Data Science @ d-fine:
Machine Learning, Text Analytics and Networks

XXXIX Heidelberg Physics Graduate Days

Heidelberg, October 10th, 2017

Introduction

Ferdinand Graf

- Manager (since 2011-11 working for d-fine)
- PhD in finance, diploma in mathematical finance (both @UNKN), and GARP financial risk manager
- Expert in rating model development and data-science
- Establishing 'text analytics' in d-fine's project portfolio

Ulf Menzler

- Senior Consultant (since 2015-05 working for d-fine)
- PhD in theoretical astrophysics @ Ruhr-University Bochum
- Project experience in Credit Risk and Machine Learning
- Currently part-time student in mathematical finance at Oxford University

Agenda

Initial setup

Digitalization becomes more important for financial corporations

Industrial changes

1. New ways of doing business (e.g. mobile devices)
2. New data awareness on both customer, institution and regulator side
3. Regulatory pressure (e.g. fraud detection, EWS)
4. Decreasing margins (e.g. due to negative / zero interest rates)
5. New entrants (e.g. digital comp. apply for banking license, crowd funding)

Digitalization

Technological changes

- Higher computing power a
- Higher data availability b
- Enhancements in methods c
- Freely available methods d
- Availability of standard-frameworks e

The market position and the profitability of banks can be improved by using new technologies and non-traditional data.

d-fine

The progressing digitalization in the financial industry increased the demand for data-driven solutions

Customer management

- Acquisition of new customers
- Improvement of customer satisfaction and loyalty
- Segmentation of customers e.g. for personalized advertising
- Strategy development for cross- and upselling

Product management

- Optimization of the product portfolio
- Product engineering
- Derivation of sophisticated product recommendation and marketing strategies
- Dynamic and customer specific pricing
- Management and analysis of product reviews
- Prediction of demand and supply

Risk/Asset management

- Analysis, modelling and management of all kinds of (financial) risks
- Sentiment analysis und topic clustering of news or other documents
- Fraud identification
- Scenario-simulations and impact analyses
- Automated and high-frequency trading

Process optimization

- Process analysis and –optimization via prioritization and scoring
- Visualization and support of processes with user friendly dashboards
- Agile project management via Kanban and Scrum
- Development of data driven digital operation methods and communication strategies

Network analysis

- Identification and analysis of relationships between corporations (e.g. supplier / customer, creditor / debtor, etc.)
- Analysis of competitors and peer-group benchmarking
- Derivation and analysis of customer network based on social media

Infrastructure

- Analysis and optimization of the (IT-) infrastructure w.r.t Data-Science aspects
- Competitive data collection and management
- Efficient information retrieval
- Fast prototyping according to „fail fast, fail cheap" and "learn and adjust"

Data becomes more and more important in the everyday life.

The structure of data depends on its recording and determines adequate methods for the data-analysis

Data

Data is a set of variables. Data only becomes information suitable for making decisions once it has been analyzed in some fashion.

In 1998, Merrill Lynch cited a rule of thumb that somewhere around 80-90% of all potentially usable business information may originate in unstructured form.

Qualitative

Observations fall into distinct categories

Nominal

Observations have no natural order between the categories (e.g. eye color, default or alive)

Ordinal

Observations exhibit an ordering between the categories (e.g. exam results, rating grades)

Quantitative

Observations are counts or measurements

Discrete

Observations are integers or can be mapped on integers (e.g. number of default)

Continuous

Observations can take on any value, usually within some range (e.g. weight, PDs)

Unstructured

Data is not organized in a pre-defined manner

Text

The written words (i.e. sequences of symbols) in a book, magazine, article etc.

Images

The visual representation of the form of an object or person

Audio

Recorded acoustic sequence over time.

Video

Recorded sequence of images potentially enriched by some audio sequence.

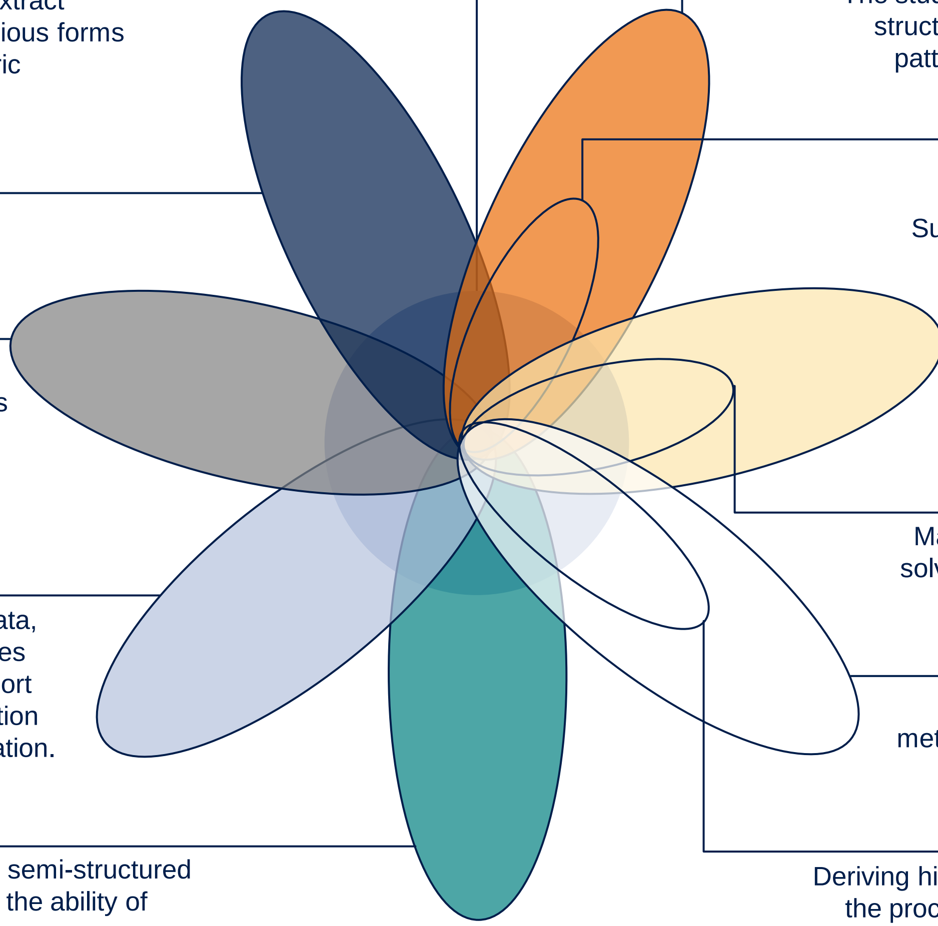
1. There is definite need for a framework how to collect, store, manage, analyse, … different data types.
2. Solutions for unstructured data must be found.

d-fine

The toolbox

Data science in a nutshell

Data science combines methods, techniques and knowhow from other (scientific) areas to gain insights from data



Data Science

Methods, processes and systems to extract knowledge or insights from data in various forms to create data products and data centric applications.

Data Mining / KDD

Explorative data analysis to create descriptive and predictive power.

Operations Research*

Application of advanced analytical methods to help make better decisions to arrives at optimal or near-optimal solutions to complex decision-making problems.

Business Intelligence

Strategies, processes, applications, data, technologies and technical architectures which are used by enterprises to support the collection, data analysis, presentation and dissemination of business information.

Big Data

Data sets consisting of **unstructured**, semi-structured and structured data with sizes beyond the ability of commonly used tools.

Mathematics

The study of topics such as quantity (numbers), structure, space, and change; and search for patterns and derivation of new conjectures.

Statistics

Collection, analysis, interpretation, presentation, and organization of data. Support or rejection of hypothesis derived from theoretical models based on data.

Computer Science

The study of the theory, engineering, and experimentation that form the basis for design and use of computers.

AI / Machine Learning

Machine mimics "cognitive" functions and solves tasks by ‘drawing conclusions‘ from observed examples.

NLP

Natural language processing (NLP) is a method to translate between computer and human languages.

Text Analytics / IR

Deriving high-quality information from text through the process of structuring, **pattern recognition** and evaluation and interpretation.

* Also risk management

Data science may be described by three collectively exhaustive thematic blocks, where our focus lies on the understanding of data

1 Construct

- Conception of (non-)relational databases models for
  - structured data
  - unstructured data (e.g. text, pictures, audio- or video files)
- Implementation
- Data collecting
  - Online-Crawler
  - interface to data provider
  - manual classification (questionnaires etc.)
  - Metadata (e.g. IP-addresses)

2 Understand

Derivation of data-driven decision rules

- **Phase I: Acquire** the necessary data or information.
- **Phase II: Analyse and manipulate** the data to gain actionable and impactful insights. This phase is closely related to **machine learning**. and may be further divided into:
  - Describe
  - Discover
  - Predict
  - Advise
- **Phase III: Evaluate** your insights and refine, evaluate again, select a different methods, techniques or data in phase I and II, until your quality standards are met.
- **Phase IV: Inspire** actions according to your insights.

3 Adopt

- Converting insights into action
- Defining or adjusting processes or strategies
- Support processes with user-friendly and interactive visualizations
  - Dynamic dashboards to support manual processes
  - Graphical user interfaces
- Comprehensive reports to control processes

Data science, especially the task ‘analyse’, operates many techniques, that will be discussed next.

The first step in the data analysis focuses on data preparation, which already can give important insights on data organization and management

Data analysis in data science

Seeks to enhance or refine raw data as well as leverage basic analytic functions.

Describe

How do I develop an understanding of the content of my data?

Discover

Predict

Advice

Processing*

How do I clean and separate my data?

- » **Filtering**
  quality checks, deduplication, outlier detection, smoothing
- » **Imputation**
  list-wise deletion, multivariate or variance neutral (Jack-knife) imputing

Aggregation*

How do I collect and summarize my data?

- » **Summary statistics**
  Box plots, scatter plots, entropy
- » **Distributional properties**
  distribution fitting, maximum likelihood, method of moments, semi-parametric methods

Enrichment*

How do I add new information to my data?

- » **Normalization / Transformation**
  fast Fourier transform, discrete wavelet transform
- » **Annotation**
  e.g. add information to keep track of source
- » **Feature addition**
  e.g. compute the value for certain operations that are used often

* Not exhaustive

d-fine

The second stage focuses on understanding interactions between observations and formulating first hypotheses

Data analysis in data science

Identifies hidden relationships or patterns.

Describe	**Discover**	Predict	Advice
	What are the key relationships in the data?		

Clustering*

How do I segment the data to find natural groupings?

- **Hierarchies**
  single-linkage, complete-linkage, average-linkage, BIRCH
- **Partitioning**
  k-Means and k-Medoids, Canopy
- **Text**
  regular expressions, stemming, topic clustering, word clouds
- **Non-elliptical clusters**
  fractal, DB scan
- **Cluster transition**
  Gaussian mixture models

Dependencies*

How do I determine which variables may be important?

- **Entities**
  graph or network representation
- **Variables**
  measures of association like correlation (Spearman, Pearson), discrimination power, Kendall's tau, concentration ratio (e.g. Herfindahl index)
- **Dimensionality reduction**
  (robust or sparse) PCA, factor analysis

Hypotheses testing*

How do I test ideas?

- **Compare two groups**
  t-test, Likelihood-Ratio-test, Lagrange-Multiplier-tests
- **Compare multiple groups**
  ANOVA, Hosmer–Lemeshow-test

* Not exhaustive

d-fine

In the third stage the focus shifts from historical observations to the dependencies between future outcomes on todays decisions

Data analysis in data science

Utilizes past observations to predict future observations.

Describe	Discover	**Predict**	Advice
		What are the likely future outcomes?	

Classification*

How do I predict group membership?

- **Decision Tree**
  rule-based classification, boosted trees, random forests
- **Lazy Learners**
  k-nearest neighbour, case-based reasoning, naïve bayesian
- **Backpropagation**
  (multilayer feed-forward) neural network, recurrent neural networks
- **Support Vector Machines**
  separability, kernels

Regression*

How do I predict a future value?

