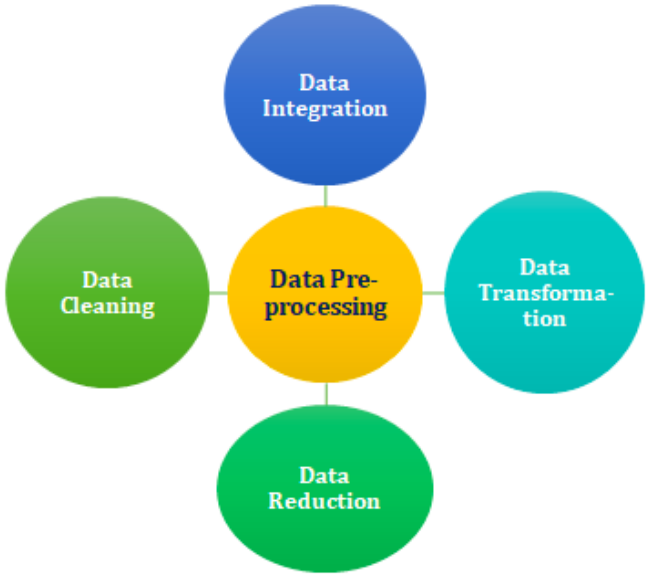
DI501 Introduction to Data Informatics

Lecture 4 – Data Preprocessing – Part V

Major Tasks in Data Preprocessing

- Data integration
  - Integration of multiple databases, data cubes, or files
- Data cleaning
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- Data reduction
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- Data transformation and data discretization
  - Normalization
  - Concept hierarchy generation



Data Reduction Strategies

- Data reduction: Obtain a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results
- Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.
- Data reduction strategies
  - Dimensionality reduction, e.g., remove unimportant attributes
    - Wavelet transforms
    - Principal Components Analysis (PCA)
    - Feature subset selection, feature creation
  - Numerosity reduction (some simply call it: Data Reduction)
    - Regression and Log-Linear Models
    - Histograms, clustering, sampling
    - Data cube aggregation
  - Data compression

Data Reduction 1:

Dimensionality Reduction

- Dimensionality reduction
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization
- Dimensionality reduction techniques
  - Principal Component Analysis
  - Wavelet transforms
  - Supervised and nonlinear techniques (e.g., feature selection)

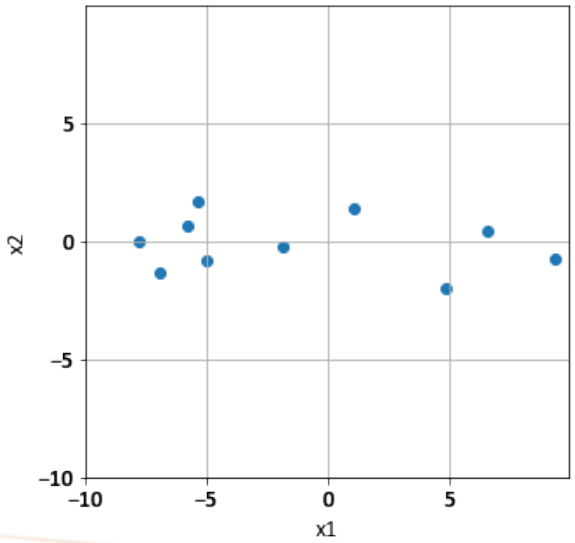
Transform or project the original data onto a smaller space.

Principal Component Analysis (PCA)

- Suppose we have a dataset comprising *n* records and each record has *m* attributes.
  - That is, each record is represented in *m* dimensional space
- If *m* is large
  - Analyses may require too much resources (e.g., memory, time)
  - It may be difficult to visualize data and see correlations between the features
    - There may be highly correlated features
  - There may be too much noise
- Can we represent data in a lower dimensional space while preserving as much information as possible?
  - For example, can we project 10-D into 2-D while keeping the most of the information in the original data?

Principal Component Analysis (PCA)

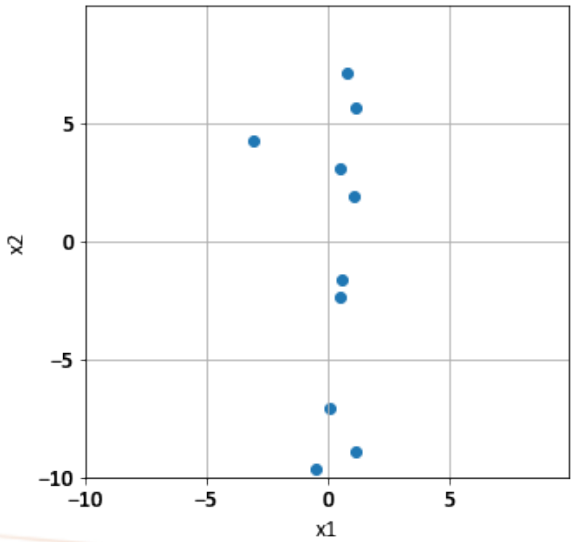
- Example 1: How can we represent each record by a single feature?
  - 2D to 1D
  - with minimum information loss!



	x1	x2
0	4.98	0.19
1	0.83	0.24
2	7.74	-1.28
3	-6.61	-1.24
4	-9.85	0.65
5	-1.32	1.48
6	-1.29	-1.01
7	1.83	-0.43
8	-8.32	-0.11
9	1.47	0.28

Principal Component Analysis (PCA)

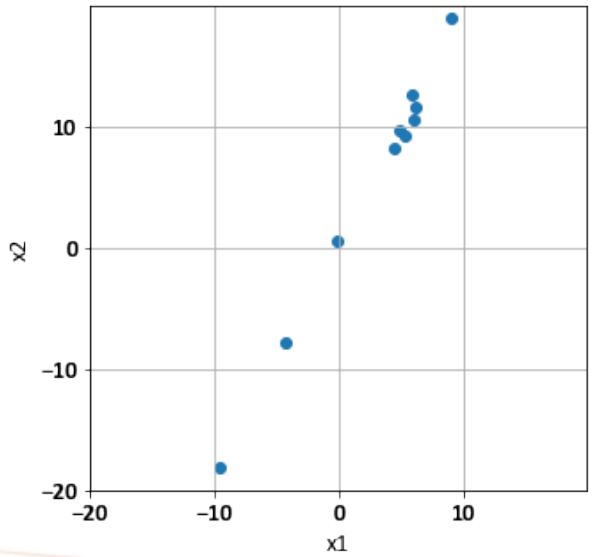
- Example 2: How can we represent each record by a single feature?
  - 2D to 1D
  - with minimum information loss!



	x1	x2
0	0.09	-7.09
1	-0.50	-9.64
2	0.50	-2.40
3	0.61	-1.59
4	1.17	5.63
5	0.82	7.12
6	-3.09	4.23
7	1.16	-8.89
8	1.08	1.88
9	0.49	3.11

Principal Component Analysis (PCA)

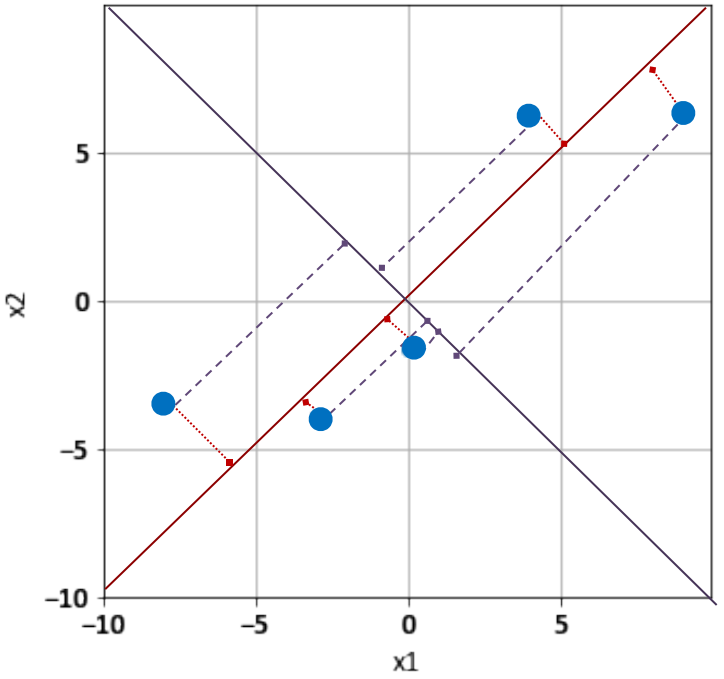
- Example 3: How can we represent each record by a single feature?
  - 2D to 1D
  - with minimum information loss!



	x1	x2
0	5.97	12.60
1	-9.61	-18.12
2	4.90	9.77
3	4.45	8.22
4	5.30	9.20
5	5.98	10.57
6	9.10	18.92
7	-4.29	-7.86
8	6.13	11.61
9	-0.11	0.62

