Principal Component Analysis

Johanni Brea

Introduction to Machine Learning

Dimensionality Reduction

Sometimes we would like to plot some high-dimensional data in two dimensions, e.g. to see if there are clusters in the data.

How should we represent the high-dimensional data in two dimensions?

Idea 1: Project the data to one of the planes spanned by the axes, i.e. plot coordinate *i* versus coordinate *j* of the data points, e.g. in pair plots.

Idea 2: Project the data onto the plane spanned by the directions along the largest variance of the data.

Table of Contents

First Principal Component

Given x_{ij} for $i = 1, \ldots, n$, $j = 1, \ldots, p$
column-wise zero mean $\bar{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij} = 0$.

First Principal Component

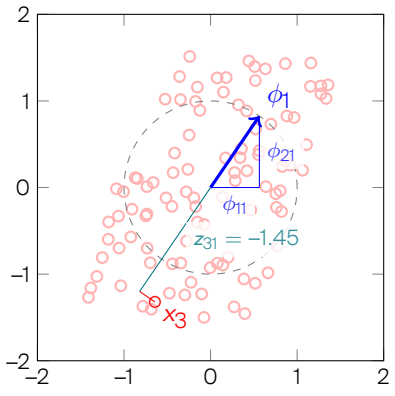
Find the direction onto which the projection of the data has the highest variance.

(projection) **scores** $z_{i1} = \langle \phi_1, x_i \rangle = \sum_{j=1}^{p} x_{ij} \phi_{j1}$

Find **loadings** $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ that

$$\underset{\phi_{11},\ldots,\phi_{p1}}{\text{maximize}} \frac{1}{n} \sum_{i=1}^{n} z_{i1}^2$$

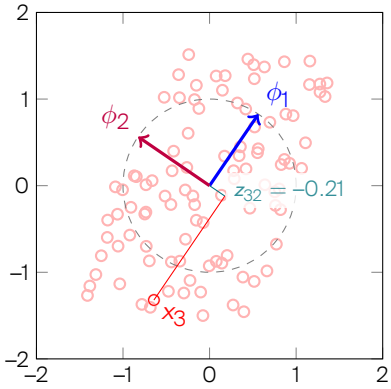
under the constraint $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.



Principal Component Analysis

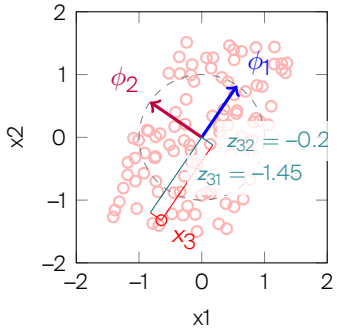
Second Principal Component
Find the direction onto which the projection of the data has the highest variance under the constraint that it is orthogonal to the first PC.

k-th Principal Component Find the direction onto which the projection of the data has the highest variance under the constraint that it is orthogonal to the first $k-1$ PCs.

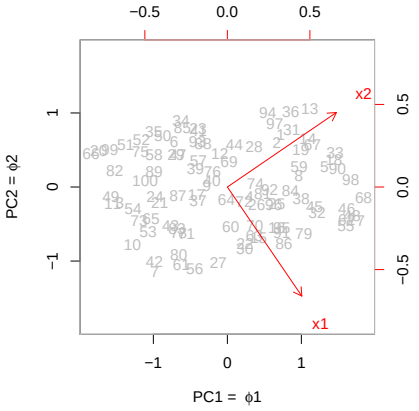


Biplots: Visualizing Scores and Loadings Simultaneously

Plot of raw data



Biplot



scores in gray: read bottom-left coordinates,
loadings in red: read top-right coordinates. (read the coordinates of the text label, e.g. x1, not the arrow tip).