- **Linear models**
  OLS, GLS (esp. logistic regression, Poisson regression, multinomial regression)
- **Time series**
  ARMA, ARCH, GARCH, (hidden) Markov model, VAR-processes, co-integration
- **Panel**
  random /.fixed-effects, Hausmann-test, Arellano–Bond estimator

Recommendation*

How do I predict relevant conditions?

- **Graph theory**
  centrality, clustering
- **Collaborative filtering**
  memory-based, model-based

* Not exhaustive

d-fine

Given an understanding of the data and the drivers of past and future outcome, measures should be defined and analysed

Data analysis in data science

Define decisions, optimize over decisions, and advise to use decision that gives the best outcome.

Describe

Discover

Predict

Advise

What course of action should I take?

Logical reasoning*

How do I sort through different evidence?

» **Expert systems**
inference engine, knowledge base
» **Reasoning system**
automated reasoning
» **Game theory**
competitive incentive systems, separating and pooling equilibria

Optimization*

How do I identify the best course of action when my objective can be expressed as a utility function?

» **Convex programming**
(In)equality constraints, Simplex algorithm
» **Stochastic search**
stochastic gradient descent, (adaptive) simulated annealing
» **Dynamic control**
Hamilton–Jacobi–Bellman

Simulation*

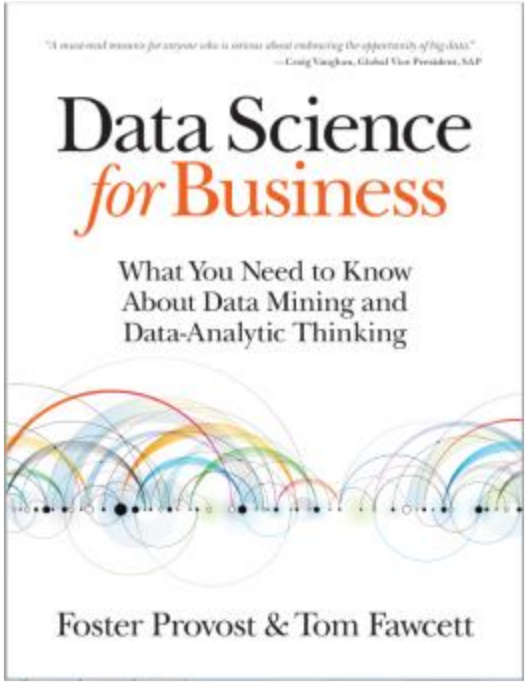
How do I characterize a system that does not have a closed-form representation?

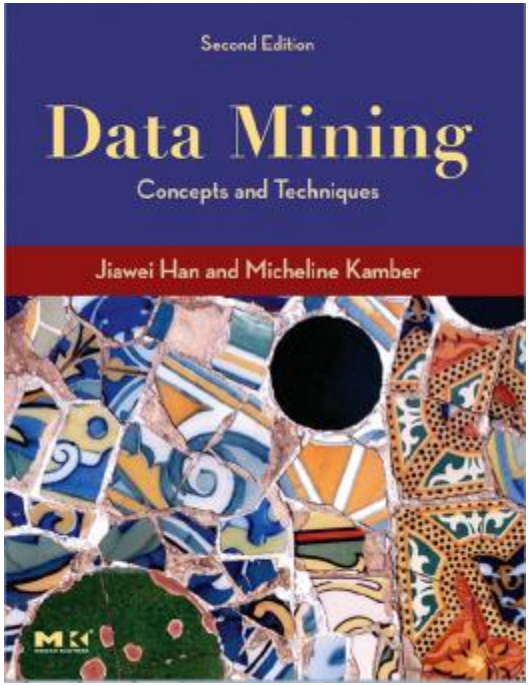
» **Discrete entities / possible states**
Markov models, discrete event simulation
» **Systems governed by factors**
ordinary/partial/stochastic differential equations
» **Path dependencies**
Mote-Carlo simulation

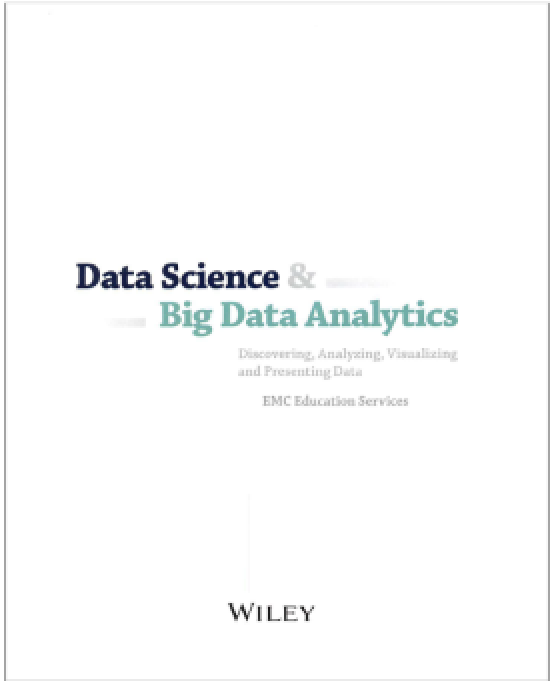
* Not exhaustive

d-fine

For more on data science (e.g. data architecture, tools and libraries, visualization) see



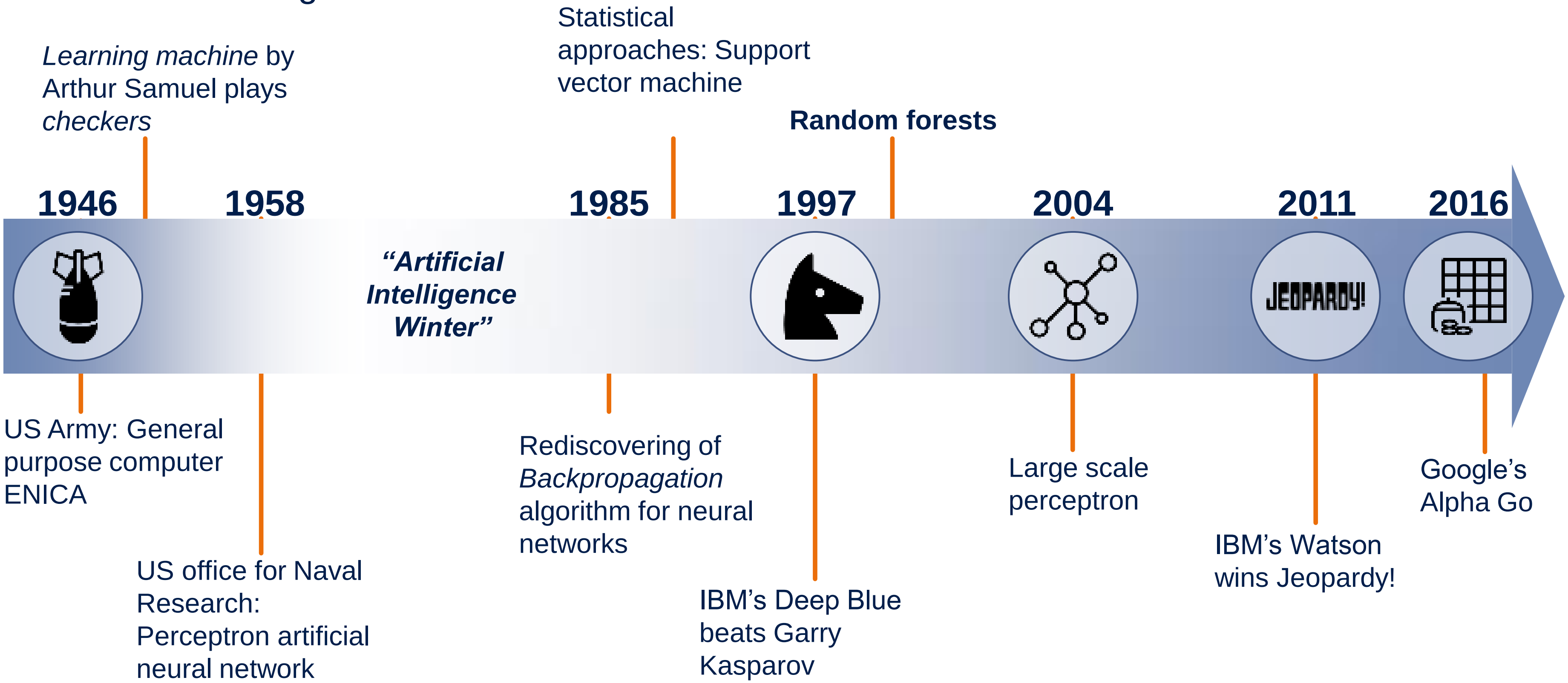




Machine learning in a nutshell

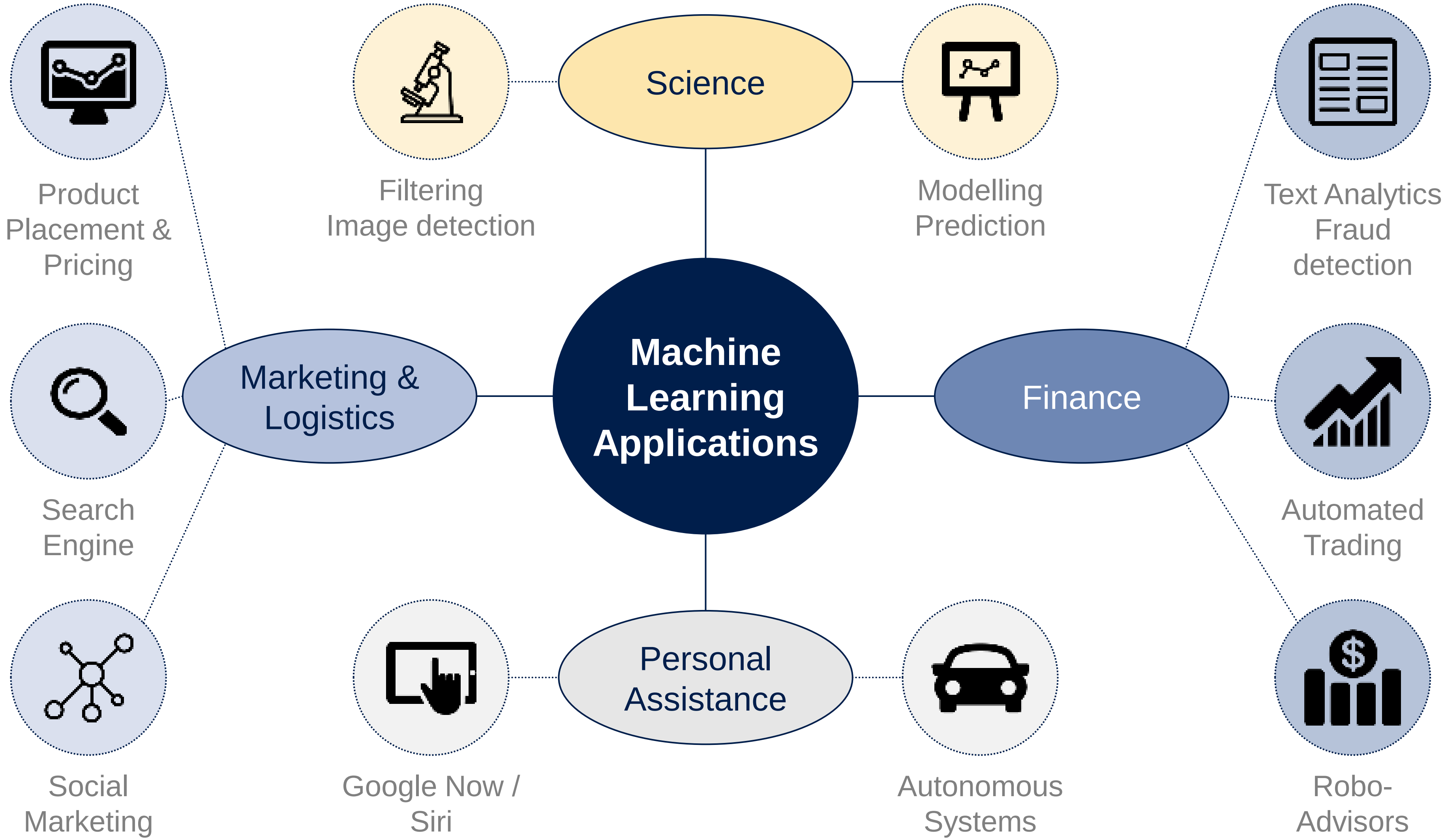
Statistical methods in data science are closely related to machine learning, which is not a new hype

