02450: Introduction to Machine Learning and Data Mining

Data and feature extraction

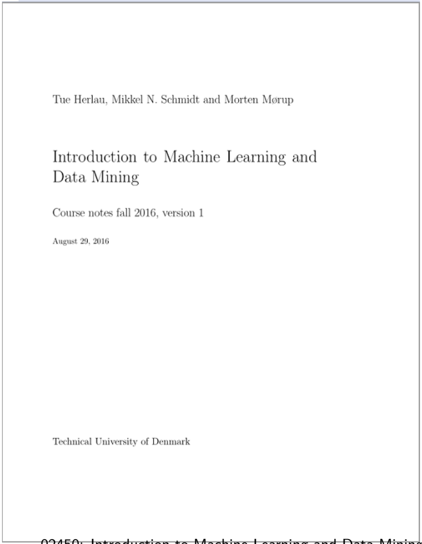
DTU Compute
Department of Applied Mathematics and Computer Science

Reading Material

Reading material:
C2, C3

Feedback Groups of the day:

- Kåre Jørgensen, Mathias Roikjær
- Martin Vieth, Marianne Louis-Hansen, Markus Færevaag


Tue Herlau, Mikkel N. Schmidt and Morten Mørup

Introduction to Machine Learning and Data Mining

Course notes fall 2016, version 1

August 29, 2016

Technical University of Denmark

Lecture Schedule

1. Introduction
   30 August: C1

Data: Feature extraction, and visualization

2. **Data and feature extraction**
   6 September: C2, C3
3. Measures of similarity and summary statistics
   13 September: C4
4. Data Visualization and probability
   20 September: C5, C6

Supervised learning: Classification and regression

5. Decision trees and linear regression
   27 September: TBA **(Project 1 due before 13:00)**
6. Overfitting and performance evaluation
   4 October: TBA
7. Nearest Neighbor, Bayes and Naive Bayes
   11 October: TBA
8. Artificial Neural Networks and Bias/Variance
   25 October: TBA
9. AUC, ensemble methods and multi-class classifiers
   1 November: TBA

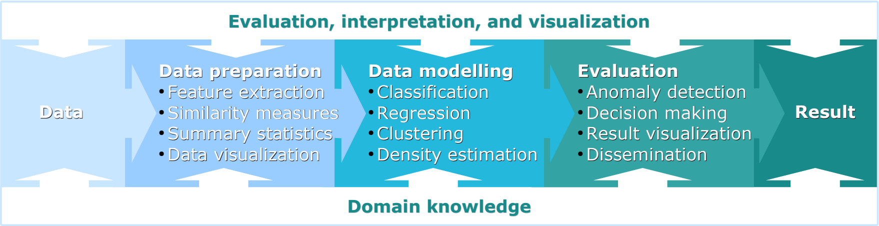
Unsupervised learning: Clustering and density estimation

10. K-means and hierarchical clustering
    8 November: TBA **(Project 2 due before 13:00)**
11. Mixture models and association mining
    15 November: TBA
12. Density estimation and anomaly detection
    22 November: TBA

Recap

13. Recap and discussion of the exam
    29 November: TBA **(Project 3 due before 13:00)**

Data modeling framework



Todays learning objectives:

Understand types of data, their attributes and data issues.
Be able to apply principal component analysis for data visualization and feature extraction.

For online help see Piazza: https://piazza.com/dtu.dk/fall2016/02450

What is data?

- Collection of data objects and their attributes
- An attribute is a property or characteristic of an object
  - Also known as variable, field, characteristic, or feature
- Collection of attributes describe an object
  - Also known as record, point, case, sample, entity, or instance

Attributes

Data objects

ID	Age	Gender	Name
1	31	F	Alex
2	24	M	Ben
3	52	F	Cindy
4	35	M	Dan
5	58	M	Eric
6	46	F	Fay
7	42	M	George

Discrete / continuous attributes

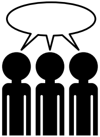
- **Discrete**
  - Finite (or countably infinite) set of values
  - Examples:
    - Zip codes
    - Counts
    - Set of words in a collection of documents
  - Often represented as integer variables
- **Continuous**
  - Has real numbers as attribute values
  - Examples:
    - Temperature
    - Height
    - Weight.
  - Often represented as floating point variables

Types of attributes

- **Nominal:** Objects belong to a category (Equal / Not equal)
  - ID numbers
  - Eye color
  - Zip codes
- **Ordinal:** Objects can be ranked (Greater than / Less than)
  - Taste of potato chips on a scale from 1-10
  - Grades
  - Height in {short, medium, tall}
- **Interval:** Distance between objects can be measured (Addition / Subtraction)
  - Calendar dates
  - Temperature in Fahrenheit and Celcius
- **Ratio:** Zero means absence of what is measured (Multiplication / Division)
  - Length
  - Time
  - Counts
  - Temperature in Kelvin

Qualitative

Quantitative



Discussion

- **Classify the following attributes**
  - a) Military rank
  - b) Angles measured in degrees
  - c) A persons year of birth
  - d) A persons age in years
  - e) Coat check number
  - f) Distance from center of campus
  - g) Number of patients in a hospital
  - h) Sea level

- **Discrete**
  - Finite (or countably infinite) set of values
- **Continuous**
  - Real number

- **Nominal** (Equal / Not equal)
  - Objects belong to a category
- **Ordinal** (Greater than / Less than)
  - Objects can be ranked
- **Interval** (Addition / Subtraction)
  - Distance between objects can be measured
- **Ratio** (Multiplication / Division)
  - Zero means absence of what is measured

Types of data sets

- **Record data**
  - Collection of data objects and their attributes
  - Representation: Table
- **Relational data**
  - Collection of data objects and their relation
  - Representation: Graph
- **Ordered data**
  - Ordered collection of data objects
  - Representation: Sequence

Record data example: Market basket data

- **Transaction data table**

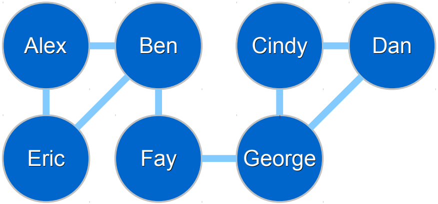
ID	Items
1	Bread, Soda, Milk
2	Beer, Bread
3	Beer, Soda, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Soda, Diaper, Milk

- **Matrix**

ID	Bread	Soda	Milk	Beer	Diaper
1	1	1	1	0	0
2	1	0	0	1	0
3	0	1	1	1	1
4	1	0	1	1	1
5	0	1	1	0	1

Relational data example: Who knows who?

