is related to positive definite functions which is the foundation of the kernel method in statistics. It focuses on the question:

$$\min_{Y\in\mathbb{R}^k} \sum_{i,j} (\|Y_i - Y_j\|^2 - d_{ij}^2)^2$$

The key observation is that the two-side centering transform of squared distance matrix D, where $D = (d_{ij}) = \|x_i - x_j\|^2$, gives the inner product matrix of the centered data matrix $(\hat{K})$. In other words, it converts a squared distance matrix D to an inner product matrix by $\hat{K} = (XH)^T(XH) = -\frac{1}{2}HDH^T$, where H is the Householder centering matrix. An eigenvector decomposition is conducted on $\hat{K}$ to find the Euclidean coordinates centered at the origin.

2.3 kernel PCA

The kernel PCA uses the idea of kernel function and reproducing kernel Hilbert spaces (RKHS). A kernel is a positive definite symmetric function whose spanned by $k_x(\cdot) = k(x, \cdot)$ for $x \in \mathbb{X}$ made up of a Hilbert space, where we can associate an inner product induced from $\langle k_x, k_y\rangle = k(x, y)$, and a reproducing kernel Hilbert space is a Hilbert space on $\mathbb{X}$ with bounded evaluation functional. Assuming data is in a sample space and there is a mapping function $\Phi(x)$ that can map the data from sample space to a higher dimension space (feature space), and then conduct PCA in the feature space and results in PCA score in a new space (PCA space). Since the mapping function, $\Phi(x)$ is unknown, so we use the kernel function to compute values in PCA space from values in sample space directly. This is benefiting from the property of kernel and here we assume the feature space is actually an RKHS. More details can be found by Mercer's Theorem and Riesz representation theorem. To perform the kernel-PCA, we simply first choose a kernel and calculate the kernel matrix:

$$K = (k_{ij}) = k(x_i, x_j) = \langle x_i, x_j\rangle$$

and then perform PCA/MDS on the kernel matrix after centering it.

2.4 Random projection

The random projection is a case of the blessing of dimensionality. The theoretical basis of this method was given as a lemma by Johnson and Lindenstrauss, which state that for any $0 < \epsilon < 1$ and integer n, let k be a positive integer such that

$$k \geq (4 + 2\alpha)(\epsilon^2/2 - \epsilon^3/3)^{-1} \ln n, \alpha > 0$$

then for any set V of n points in $\mathbb{R}^p$, there is a map $f : \mathbb{R}^p \to \mathbb{R}^k$ such that for all $u, v \in V$

$$(1 - \epsilon)\|u - v\|^2 \leq \|f(u) - f(v)\|^2 \leq (1 + \epsilon)\|u - v\|^2$$

Based on this lemma, we can apply random projections to achieve a uniformly almost isometric embedding or a Lipschitz mapping from metric space $\mathbb{X}$ to $\mathbb{Y}$, and overcome the dataset adaptive property of classical MDS to keep the mapping be universal.

3 Results and discussions

This section shows partly calculated results for illustration purposes. Actually more computations are done and their results can be found in the published folder whose linage has been shown in the introduction part.

3.1 Eigenvalues of PCA

Eigenvalues are calculated by Eigen-decomposition of the centered empirical variance matrix and they imply the volume of information contained in the corresponding eigenvector directions. Figure 1 shows the first 20 singular values (eigenvalues) calculated from the SNPs dataset. It is noticeable that the first few eigenvalues are relatively larger than the following. The first three eigenvalues account for 0.87%, 0.74% and 0.54% of the sum of all eigenvalues. Even if we should consider more eigenvalues/eigenvectors to have a precise analysis of data, we only take the first two for the purpose of visualization and to have a brief insight into data.

