Introduction to Machine Learning

PCA

Bernd Bischl, Christoph Molnar, Daniel Schalk, Fabian Scheipl

Department of Statistics – LMU Munich

Introduction

SUGGESTED LITERATURE

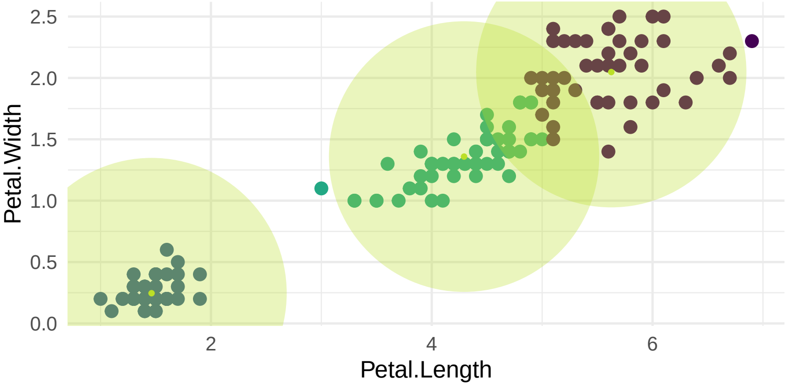
- Hastie, T., Tibshirani, R., Friedman, J. (2009): The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.
- James, G., Witten, D., Hastie, T., Tibshirani, R. (2013): An Introduction to Statistical Learning with Applications in R. Springer.
- Aggarwal, C. C., & Reddy, C. K. (Eds.). (2013). Data Clustering: Algorithms and Applications. CRC press.

UNSUPERVISED LEARNING

- Supervised machine learning deals with *labeled* data, i.e., we have input data x and the outcome y of past events.
- Here, the aim is to learn relationships between x and y.
- Unsupervised machine learning deals with data that is *unlabeled*, i.e., there is no real output y.
- Here, the aim is to search for patterns within the inputs x.

CLUSTERING TASK

Goal: Group data into similar clusters (or estimate fuzzy membership probabilities)



CLUSTERING: CUSTOMER SEGMENTATION

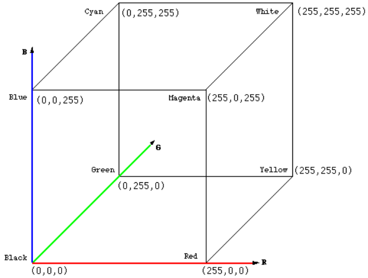
- In marketing, customer segmentation is an important task to understand customer needs and to meet with customer expectations.
- Customer data is partitioned in terms of similiarities and the characteristics of each group are summarized.
- Marketing strategies are designed and prioritized according to the group size.

Example Use Cases:

- Personalized ads (e.g., recommend articles).
- Music/Movie recommendation systems.

CLUSTERING: IMAGE COMPRESSION

- An image consists of pixels arranged in rows and columns.
- Each pixel contains **RGB** color information, i.e., a mix of the intensity of 3 **primary colors**: **R**ed, **G**reen and **B**lue.
- Each primary color takes intensity values between 0 and 255.

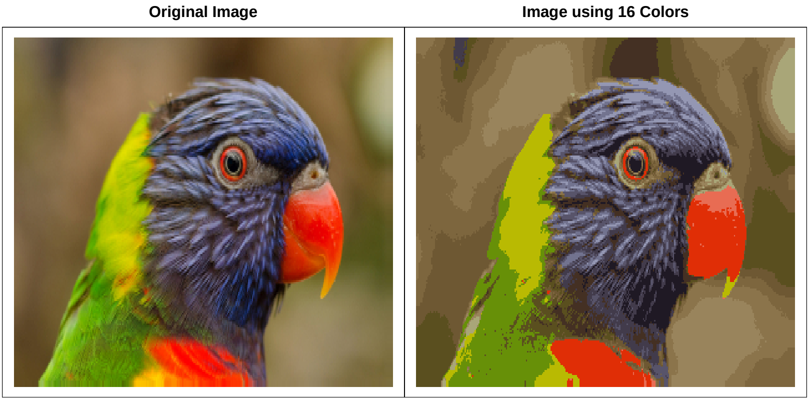


Source: By Ferlixwangg CC BY-SA 4.0, from Wikimedia Commons.

CLUSTERING: IMAGE COMPRESSION

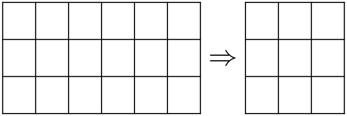
An image can be compressed by reducing its color information, i.e., by replacing similar colors of each pixel with, say, k distinct colors.