First Principal Component Rewritten in Matrix Notation

$$z_1 = X\phi_1$$

data	X	$n \times p$ matrix	row i contains observation i
loadings	ϕ_1	$p \times 1$ column vector	first PC
scores	z_1	$n \times 1$ column vector	first PC scores of all observation

$$\text{Find } \phi_1 \text{ that } \underset{\|\phi_1\|=1}{\text{maximizes}} \frac{1}{n} z_1^T z_1 = \underset{\|\phi_1\|=1}{\text{maximizes}} \frac{1}{n} \phi_1^T X^T X \phi_1$$

The solution of this optimization problem is the eigenvector of the largest eigenvalue of the covariance matrix $X^T X$.

PCA in Matrix Notation and Relation to SVD

$$Z = X\Phi$$

data	X	$n \times p$ matrix	row i contains observation i
loadings	Φ	$p \times p$ matrix	column j contains PC j
scores	Z	$n \times p$ matrix	column j = scores of PC j for all observations

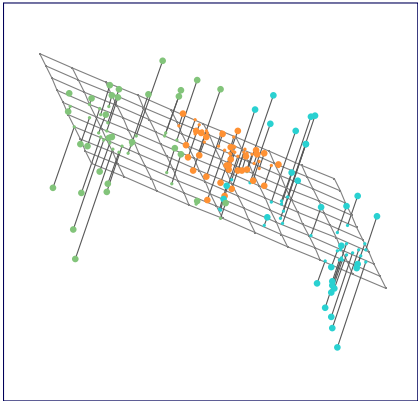
The columns of the loading matrix Φ are eigenvectors of $X^T X$, i.e. $X^T X \phi_i = \lambda_i \phi_i$ or $X^T X \Phi = \Phi \Lambda$ with $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_p)$.

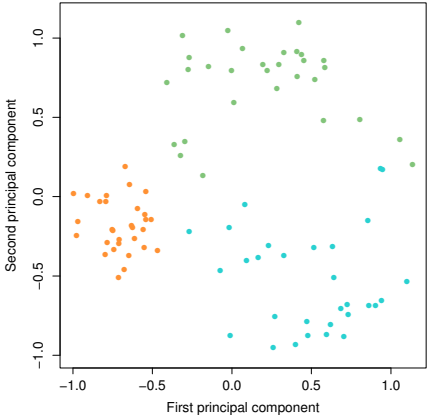
PCA is closely linked to **Singular Value Decomposition** $X = U \Sigma V^T$ where U and V are unitary matrices and Σ is a diagonal matrix. One can show that (see exercises)

$$Z = U\Sigma \quad \text{and} \quad \Phi = V$$

Table of Contents

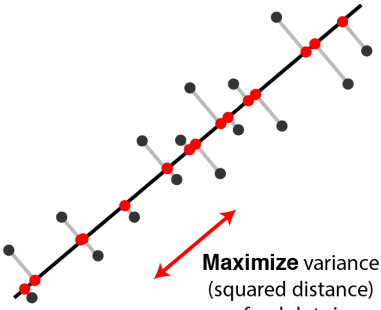
PCA Provides Linear Subspaces Closest to the Data



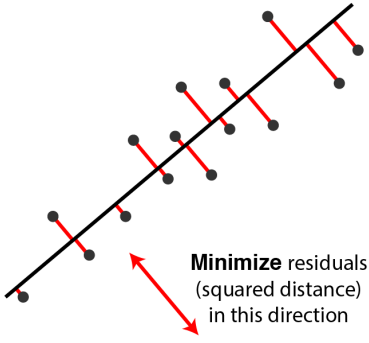


The data varies more within the plane than perpendicular to the plane.