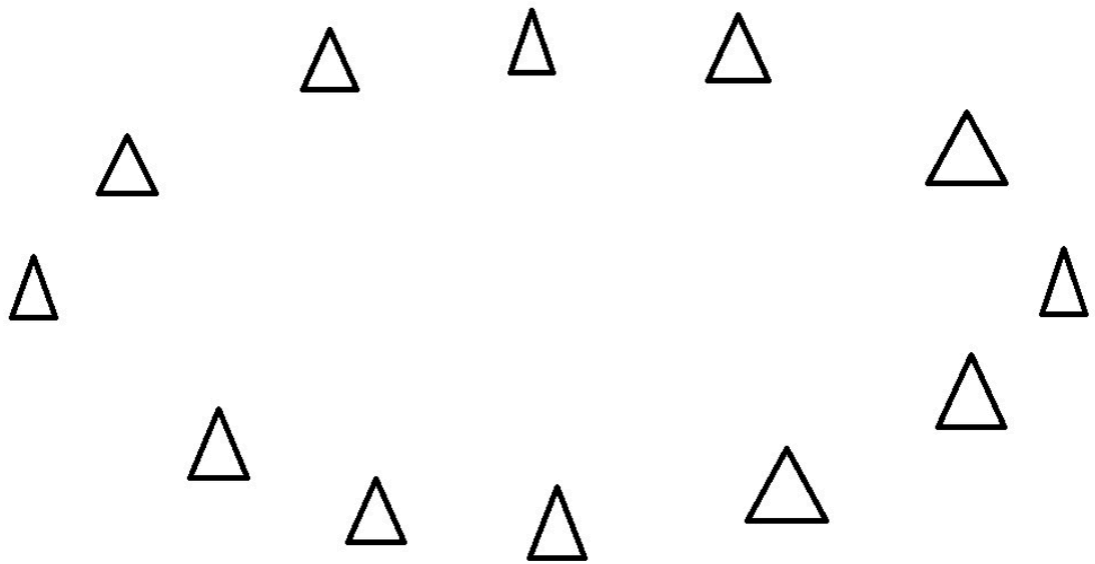
Principal Component Analysis (PCA)

- Idea: Orthogonal projection of the data onto a lower-dimensional space such that
  - Variance of the projected data is minimized.
  - Sum of the mean squared distance between the data points and their projections is minimized.
- Principle components
  - The first: points in the direction of largest variance.
  - Subsequent: orthogonal to the previous one(s) and points in the direction of the largest variance of the residual subspace



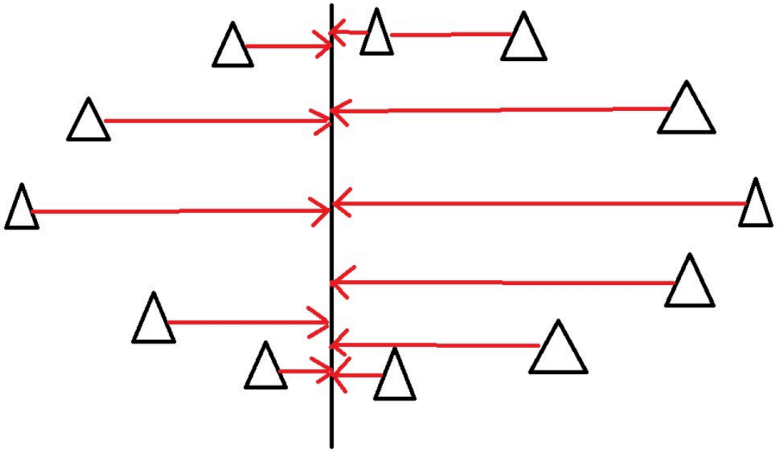
Principal Component Analysis (PCA)

- Imagine that the triangles are points of data. Can you find the direction where there is most variance?

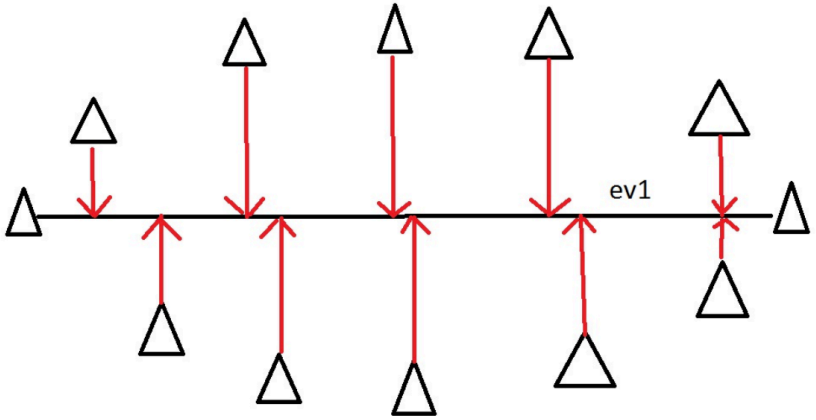


Principal Component Analysis (PCA)

- Can you find the direction where there is most variance?



It has a smaller variance in this projection.



It has a large variance in this projection.

Principal Component Analysis (PCA)

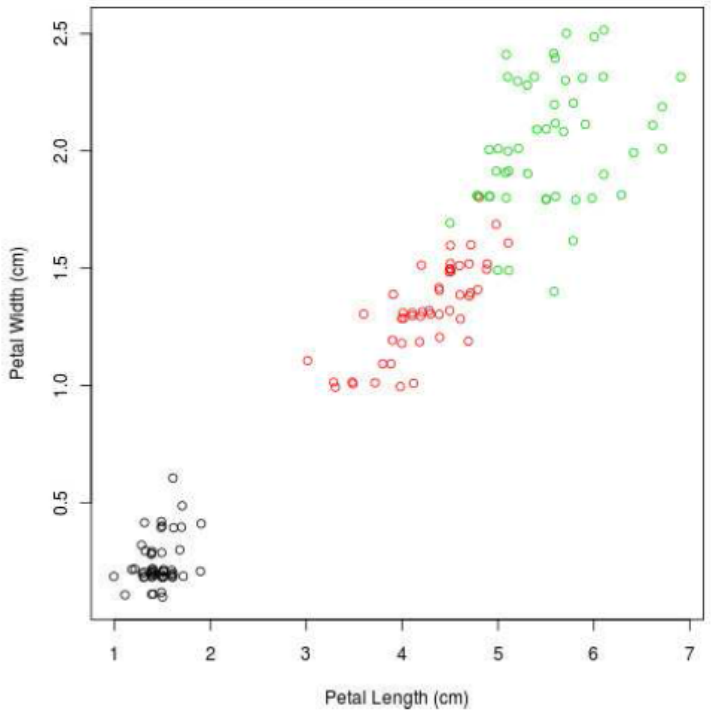
- Let $\boldsymbol{X}$ be the original dataset where each column is a single unit $\mathbf{x}_i$ and each row corresponds to a feature.
  - $\boldsymbol{X}$ is $m \times n$ matrix where m = # of features and n = # of samples
  - Suppose the rows of $\boldsymbol{X}$ have zero mean
- The covariance of the features in our dataset will be
  - $\boldsymbol{C_X} = \frac{1}{n-1}\boldsymbol{XXT}$
  - Note that $\boldsymbol{CX}$ is symmetric.
- We want to find a linear transformation $\boldsymbol{P}$ such that
  - $\boldsymbol{Y} = \boldsymbol{PX}$ and $\boldsymbol{C_Y} = \frac{1}{n-1}\boldsymbol{YYT}$ is diagonal
  - i.e., the new set of features will not be correlated
- $\boldsymbol{C_Y} = \frac{1}{n-1}\boldsymbol{YYT} = \frac{1}{n-1}\boldsymbol{PX}(\boldsymbol{PX})\boldsymbol{T} = \frac{1}{n-1}\boldsymbol{PXXTPT} = \boldsymbol{PCXPT}$

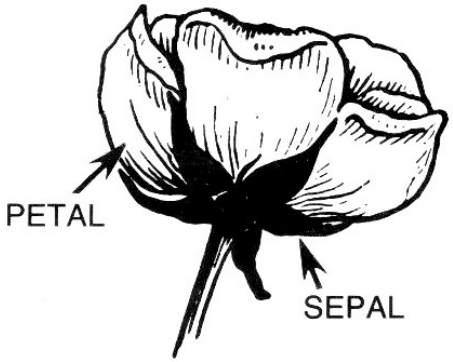
Principal Component Analysis (PCA)

- A symmetric matrix is diagonalized by an orthogonal matrix of its eigenvectors
  - i^{th} diagonal element will be the i^{th} eigenvalue.
- Let $\boldsymbol{A}$ be an $n \times n$ matrix
  - Scalar λ and $n \times 1$ vector $\boldsymbol{v}$ pairs satisfying $\boldsymbol{Av} = \boldsymbol{\lambda v}$ are the eigenvalues and the corresponding eigenvectors of $\boldsymbol{A}$.
  - there are *n* eigenvalues (some may be the same).
- Hence, we can select the matrix $\boldsymbol{P}$ to be a matrix where each row $\boldsymbol{pi}$ is an eigenvector of $\boldsymbol{XXT}$

Principal Component Analysis (PCA)

- Find a projection that captures the largest amount of variation in data
- It is about stretching and rotating the data.
- PCA decorrelates the vector features (a feature corresponds to an element position) which might sometimes help in fitting a model.