Example:



DIMENSIONALITY REDUCTION TASK

Goal: Describe data with fewer features (reduce number of columns).
$\Rightarrow$ there will always be an information loss.



Unsupervised Methods:

- Principle Component Analysis (PCA).
- Factor Analysis (FA).
- Feature filter methods.

Supervised Methods:

- Linear Discriminant Analysis (LDA).
- Feature filter methods.

Principal Component Analysis

NORMALIZING DATA

A variable X can be normalized by substracting its values with the mean $\bar{X}$ and dividing by the standard deviation s_X, e.g. $\tilde{X} = \frac{X - \bar{X}}{s_X}$.

Example:
Consider the following body heights measured in different units:

	Person A	Person B	Person C	mean	sd
body height (cm)	180.00	172.00	175.00	175.67	4.04
body height (m)	1.80	1.72	1.75	1.76	0.04
body height (feet)	5.91	5.64	5.74	5.76	0.13

After normalizing, we always obtain the normalized body height (no matter which unit was used):

	Person A	Person B	Person C	mean	sd
normalized body height	1.07	-0.91	-0.16	0.00	1.00

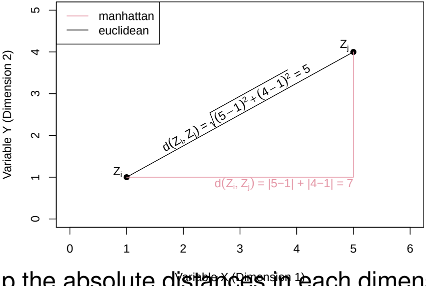
NORMALIZING DATA

Normalizing all variables in a data set, can have several advantages:

- It puts all variables into *comparable* units, i.e., we make sure that all normalized variables have mean 0 and standard deviation of 1.
- It can avoid numerical instabilites in several algorithms, e.g. if a variable has very low / high values.
- It helps in computing meaningful *distances* between observations.

NORMALIZING DATA:DISTANCES

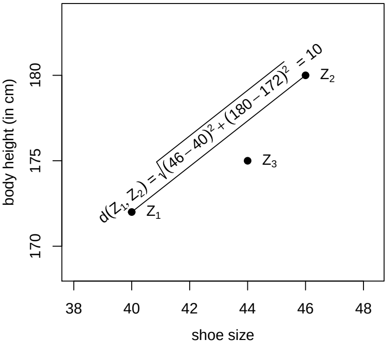
There are many ways to define the distance between two points, e.g., $Z_i = (X_i, Y_i)$ and $Z_j = (X_j, Y_j)$:



- manhattan: sum up the absolute distances in each dimension.
- euclidean: remember Pythagoras theorem from school?

NORMALIZING DATA:DISTANCES

It is often a good idea to *normalize* the data before computing distances, especially when the scale of variables is different, e.g. the euclidean distance between the point Z_1 and Z_2:



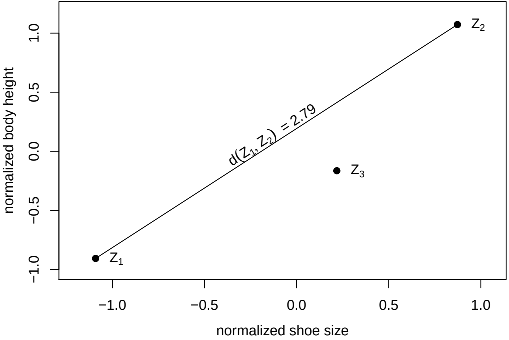
On the right plot, the distance is dominated by “shoe size”.

NORMALIZING DATA: DISTANCES

The normalized variable $\tilde{X}_{\texttt{shoe.size}}$ is computed by <!– Normalization of the `shoe.size` variable means: –>

$$\tilde{X}_{\texttt{shoe.size}} = \frac{X_{\texttt{shoe.size}} - \bar{X}_{\texttt{shoe.size}}}{s_{X_{\texttt{shoe.size}}}}.$$

Distances based on normalized data are better comparable and **robust** in terms of linear transformations (e.g., conversion of physical units).