Connection Between the Two Interpretations of PCA



Maximize variance (squared distance) of red dots in this direction



Minimize residuals (squared distance) in this direction

Reinterpretation of Loadings and Scores

Lossless transformation: change of basis

standard basis	$\mathbb{I}$	$p \times p$	columns are standard basis vectors
data	X	$n \times p$	rows are coordinates of points in standard basis
loadings	Φ	$p \times p$	columns are new basis vectors
scores	$Z = X\Phi$	$n \times p$	rows are coordinates of points in new basis
reconstruction	$X = Z\Phi^T$	$n \times p$	rows are coordinates of points in standard basis

Lossy transformation: projection to lower-dimensional space

loadings	Φ_L	$p \times L$	$L = 1, \ldots, p$ first columns of Φ are new basis vectors
scores	$Z_L = X\Phi_L$	$n \times L$	rows are coordinates of points in new basis
reconstruction	$X_L = Z_L\Phi_L^T$	$n \times p$	reconstructed coordinates in standard basis

Φ_L minimizes $\|X - X\Phi_L\Phi_L^T\|_2^2$.

Quiz

Correct or wrong?

- The scores $z_{11} \approx -1.1$ and $z_{12} \approx 0.1$.
- The data has the highest variance in direction $\phi_1 \approx (0.65, -0.55, 0.5)$.
- The two-dimensional linear subspace that is closest to the data is parallel to the X1-X2-plane..

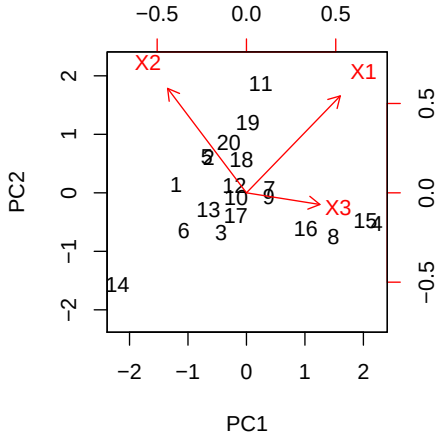
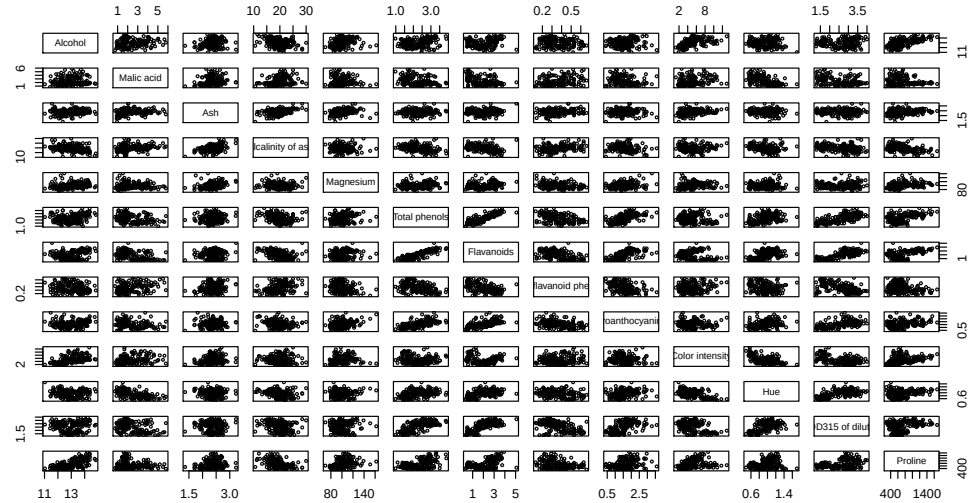