- **Graph**

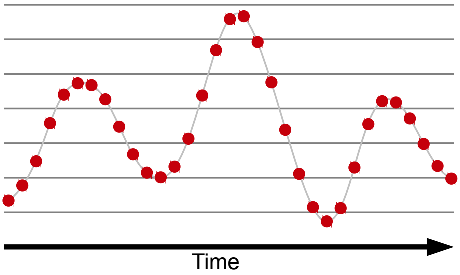


- **Matrix**

	A	B	C	D	E	F	G
A	0	1	0	0	1	0	0
B	1	0	0	0	1	1	0
C	0	0	0	1	0	0	1
D	0	0	1	0	0	0	1
E	1	1	0	0	0	0	0
F	0	1	0	0	0	0	1
G	0	0	1	1	0	1	0

Ordered data example: Time series

- **Sequence**



- **Matrix**

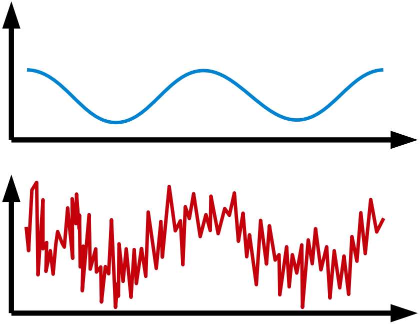
Time	Value
0	1.3
1	1.8
2	2.5
3	3.6
4	4.4
5	4.7
6	4.6
7	4.3
8	2.4
9	2.1
10	2.0
11	2.3
12	3.1

Data quality

- **Data is of high quality if they**
  - Are fit for their intended use
  - Correctly represent the phenomena they correspond to
- **Examples of quality problems**
  - Noise
  - Outliers
  - Missing values

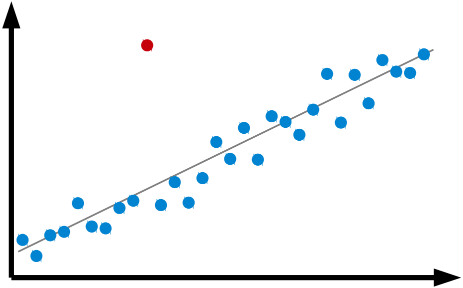
Noise

- **Definition**
  - Unwanted pertubation to a signal
  - Unwanted data
- **Reasons for noise**
  - Limits in measurement accuracy
  - Interference from other signals
  - Measurement of attributes not related to the data modeling task
- **Handling noise**
  - Exclude noisy attributes
  - Remove noise by filtering
  - Include a model of the noise



Outliers

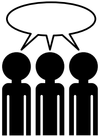
- **Definition**
  - Data objects which are significantly different from most others
- **Reasons for outliers**
  - Measurement error
  - Natural property of data
- **Handling outliers**
  - Identify & exclude outliers
  - Model the outliers



Missing values

- **Definition**
  - No value is stored for an attribute in a data object
- **Reasons for missing values**
  - Information is not collected
    - People decline to give their age
  - Attribute is not applicable
    - Annual income is not applicable to children
- **Handling missing values**
  - Eliminate data objects
  - Estimate missing values (e.g. an average)
  - Ignore the missing value in analysis

ID	Age	Gender	Name
1	31	F	Alex
2	(?)	M	Ben
3	52	F	Cindy
4	35	(?)	Dan
5	(?)	M	Eric
6	(?)	F	Fay
7	42	M	(?)



Discussion

- A group of people were asked to write how many children they have
  - Their response was this

- A research assistant typed the results into a table
  - His table looked like this

Children	3	1	0	2	7	5	15	0	1	3	-2	0	0	0	1

- Are there any data quality issues?
  - Noise?
  - Outliers?
  - Missing values?
- Why have these issues occured, and how should they be handled?

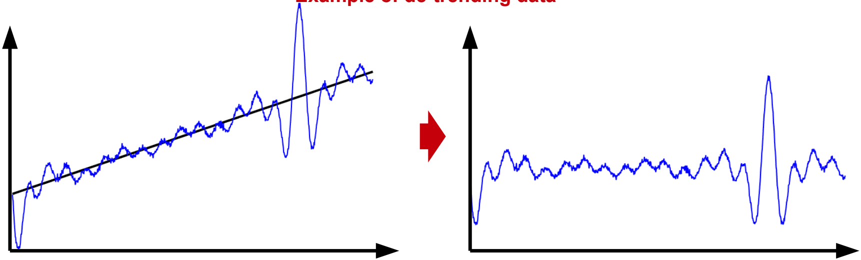
Dataset manipulation

- **Sampling**
  - Selecting a representative subset of data points
- **Feature subset selection**
  - Choose a subset of attributes
- **Feature extraction/transformation**
  - Create new features from existing attributes
  - Discretization and binarization
  - Apply a fixed transformation to an attribute
  - Aggregation several attributes into a single attribute
- **Dimensionality reduction**
  - Project data to a low-dimensional subspace **(PCA in lecture 2)**

Feature processing

- Eliminating, suppressing, or attenuating certain aspects of the data
  - Noise removal in audio signals
  - Elimination of common words in text documents
  - Removal of background in images
  - Removal of examples which are corrupted
  - De-trending data (if it is not stationary)

Example of de-trending data



Common feature transformations

ID	MPG	Cylinders	Horsepower	Weight	Year	Safety	Acceleration	Origin
1	18	8	150	3436	70	4	11	France
2	28	4	79	2625	82	4	18.6	USA
3	26	4	79	2255	76	3	17.7	USA
3	29	4	70	1937	76	1	14.2	Germany
4	NaN	8	175	3850	70	2	11	USA
5	24	4	90	2430	70	3	14.5	Germany
6	17.5	6	95	3193	76	4	17.8	USA
7	25	4	87	2672	70	-100	17.5	France
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
142	15	8	198	4341	70	2	10	USA

$$\boldsymbol{X} = \begin{bmatrix} 18 & 8 & 150 & 3436 & 70 & 4 & 11 & 3 \\ 28 & 4 & 79 & 2625 & 82 & 4 & 18.6 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 15 & 8 & 198 & 4341 & 70 & 2 & 10 & 1 \end{bmatrix}$$