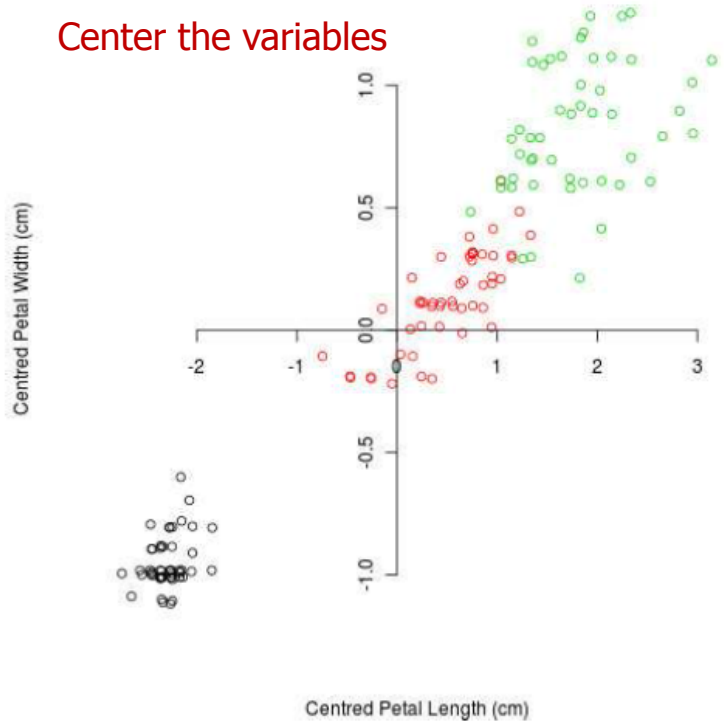


Data set belongs to Iris data set,
There are 4 dimensions:
sepal length
sepal width
petal length
petal width
Here are the petal measurements (the different colors are different species)

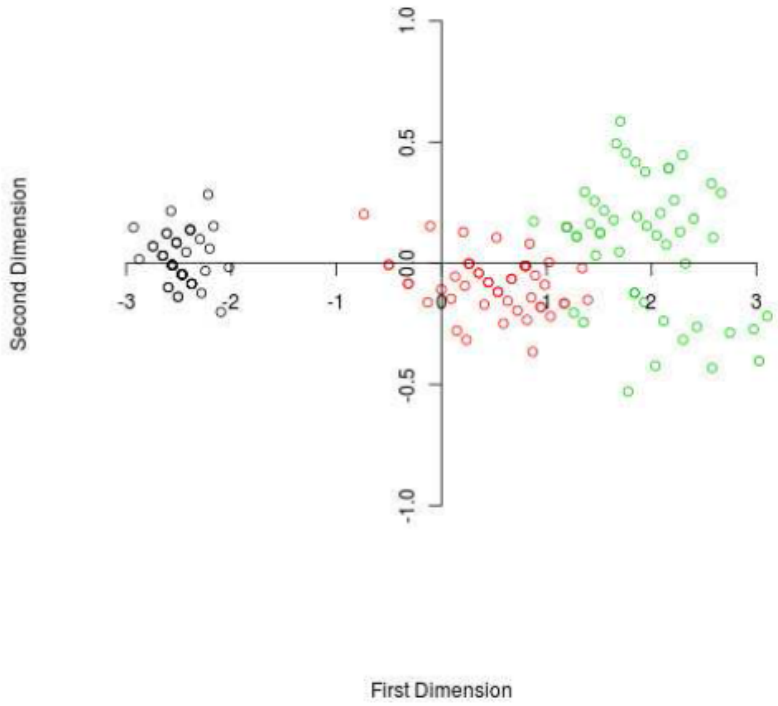
Principal Component Analysis (PCA)

- The purpose of PCA is to represent as much of the variation as possible in the first few axes. To do this we first, center the variables to have a mean of zero and then rotate the data (or rotate the axes):

Center the variables



Rotate the axis

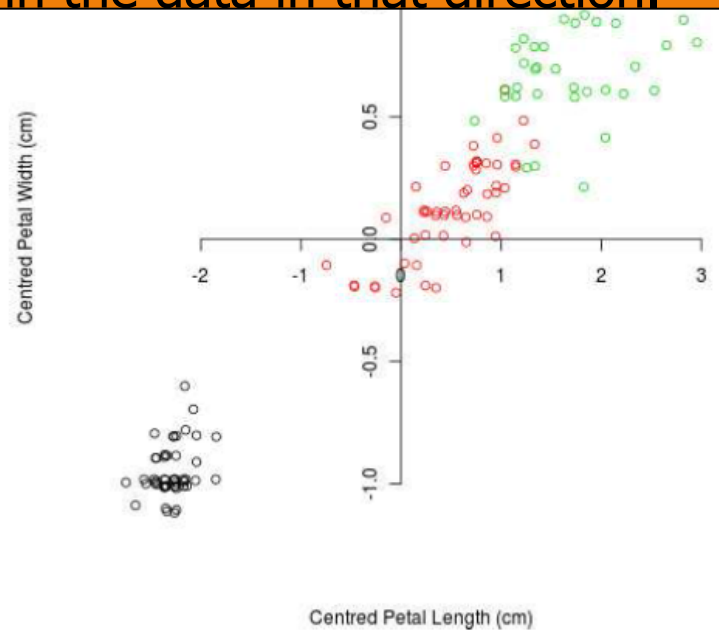


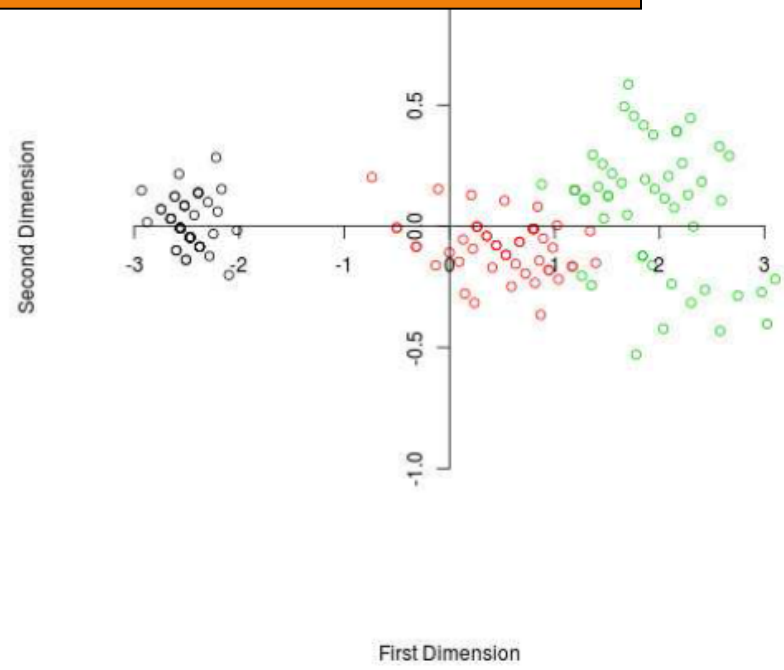
The eigenvector with the highest eigenvalue is the principal component.
Principal components are the underlying structure in the data.
They are the directions where there is the most variance, the directions where the data is most spread out.
Every eigenvector has a corresponding eigenvalue.
An eigenvector is a direction.
An eigenvalue is a number telling you how much variance there is in the data in that direction.

A)

ible in

mean of



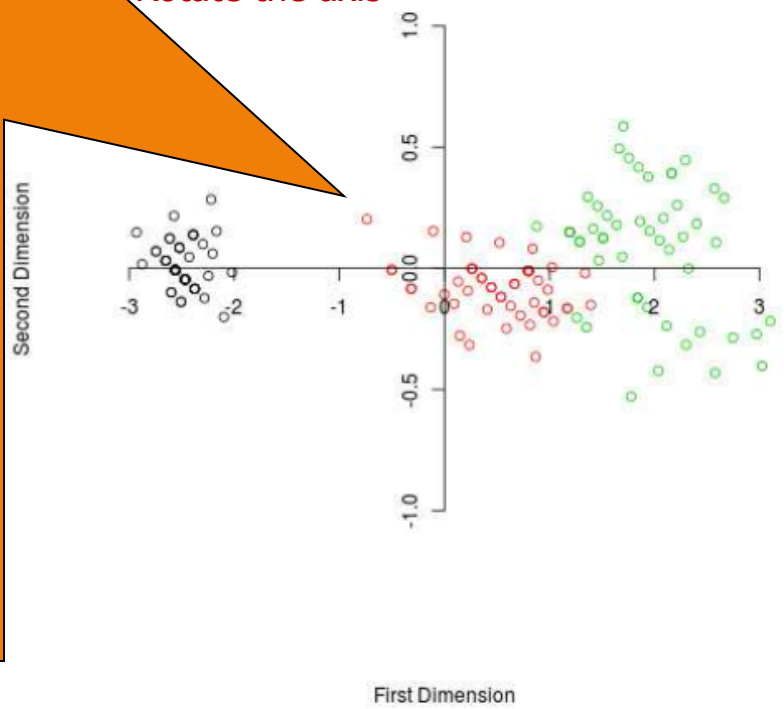


Principal Component Analysis (PCA)