Table of Contents

Exploring the Wine Dataset

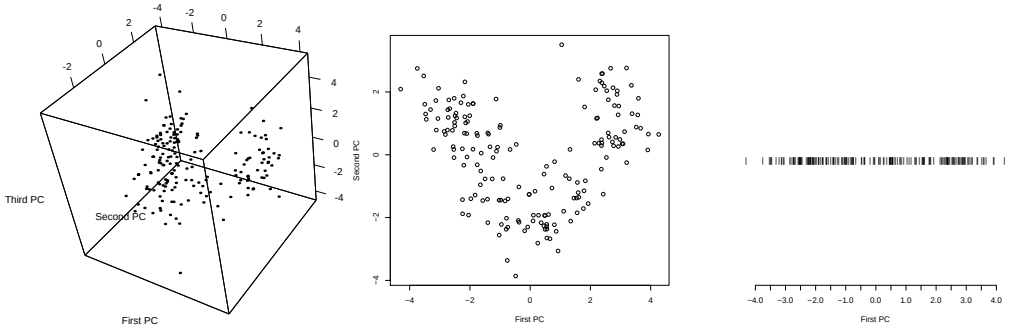
- 178 different wines from the same region in Italy were chemically analyzed.
- 13 different measurements per wine: 1) Alcohol 2) Malic acid 3) Ash 4) Alcalinity of ash 5) Magnesium 6) Total phenols 7) Flavanoids 8) Nonflavanoid phenols 9) Proanthocyanins 10) Color intensity 11) Hue 12) OD280/OD315 of diluted wines 13) Proline

Exploring the Wine Dataset



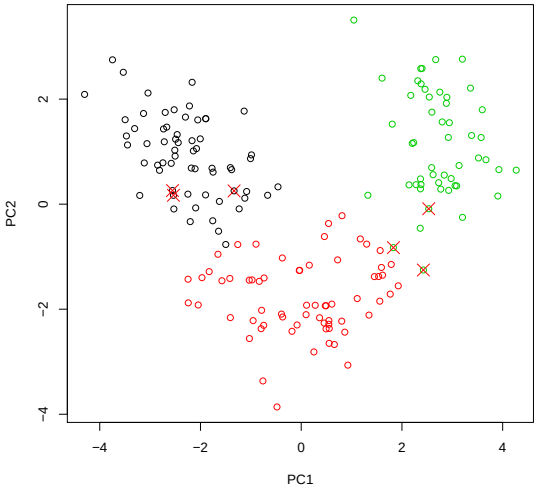
Data: $n = 178$, $p = 13$

Linear Subspaces in the Wine Data



The 3D plot is obtained by projecting the 13D data onto the linear 3D subspace that is closest to the data. From there onwards we project onto the 2D and 1D subspaces closest to the data to obtain the 2D and 1D plots.

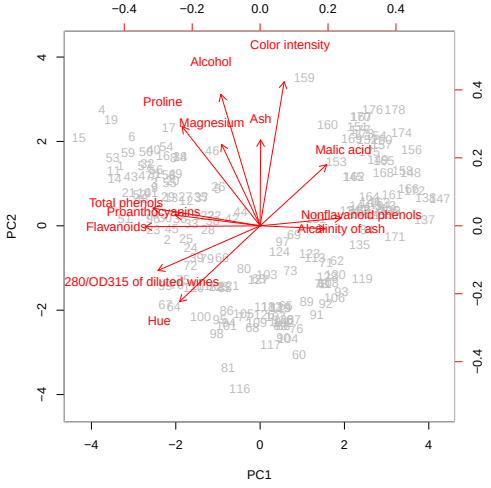
Clustering the Wine Dataset



K-means clustering

- The colors on in the image on the left indicate the classes found by 3-means clustering.
- The data actually came from wines of 3 different cultivars. The red crosses indicate data points that actually came from the red wine cultivar. All other observations were “correctly” clustered (without having seen a single class label in the training set!)

A Biplot of the Wine Data



PC1, PC2 scores of all observations in grey.

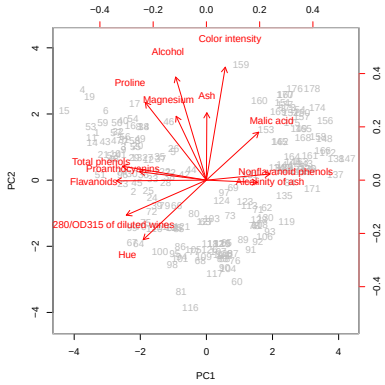
Loadings in red (axis and the top and right).