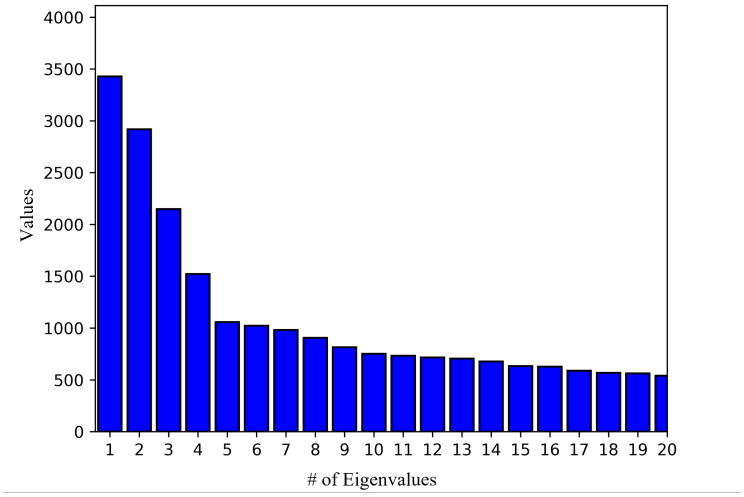

Figure 1: Sorted eigenvalues of the empirical variance matrix

3.2 PCA and MDS based on full dataset

The PCA and MDS are calculated from the whole dataset and their results are shown in Figure 2. The left figure shows the result of PCA, the middle one shows the results of MDS and the right one

shows the PCA followed by MDS. It is clear both PCA and MDS have good performance even only considering the first two components. Nationality is distinguished clearly from scores of the first two principal directions, and this implies a good linearship between SNPs and people's nationality.

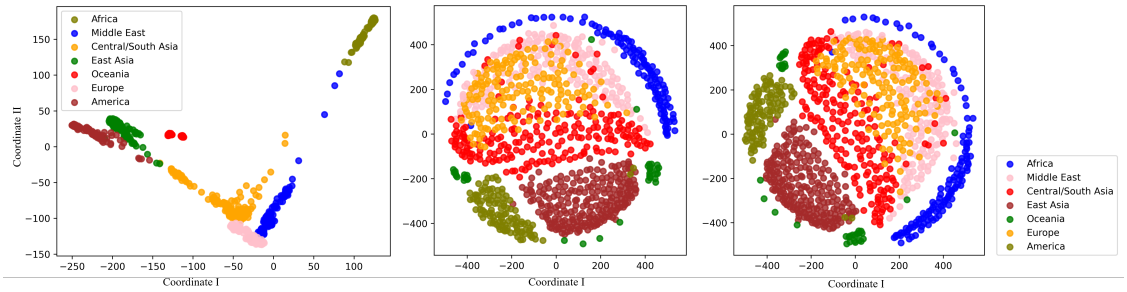

Figure 2: Direct PCA (left), MDS (middle), and MDS followed by PCA (right) on data

It is noticeable that implementing PCA on MDS results in only a coordinate rotation. This can be seen from their algorithm that they are both based on the eigen-decomposition of the empirical variance matrix.

3.3 Kernel-PCA

The kernel-PCA mainly focuses on non-linear relationships, but kernel-PCA is still implemented due to curiosity. The radial basis function kernel is selected with various variances (σ). The σ is chosen from $10^{-3}, 10^{-2}, \ldots, 10^{6}$ times the inverse of feature size (e.g. $488,919^{-1}$), and results shown in Figure 3.