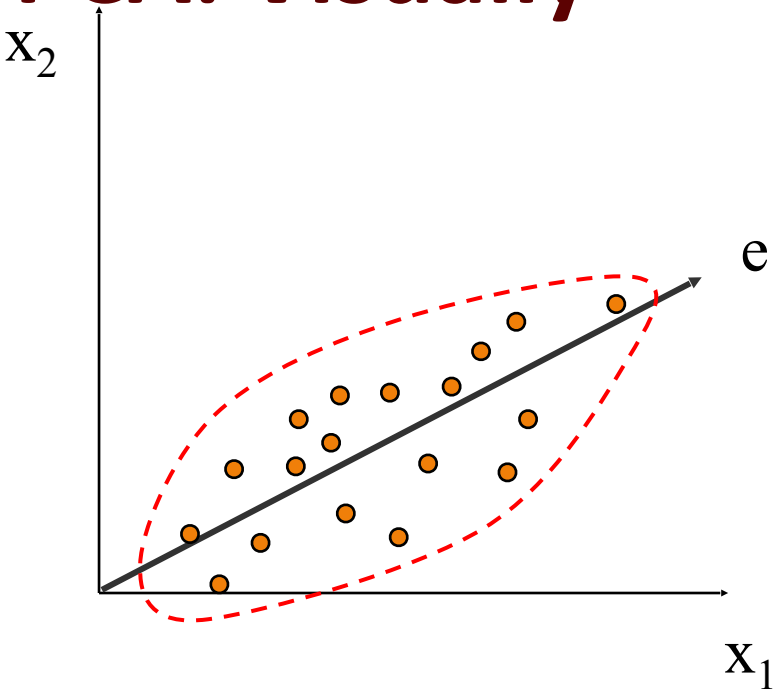
- The ... much of the variation as possible in ... enter the variables to have a mean of ... te the axes):

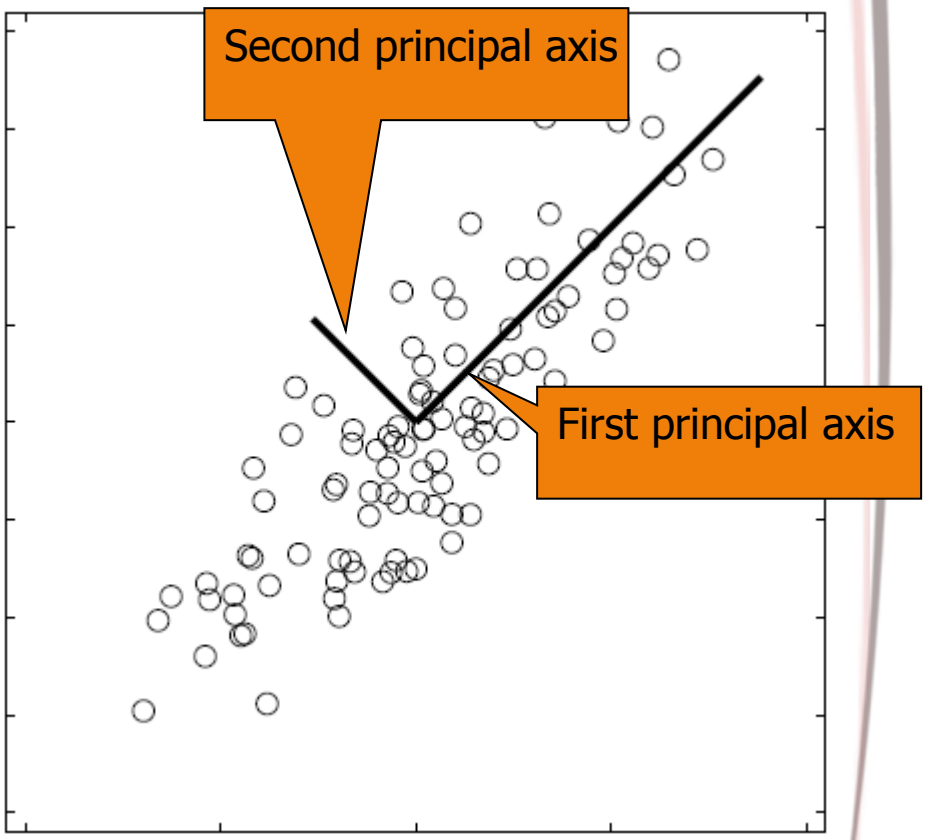
The rotation is done so that the first axis contains as much variation as possible, the second axis contains as much of the remaining variation etc.
Thus if we plot the first two axes, we know that these contain as much of the variation as possible in 2 dimensions.
As well as rotating the axes, PCA also re-scales them: the amount of re-scaling depends on the variation along the axis.
This can be measured by the Eigenvalue, and it's common to present the proportion of total variation as the Eigenvalue divided by the sum of the Eigenvalues, e.g. for the data above the first dimension contains 99% of the total variation.

Rotate the axis



PCA: Visually





- Data points are represented in a rotated orthogonal coordinate system: the origin is the mean of the data points and the axes are provided by the eigenvectors.

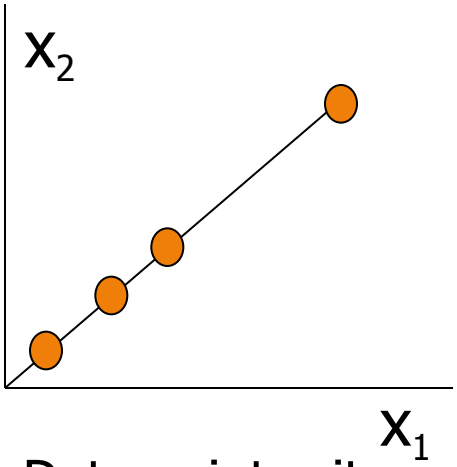
PCA

- What PCA asks: Is there another basis, which is a linear combination of the original basis, that best re-expresses our data set?
- PCA is limited to re-expressing the data as a linear combination of its basis vectors.
- Let **X** and **Y** be *m*×*n* matrices related by a linear transformation **P**. **X** is the original recorded data set and **Y** is a re-representation of that data set.

$$\mathbf{PX}=\mathbf{Y}$$

- This equation represents a change of basis and thus can have many interpretations.
- **P** is a matrix that transforms **X** into **Y**.
- Geometrically, **P** is a rotation and a stretch which again transforms **X** into **Y**.
- The rows of **P**, $\{p_1, \ldots, p_m\}$, are a set of new basis vectors for expressing the columns of **X** (principal components).
- We select the matrix **P** to be a matrix where each row p_i is an eigenvector of $\mathbf{XX}^T$.
- In practice computing *PCA* of a data set **X** entails (1) subtracting off the mean of each measurement type and (2) computing the eigenvectors of $\mathbf{XX}^T$.

PCA: Example



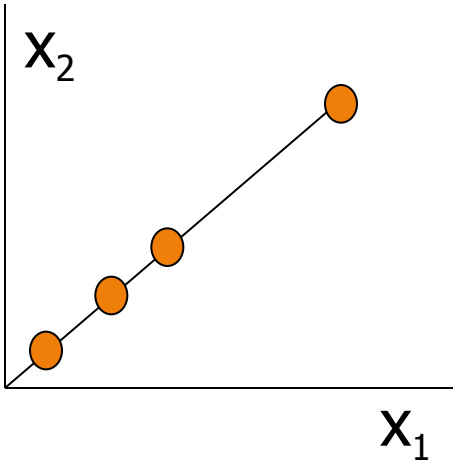
x1	x2	Single coordinate value
1	1	1.414
3	3	4.243
4	4	5.657
8	8	11.314

Data points sit on the straight line $x_2=x_1$

So each data point can be described by a single «principal coordinate» with no loss of information.

The main idea behind principal component analysis is to derive a linear function *y* for each of the vector variables x_i. This linear function possesses an extremely important property; namely, its variance is maximized.

PCA: Example



x1	x2	Single coordinate value
1	1	1.414
3	3	4.243
4	4	5.657
8	8	11.314

$$\begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 1 & 3 & 4 & 8 \\ 1 & 3 & 4 & 8 \end{bmatrix} = \begin{bmatrix} 1.414 & 4.243 & 5.657 & 11.314 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

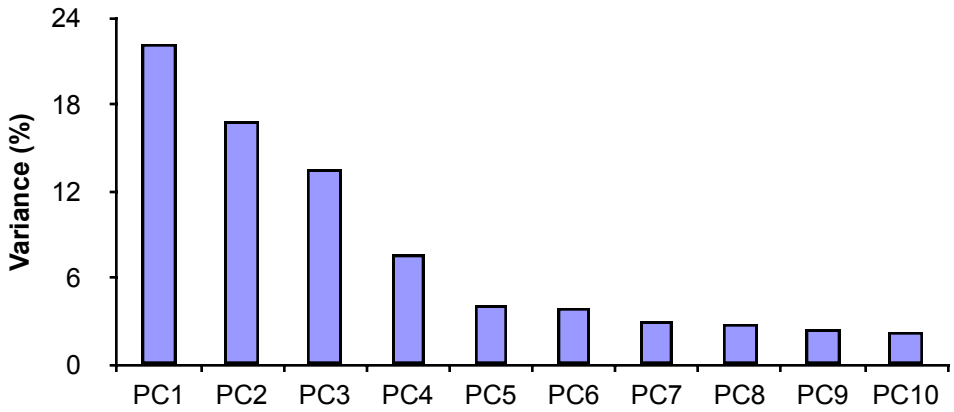
The inverse mapping for 1.414 is: $[1.414] \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = [1\ 1]$

PCA: Steps

- Given *n* data vectors from *m*-dimensions, find $p \leq m$ orthogonal vectors (principal components) that can be best used to represent data
  - Subtract mean from each feature (i.e., rows have zero mean)
  - Compute *p* orthonormal (unit) vectors, i.e., principal components
  - Each input data (vector) is a linear combination of the *p* principal component vectors
  - The principal components are sorted in order of decreasing "significance" or strength
  - Since the components are sorted, the size of the data can be reduced by eliminating the weak components, i.e., those with low variance (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data)
- Works for numeric data only

PCA: Principal Components

- Can ignore the components of lesser significance.