- » Machine learning is not new. Early inventions were driven by the military.
- » The Internet age: IBM, Google, Amazon and Facebook are leading to a renaissance of machine learning.



d-fine

ML is no longer limited to artificial-intelligence researchers and born-digital companies like Amazon, Google, and Facebook



d-fine

Machine learning includes a large variety of methods and becomes valuable when problems and/or solutions are highly complex

Machine learning gives computers the ability to learn without being explicitly programmed

Learning system

1 **Supervised learning** (learning by examples)

2 **Unsupervised learning** (learning by observation**)**

3 **Reinforcement learning** (learning by feedback)

i **Shallow learning** (Feature definition excluded)

ii **Deep learning** (Feature definition included)

Useful when*

A **Humans** are unable to explain their **expertise** or expertise does not exist

B The **solutions** need to be adapted to particular cases or change over time

Methods*

a **Regression** (ordered value)

b **Classification** (no ordering required)

c **Clustering** (no predefined classes)

d **Density estimation** (esp. nonparametric)

e **Dimensionality reduction**

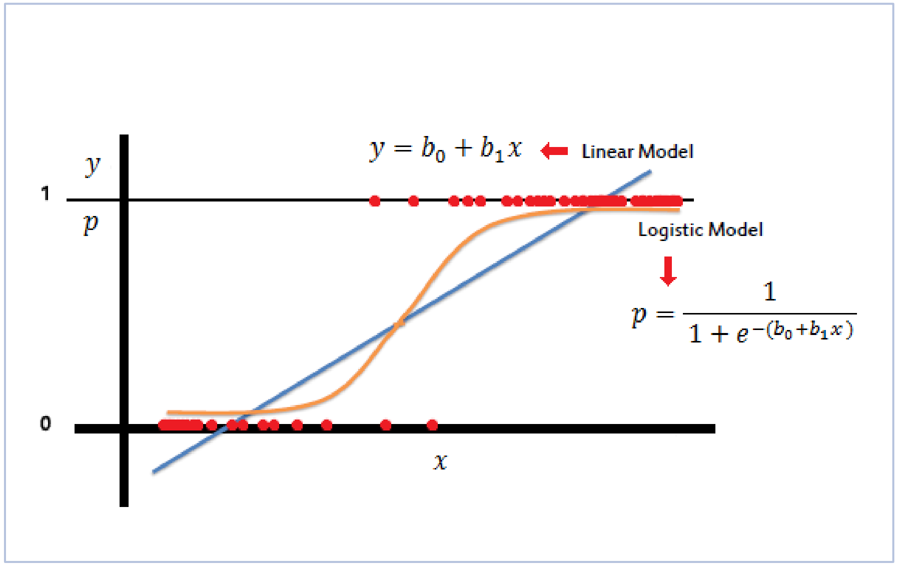
* Not exhaustive

A basic statistical model: Logistic regression

A classification problem in finance is the estimation of the default probability of a debtor

- » **Objective:** For e.g. the calculation of the regulatory capital requirements, bank have models to estimate the **default probability** of a single debtor. However this probability is **not observable**, an individual debtor can only be “alive” or “default”. Thus linear regression fails in this scenario.
- » **Target variable:** The binary default variable y (y=1 : default, y=0 : no default)
- » **Training set:** Historic creditor data e.g.:

Name	Age	Income	Default
A	32	46.000€	0
B	26	31.000€	1
C	54	60.000€	0



- » **Predictor ansatz:**
  $p(y = 1 \mid \boldsymbol{x}) = f_L(b_0 + b_1 x_1 + \cdots) = f_L(\boldsymbol{bx})$,
  where $f_L(s) = \frac{1}{1+\exp(-s)}$.
  This ansatz ensures that the probability is confined in the interval $[0,1]$.
- » **Estimation method:** Maximum Likelihood Estimation (MLE) and gradient ascent.

A linear model can not predict a probability confined to the interval 0 to 1. Using a sigmoid “transfer” function the regression method can be adapted to meet this requirement.

d-fine

Maximum likelihood estimation (1/2)

The general setup

» Assuming we have n independent observations $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ with an identical underlying distribution dependent describes by the parameter vector $\boldsymbol{b}$.

» Then, the joined density function of all observations is the product of the individual density functions, i.e.

$$f(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n; \boldsymbol{b}) = \prod_{i=1}^{n} f_{X_i}(\boldsymbol{x}_i; \boldsymbol{b}) \stackrel{\text{def}}{=} L(\boldsymbol{b}; \boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$$

» Since the n observations are given, the joint density function may be seen solely as a function of b. This is called the likelihood function $L(\boldsymbol{b};\ \boldsymbol{x}_1, \dots, \boldsymbol{x}_n\ )$.

» The parameter vector $\widehat{\boldsymbol{b}}$ that maximizes the likelihood function implies the best fit of the underlying distribution w.r.t to the observations.

» However, it is often useful to maximize the logarithmic likelihood $l(\boldsymbol{b}) = \log L(\boldsymbol{b}; \boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$

» To illustrate that let us at first assume that feature and target variables are connected by

$$y_i = \boldsymbol{b}\boldsymbol{x}_i + \varepsilon_i$$

where ε is a random noise distributed according to a Gaussian distribution $f_\varepsilon(z)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{(z)^2}{2\sigma^2})$ which implies $f_{X_i}(\boldsymbol{x}_i; \boldsymbol{b})=\frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{(y_i-\boldsymbol{b}\boldsymbol{x}_i)^2}{2\sigma^2})$.

The maximum likelihood estimation assumes an inherent density function of the observed parameters.

Maximum likelihood estimation (2/2)

Maximization of the likelihood leads to the “gradient ascent/descent rule”

- The likelihood function is hence given by:

$$l(\boldsymbol{b}) = \log \prod_{i=1}^{n} \frac{\exp\left(-\frac{(y_i - \boldsymbol{b}\boldsymbol{x}_i)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}} = \sum_{i=1}^{n} \log \frac{\exp\left(-\frac{(y_i - \boldsymbol{b}\boldsymbol{x}_i)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}} = n \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \boldsymbol{b}\boldsymbol{x}_i)^2$$

- $l(\boldsymbol{b})$ is maximized if $\frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{b}\boldsymbol{x}_i)^2$ is minimizes, implying the least mean squares estimator for $\boldsymbol{b}$.
- Coming back to our logistic regression problem, knowing that

$$P(Y = 1 \mid \boldsymbol{x}) = f_L\,(\boldsymbol{b}\boldsymbol{x}) \text{ and } P(Y = 0 \mid \boldsymbol{x}) = 1 - f_L\,(\boldsymbol{b}\boldsymbol{x})$$

  we can construct an individual density function as

$$P(Y = y \mid \boldsymbol{x}) = (f_L(\boldsymbol{b}\boldsymbol{x}))^y (1 - f_L\,(\boldsymbol{b}\boldsymbol{x}))^{1-y}$$

- Then, the log likelihood is:

$$l(\boldsymbol{b}) = \log L(\boldsymbol{b}) = \sum_{i=1}^{n} y_i \log f_L(\boldsymbol{b}\boldsymbol{x}_i) + (1 - y_i) \log(1 - f_L(\boldsymbol{b}\boldsymbol{x}_i)).$$

- To maximize the log likelihood we now use the **gradient ascent** rule

$$b_j := b_j + \alpha \frac{\partial}{\partial b_j} l(\boldsymbol{b})$$

- Using the differentiation rule of the logistic function $f_L{}'(s) = f_L(s)\,(1 - f_L(s))$ the contained derivative can be evaluated as $\frac{\partial}{\partial b_j} l(\boldsymbol{b}) = \big(y - f_L(\boldsymbol{b}\boldsymbol{x})\big)x_j$

d-fine

The perceptron as a logical building block

From logistic regression to logical operations

» We end again with a stochastic gradient ascent rule

$$b_j := b_j + \alpha\big(y - f_L(\boldsymbol{bx})\big)x_j$$

where $f_L(s)$ is a non-linear function.

» Changing the logistic function $f_L(s)$ in to a step function $H(s)$, i.e.

$$H(s) = \begin{cases} 1, s \geq 0 \\ 0, s < 0 \end{cases}$$

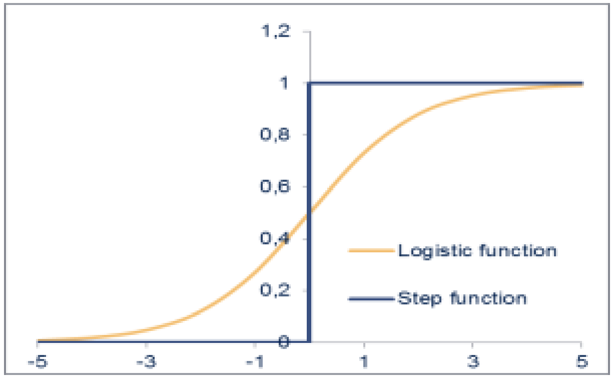
brings us to the so-called **perceptron** and the **perceptron learning rule**, i.e.

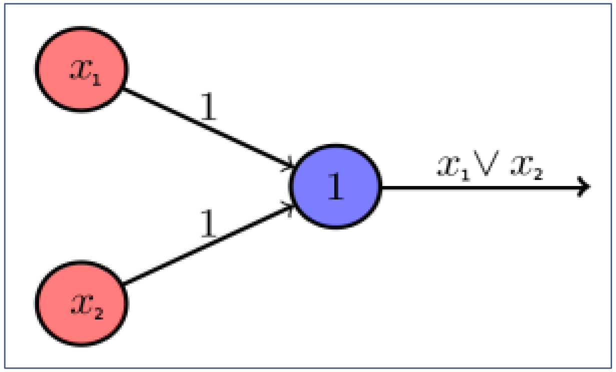
$$b_j := b_j + \alpha\big(y - H(\boldsymbol{bx})\big)x_j.$$

» Historically it was thought that a perceptron resembles the way a human neuron works, as it transfers a signal (feature variable) to a non-zero output ($y = 1$) only when it overcomes a certain threshold.

» A single perceptron connected to two input variables can realize the logic OR-function.