NORMALIZING: COVARIANCE VS. CORRELATION

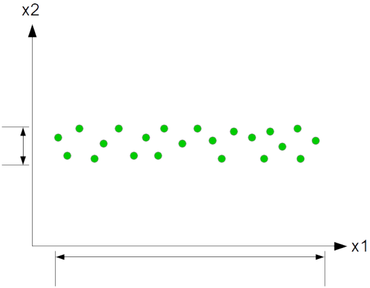
The **variance** of a normalized variable is always 1, its mean is always 0. The **covariance** of two normalized variables $\tilde{X} = \frac{X - \bar{X}}{s_X}$ and $\tilde{Y} = \frac{Y - \bar{Y}}{s_Y}$ is the same as the **correlation** of the non-normalized variables X and Y. One can proof this with the help of

$$s_{\tilde{X}\tilde{Y}} = \frac{1}{n-1} \sum_{i=1}^{n} (\tilde{x}_i - \bar{\tilde{x}})(\tilde{y}_i - \bar{\tilde{y}}) = \ldots = \frac{1}{n-1} \sum_{i=1}^{n} \frac{(x_i - \bar{x})}{s_X} \frac{(y_i - \bar{y})}{s_Y} = r_{XY}.$$

PCA INTUITION

Motivational example I:

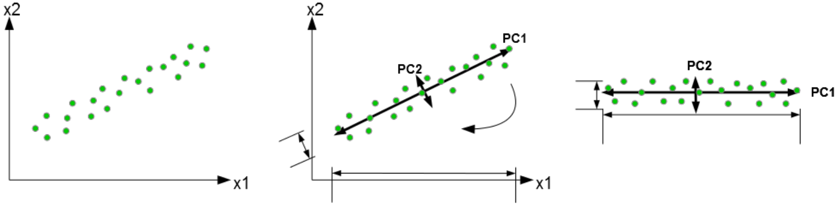
- Variable x_1 explains most of the variation.
- Variable x_2 has a lower variance than x_1.
- If we disregard x_2 and project the points into the 1-dimensional space of x_1, we do not lose much information w.r.t. variability.



PCA INTUITION

Motivational example II:

- x_1 and x_2 are correlated and have similar variances.
- Find a new orthogonal axes (e.g. PC1 and PC2), where PC1 explains most of the variation.
- Rotate the points and consider PC1 and PC2 as new coordinate system (situation as in the previous example).
- We can now project points onto PC1 and disregard PC2 (hopefully without losing much information).



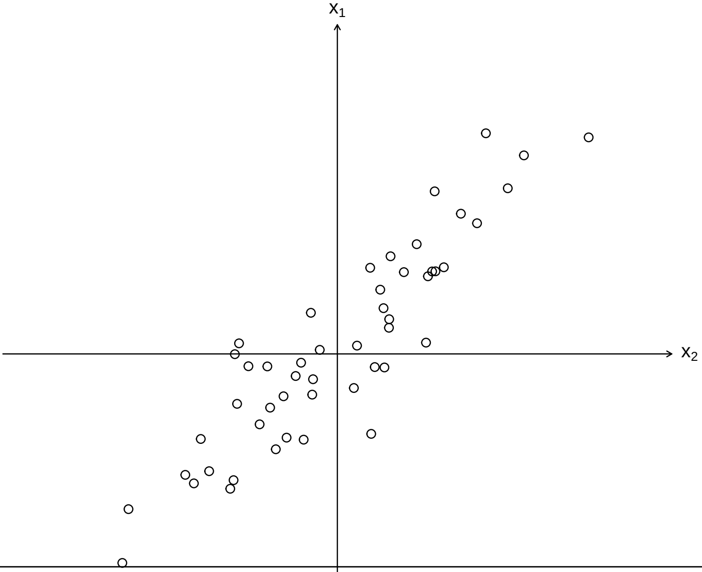
PCA INTUITION

General procedure:

1. Rotate the original p-dimensional coordinate system until the first PC that explains most of the variation is found.
2. Fix the first PC and proceed with rotating the remaining $p - 1$ coordinates until the second PC (which is orthogonal to the first PC) is found that explains most of the *remaining* variation, etc.
3. We can reduce the dimensions by projecting the points onto the first, say $k < p$, PC.