- You do lose some information, but if the eigenvalues are small, you don't lose much
  - m dimensions in original data
  - calculate m eigenvectors and eigenvalues
  - choose only the first p eigenvectors, based on their eigenvalues
  - final data set has only p dimensions

Example

- A student asked 1000 people a set of questions about their personality.
  - 50 questions were asked to each participant.
  - So we have 50 dimensions.
  - There will be 50 eigenvectors/values at the end.
  - Assume that the eigenvalues of the data set (in descending order) are as follows:
    - 52, 43,20,19,4,2,1,0.5,0.2,...
    - There are four significant components
    - The corresponding dimensions can be selected in further analysis.
    - We found the significant dimensions.

Kernel PCA

- PCA is designed to model linear variabilities in high dimensional data.
- For non-linear dimensionality reduction, Kernel PCA can be used.
- Kernel PCA finds principal components which are nonlinearly related to the input space by performing PCA in the space produced by the nonlinear mapping, where the low-dimensional latent structure is, hopefully, easier to discover.

Ref: http://www.math.uwaterloo.ca/~aghodsib/courses/f06stat890/readings/tutorial_stat890.pdf

PCA: Example

- Morchid et al. (2014) studied the behavior of tweets that have been massively retweeted in a short period of time. They first analyzed specific tweet features through a PCA.
- A corpus of 6 million tweets was collected using the official Twitter API.

Description of tweet features for a user **U**.

	Description	Twitter API names
Content features		
Retweet	# of sharing	*retweet_count*
Hashtag	# of topics in a tweet	*hashtags*
Mention	# of cited usernames	*text*
Url	# of contained URLS	*urls*
User features		
Days	# of days **U** created its account	**U**/*created_at*
Favorite	# of favorite tweets by **U**	**U**/*favourites_count*
Follower	# of users who follow **U**	**U**/*followers_count*
Followee	# of friends of **U**	**U**/*friends_count*
Status	# of tweets wrote by **U**	**U**/*statuses_count*

Morchid, M., Dufour, R., Bousquet, P. M., Linares, G., & Torres-Moreno, J. M. (2014). Feature selection using principal component analysis for massive retweet detection. Pattern Recognition Letters, 49, 33-39.

PCA: Example

- Understanding retweet behavior

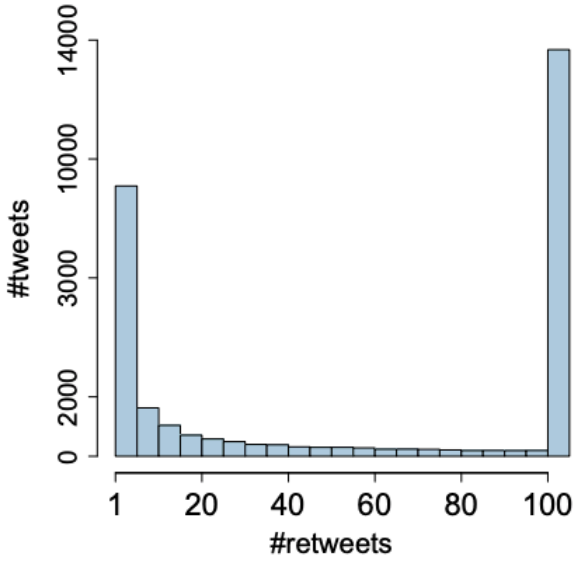


Fig. 1. Tweet distribution depending on the number of retweets.

Columns 3 and 4 of Table 2 respectively represent the percentage of the total variance and the cumulative variance accounted for from the original dataset. These factors are sorted in descending order depending on their eigenvalues.

Table 2
Principal components from a PCA.

Component (factor f_i)	Eigenvalues ev_i	Eigenvalues % variance $v(f_i)$	Cumulative % variance
1	1.65	18.41	18.41
2	1.31	14.57	32.98
3	1.11	12.33	45.31
4	1.00	11.02 ($\simeq \frac{1}{9}$)	56.34
5	0.94	10.50	66.84
6	0.87	9.71	76.55
7	0.76	8.47	85.02
8	0.70	7.79	92.82
9	0.64	7.17	100

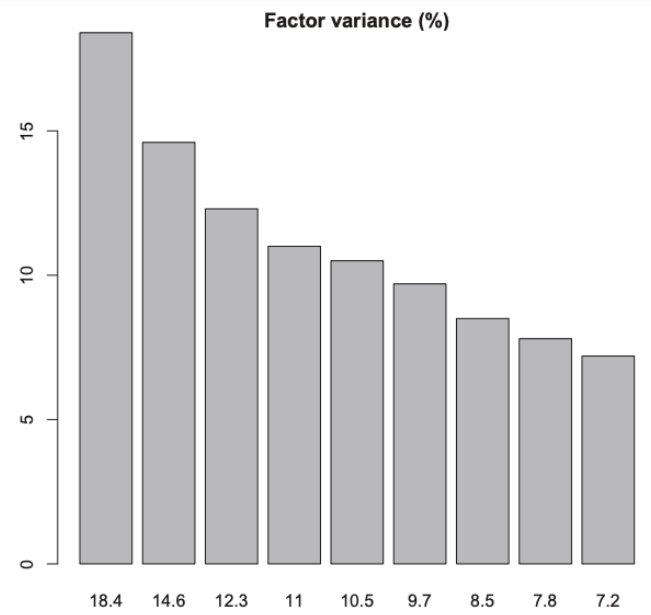


Fig. 3. Proportion of the total variance contribution for each of the 9 studied features.

Morchid, M., Dufour, R., Bousquet, P. M., Linares, G., & Torres-Moreno, J. M. (2014). Feature selection using principal component analysis for massive retweet detection. Pattern Recognition Letters, 49, 33-39.

PCA: Example

- Understanding retweet behavior

Table 2
Principal components from a PCA.

Component (factor f_i)	Eigenvalues ev_i	Eigenvalues % variance $v(f_i)$	Cumulative % variance
1	1.65	18.41	18.41
2	1.31	14.57	32.98
3	1.11	12.33	45.31
4	1.00	11.02 ($\simeq \frac{1}{9}$)	56.34
5	0.94	10.50	66.84
6	0.87	9.71	76.55
7	0.76	8.47	85.02
8	0.70	7.79	92.82
9	0.64	7.17	100

They first looked into the number of components with Kaiser criterion (The eigenvalue-one criterion): You retain and interpret any component with an eigenvalue greater than 1.00.

* In this case, only four components meet this criteria.

They also looked into the proportion of variance accounted for.
In the original space representation, each of the 9 features contains about 11% (1/9) of the total variance explained by all the features. The principal component retained should then explain at least 11% of the total variance. The variance of a factor f_i is performed as follows:

$$v(f_i) = \frac{ev_i}{\sum_{j=0}^{n} ev_j}$$

As a result, the application of these two rules allows the selection of only the first 4 factors.

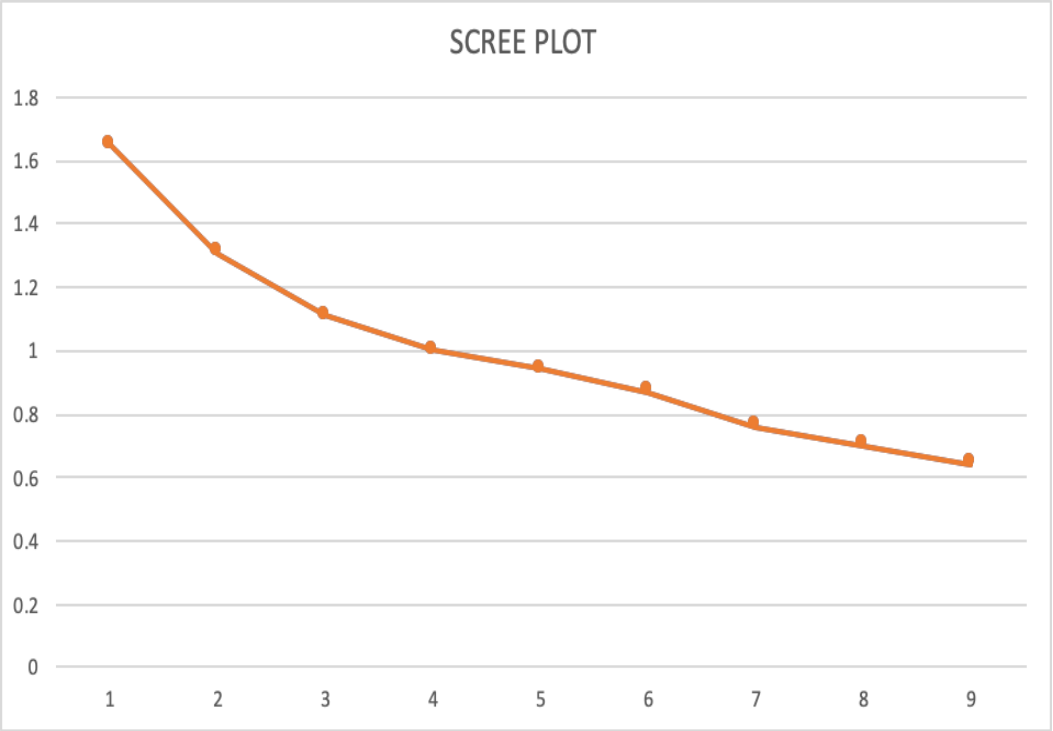
Morchid, M., Dufour, R., Bousquet, P. M., Linares, G., & Torres-Moreno, J. M. (2014). Feature selection using principal component analysis for massive retweet detection. Pattern Recognition Letters, 49, 33-39.