» In Machine Learning we are not so much concerned about the inherent statistical distributions as in statistics



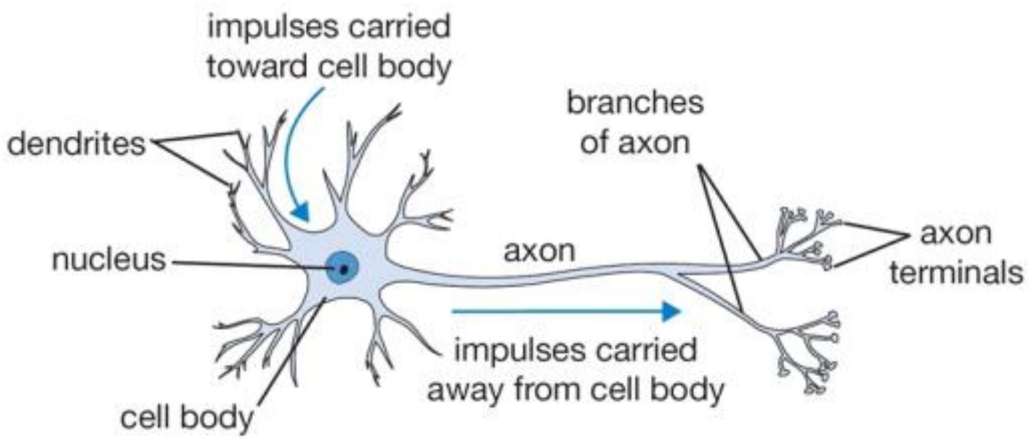


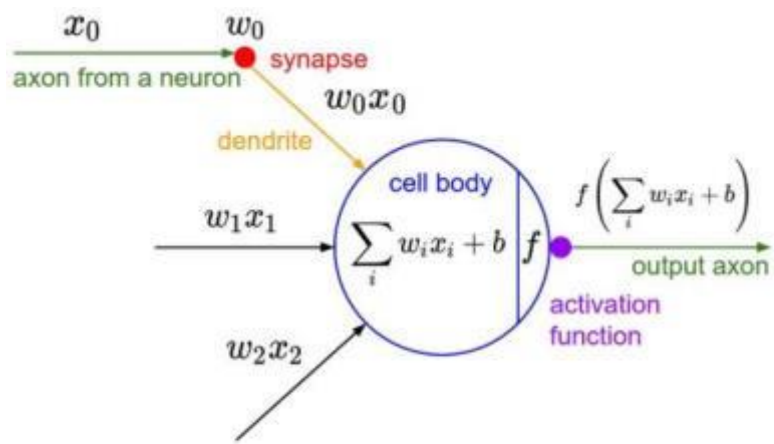
Interchanging the transfer function of logistic regression with a step-function lets us view regression as "a concept for logical decision making" based on the input parameters.

d-fine

(Deep) neural networks

» Neural networks can be understood as a mathematical model of neurons in a brain[1]. Of course, this model is too simplistic to account for the processes in a real brain.





» Neurons "fire" electrical impulses along so-called axons to other neurons, thereby producing a dense and complicated web of interacting units.

» In the mathematical model of a neuron the signal x from another neuron undergoes first an affine transformation (the parameters of those are called weights), which is the input to a non-linear function called an activation function.

» By building a network of such mathematical neurons one obtains a so-called **neural network**.

[1]See e.g. Bengio, Yoshua, et al. "Towards biologically plausible deep learning." *arXiv preprint arXiv:1502.04156* (2015 for a discussion.
Image credit: http://cs231n.stanford.edu/

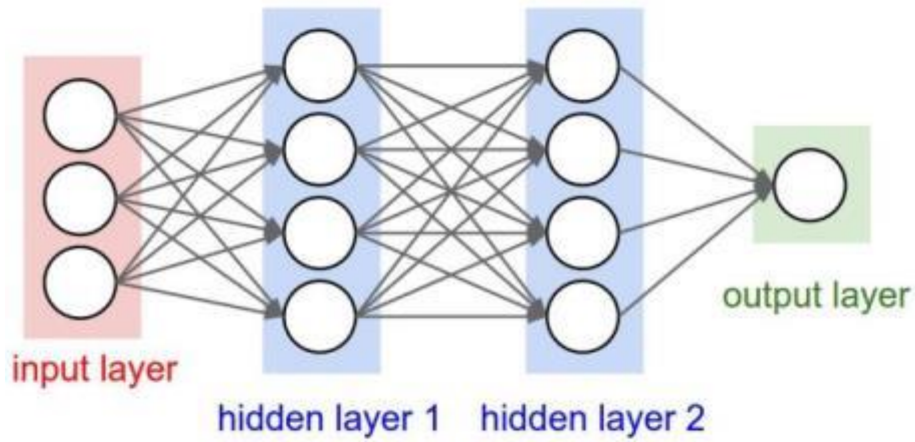
d-fine

General overview of neural networks

» Typical[1] neural networks models come with various so-called layers.

» There are various activation functions used in the literature and in practice.

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer NN	



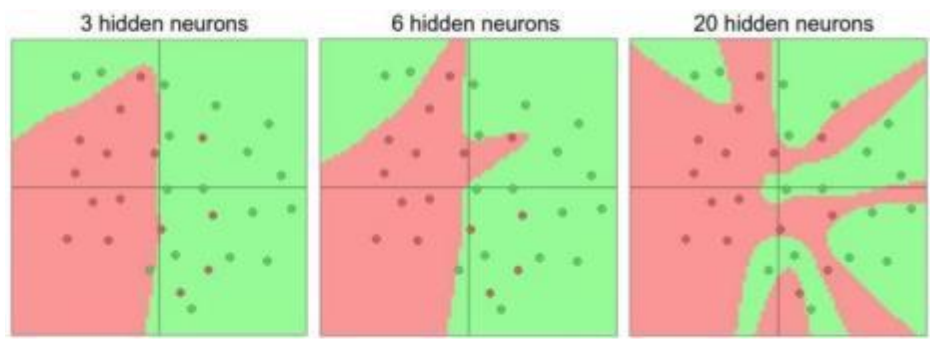


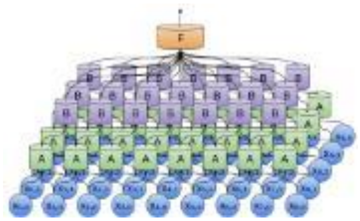
Image credit: http://cs231n.stanford.edu/

Nowadays neural networks are build with 100s of layers leading to a high capacity[2].

[1]There are also other types like Hopfian networks or (Deep) Boltzmann Machines which are not discussed here
[2]http://playground.tensorflow.org

There are several deep neural architectures for (un-) supervised learning tasks

Convolutional (CNN)

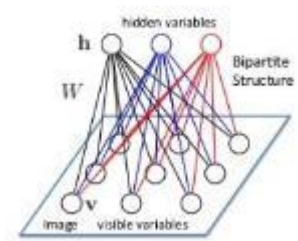


Mainly used for static data such as images

- » The static length input features are “convoluted”
- » Each layer learns a more abstract representation of the data
- » Equivalent to renormalization group flow in physics

Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

Boltzmann machines (and variants)



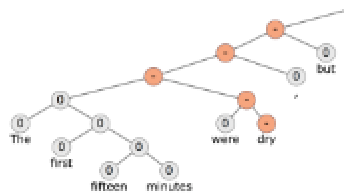
Mainly used for static data

- » The neural net consists of a visible and hidden part
- » The hidden part can learn arbitrary complex representations of the inputs
- » Comparable to HMMs

Image credit: http://prog3.com/sbdm/blog/zouxy09/article/details/8781396

Architectures can be 100s of layers deep!

Recursive

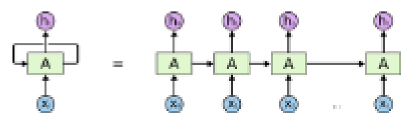


- » The “input features” come in a natural hierarchy or tree-like structure
- » The neural net is applied all along the hierarchy

Image credit: http://colah.github.io/posts/2014-07-NLP-RNNs-Representations/

Mainly used for hierarchical and tree-like data such as language

Recurrent (RNN)



Mainly used for sequential data such as language and time series

- » The “input features” are sequential and of different length
- » The neural net is recurring
- » The neural net can learn long term dependencies (c.f. LSTM, GRUs)

Image credit: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

d-fine

Deep learning networks

» Artificial neurons can have an arbitrary transfer function $\varphi(\boldsymbol{\omega}_{ij}\boldsymbol{x}_j)$ and unlimited inputs.

» A layered network of connected neurons is a powerful tool to predict complex non-linear dependencies in data.

But how can a neural network be trained?

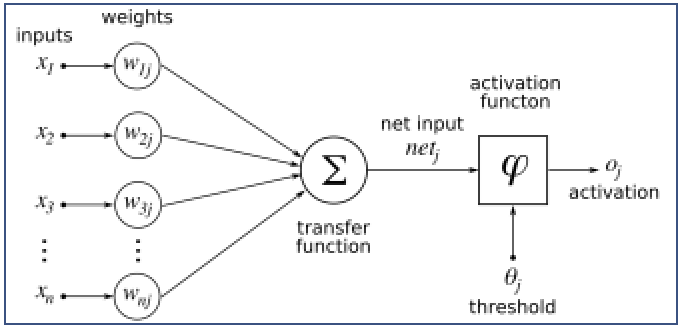
» One-layer network training: perceptron learning rule

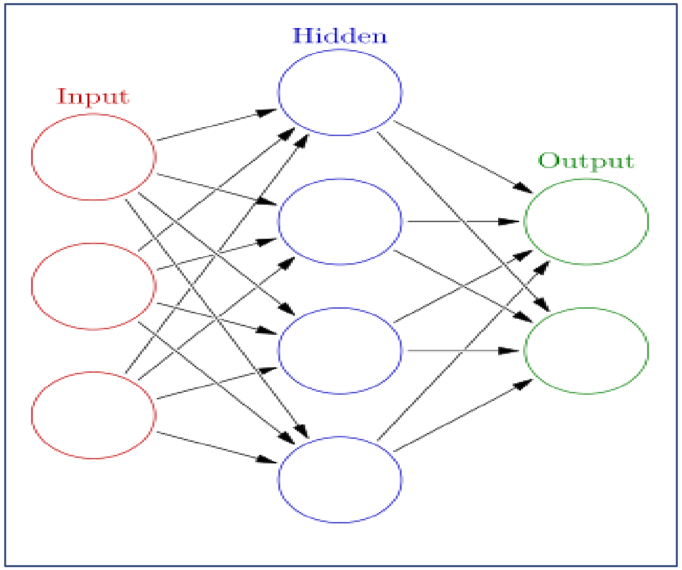
» A training method for multi-layer networks is **backpropagation**:

1. Propagation of input features to output variables o
2. Estimation of the **mean square error** $E = \frac{1}{2}(t-o)^2$, where t is the training target variable
3. Back-propagation of the error through the network with weight adjustment corresponding to **gradient descent**

In detail: The weight ω_{ij} between the i-th and j-th neuron has to be updated via: $\omega_{ij} := \omega_{ij} - \alpha \frac{\partial E}{\partial \omega_{ij}} = \omega_{ij} + \alpha\, \delta_j o_i$, where

$$\delta_j = \begin{cases} \varphi'(\boldsymbol{\omega}_{ij}\boldsymbol{x}_j)(t_j - o_j), \text{if } j \text{ is output neuron} \\ \varphi'(\boldsymbol{\omega}_{ij}\boldsymbol{x}_j) \sum_k \delta_k \omega_{jk}, \text{if } j \text{ is hidden/input neuron} \end{cases}$$



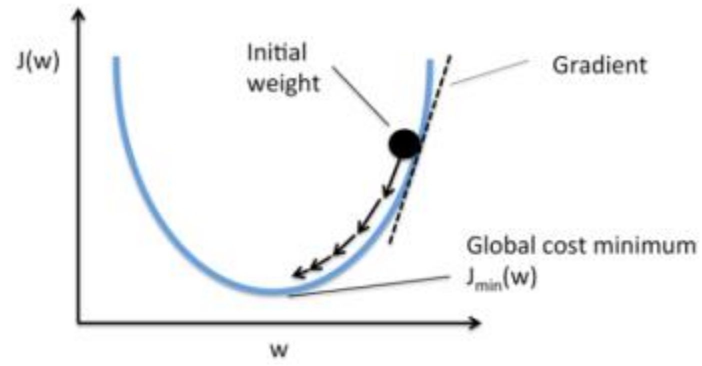


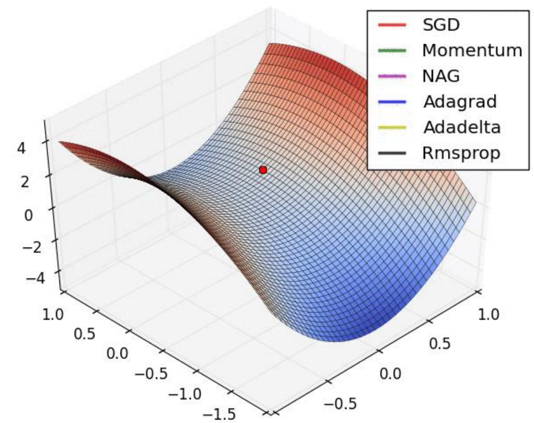
How can neural networks be trained?