Standardize:

$$\boldsymbol{X} = \begin{bmatrix} \cdots & (X_{1j} - \hat{\mu}_j)/\hat{\sigma}_j & \cdots \\ \cdots & (X_{2j} - \hat{\mu}_j)/\hat{\sigma}_j & \cdots \\ & \vdots & \\ \cdots & (X_{Nj} - \hat{\mu}_j)/\hat{\sigma}_j & \cdots \end{bmatrix}$$

$$\hat{\mu}_j = \frac{1}{N}\sum_{i=1}^N X_{ij}, \quad \hat{\sigma}_j = \sqrt{\frac{1}{N-1}\sum_{i=1}^N (X_{ij} - \mu_j)^2}$$

Binarize/threshold:

$$\boldsymbol{X} = \begin{bmatrix} \cdots & 1_{[\theta,\infty[}(x_{1j}) & \cdots \\ \cdots & 1_{[\theta,\infty[}(x_{2j}) & \cdots \\ & \vdots & \\ \cdots & 1_{[\theta,\infty[}(x_{Nj}) & \cdots \end{bmatrix}$$

$$1_{[\theta,\infty[}(x) = 1 \text{ if } x \geq \theta \text{ otherwise } 0$$

One-out-of-K coding

One-out-of K coding

X=

Age	Height	Weight	Nationality
-0.2248	-0.4762	-0.2097	'Sweden'
-0.5890	0.8620	0.6252	'Sweden'
-0.2938	-1.3617	0.1832	'Sweden'
-0.8479	0.4550	-1.0298	'Sweden'
-1.1201	-0.8487	0.9492	'Norway'
2.5260	-0.3349	0.3071	'Norway'
1.6555	0.5528	0.1352	'Norway'
0.3075	1.0391	0.5152	'Norway'
-1.2571	-1.1176	0.2614	'Norway'
-0.8655	1.2607	-0.9415	'Sweden'
-0.1765	0.6601	-0.1623	'Norway'
0.7914	-0.0679	-0.1461	'Denmark'
-1.3320	-0.1952	-0.5320	'Denmark'
-2.3299	-0.2176	1.6821	'Sweden'
-1.4491	-0.3031	-0.8757	'Sweden'
0.3335	0.0230	-0.4838	'Sweden'
0.3914	0.0513	-0.7120	'Denmark'
0.4517	0.8261	-1.1742	'Sweden'
-0.1303	1.5270	-0.1922	'Norway'
0.1837	0.4669	-0.2741	'Denmark'

Age	Height	Weight	Denmark	Norway	Sweden
-0.2248	-0.4762	-0.2097	0	0	1
-0.5890	0.8620	0.6252	0	0	1
-0.2938	-1.3617	0.1832	0	0	1
-0.8479	0.4550	-1.0298	0	0	1
-1.1201	-0.8487	0.9492	0	1	0
2.5260	-0.3349	0.3071	0	1	0
1.6555	0.5528	0.1352	0	1	0
0.3075	1.0391	0.5152	0	1	0
-1.2571	-1.1176	0.2614	0	1	0
-0.8655	1.2607	-0.9415	0	0	1
-0.1765	0.6601	-0.1623	0	1	0
0.7914	-0.0679	-0.1461	1	0	0
-1.3320	-0.1952	-0.5320	1	0	0
-2.3299	-0.2176	1.6821	0	0	1
-1.4491	-0.3031	-0.8757	0	0	1
0.3335	0.0230	-0.4838	0	0	1
0.3914	0.0513	-0.7120	1	0	0
0.4517	0.8261	-1.1742	0	0	1
-0.1303	1.5270	-0.1922	0	1	0
0.1837	0.4669	-0.2741	1	0	0

Bag of words

- First three sentences on **wikipedia.org**
  - The bag-of-words model is a simplifying assumption used in natural language processing and information retrieval
  - In this model, a text (such as a sentence or a document) is represented as an unordered collection of words, disregarding grammar and even word order
  - The bag-of-words model is used in some methods of document classification

(Image source: https://pixabay.com/p-297223/)

Bag of words

- First three sentences on **wikipedia.org**
  - The bag-of-words model is a simplifying assumption used in natural language processing and information retrieval
  - In this model, a text (such as a sentence or a document) is represented as an unordered collection of words, disregarding grammar and even word order
  - The bag-of-words model is used in some methods of document classification

- We will treat **this text** as a data set and create a bag-of-words model of it

Bag of words

- Elimination of common words (so-called stop words)
  - The bag-of-words model is a simplifying assumption used in natural language processing and information retrieval
  - In this model, a text (such as a sentence or a document) is represented as an unordered collection of words, disregarding grammar and even word order
  - The bag-of-words model is used in some methods of document classification

Bag of words

- Representation as matrix

Word	Sentence 1	Sentence 2	Sentence 3
bag-of-words	1		1
model	1	1	1
simplifying	1		
assumption	1		
natural	1		
language	1		
processing	1		
information	1		
retrieval	1		
text		1	
sentence		1	
document		1	1
represented		1	
unordered		1	
collection		1	
words		1	
disregarding		1	
grammar		1	
word		1	
order		1	
methods			1
classification			1

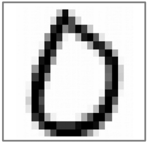
Bag of words

- Stemming

Word	Sentence 1	2	3
bag-of-word*	1		1
model*	1	1	1
simplif*	1		
assum*	1		
natural*	1		
languag*	1		
process*	1		
information*	1		
retriev*	1		
text*		1	
sentence*		1	
document*		1	1
represent*		1	
unorder*		1	
collect*		1	
word*		2	
disregard*		1	
grammar*		1	
order*		1	
method*			1
classif*			1

Image representation

- **Example: Handwritten digits**
- Preprocessing
  - Digitalization
  - Centering
  - Rotation
  - Scaling



28×28

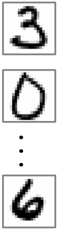
$$M_0 = \begin{bmatrix} 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0.3 & 1 & 0.2 & 0 & \dots & 0 \\ \vdots & & & & & & & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

- Vectorization 1×784

$$x_0 = \begin{bmatrix} 0 & \cdots\cdots & 0 & 0.3 & 1 & 0.2 & 0 & \cdots\cdots & 0 \end{bmatrix}^\top$$

- Matrix representation of data set

$$\boldsymbol{X} = \begin{bmatrix} \text{———} & \boldsymbol{x}_1 & \text{———} \\ \text{———} & \boldsymbol{x}_2 & \text{———} \\ & \vdots & \\ \text{———} & \boldsymbol{x}_N & \text{———} \end{bmatrix}$$

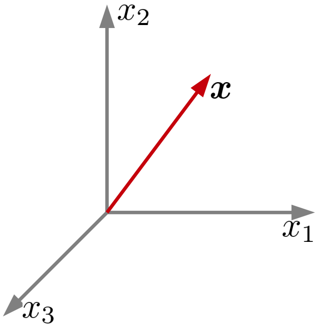


If each image is 28×28 pixels then $\boldsymbol{X}$ is a $N \times 784$ matrix.