The first PC corresponds roughly to the total level of phenols (with subtypes Flavanoids, Non-Flavanoids and Proanthocyanins). The Alcalinity of the ash is strongly correlated.

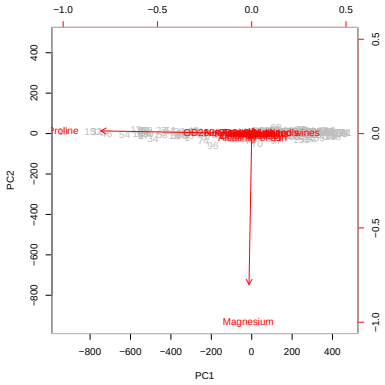
The second PC corresponds roughly to the amount of ash, with e.g. the color intensity strongly correlated.

Scaling Matters

Scaled



Unscaled



The direction of highest variance may depend on the choice of units (meters versus millimeters). It is common to preprocess each predictor to have mean 0 and variance 1.

Table of Contents

PCA Decorrelates

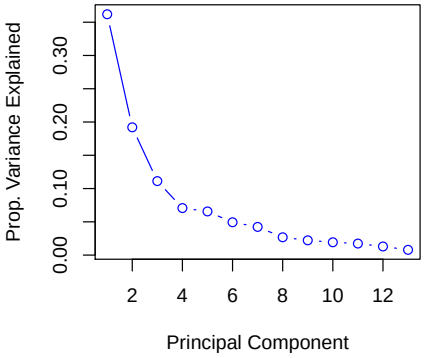
The covariance matrix of the scores is given by the diagonal matrix Λ with the eigenvalues $\lambda_1, \ldots, \lambda_p$ of the covariance matrix $X^T X$ on the diagonal (sorted from largest to smallest).

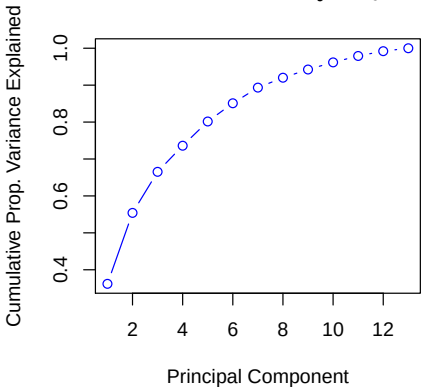
$$Z^T Z = \Phi^T X^T X \Phi = \Phi^T \Phi \Lambda = \Lambda$$

$$\text{This means } \sum_{i=1}^{n} z_{ik} z_{il} = \begin{cases} 0 & k \neq l \\ \lambda_k & k = l \end{cases}$$

How Many Principal Components Matter?

Proportion of Variance Explained (PVE) of PC m: $\text{PVE}_m = \frac{\lambda_m}{\sum_{j=1}^{p} \lambda_p} = \frac{\sum_{i=1}^{n} z_{im}^2}{\sum_{i=1}^{n} \sum_{j=1}^{p} x_{ij}^2}$

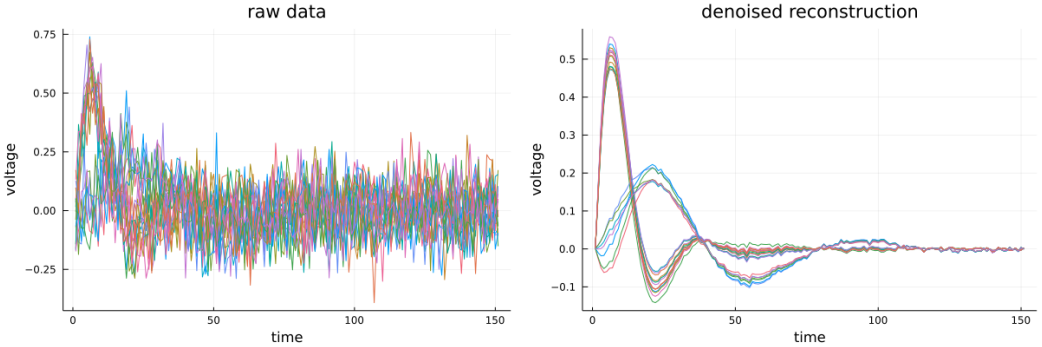




How Many Principal Components Matter?