- A neural network is parametrized by its topology, its activation functions and its weights.
- In a real world application one chooses a topology and types of activation functions.
- The weights w are then derived from training data by optimizing an objective function $J(w)$.
- Example: Supervised learning task:

> Given a set of N observations x_i with labels y_i the weights $\bar{w}$ are fixed by:
>
> $$\bar{w} = argmin_w \sum_i J(w, x_i, y_i) = argmin_w \sum_i J(NN(w, x_i); y_i)$$
>
> where NN denotes the output of the Neural Network (or almost any other ML algorithm).

- Therefore the learning problem is mapped to an optimization problem of an in general non-convex objective function.
- The optimization problem for neural networks is in almost all cases tried to be solved by the 17th century technique of gradient descent with some modern twists[1].

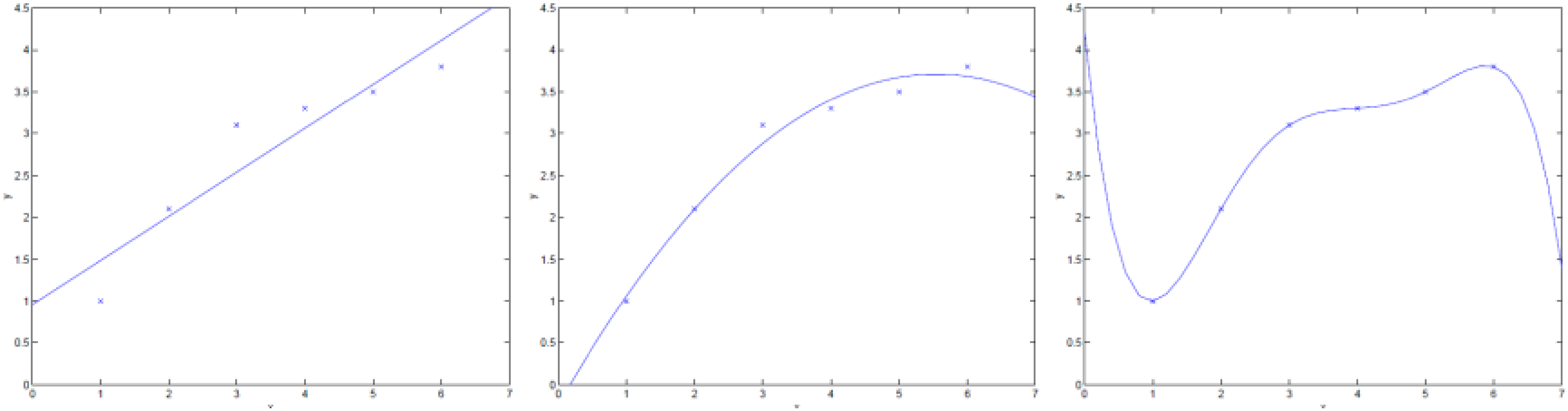




Images credit: Alec Radford

[1]See e.g. http://sebastianruder.com/optimizing-gradient-descent/ for a good overview.

Bias-variance trade-off



- Fitting with high order polynomials $b_0 + b_1 x + b_2 x^2 + \ldots + b_5 x^5$ leads to a lower total error compared to simple linear model.
- **The bias** is error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs → underfitting.
- **The variance** is error from sensitivity to small fluctuations in the training set. High variance can cause overfitting: modelling the random noise in the training data, rather than the intended outputs.
- With more and more complex models and more parameters we tend to over-fit the Noise and mask the true signal.
- Model selection: We want to choose the best trade between bias and variance.

d-fine

Model selection – basics

Goal: Pick the “best” Model

- » First Idea: Find the Model with smallest training error
- » Does not work: Will always prefer high **variance** models with **maximum parameter number**
- » New Idea: **simple cross validation**
- » Split the data into a **training set** with ca. 70% of the data and a **validation set** with 30% of data.
- » Fit models to training set and measure error on validation set.
- » **Problem:** We loose 30% of our data.
- » **Problem 2:** Depending on the Split our MSE can differ in level.

Our Example

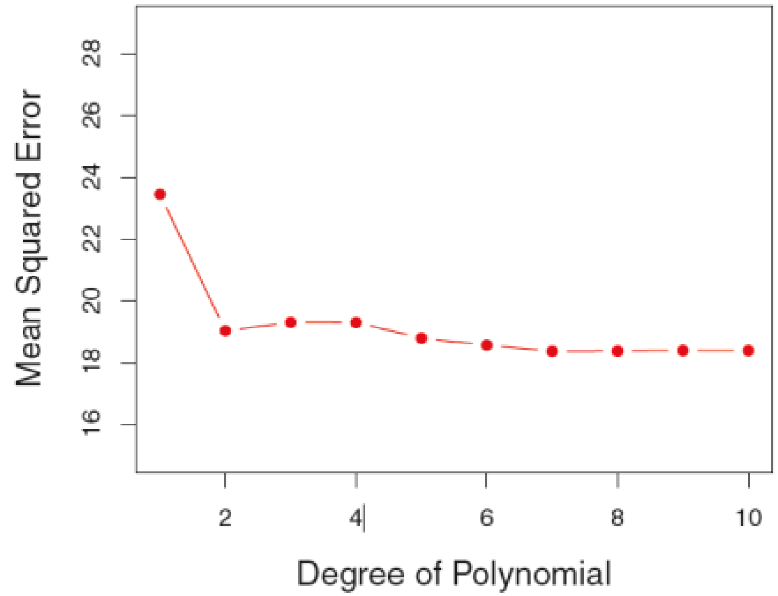


Image credit: An introduction to Statistical Learning. Springer.

- » For our polynomial problem we plot the **validation MSE.**
- » In accordance with our intuition we see that 2 seems to be a good choice with not much more to gain with higher orders.

Separating the whole data set into training and validation set is a useful concept to quantify the model error on “unseen” data.

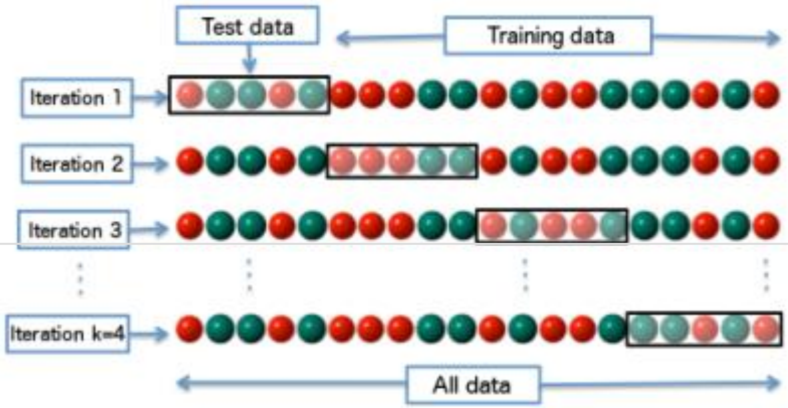
d-fine

Model selection – k-fold cross validation

Goal: Pick the "best" Model

How can we recycle more of our data?

1. Split the data into k-subsets
2. Fit the models using k-1 subsets and measure the MSE on the remaining subset
3. Average the results over all possible choices for the k-1 subsets



Our Example

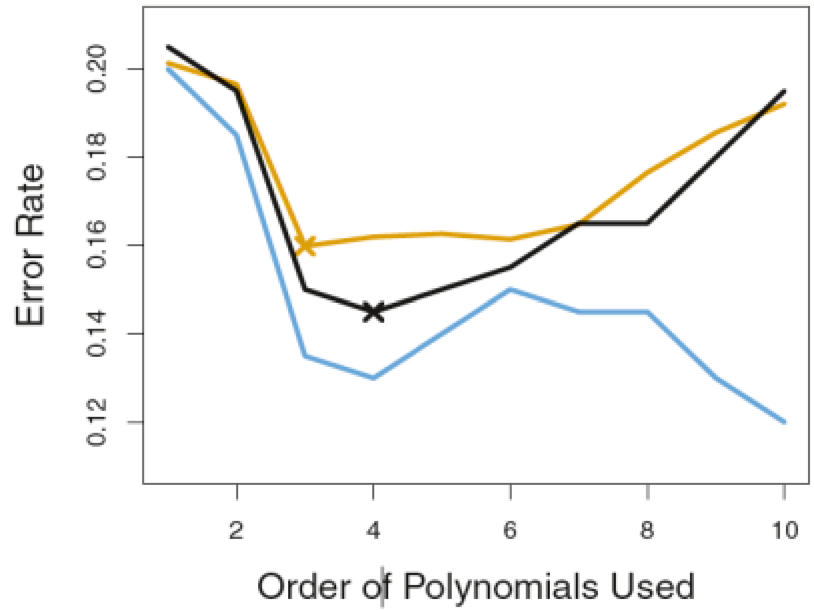


Image credit: An introduction to Statistical Learning. Springer.

- » We typically See a U-shape
- » Shown here: Three curves for different problems

(The black line is the problem we presented before)

Methods such as k-fold cross validation can effectively reduce the generalization error and recycle as much of the data as possible.

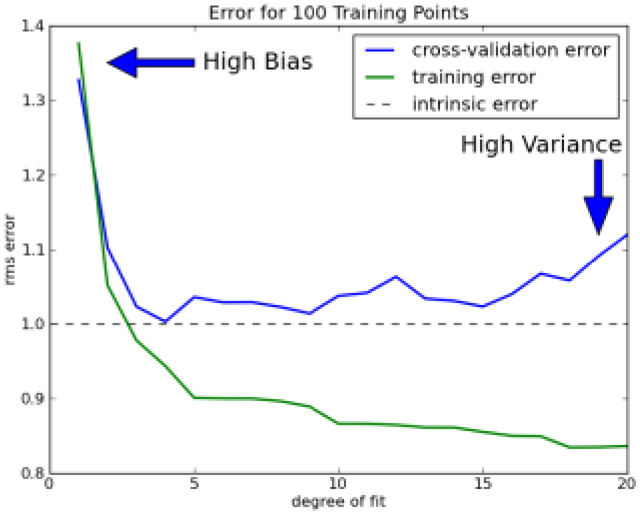
d-fine

Deep neural networks in their naïve form have various problems

» Vanishing gradient problem[1]
  › Neural networks are usually trained by various incarnations of gradient descent, e.g.

$$w_{i+1} = w_i - \gamma \cdot \left.\frac{\partial J(w)}{\partial w}\right|_{w_i}$$

  › By the chain rule this leads to products of derivatives of activation functions
  › As most activation functions take values in $[-1, 1]$ these products become very small for deep networks



» Overfitting
  › Deep neural networks typically have millions of free parameters
  › Without care this can typically lead to the overfitting phenomenon

» Slow training
  › To optimize a non-convex function with millions of terms and millions of variables is computationally very expensive
  › Without special hardware the training of deep neural nets is not feasible