Vector space representation

- All these data objects have a vector space representation

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix}$$



Plan for the rest of today:

- Data represented as vectors and matrices
  - Linear algebra useful for manipulating and analyzing data
- We will derive the Principal Component Analysis (PCA) and discuss the Singular Value Decomposition (SVD)
  - First a (brief) highlight of linear algebra
  - PCA is very important for data visualization

Vectors and matrices

- Common matrix notation

$$\boldsymbol{A}, A, \overline{\overline{A}}$$

$$\boldsymbol{A} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,M} \\ \vdots & & \vdots \\ a_{N,1} & \cdots & a_{N,M} \end{bmatrix} \in \mathbb{R}^{N \times M}$$

- Common vector notation

$$\boldsymbol{x}, x, \overline{x}, \vec{x}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^{N}$$

Matrix Multiplication

- Two matrices can be multiplied $AB = C$
  - if the number of columns in the first equals the number of rows in the second

$$\underset{L \times M}{A} \times \underset{M \times N}{B} = \underset{L \times N}{C}$$

$$\overset{3\times 4 \text{ matrix}}{\begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \color{blue}{1} & \color{blue}{2} & \color{blue}{3} & \color{blue}{4} \end{bmatrix}} \overset{4\times 5 \text{ matrix}}{\begin{bmatrix} \cdot & \cdot & \cdot & \color{red}{a} & \cdot \\ \cdot & \cdot & \cdot & \color{red}{b} & \cdot \\ \cdot & \cdot & \cdot & \color{red}{c} & \cdot \\ \cdot & \cdot & \cdot & \color{red}{d} & \cdot \end{bmatrix}} = \overset{3\times 5 \text{ matrix}}{\begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & x_{3,4} & \cdot \end{bmatrix}}$$

$$x_{3,4} = \color{blue}{1} \cdot \color{red}{a} + \color{blue}{2} \cdot \color{red}{b} + \color{blue}{3} \cdot \color{red}{c} + \color{blue}{4} \cdot \color{red}{d}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = ?$$

Matrix transpose

- The transpose of a matrix

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 4 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad \boldsymbol{A}^\top = \begin{bmatrix} 1 & 3 & 7 \\ 2 & 4 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

- Transpose of a sum

$$(\boldsymbol{A} + \boldsymbol{B})^\top = \boldsymbol{A}^\top + \boldsymbol{B}^\top$$

- Transpose of a product

$$(\boldsymbol{A}\boldsymbol{B})^\top = \boldsymbol{B}^\top \boldsymbol{A}^\top$$

$$(\boldsymbol{A}\boldsymbol{x})^\top \boldsymbol{y} = \boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{y} = \boldsymbol{x}^\top (\boldsymbol{A}^\top \boldsymbol{y})$$

The identity matrix

- Ones on the diagonal and zeros everywhere else

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \qquad \boldsymbol{I}^\top = \boldsymbol{I}$$

- Multiplying by the identity does not change anything

$$\boldsymbol{I}\boldsymbol{A} = \boldsymbol{A}$$

$$\boldsymbol{I}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\boldsymbol{I}_2\boldsymbol{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Matrix inverses

- For a square matrix, the inverse satisfies

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{I}$$

- Inverse of a product of square matrices

$$(\boldsymbol{A}\boldsymbol{B})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$$

- Transpose of inverse

$$(\boldsymbol{A}^{-1})^{\top} = (\boldsymbol{A}^{\top})^{-1}$$

Norms

- The (Euclidian) norm of a vector measures it's length (magnitude):

$$\|\boldsymbol{x}\|_2 = \sqrt{\sum_{n=1}^{N} |x_n|^2} = \sqrt{\boldsymbol{x}^\top \boldsymbol{x}}$$

- The Frobenius norm of a matrix measures it's magnitude:

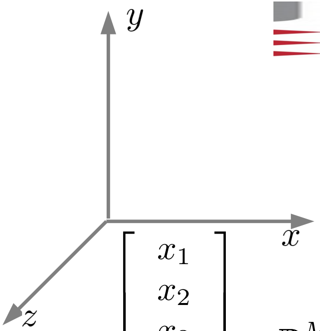
$$\|\boldsymbol{X}\|_F^2 = \sum_{i,j} x_{i,j}^2 = \mathrm{trace}(\boldsymbol{X}\boldsymbol{X}^T) = \mathrm{trace}(\boldsymbol{X}^T\boldsymbol{X})$$

Where trace takes the sum of the diagonal elements, i.e. $\mathrm{trace}(\boldsymbol{A}) = \sum_{i=1}^{N} a_{i,i}$

Vector spaces

- A M-dimensional vector space is just $\mathbb{R}^M$
- This is the set of all M-dimensional vectors
- A vector space is closed under **linear combinations**

$$a_1\boldsymbol{x}_1 + a_2\boldsymbol{x}_2 + \cdots + a_n\boldsymbol{x}_n$$



$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^M$$

$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$	Vectors
$\mathrm{a}_1, \ldots, a_n$	Numbers