- The Proportion of Variance Explained is decreasing, i.e. $\text{PVE}_m \geq \text{PVE}_{m+1}$.
- Sometimes the first k components reach a cumulative proportion of variance explained close to 1 and all the other components do not add much anymore.
- The question of how many principal components matter is not well defined.
- In practice, we tend to look at the first few components in order to find interesting patterns in the data.
- If the signal is along the first few principal components and the noise orthogonal to that, we can use this for compression and denoising.

Denoising



This is an artificial dataset with noise. For the reconstruction we assume that the variance along the first two principal components is the signal and the rest is noise, i.e. for the reconstruction we use $L = 2$ and $X_L = Z_L \Phi_L^T$.

Lossy Image Compression with PCA

A dataset with $n = 400$ grayscale images of faces with $p = 64 \times 64$ pixels.
This corresponds to $400 \times 64^2 = 1'638'400$ pixel values.

Idea: take each image as a 64 x 64 = 4096 dimensional vector, compute the first few principal components Φ_L and scores Z_L and store the data in this format.

If $L = 200$ is sufficient, we store $4096 \times 200 + 400 \times 200 = 899'200$ values. This corresponds to a compression factor of $\frac{1'638'400}{899'200}$. If needed, we can reconstruct the images $X_L = Z_L \Phi_L^T$.

In general the compression ratio is $\frac{n \times p}{(p + n) \times L}$.

Lossy Image Compression with PCA

Loadings

PC 1 PC 2 PC 3 PC 4 PC 5

PC 256 PC 257 PC 258 PC 259 PC 260

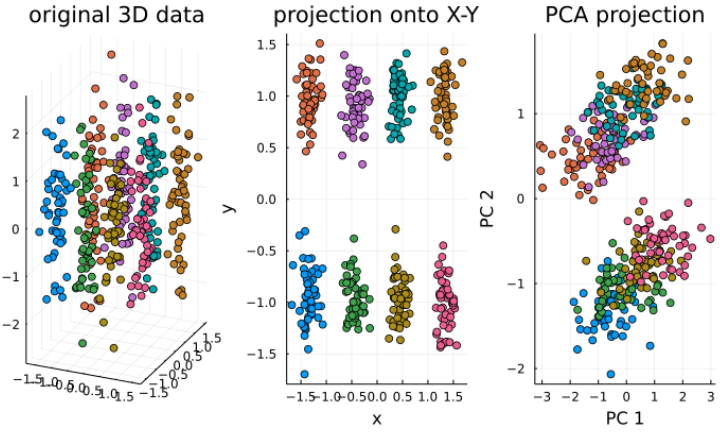
Original and Reconstruction with $L = 200$

orginal reconstruction

Table of Contents

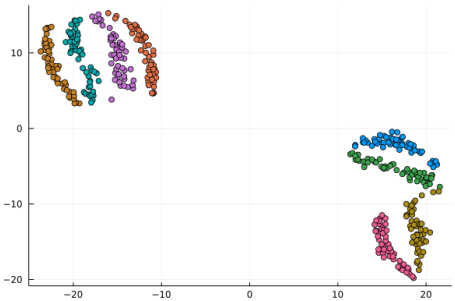
Limitations of PCA

Neighbourhood-relationships can get lost in PCA due to projections.