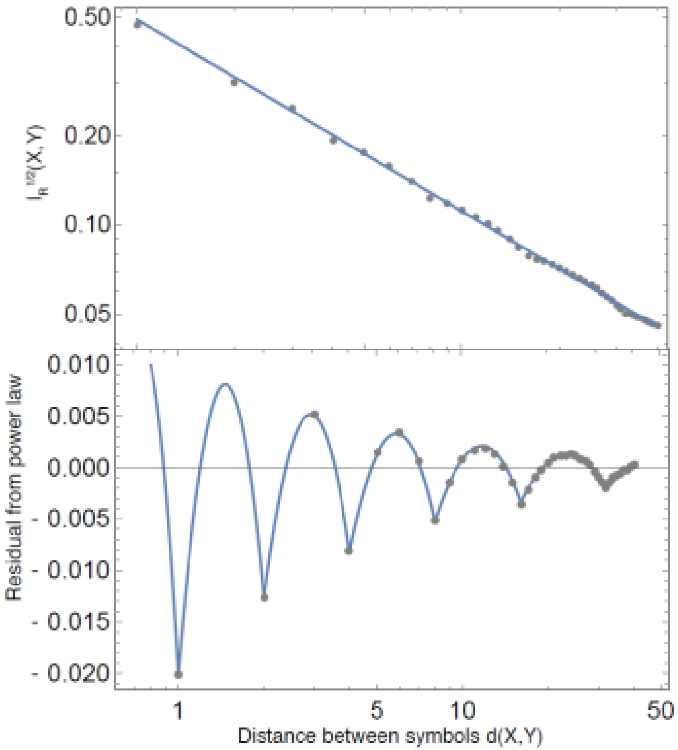
» Lots of (labelled) data is needed for training
  › Without expert knowledge, which could either be built into the topology of the net or into constraints on the weights and/or the objective function, lots of labelled data is needed to bring deep neural nets into a regime of good behaviour with respect to generalization

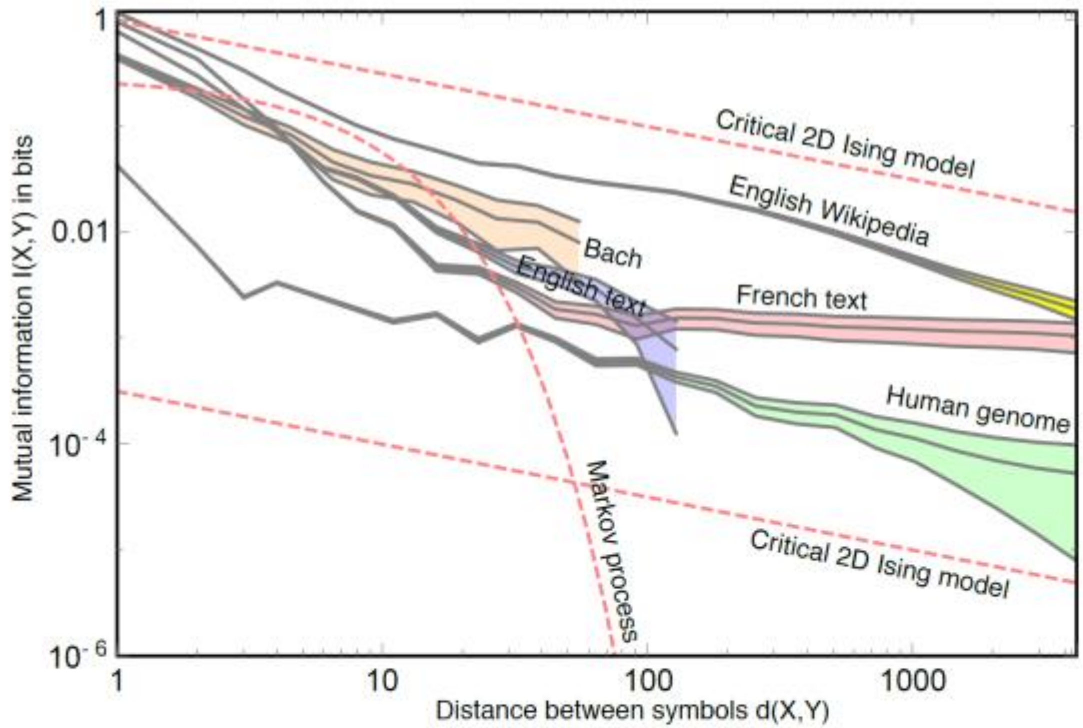
[1]Hochreiter, Sepp. "Untersuchungen zu dynamischen neuronalen Netzen."*Diploma, Technische Universität München* (1991): 91.

Why should neural networks be deep after all? (1/2)

Example: natural language

- Studying the empirical mutual information (kind of a two-point function) between symbols in natural written language unveils a power-law behaviour!
- These long range interactions can even theoretically not be modelled by simple shallow models like Hidden Markov Models (HMMs).





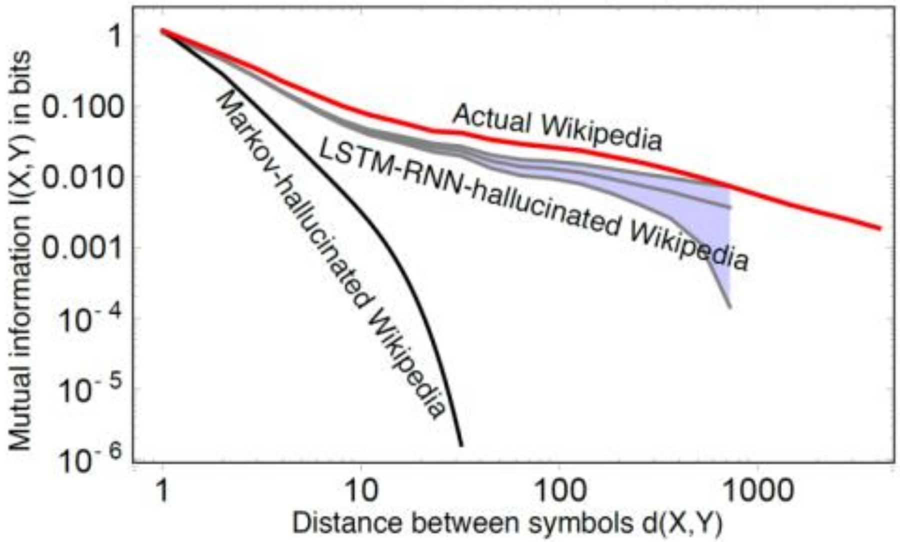
Lin, Henry, and Max Tegmark. "Critical Behavior from Deep Dynamics: A Hidden Dimension in Natural Language." *arXiv preprint arXiv:1606.06737*(2016), Lin, Henry W., and Max Tegmark. "Why does deep and cheap learning work so well?." *arXiv preprint arXiv:1608.08225* (2016).

Why should neural networks be deep after all? (2/2)

Example: natural language

» Hallucinating Wikipedia entries with a deep recurrent neural architecture captures the long range interactions present in natural language.

» This is not an accident, as it was argued that Deep Neural architectures are related to a well-known set of ideas in physics, namely the **Renormalization Group** (RG).



» This would explain two empirically observed properties of Deep Neural Networks:

› These types of models are able to extract high level features from microscopic data (e.g. raw pixels to categories of objects) as they flow to fixed points under the RG-flow (universality).

› The “two-point functions” of Deep Neural Networks in general exhibit a power law decay near their critical points.

» Nonetheless can deep models sometimes be approximated by simpler shallow models!

Deep Learning can be mapped onto the Renormalization Group known from physics.

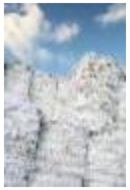
[1]Mehta, Pankaj, and David J. Schwab. "An exact mapping between the variational renormalization group and deep learning." *arXiv preprint arXiv:1410.3831* (2014), https://charlesmartin14.wordpress.com/2015/04/01/why-deep-learning-works-ii-the-renormalization-group/

d-fine

What rescues deep learning?

- Deep neural nets are hard to train due to what is known the “vanishing/exploding gradient problem“
- In the 90s this (among other things) led to a period called the AI-winter and almost to an abandonment of the idea of neural nets[1]. Progress during the last 10 years has made it possible to train very deep nets with hundreds of layers
- Responsible for this progress are mainly:

Growth of available computing power: Clusters of (C,G,T)PUs



Availability of large amounts of (labelled) data

Methodological breakthroughs (pre-training, dropout, LSTMs/GRUs, ReLUs, stochastic depth training, convnets, …)

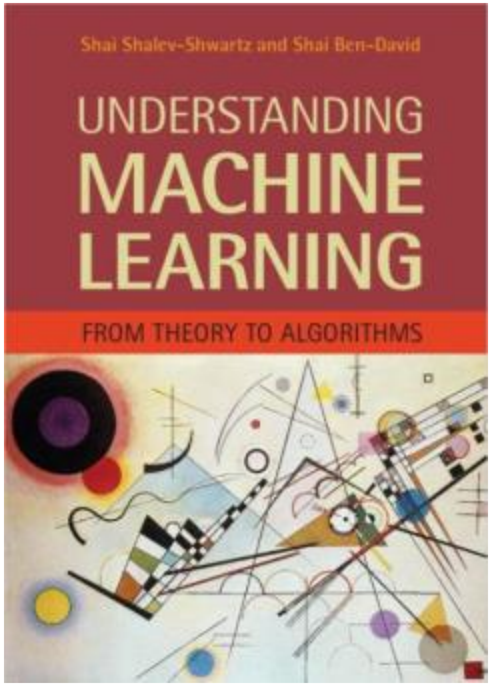
- With these techniques Deep Neural Nets have reached super-human abilities in many areas, including image recognition, geolocating by images, game playing, sentiment analysis, …
- There are several software frameworks available for the training of deep neural nets

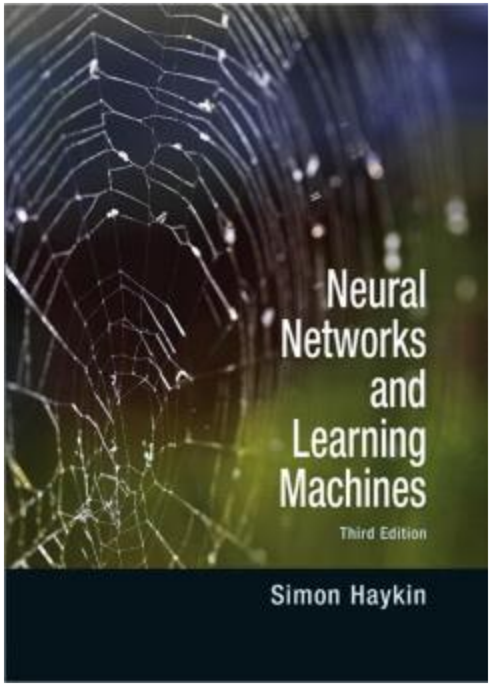
Framework	TensorFlow	theano	torch	Microsoft CNTK	Caffe	dmlc mxnet	H2O	DEEPLEARNING4J	neon
Developer	Google	U. Montreal	Collobert, et al.	Microsoft	U. Berkeley	DMLC	H2O.ai	Skymind	Nervana
Language(s)	C++, Python	Python	C, Lua	C++	C++, Python	C++, Python, R, Julia, …	Java, Scala, Python, R	Java, Scala, C	Python