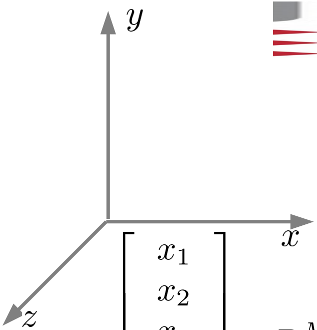
Vector spaces

- A M-dimensional vector space is just $\mathbb{R}^M$
- This is the set of all M-dimensional vectors
- A vector space is closed under **linear combinations**

$$a_1\boldsymbol{x}_1 + a_2\boldsymbol{x}_2 + \cdots + a_n\boldsymbol{x}_n$$

$\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ Vectors

$a_1, \ldots, a_n$ Numbers



$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^M$$

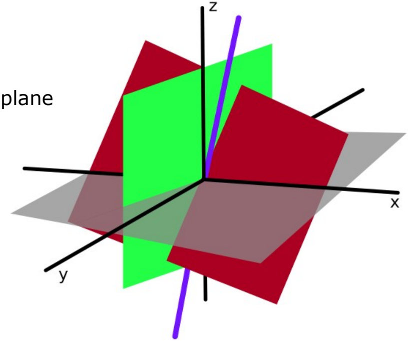
Subspaces

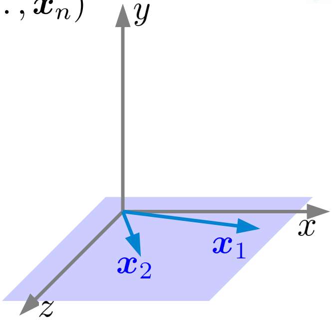
- A **subspace** generalizes the concept of a line/plane
- If we consider n vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ the **span** is then all linear combinations

$$\boldsymbol{z} = a_1\boldsymbol{x}_1 + a_2\boldsymbol{x}_2 + \cdots + a_n\boldsymbol{x}_n$$

and it is said to be a **subspace**

$$\mathrm{V} = \mathrm{span}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$$





Basis of a (sub)space

- Vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are said to be **linearly independent** if

$$\mathbf{0} = a_1\boldsymbol{x}_1 + a_2\boldsymbol{x}_2 + \cdots + a_n\boldsymbol{x}_n$$

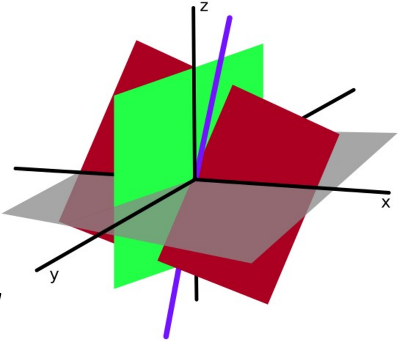
implies $a_1 = a_2 = \cdots = a_n = 0$

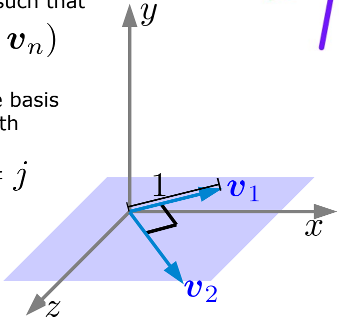
- A **basis** of a vector space V are n linearly independent vectors such that

$$\mathrm{V} = \mathrm{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$$

- A basis is **orthonormal** if the basis is orthogonal and of unit length

$$\boldsymbol{v}_i^T \boldsymbol{v}_j = 0 \text{ for } i \neq j$$
$$\|\boldsymbol{v}_i\| = 1$$





Basis of a (sub)space

- A **basis** of a vector space V are n linearly independent vectors such that

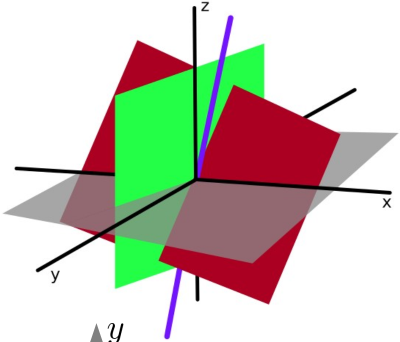
$$\mathrm{V} = \mathrm{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$$

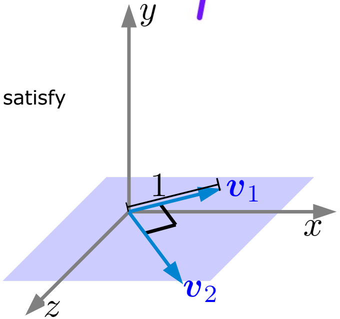
- We collect the basis into a matrix

$$\boldsymbol{V} = \begin{bmatrix} | & | & & | \\ \boldsymbol{v}_1 & \boldsymbol{v}_2 & \text{---} & \boldsymbol{v}_N \\ | & | & & | \end{bmatrix}$$

- If the basis is orthonormal the matrix satisfy

$$\boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I}, \quad \boldsymbol{V}^\top = \boldsymbol{V}^{-1}$$





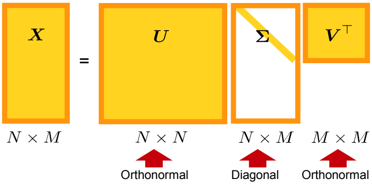
Visualization of hand written digits

- Data matrix

$$\boldsymbol{X} = \begin{bmatrix} — & \boldsymbol{x}_1 & — \\ — & \boldsymbol{x}_2 & — \\ & \vdots & \\ — & \boldsymbol{x}_N & — \end{bmatrix}$$

If each image is 28 x 28 pixels then X is a N x 784 matrix

- Principal component analysis



Eigenvectors and eigenvalues

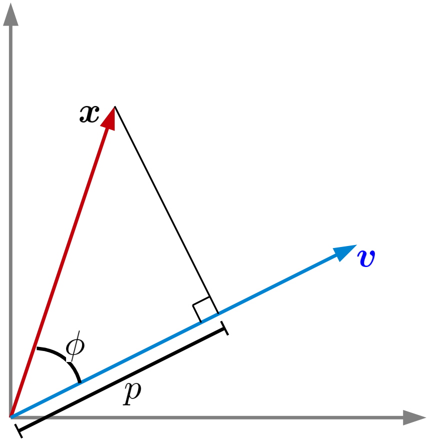
- Suppose

$$\boldsymbol{A}\boldsymbol{v} = \lambda \boldsymbol{v}, \qquad \boldsymbol{A} \text{ is a } N \times N \text{ matrix}$$

  - We say $\boldsymbol{v}$ is an eigenvector with eigenvalue λ
  - If $\boldsymbol{A}$ is symmetric: $\boldsymbol{A} = \boldsymbol{A}^\top$ then $\boldsymbol{A}$ has N linearly independent eigenvectors with positive eigenvalues