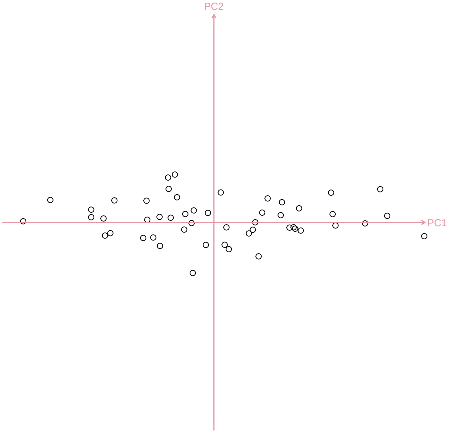
PCA INTUITION: FIND FIRST PC

Test



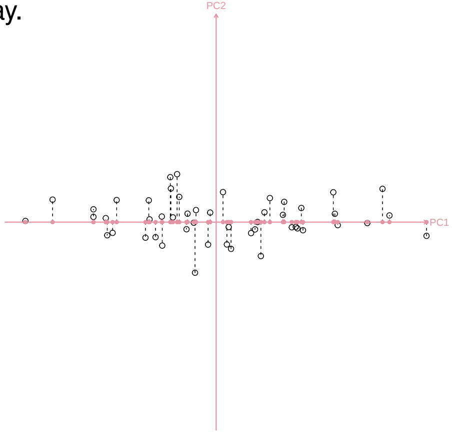
PCA INTUITION: REDUCE DIMENSIONALITY

Rotate the points and use PC1 and PC2 as new coordinate system.
Here, the PC1 axis explains most of the variance:



PCA INTUITION: REDUCE DIMENSIONALITY

Dimensionality can be reduced by projecting the points onto the PC1 (and by disregarding PC2). The hope is that we won't lose much information this way.



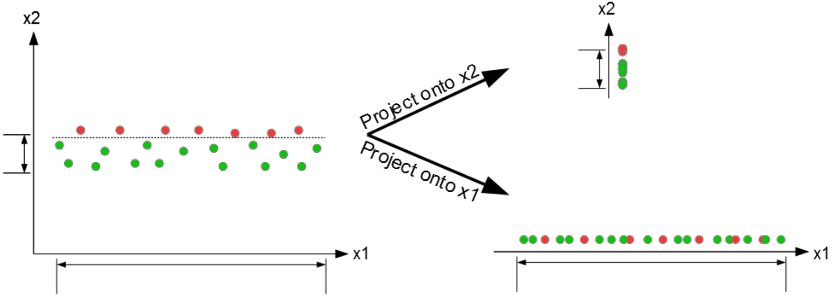
PCA INTUITION: SUMMARY

Idea: Transform an original set of correlated metric variables to a new set of uncorrelated (orthogonal) metric variables, called principal components (PC), that explain the variability in the data.

- The objective is to investigate if only a few PC account for most of the variability in the original data.
- If the objective is fulfilled, we can use fewer PCs to reduce the dimensionality.
- The PCs remove collinearity of the input variables as they are orthogonal to each other.

PCA INTUITION: FINAL REMARKS

- PCA is used for dimensionality reduction by disregaring dimensions with lower variability.
- There is always an information loss, especially for other criteria.
- E.g., dimensionality reduciton can worsen the classification accuracy when the task is to classify two groups:



DERIVING THE FIRST PC MATHEMATICALLY

Aim: Find a new set of variables (PC scores) $\mathbf{pc}_1, \dots, \mathbf{pc}_p$ based on the original data $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_p]$ so that

- each PC score $\mathbf{pc}_1, \dots, \mathbf{pc}_p$ is a linear combination of the original metric variables with coefficient weights (so-called **loading vectors**) $\mathbf{a}_1, \dots, \mathbf{a}_p$, i.e.

$$\mathbf{pc}_j = a_{j1}\mathbf{x}_1 + a_{j2}\mathbf{x}_2 + \dots + a_{jp}\mathbf{x}_p = \mathbf{X}\mathbf{a}_j.$$

- the set is mutually uncorrelated: $Cov(\mathbf{pc}_j, \mathbf{pc}_k) = 0, \ \forall j \neq k$.
- the variances of the PC scores decrease:

$$\lambda_1 > \lambda_2 > \dots > \lambda_p, \quad \text{where } \lambda_k := Var(\mathbf{pc}_k).$$

DERIVING THE FIRST PC MATHEMATICALLY

We look for the loading vector $\mathbf{a}_1 = (a_{11}, a_{21}, \dots, a_{p1})^\top$ that maximizes the variance of $\mathbf{pc}_1$:

$$\max_{\mathbf{a}_1} \; Var(\mathbf{pc}_1) = Var(\mathbf{X}\mathbf{a}_1) = \mathbf{a}_1^\top \Sigma \mathbf{a}_1$$

subject to the normalization constraint $\mathbf{a}_1^\top \mathbf{a}_1 = \sum_{k=1}^{p} a_{k1}^2 = 1$.

The constraint is required for identifiability reasons, otherwise we could maximize the variance by just increasing the values in $\mathbf{a}_1$.

Repeat this maximization step for the other PCs and additionally use the orthogonality constraint, i.e. for the second PC:

$$\mathbf{a}_2^\top \mathbf{a}_1 = 0.$$