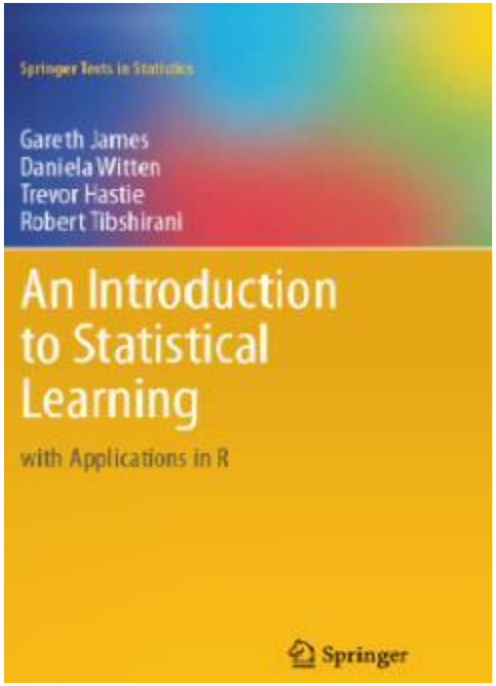
[1]Nonetheless remarkable progress was made during the 90s by people like Jürgen Schmidthuber (ETH), Geoffrey Hinton (Google), Yann LeCun (Facebook), Yosuah Bengio (U. Montreal), Andrew Ng (Baidu) and others

d-fine

For more on machine learning (deep learning, boosting, dynamic programming) see



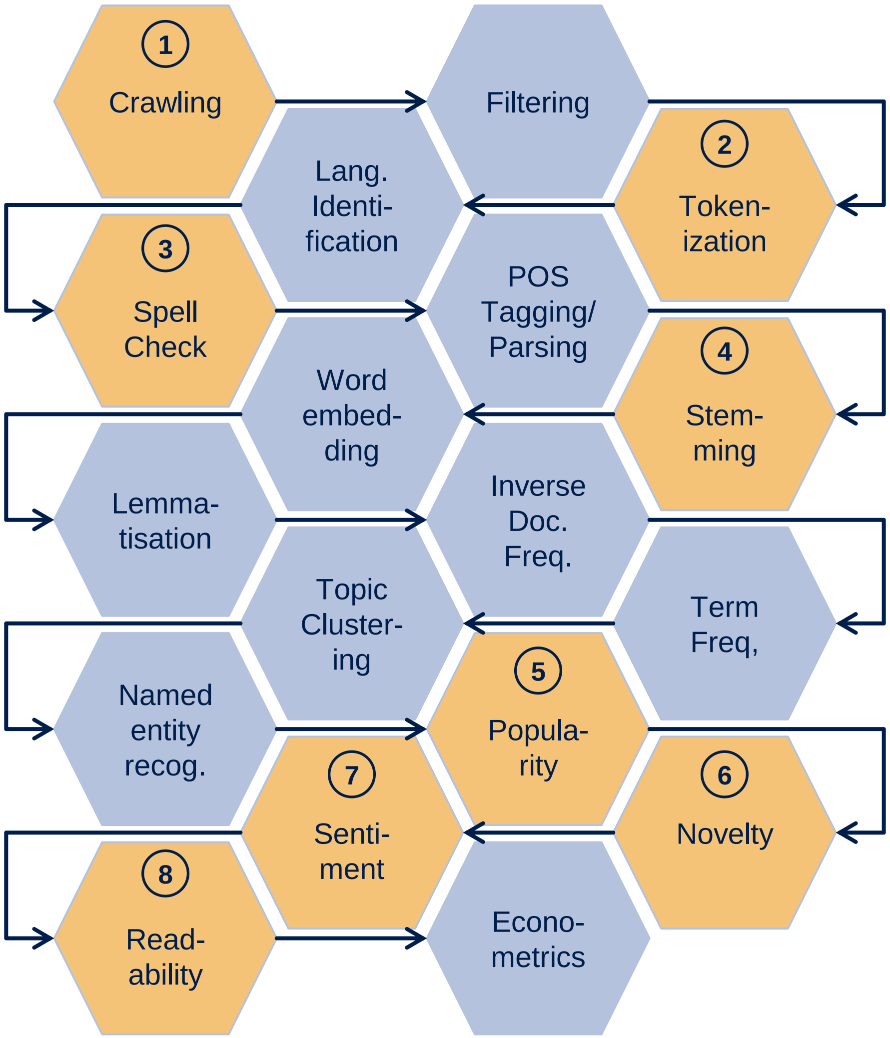




15 minutes break

Text analytics in a nutshell

The analysis of unstructured text can be structured in different phases, which make use of individual concepts



1. Connect to an online archive, download relevant articles and do quality assurance (e.g. download succeeded, delete advertisements)
2. Store the news in a local DB in an efficient and comprehensive way (e.g. **elasticsearch**)
3. Use e.g. the **Levenshtein distance** to find the best match for spelling errors
4. Linguistic rules to map words modified by conjugations or declensions on its stem, see e.g. **Porter or Snowball stemmer**.
5. Identify references (e.g. with **regular expressions**), build a network and measure the popularity
6. Apply the **vector space model**, i.e. represents document as vectors and compare them by making use of the corresponding angles
7. Use word lists (e.g. General Inquirer) or toolkits (SentiWordnet, SentimentAnalyzer, VADER)
8. Use e.g. the **Gunning Fog** index (i.e. estimated years of education needed to understand the text) based on the number of syllables, words and sentences

d-fine

Regular Expressions – extremely useful to pre-process raw data and for search queries

Remarks on regular expressions

- A regular expression is a pattern describing a certain amount of text. It defines a search pattern by a sequence of characters.
- The use of regular expressions enables to get results with just one operations instead of many.
- Established in the 1950s by Stephen Cole Kleene to formalize the description of a regular language in theoretical computer science.
- Embedded in many programming languages and software packages.

Examples of regular expressions

Character classes:

```
.                    any character except newline
\w \d \s             word, digit, whitespace
\W \D \S             not word, digit, whitespace
[abc]                any of a, b or c
[^abc]               not a, b or c
[a-g]                character between a and g
```

Anchors:

```
^abc$                start / end of the string
\b \B                word, not-word boundary
```

Escaped characters:

```
\. \* \\             escaped special characters
\t \n \r             tab linefeed carriage-return
```

Groups:

```
(abc)                capture group
\1                   backreference to group #1
(?:abc)              non-capturing group
```

Lookaround:

```
(?=abc)  (?!abc)     positive negative lookahead
(?<=abc) (?<!abc)    negative positive lookbehind
```

Quantifiers:

```
a* a+ a? Greedy      0 or more, 1 or more, 0 or 1
a*? a+? a?? Lazy     0 or more, 1 or more, 0 or 1
a{5} a{2,}           exactly five, two or more
a{1,3}               between one and five
a+? a{2,}            matches as few as possible
```

Alternation:

```
ab|cd                matches ab or cd
(?(?=ab)cd|ef)       if-then condition
```

Examples:

Grabbing HTML Tags:

```
<([A-Z][A-Z0-9]*)\b[^>]*>(.*?)</\1>
```

Identify duplicate lines in a csv:

```
(?<=,|^)([^,]*)(,\1)+(?=,|$)
```

d-fine

Large text-based datasets require new (NoSQL-based) storage techniques

E.g. news are weakly structured but possess (usually) an URL, publication date, headline and content

Boeing boosts share buyback to $14 billion, hikes dividend

Mon Dec 14, 2015 5:11pm EST

Boeing Co (BA.N) raised its share repurchase authorization to $14 billion from $12 billion and also increased its quarterly dividend, a sign of confidence in its cash outlook despite plans to cut production. The planemaker, which had $5.25 billion remaining under the previous buyback plan, said it raised its dividend to $1.09 per share from 91 cents. Boeing's shares rose 1.2 percent to $144.75 in after market trade on Monday, more than recovering their losses in regular trading. The company had said in October it could cut production by as much as 15 percent on its 777 long-range, widebody jetliner, one of its most profitable planes and a key source of cash. The talk of a possible slowdown came as Boeing posted narrower losses on its 787 Dreamliner and voiced confidence in that plane's ability to generate cash and fill the gap. Boeing is banking on the 787, its newest jet in production, to begin generating cash flow in the current quarter. The company had previously raised its share repurchase authorization and increased its dividend in December last year. Boeing said on Monday that it had finished its stock repurchases for 2015, having spent $6.75 billion. It expects to start buying back shares in January.

(Reporting by Radhika Rukmangadhan in Bengaluru; Editing by Savio D'Souza)

http://www.reuters.com/article/us-boeing-buyback-idUSKBN0TX2I520151214

Document	Date	Headline	Content	URL
1	Dec 14, 2015, 5:11pm EST	Boeing boosts share […]	Boeing Co (BA.N) raised its share […]	http://www.reuters.com/article/us-boeing-[...]
…	…	…	…	…

Document	Boeing	Co	raised	its	share	repurchase	authorization	…
1	4	1	3	11	3	2	2	…
…	…		…	…	…	…	…	…

The Term-Document-Matrix approach allows **efficient searching & scoring (e.g. via term frequency and inverse document frequency)**, but does not always preserve the structure of the text.

How can we compare text? The Levenshtein (or minimum-edit) distance as a measures of similarity between words (or strings)

The minimum number of single-character edits* to change one word into the other measures similarity

» Let w_k denote a word with $|w_1| \in \mathbb{N}$ letters. Let $w_k(i)$ denote the i^{th} letter. Then, the Levenshtein distance between two words w_1 and w_2 is given by $Edit_{w_1,w_2}(|w_1|,|w_2|)$, where $Edit_{w_1,w_2}(\cdot,\cdot)$ is recursively defined by:

$$Edit_{w_1,w_2}(i,j) = \begin{cases} \max(i,j), & \text{if} \min(i,j) = 0 \\ \min \begin{cases} Edit_{w_1,w_2}(i-1,j)+1 \\ Edit_{w_1,w_2}(i,j-1)+1 \\ Edit_{w_1,w_2}(i-1,j-1)+\mathbb{I}(w_1(i) \neq w_2(j)) \end{cases}, & \text{otherwise} \end{cases}.$$

» The Levenshtein distances can also be computed by a bottom-up dynamic programming algorithm.

» As an example, consider the words-pairs:.

› ‘pear’ and ‘peach’

› ‘bottom’ and ‘tom’

» Using individual penalties for insertions, deletions and substitutions allows to fine-tune the algorithms.

		P	E	A	C	H
	0	1	2	3	4	5
P	1	0	1	2	3	5
E	2	1	0	1	2	4
A	3	2	1	0	1	3
R	4	3	2	1	1	**2**

		B	O	T	T	O	M
	0	1	2	3	4	5	6
T	1	1	2	2	3	4	5
O	2	2	1	2	3	3	4
M	3	3	2	3	4	4	**3**

Simple spell checking may be based on the Levenshtein distances for words that do not appear in a given dictionary, but the approach is not feasible to compare documents.

* i.e. insertions, deletions or substitutions

d-fine

The vector space model for documents is closely related to the Term-Document-Matrix and allows simple algebraic operations

Documents can be seen as vectors and the Euclidean norm allows to measure similarities

1

News 1:
This news stands for
(1.00) (0.99) (0.97) (0.95)
a simple wildcard for
(0.92) (0.90) (0.87) (0.83)
more meaningful news
(0.79) (0.74) (0.70)

News 2:
Here is another article
(1.00) (0.99) (0.97) (0.95)
without meaningful content
(0.92) (0.90) (0.87)
standing for some news
(0.83) (0.79) (0.74) (0.70)

2

Word	a	Another	article	content	for	here	is	meaningful	more	news	simple	some	standing	stands	this	wildcard	without
News1	.92	0	0	0	1.78	0	0	.74	.79	1.69	0.9	0	0	.97	1	.87	0
News2	0	.97	.95	.87	.79	1	.99	0.9	0	0.7	0	.74	.83	0	0	0	.92

3

$$\cos(\mathrm{x},\mathrm{y}) = \frac{\langle x, y\rangle}{\|x\| \cdot \|y\|} = \frac{3.2552}{9.66} = 0.3370$$

- *n*-grams (i.e. sequence of *n* words) may be considered to recognize combined phrases e.g. "machine learning", "big data", …
- Discard Stop-words*
- More sophisticated weights
  - Term Frequency, i.e. the number of times a term occurs in a document relative to the length of documents
  - Inverse Document Frequency, i.e. diminishes the weight of terms that occur very frequently in document
- Reduce the dimensionality by
  - linguistic stemming algorithms
  - statistical methods for word embedding
- Given classified documents, this approach may be applied to find for unclassified document the best matching classified document(s) (e.g. k-nearest-neighbor)

Natural languages contain 10.000+ distinct words per language (and still counting), which implies an infeasible dimensionality.

* terms that inherit no intrinsic meaning

The dimensionality of the vector space may be reduced with the Porter Stemmer algorithm

Remarks on the Porter Stemmer

- Every word can be represented as **C?(VC){m}V?** where
  - C is a sequence of consonant
  - V is a sequence of vowels
  - (.){m} denotes an m-times repetition of the expression in the brackets
  - ? denotes optionality for the preceding expression
  - * denotes wildcard
- With this notation, there are five steps to take to cut a word to its stem (see right for an extraction of the rules)
- However, word stems are not always real words and stemming rules might fail for some words (e.g. European / Europe or matrices / matrix)
- Clever stemming reduces the dimension by factor 10!