Projection

- Projection onto a vector



- Angle between vectors

$$\cos(\phi) = \frac{\boldsymbol{v}^\top \boldsymbol{x}}{\|\boldsymbol{x}\|_2 \|\boldsymbol{v}\|_2}$$

- Length of projection

$$p = \|\boldsymbol{x}\|_2 \cos(\phi) = \frac{\boldsymbol{v}^\top \boldsymbol{x}}{\|\boldsymbol{v}\|_2}$$

- Projection onto unit vector

$$p = \boldsymbol{v}^\top \boldsymbol{x}$$

Projection onto subspace

- **Projection onto a subspace**
  - Subspace of dimension n defined by a orthonormal basis matrix $\boldsymbol{V}$
  - Projection of $\boldsymbol{x}$ (M dimensional) onto V given by

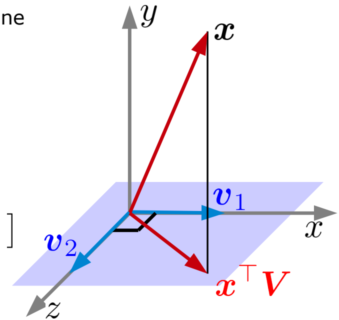
$$\boldsymbol{b}^T = \boldsymbol{x}^T \boldsymbol{V}$$

  - 'Reconstruction' can be found as: $\boldsymbol{x}' = \boldsymbol{V}\boldsymbol{b}$

Example: Projection of 3-D vector onto the (x,z) plane

$$\boldsymbol{V} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad \boldsymbol{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\boldsymbol{x}^\top V = \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} x & z \end{bmatrix}$$



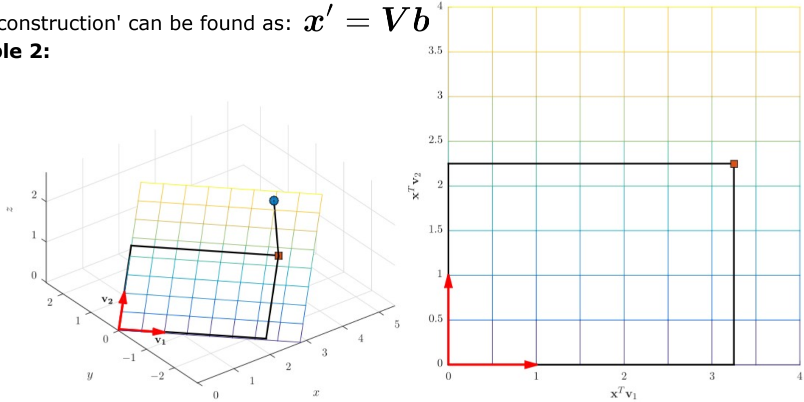
Projection onto subspace

- **Projection onto a subspace**
  - Subspace of dimension n defined by a orthonormal basis matrix $\boldsymbol{V}$
  - Projection of $\boldsymbol{x}$ (M dimensional) onto V given by

$$\boldsymbol{b}^T = \boldsymbol{x}^T \boldsymbol{V}$$

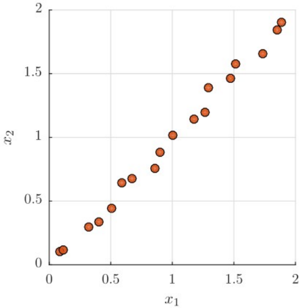
  - 'Reconstruction' can be found as: $\boldsymbol{x}' = \boldsymbol{V}\boldsymbol{b}$

Example 2:

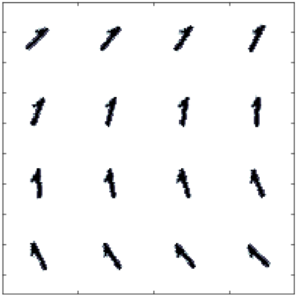


PCA for high dimensional data

- Much data is high-dimensional
- We want to find a **lower**-dimensional representation of the **high**-dimensional data



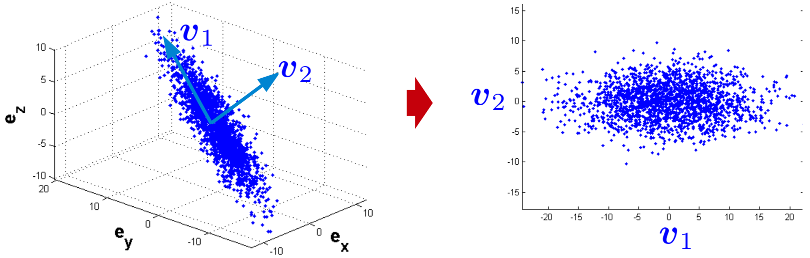
(2 dimensional but really 1 dimensional)



(784 dimensional but really 1 dimensional)

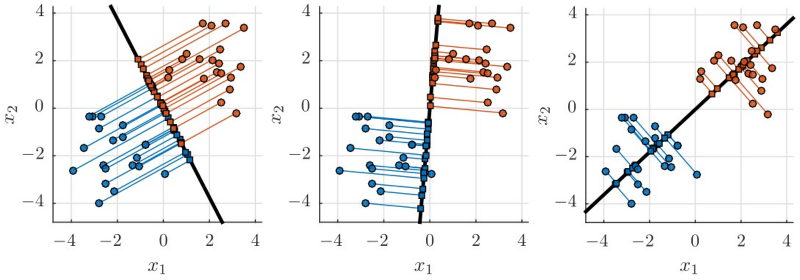
PCA for high dimensional data

- Much data is high-dimensional
- We can **project high** dimensional data to a **lower** dimensional **subspace**
- But what is a good projection?