PCA: Example

Another method for selecting number of components:
Scree Test: Plot the eigenvalues associated with each component and look for a “break” between the components with relatively large eigenvalues and those with small eigenvalues. The components that appear before the break are assumed to be meaningful and are retained for rotation; those appearing after the break are assumed to be unimportant and are not retained.

In the scree plot, the first three components appear to be more significant.

They selected the number of components with Kaiser criterion (The eigenvalue-one criterion): You retain and interpret any component with an eigenvalue greater than 1.00

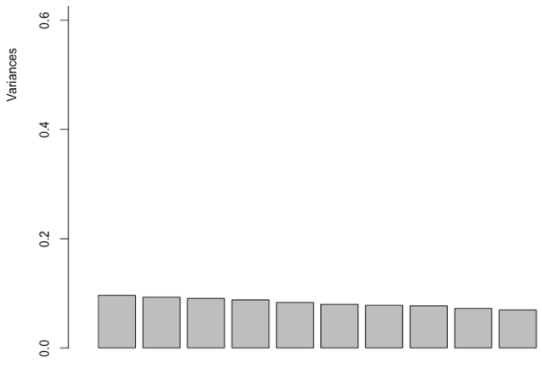


Morchid, M., Dufour, R., Bousquet, P. M., Linares, G., & Torres-Moreno, J. M. (2014). Feature selection using principal component analysis for massive retweet detection. Pattern Recognition Letters, 49, 33-39.

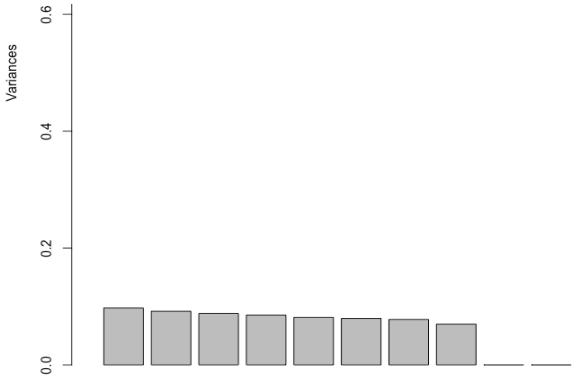
PCA: Example 2

Classroom Exercise

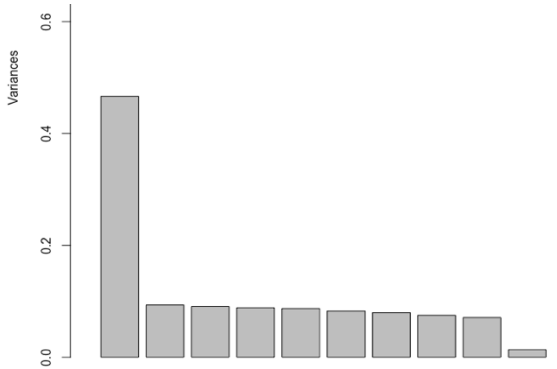
- Consider the following four PCA plots (scree plots). The dataset used for generation of these plots included 10 dimensions and 1000 number of points:



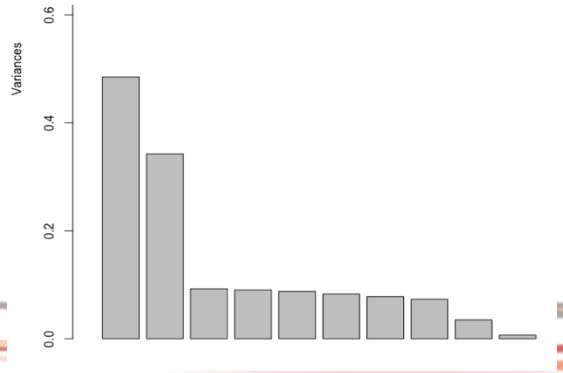
(a)



(b)



(c)



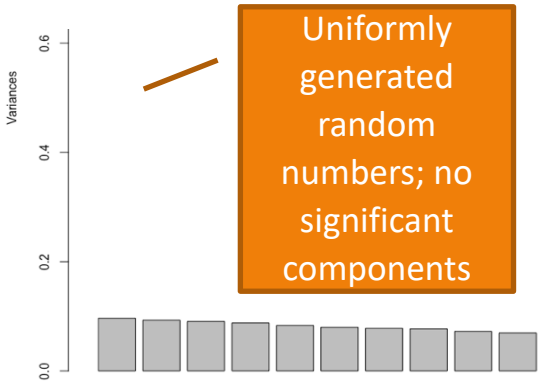
(d)

a. Explain how many components you might select for each PCA result.
b. One of the plots was generated using a dataset where multicollinearity was observed in some dimensions (i) and one was generated using uniformly generated random numbers between 0 and 1 (ii). Match each of these cases with one of the above plots and give your reasons.

PCA: Example 2

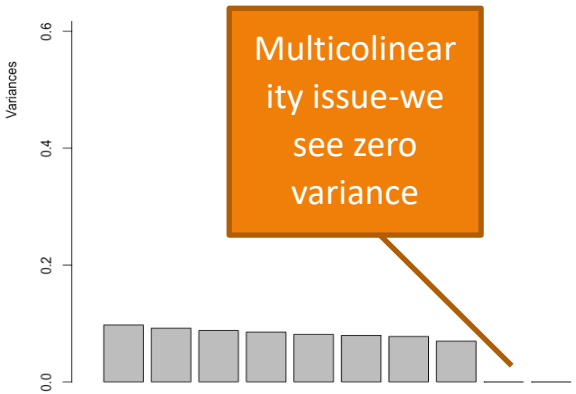
Classroom Exercise

- Consider the following four PCA plots (scree plots). The dataset used for generation of these plots included 10 dimensions and 1000 number of points:



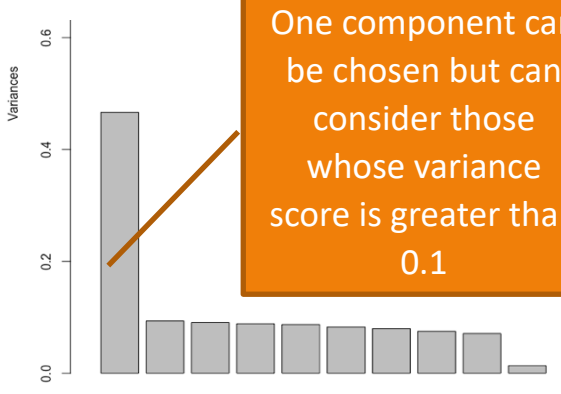
Uniformly generated random numbers; no significant components

(a)



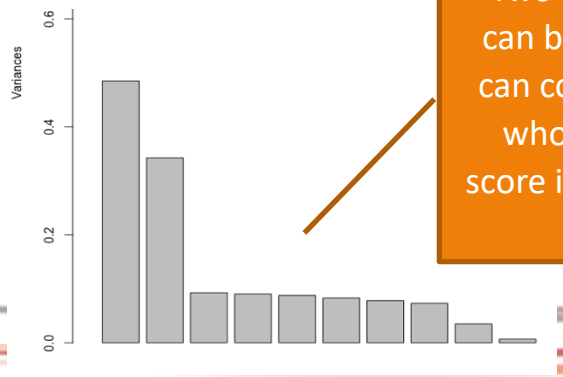
Multicolinearity issue-we see zero variance

(b)



One component can be chosen but can consider those whose variance score is greater than 0.1

(c)



Two components can be chosen but can consider those whose variance score is greater than 0.1

(d)

Attribute Subset Selection

- Another way to reduce dimensionality of data
- Redundant attributes
  - Duplicate much or all of the information contained in one or more other attributes
  - E.g., purchase price of a product and the amount of sales tax paid
- Irrelevant attributes
  - Contain no information that is useful for the data mining task at hand
  - E.g., students' ID is often irrelevant to the task of predicting students' GPA

Attribute Subset Selection

- Many successful methods in the literature e.g. Minimum Redundancy Feature Selection (mRMR), mutual information based criterion...
- Mutual information: $$I(A,B) = \sum_{a,b} p(a,b) \cdot \log\left(\frac{p(a,b)}{p(a)\,p(b)} \right)$$

This definition is related to the Kullback-Leibler distance between two distributions.
Measures the dependence of the two distributions.
In feature selection, choose the features that minimize I(A,B) to ensure they are not related.
Mutual information, and correlation are effective methods but mutual information contains information about all dependence—linear and nonlinear—and not just linear dependence as the correlation coefficient measures.