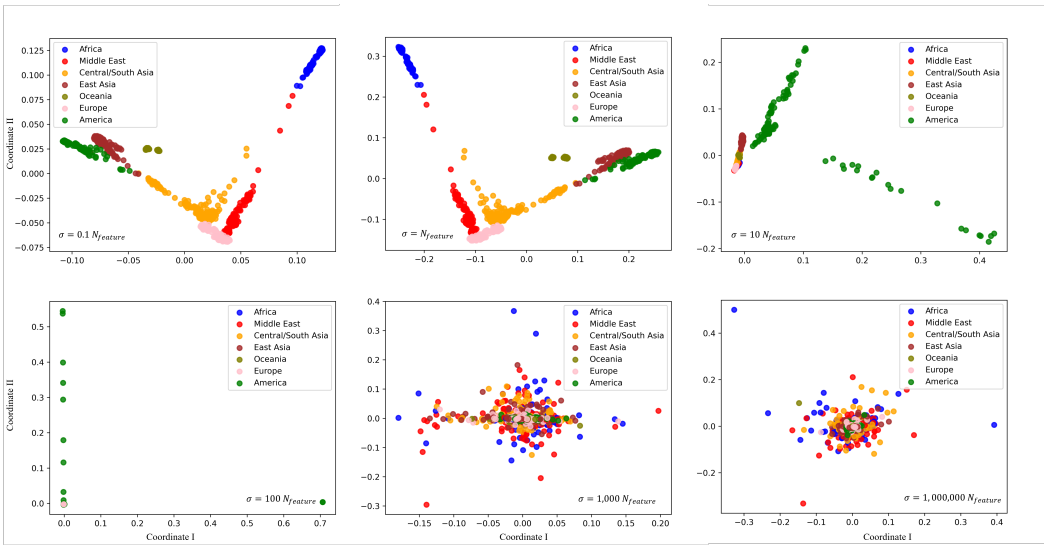

Figure 3: Kernel-PCA with various sigma (σ), the upper-left with $\sigma = 0.1N_F$, the upper-middle with $\sigma = N_F$, the upper-right with $\sigma = 10N_F$, the lower-left with $\sigma = 10^2 N_F$, the lower-middle with $\sigma = 10^3 N_F$, the lower-right with $\sigma = 10^6 N_F$, where $N_F = 488,919^{-1}$

It is clear that the performance of kernel-PCA becomes worse with σ increase. In the case of σ lower than N_F, results are good and similar to direct PCA, but latter data points start to concentrate on one point. This can be interpolated by that as variance increases more data is regarded as similar (the kernel function will result in similar results by fitting in different arguments). In cases of small

variances, kernel functions are sensitive to small variances of arguments and data become far away from each other in the corresponding RKHS.

3.4 Random projections combined with PCA and MDS

A Gaussian random projection matrix is used here with different dimensions (P). The cases of results of random projections with PCA and $P = 10^2, 5\times10^2, 10^3, 2\times10^3, 6\times10^3$ and 1.2×10^4 are shown in Figure 4. Notice that the required P is $11,228$ with $\epsilon = 0.1$ from Johnson and Lindenstrauss lemma.

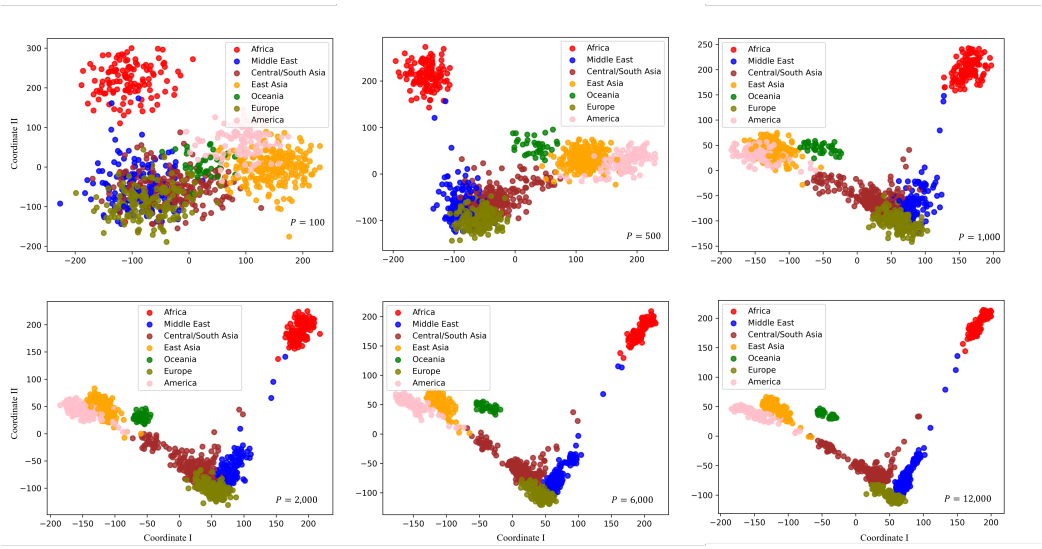

Figure 4: Random projections (with different dimension P) with PCA, the upper left $P = 10^2$, the upper middle $P = 5 \times 10^2$, the upper left $P = 10^3$, the lower left $P = 2 \times 10^3$, the lower middle $P = 6 \times 10^3$ and the lower left $P = 1.2 \times 10^4$

Besides, random projections accompanying MDS and with the same set of dimensions of random projection matrix are done and results are shown in Figure 5.