PCA for high dimensional data

- Much data is high-dimensional
- We can project high dimensional data to a lower dimensional subspace
- But what is a good projection?
- Select projection that maximizes the variance of the projected data

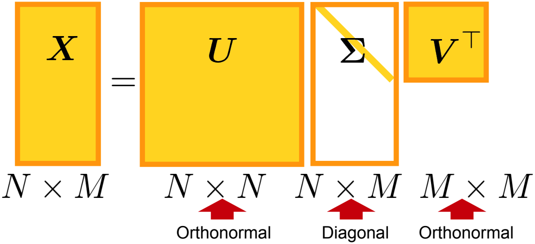


Singular Value Decomposition (SVD)

(Eugino Beltrami & Camille Jordan, independently, 1873-1874)

Any N x M matrix can be decomposed as follows:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$$



$\sigma_1, \ldots, \sigma_M$ is known as the singular values

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_M \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_N \end{bmatrix} \qquad \boldsymbol{V} = \begin{bmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_M \end{bmatrix}$$

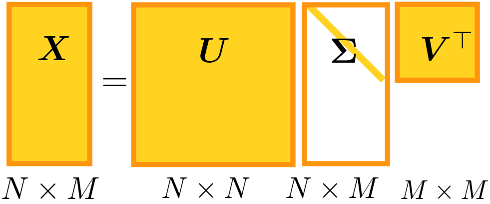
$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_M \geq 0$$

$$\text{if } i \neq j\text{:}\ \boldsymbol{\Sigma}_{i,j} = 0, \quad \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_{N\times N}, \quad \boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I}_{M\times M}$$

Principal component analysis (PCA)

(Karl Pearson, 1901)

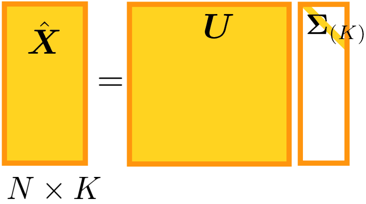
1) Subtract the mean from each attribute
2) Apply singular value decomposition (SVD) $\boldsymbol{X} = \boldsymbol{U\Sigma V}^\top$



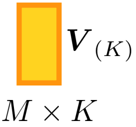
3) Select first K columns of $\boldsymbol{V}$ (the PCA projection operation) and first K columns of $\boldsymbol{\Sigma}$.

$$\boldsymbol{V}_{(K)} = \begin{bmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_K \end{bmatrix}$$

$$(\boldsymbol{X}^\top \boldsymbol{X})\boldsymbol{v}_k = \sigma_k^2 \boldsymbol{v}_k$$



(PCA components or PCA projection of the data)



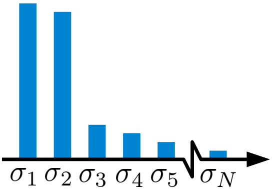
(PCA loadings)

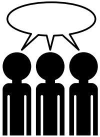
Principal component analysis $X = U\Sigma V^\top$

- Entries in the diagonal matrix Σ are called **singular values**
  - They are sorted (largest first)
  - Indicate how much variability is explained by the corresponding component
    - 1st component explains most of the variability
    - 2nd component explains most of the remaining variability
    - Etc.

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_N \end{bmatrix} \qquad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N \geq 0$$

- Singular value spectrum





- Show that

$$\|\boldsymbol{X}\|_F^2 = \sum_i \sigma_i^2$$

where

$$\sigma_i = \boldsymbol{\Sigma}_{i,i}$$

Fraction of the variation in the data explained by the i[th] principal component is given by:

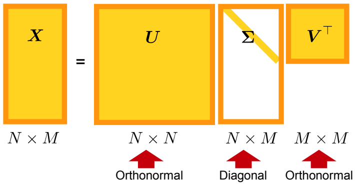
$$\frac{\sigma_i^2}{\sum_j \sigma_j^2}$$

And by the first K principal components

$$\frac{\sum_{i=1}^K \sigma_i^2}{\sum_j \sigma_j^2}$$

Hints:

$$\boldsymbol{X} = \boldsymbol{U\Sigma V}^\top$$



$$\|\boldsymbol{X}\|_F^2 = \mathrm{trace}(\boldsymbol{X}\boldsymbol{X}^\top)$$

$$\mathrm{trace}(\boldsymbol{AB}) = \mathrm{trace}(\boldsymbol{BA})$$

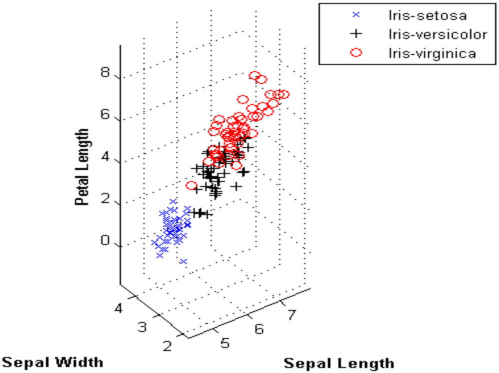
Fishers Iris Data

Three types of flowers:
Iris Setosa, Iris Versicolor, Iris Virginica

Flower ID	Attribute			
	Sepal Length	Sepal Width	Petal Length	Petal Width
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
3	4.7	3.2	1.3	0.2
4	4.6	3.1	1.5	0.2
.	.	.	.	.
.	.	.	.	.
150	5.9	3.0	5.1	1.8

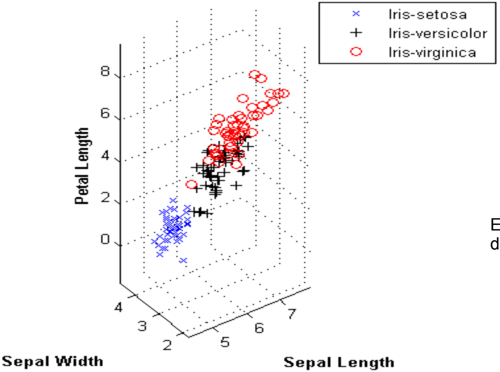
We will presently consider the first 3 attributes, i.e. Sepal length, Sepal Width and Petal Length.

3D scatter plot of the data



What fraction of the total variation in the data will the first principal component account for?

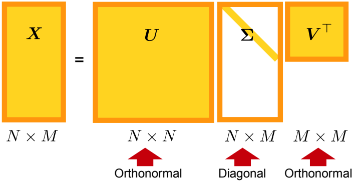
3D scatter plot of the data



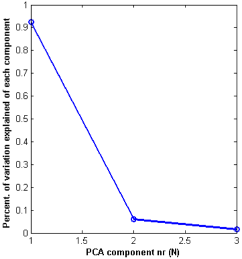
What fraction of the total variation in the data will the first principal component account for?