Attribute Subset Selection

An Example

- Suppose we roll 3 dice (Dice1, Dice2, Dice3)
- Outcomes:

Dice1	Dice2	Dice3
2	2	5
6	6	1
1	1	6
3	3	6
4	4	3
5	5	2
2	2	4
4	4	5
3	3	4
1	1	1
6	6	2
5	5	3

$$I(A,B) = \sum_{a,b} p(a,b) \cdot \log\left(\frac{p(a,b)}{p(a)p(b)}\right)$$

Attribute Subset Selection

An Example

- Dice1 and Dice3 are fair but Dice2 is loaded and it gives exactly the same value as Dice1

Mutual Information (1&2):

A (Dice1)	B (Dice2)	
1	1	0.43082708
2	2	0.43082708
3	3	0.43082708
4	4	0.43082708
5	5	0.43082708
6	6	0.43082708
I(A,B)		2.5849625

Mutual Information (1&3):

A (Dice1)	C (Dice3)	
1	6	0.13208021
1	1	0.13208021
2	5	0.13208021
2	4	0.13208021
3	6	0.13208021
3	4	0.13208021
4	3	0.13208021
4	5	0.13208021
5	3	0.13208021
5	2	0.13208021
6	1	0.13208021
6	2	0.13208021
I(A,C)		1.5849625

The mutual information between Dice1 and Dice2 is higher than the mutual information between Dice1 and Dice3 indicating that Dice1 and Dice2 are highly related.

Attribute Subset Selection

Minimum Redundancy Feature Selection (mRMR)

Maximize relevance according to the target variable based on mutual information and choose a variable where the mutual information between the variable x and the others is the least minimum (minimize redundancy).

- **Minimize Redundancy**:

$$\min W_I, \qquad W_I = \frac{1}{|S|^2} \sum_{i,j \in S} I(i,j)$$

S is the set of features.

$I(i,j)$ is **mutual information** between features i and j

- **Maximize Relevance**:

$$\max V_I, \qquad V_I = \frac{1}{|S|} \sum_{i \in S} I(h,i)$$

h = target classes (e.g. types of different cancers, or annotations)

Additive combination: max(V-W)

Attribute Subset Selection

Minimum Redundancy Feature Selection (mRMR)

- How to interpret?
  - mRMR scores should be sorted in decreasing order.
  - You find a cut off point (the maximum drop out point) and select the features above the point.
- **Example**:

Feature No	(i)	(i)	p-value from (i)
X1	0.887	0.777	< 2.2e-16
X2	-0.031	-0.005	0.32
X3	0.859	0.036	< 2.2e-16
X4	0.005	-0.003	0.85
X5	0.439	0.159	< 2.2e-16
X6	0.439	-0.2855	< 2.2e-16

Given a six-dimensional dataset with all having continuous data types, I obtained the Pearson correlation coefficients between the target variable and each input variable (i) and MRMR results (ii) as can be seen in the above Table where the correlation values between X1 and X3, X5 and X6 are higher than 0.7 whereas the correlations between the other variables are lower than 0.6.

Comment on the results of MRMR and correlation results. Which features can be chosen for modelling? Give your reasons.

Attribute Subset Selection

Minimum Redundancy Feature Selection (mRMR)

- **Example**:

Feature No	(i)	(i)	p-value from (i)
X1	0.887	0.777	< 2.2e-16
X2	-0.031	-0.005	0.32
X3	0.859	0.036	< 2.2e-16
X4	0.005	-0.003	0.85
X5	0.439	0.159	< 2.2e-16
X6	0.439	-0.2855	< 2.2e-16

Regarding the correlation results, we consider those whose p-values are significant such as less than 0.01. The correlation results show the following features are highly correlated with the target variable in descending order: X1, X3, X5, X6. However, for choosing the relevant features, we should not totally rely on the correlation results. Because there could be multicollinearity. Based on the given correlation results between the features, I can select X1 and X5.
MRMR can be more reliable and complementary to be used with correlation results. When I sort the MRMR results in descending order, the most significant features can be selected based on a cut-off point: X1, X5.
We don't choose X2 and X4 based on two analyses. Because they are insignificant. However, note that these analyses consider the linear relationship among the variables.

Data Reduction 2:

Numerosity Reduction

- Reduce data volume by choosing alternative, smaller forms of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Ex.: Regression – estimate the parameters of the model that relates variables in the data.
- Non-parametric methods
  - Do not assume models
  - Major families: histograms, clustering, sampling, …

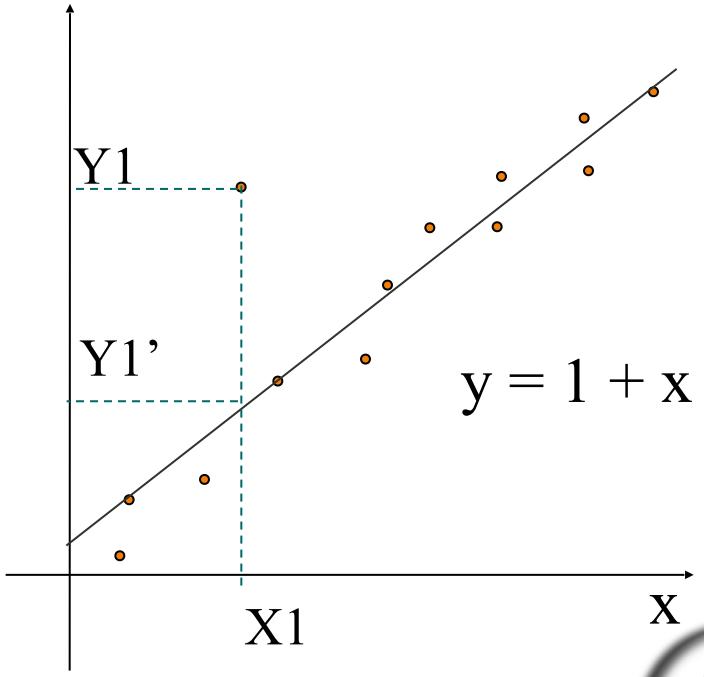
Parametric Data Reduction:

Regression and Log-Linear Models

- Linear regression
  - Data modeled to fit a straight line
  - Often uses the least-square method to fit the line
- Multiple regression
  - Allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- Log-linear model
  - Approximates discrete multidimensional probability distributions

Parametric Data Reduction:
Regression and Log-Linear Models

- Regression analysis: A collective name for techniques for the modeling and analysis of numerical data consisting of values of a dependent variable (also called response variable or measurement) and of one or more independent variables (aka. explanatory variables or predictors)
- The parameters are estimated so as to give a "best fit" of the data
- Most commonly the best fit is evaluated by using the least squares method, but other criteria have also been used
- Used for prediction (including forecasting of time-series data), inference, and hypothesis testing,



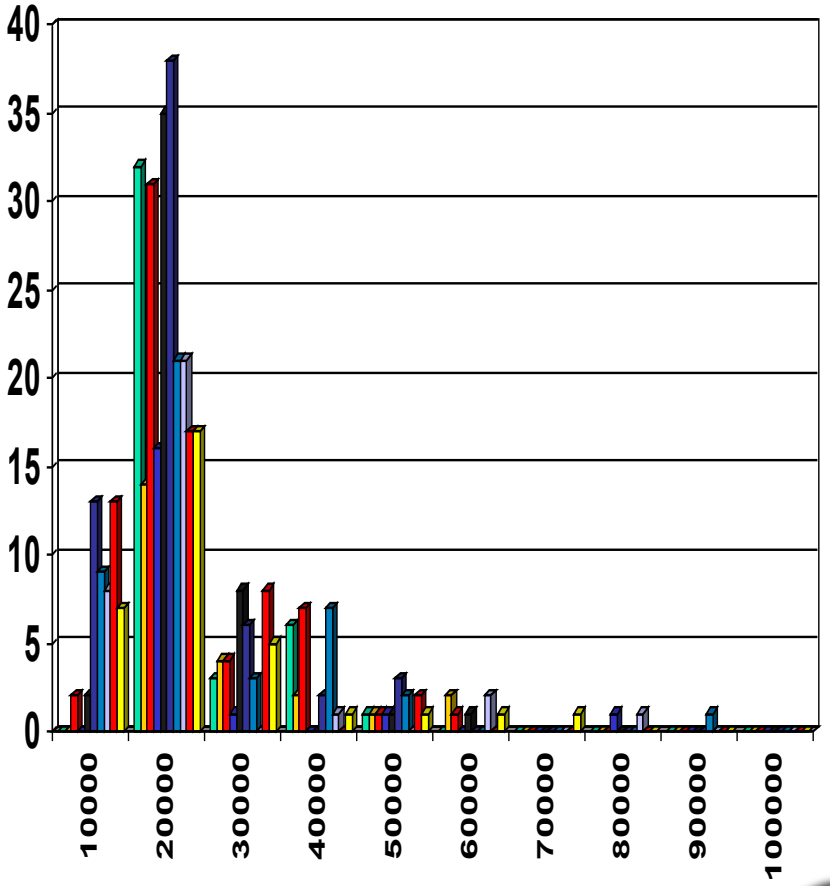
Parametric Data Reduction:

Regression and Log-Linear Models

- Linear regression: $Y = b_0 + b_1 X$
  - Two regression coefficients, b_0 and b_1, specify the line and are to be estimated by using the data at hand (i.e., $Y^{(1)}$, $Y^{(2)}$, …, $X^{(1)}$, $X^{(2)}$, ….) using the least squares criterion
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2 + … + b_n X_n$
  - Multiple independent variables
  - Many nonlinear functions can also be transformed into the above form by using polynomial expansion
- Log-linear models:
  - Approximate discrete multidimensional probability distributions
  - Estimate the probability of each point (tuple) in a multi-dimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations
  - Specify how the cell counts depend on the levels of categorical variables. They model the association and interaction patterns among categorical variables (It is an extension of the familiar chi-square test for independence in two-way contingency tables.)