Alternative 1: t-SNE

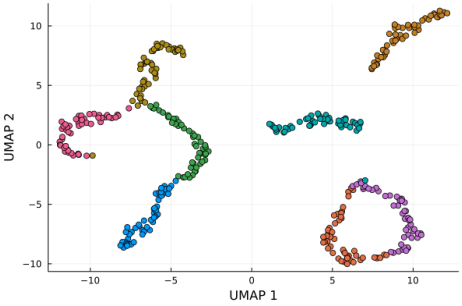
t-Distributed Stochastic Neighbor Embedding (t-SNE)



- t-SNE tries to preserve local neighbourhood-relationships, but the global structure disappears.
- t-SNE is a gradient descent procedure to find low-dimensional coordinates with the property that the probabilities of being neighbours in the low-dimensional space roughly match the probabilities of being neighbours in the high-dimensional space.
- Details can be found here: `https://lvdmaaten.github.io/tsne` and `https://distill.pub/2016/misread-tsne`

Alternative 2: UMAP

Uniform Manifold Approximation and Projection (UMAP)

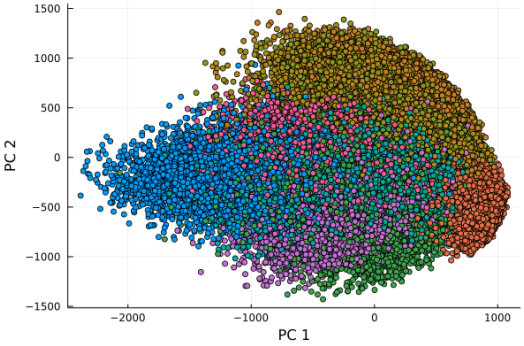


- UMAP also tries to preserve local neighbourhood-relationships, but the global structure disappears.
- It is usually faster than t-SNE.
- Details can be found here: `https://github.com/lmcinnes/umap`

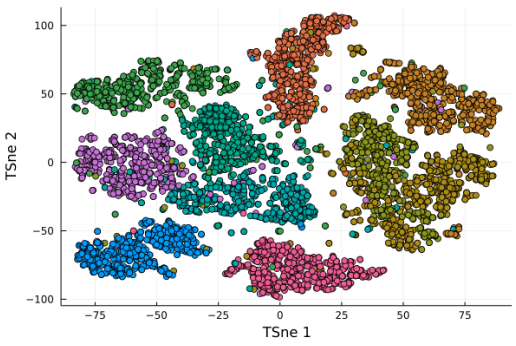
Limitations of PCA

Also in real data there may be clusters that cannot be seen in the first few PCs.

PCA on MNIST images

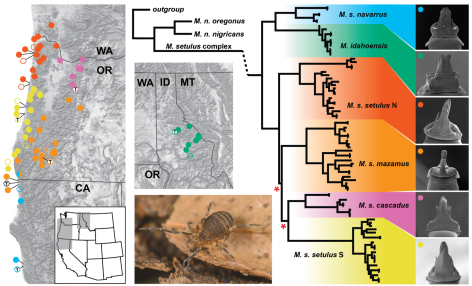


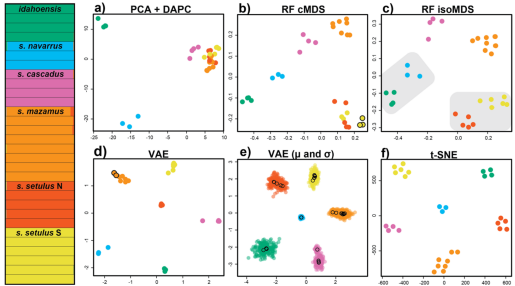
t-SNE on MNIST images



Species Delimitation with Unsupervised Machine Learning

Species Delimitation based on SNP data, dimensionality reduction and clustering:
if different methods find the same clusters, we can be somewhat confident
that the individual clusters are meaningful.





A demonstration of unsupervised machine learning in species delimitation, Derkarabetian et al. 2019

Table of Contents

Principal Component Regression

Unsupervised Feature Learning

PCA can be used to reduce the dimensionality of the data before supervised learning.