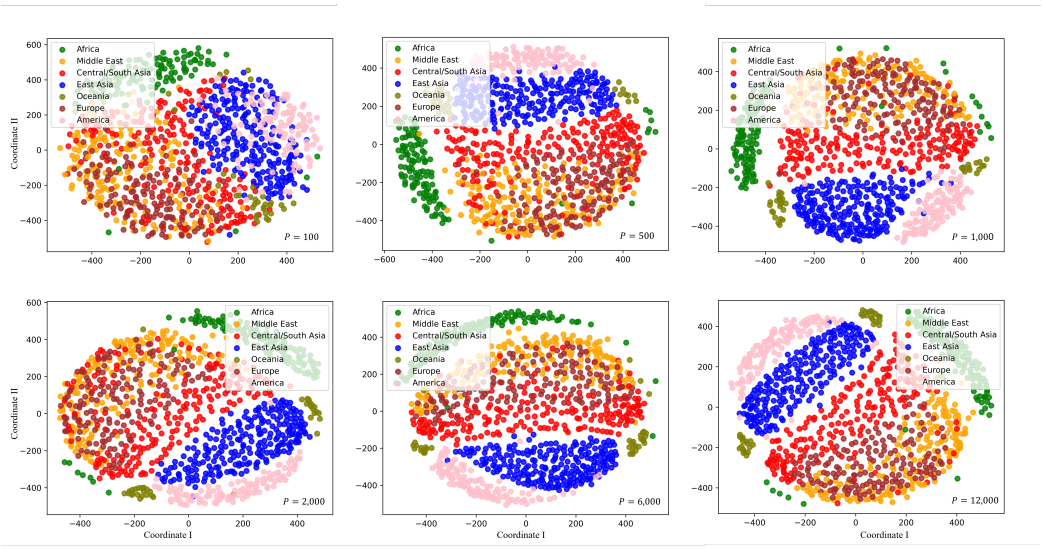

Figure 5: Random projections (with different dimension P) with MDS

It is clear that random projections work well and it provides similar results as that of direct PCA/MDS. With P decreases, the performance becomes worse but can still generate a good pattern. It is benefited from that random projections provide a universal control on the almost isometric embedding where performance is governed by the Johnson and Lindenstrauss lemma as stated in section 2.4.

Table 1 shows the CPU time consumed by different dimensions of the Gaussian random projection matrix with PCA and the classical PCA without implementing random projections. Even results show that the CPU time is not necessary to reduce with decreased dimensions of random projection matrix, algorithms are significantly faster than the case without implementing random projections.

Table 1: CPU time consumed by PCA with different dimensions of Gaussian random projection matrix

Algorithms	CPU times (s)
Full PCA	62.63462
Random Projection (P=1,000)	0.00917
Random Projection (P=2,000)	0.03931
Random Projection (P=4,000)	0.15000
Random Projection (P=6,000)	0.26363
Random Projection (P=8,000)	0.13827
Random Projection (P=10,000)	0.38228
Random Projection (P=12,000)	0.29808

4 Conclusions

This article shows the results of implementing PCA, MDS, kernel-PCA, and random projections on SNPs data and gives a brief interpolation. From the results of both PCA and MDS it can be shown that the SNPs inherit a linear relationship with people‘s mitigation history. The kernel-PCA with the radial basis function kernel performs worse with the variance of the kernel function increase because data become more similar in the view of the chosen kernel function. Random projections show a significant power in saving computation costs and it do provide a universal embedding control as illustrated by the Johnson and Lindenstrauss lemma.

References

[1] Jun Z. Li, Devin M. Absher, Hua Tang, Audrey M. Southwick, Amanda M. Casto, Sohini Ramachandran, Howard M. Cann, Gregory S. Barsh, Marcus Feldman, Luigi L. Cavalli-Sforza & DRichard M. Myers (2008) Worldwide Human Relationships Inferred from Genome-Wide Patterns of Variation. *Science* **319**(5866):1100-1104.

[2] Yao Yuan (2023). *Geometric and Topological Data Reduction: A Mathematical Introduction to Data Science*. URL:`https://yao-lab.github.io/bookdatasci/`