1) Subtract the mean
2) Apply singular value decomposition (SVD)

$$X = U\Sigma V^\top$$

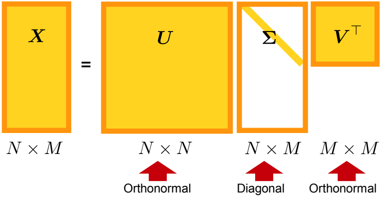


Evaluate the singular values to determine how much of the dynamics is lost when reducing the dimensionality

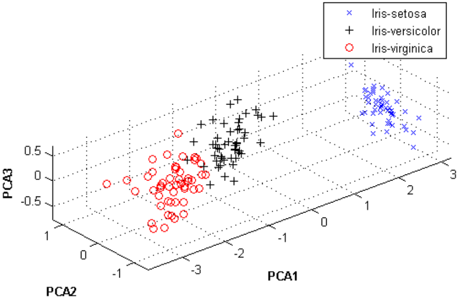


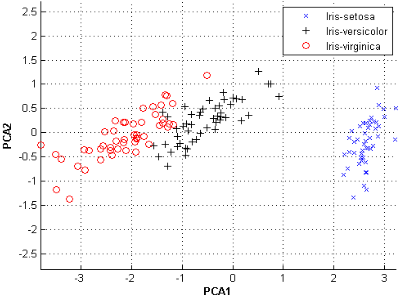
Visualizing the data projected onto the space of the principal components

$$X = U\Sigma V^\top$$



$$PCA1: \boldsymbol{p}_1 = \boldsymbol{X}\boldsymbol{v}_1 = \boldsymbol{u}_1 \sigma_1$$
$$PCA2: \boldsymbol{p}_2 = \boldsymbol{X}\boldsymbol{v}_2 = \boldsymbol{u}_2 \sigma_2$$
$$PCA3: \boldsymbol{p}_3 = \boldsymbol{X}\boldsymbol{v}_3 = \boldsymbol{u}_3 \sigma_3$$





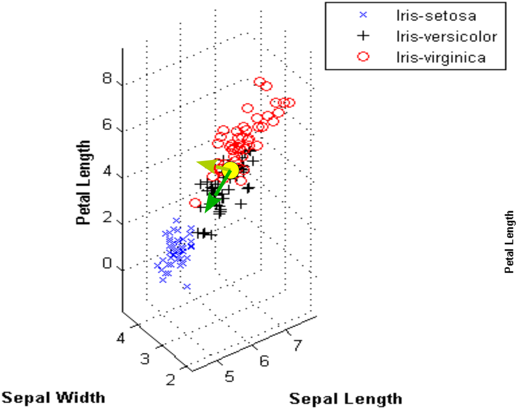
The principal directions V

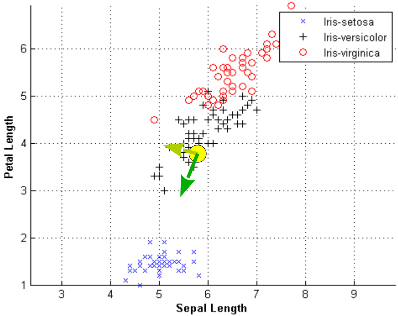
Sepal Length
Sepal Width
Petal Length

$$V = \begin{bmatrix} -0.39 & -0.64 & -0.66 \\ 0.09 & -0.74 & 0.66 \\ -0.92 & 0.20 & 0.35 \end{bmatrix}$$

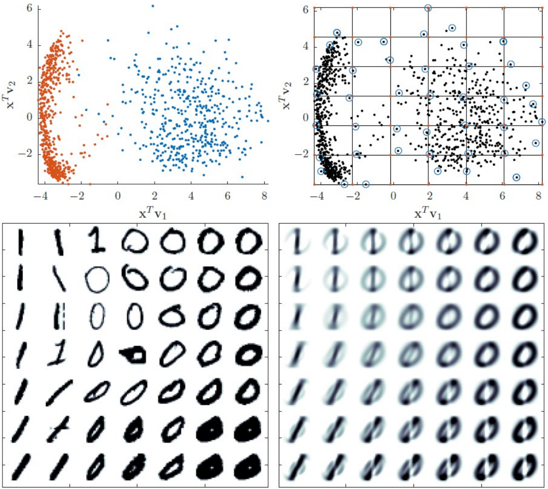
$$\boldsymbol{\mu} = \begin{bmatrix} 5.8 \\ 3.1 \\ 3.8 \end{bmatrix}, \quad \boldsymbol{v}_1 = \begin{bmatrix} -0.39 \\ 0.09 \\ -0.92 \end{bmatrix}, \quad \boldsymbol{v}_2 = \begin{bmatrix} -0.64 \\ -0.74 \\ 0.20 \end{bmatrix}$$

Sepal Length
Sepal Width
Petal Length





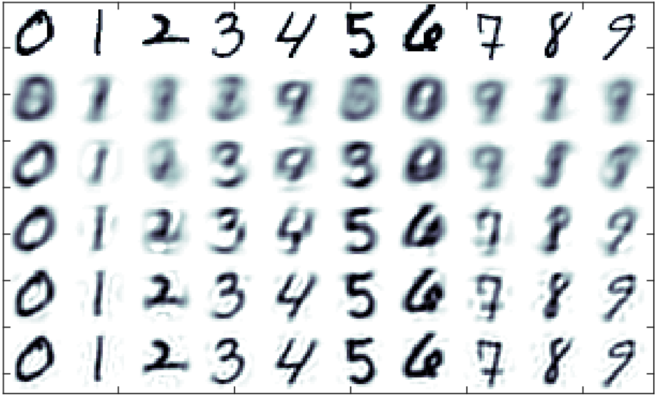
Visualization of hand written digits



Compression

Only include a few components: $\boldsymbol{x}' = \boldsymbol{V}\boldsymbol{b}$

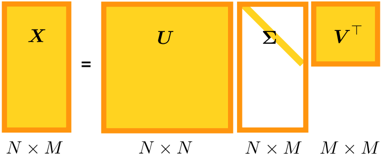
n=2,5,20,50,100



Data and Domain driven feature extraction

PCA is an example of a data driven approach for feature extraction

i.e., we define from data the features extracted in terms of the projections $V^{(PCA)}$ that preserve most of the variance in the data



The fourier transform is an example of a domain driven approach for feature extraction

i.e., *in the analysis of sound good features are often to use spectral representations. These can be found by projecting the data using the so-called fourier transform matrix $V^{(FFT)}$ where the components are defined as specific frequencies such that the projection of the data onto these frequencies defines the extend to which these frequencies are present in the data. (you can learn much more about this in 02451 Digital Signal Processing)*

$V^{(\mathrm{FFT})} =$