Instead of fitting $y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$ fit

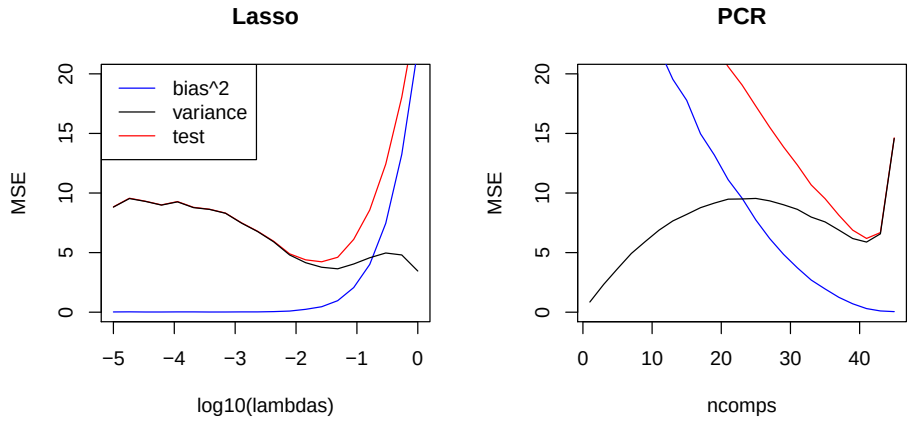
$$y_i = \beta_0 + \beta_1 z_{i1} + \cdots + \beta_m z_{im}$$

with $m < p$.

In this case we can choose m with cross-validation.

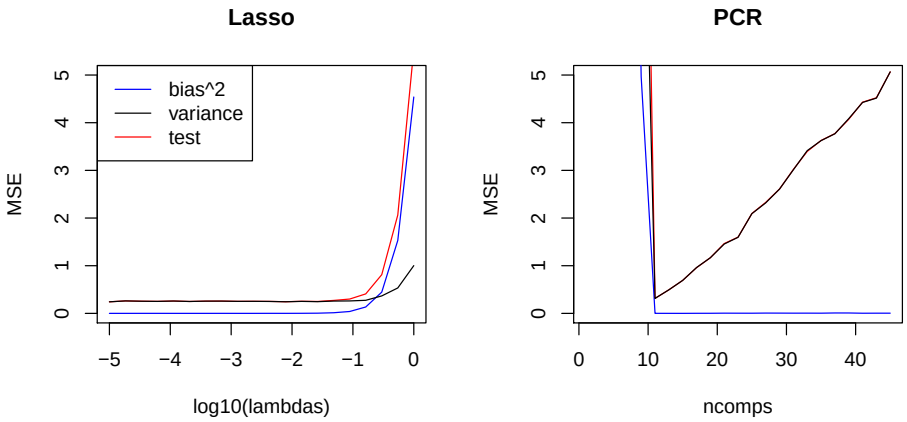
Choosing the Number of PCs in PCR

Choosing a small number of PCs in PCR acts as a regularizer that reduces the variance but may increase the bias



PCR versus Ridge Regression and the Lasso

Typically the Lasso (L1) or Ridge Regression (L2) are at least as good as PCR.



Summary

- Principal component analysis can be seen as
  1. finding the directions of largest variance in the data,
  2. finding linear subspaces that are closest to the data.
- The scores and loadings of PCA can be computed with Singular Value Decomposition or by finding the eigenvectors of the covariance matrix.
- Data with different units should be rescaled before applying PCA.
- The covariance matrix of the scores is diagonal.
- The (cumulative) proportion of variance explained can be used to see roughly how many PCs matter.
- PCA can fail to represent clusters faithfully; t-SNE may be more useful in this case.
- PCA can be used to reduce the dimensionality of a supervised learning problem, with the number of PCs acting as a regularization tuning parameter.

Suggested Reading

- 12.2 Principal Components Analysis
- 6.3.1 Principal Components Regression