Histogram Analysis

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)

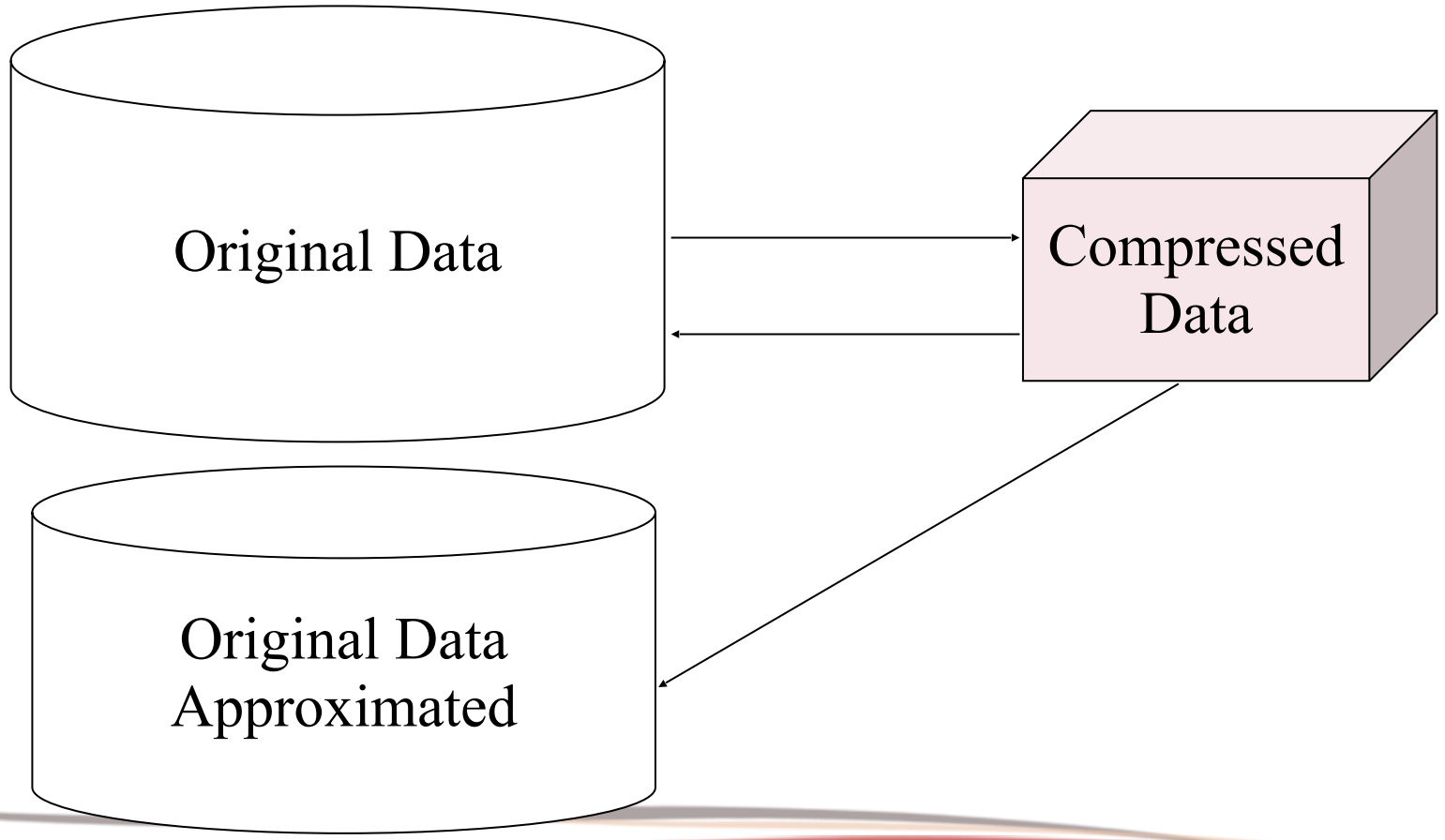


Clustering

- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid and diameter) only
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms

Data Reduction 3:

Compression



Data Reduction 3:

Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless, but only limited manipulation is possible without expansion
- Audio/video compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence (not audio)
  - Typically short and vary slowly with time
- Dimensionality and numerosity reduction may also be considered as forms of data compression

Data Reduction 4:

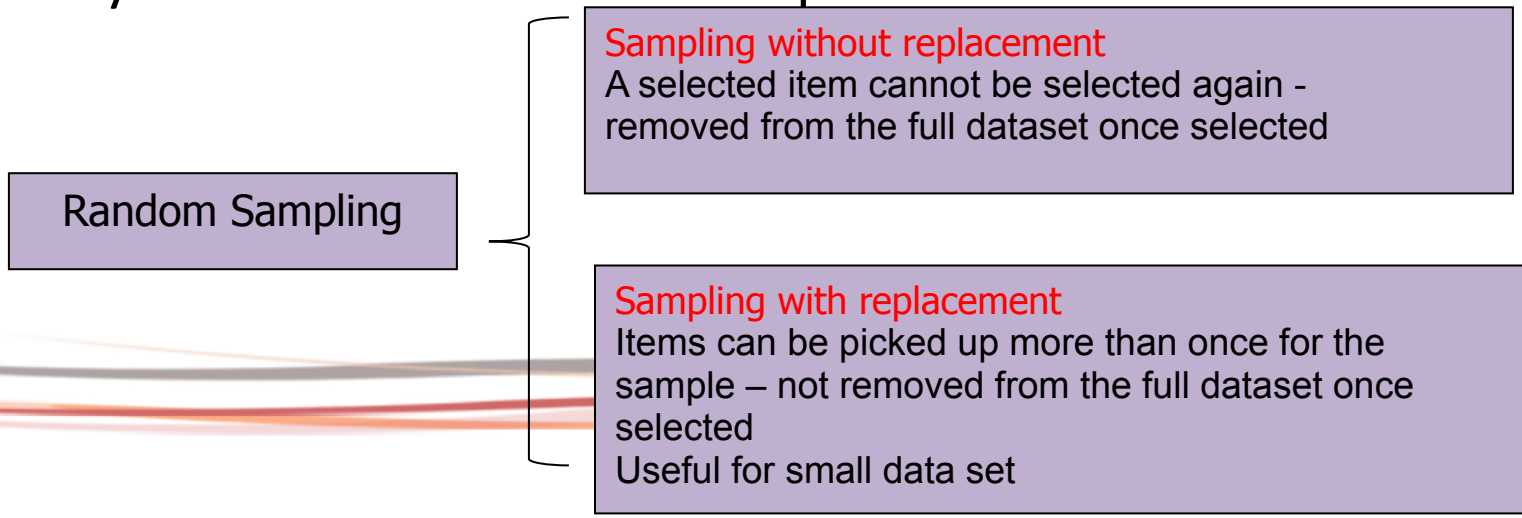
Sampling

- Sampling is a technique employed for selecting a subset of the data
- Why is it used?
  - It may be too expensive or too time consuming to process all data
  - To measure a classifier's performance the data may be divided in a training set and a test set
  - To obtain a better balance between class distributions

Sampling

Simple Random Sampling

- Simple Random Sampling:
  - Every sample of size n has the same chance of being selected
  - Perfect random sampling is difficult to achieve in practice
  - Use random numbers
- Drawback: by bad luck, all examples of a less frequent (rare) class may be missed out in the sample



Sampling

- Stratified sampling
  - Split the data into several partitions (strata); then draw random samples from each partition
  - Each strata may correspond to each of the possible classes in the data
  - The number of items selected from each strata is proportional to the strata size
  - However, stratification provides only a primitive safeguard against uneven representation of classes in a sample

Sampling

- Sampling with skewed classes (imbalanced datasets): Data often include classes including very few data points.
  - Ex: credit card fraud detection, intrusion detection in networks
- Many algorithms are not able to perform well with imbalanced datasets.
- Solutions:
  - Under-sampling balances the dataset by reducing the size of the abundant class.
  - Oversampling balances the dataset by increasing the size of rare examples. Rather than getting rid of abundant samples, new rare samples are generated by using e.g. repetition, bootstrapping or SMOTE (Synthetic Minority Over-Sampling Technique).

Summary

- Data quality: accuracy, completeness, consistency, timeliness, believability, interpretability
- Data cleaning: e.g. missing/noisy values, outliers
- Data integration from multiple sources:
  - Entity identification problem; Remove redundancies; Detect inconsistencies
- Data reduction
  - Dimensionality reduction; Numerosity reduction; Data compression
- Data transformation and data discretization
  - Normalization; Concept hierarchy generation