Examples for stemming rules

Step 1a*
```
SSES -> SS caresses -> caress
IES -> I ponies -> poni
S -> '' cats -> cat
```

Step 1b*
```
(m>0) EED -> EE agreed -> agree BUT feed -> feed
(*v*) ED -> '' plastered -> plaster BUT bled -> bled
(*v*) ING -> '' motoring -> motor BUT sing -> sing
```

Step 1c*
```
(*v*) Y -> I happy -> happi BUT sky -> sky
```

Step 2*
```
(m>0) ATIONAL -> ATE relational -> relate
(m>0) TIONAL -> TION conditional -> condition
      BUT rational -> rational
```

Step 3*
```
(m>0) ICATE -> IC triplicate -> triplic
(m>0) ATIVE -> '' formative -> form
(m>0) ALIZE -> AL formalize -> formal
```

Step 4*
```
(m>1) AL -> '' revival -> reviv
(m>1) ANCE -> '' allowance -> allow
(m>1) ENCE -> '' inference -> infer
```

Step 5a*
```
(m>1) E -> '' probate -> probat BUT rate -> rate
```

Step 5b*
```
(m > 1 and *d and *L) -> single letter
      controll -> control
```

There is need for more robust and more sophisticated (e.g. language-independent) methods.

* Not exhaustive

d-fine

Statistical approaches may also be used to reduce the dimensionality (1/2)

“You shall know a word by the company it keeps” (J. R. Firth 1957)

- » The discrete word representation implies for some tasks an unfeasible dimensionality, i.e. natural languages contain 10.000+ distinct words per language (and still counting)
- » However, the dimensionality may be reduces by adding structure to text since
  - › some completely distinct words may have (exactly/almost) the same meaning (synonyms), and
  - › some word groups have the same word stem (conjugation, declension) (-> see also Porter stemmer)
- » Instead of capturing co-occurrence counts directly predict surrounding words of every word
- » Example for a co-occurrence matrix with window 1:

I like deep learning
I like NLP
I enjoy flying
I enjoy NLP, too

Counts	I	like	enjoy	deep	learning	NLP	flying	too
I	0	2	2	0	0	0	0	0
Like	2	0	0	1	0	1	0	0
Enjoy	2	0	0	0	0	1	1	0
Deep	0	1	0	0	1	0	0	0
Learning	0	0	0	1	0	0	0	0
NLP	0	1	1	0	0	0	0	1
flying	0	0	1	0	0	0	0	0
too	0	0	0	0	0	1	0	0

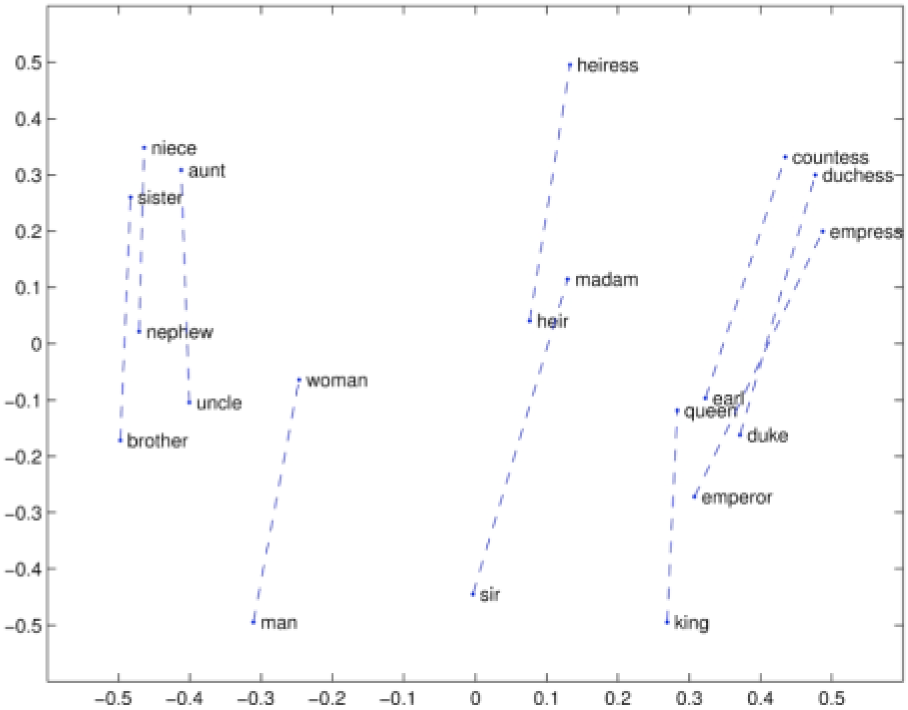
“like” and “enjoy” may be synonyms

- » Some terms consists of more than one word (e.g. machine learning, banking crisis) so that larger windows may be more appropriate.

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation." *EMNLP*. Vol. 14. 2014.
Collobert, Ronan, et al. "Natural language processing (almost) from scratch." *Journal of Machine Learning Research* 12.Aug (2011): 2493-2537.

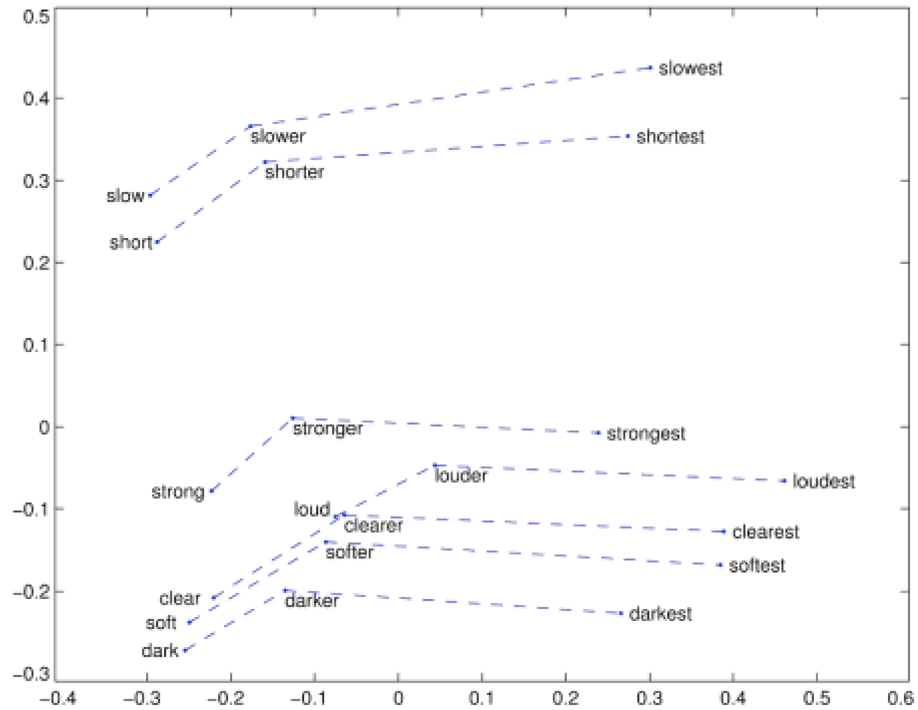
Statistical approaches may also be used to reduce the dimensionality (2/2)

Gender analogies



$$v(king) - v(man) + v(woman) \simeq v(queen)$$

Degrees of adjectives



$$v(slow) + v(1) \simeq v(slower)$$

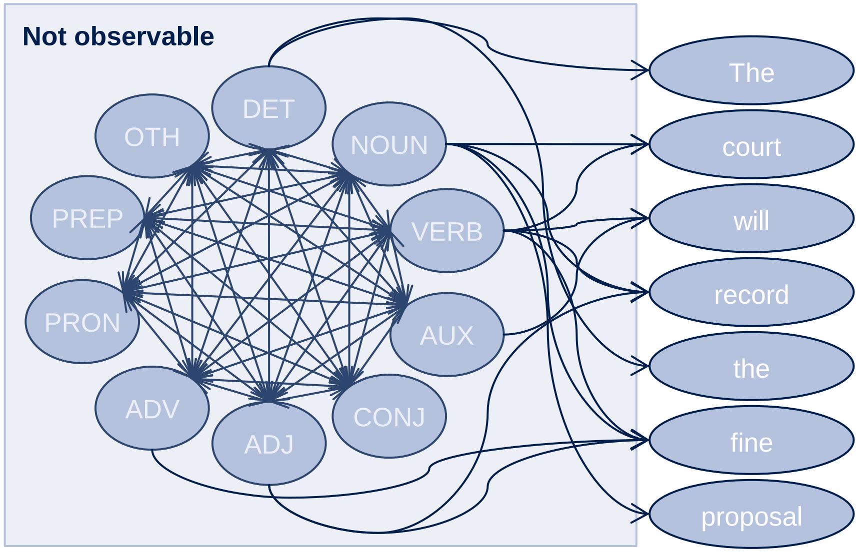
Irrelevant dimensions may be removed.

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation." *EMNLP*. Vol. 14. 2014.

More structure can be added to text by statistical Part-Of-Speech tagging

Hidden-Markov-Model for Part-Of-Speech tagging

- The meaning of a word may depend on its function in the sentence (e.g. court, will, fine, sound, record)
- Consider a hidden stochastic process for word tags (e.g. the Brown corpus). Each tag emissions an observable word if the process arrives.
- Based on a tagged trainings corpus, the **transition** and **emission probabilities** can be estimated.
- Given a word sequence, the most likely path of the hidden process is used for tagging the words.



COURT

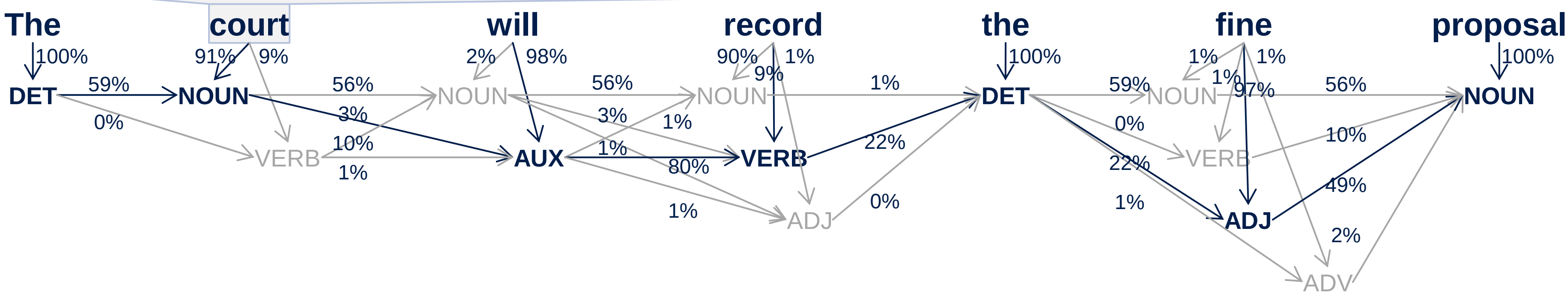
Noun

| 26% idiom-noun: "Supreme court"

| 65% noun-adj: A place where trials are held and the law is carried

VERB

| 9% verb: To woo or try to get, seek



Popular sentiment dictionaries are a good starting point but might be to general / not specific enough for some purposes

1 General Inquirer

- General purpose, 182 categories (e.g. Positive, Negative, Hostile, Strong, Power, Weak, Active, Passive)
- the dictionary also contains part-of-speech tags for each word (e.g. Noun, CONJ, DET, PREP)
- Available fee of charge via http://www.wjh.harvard.edu/~inquirer/

2 Sentiment Word Lists

- Financial / economic background, i.e. constructed in 2009 with 10-K fillings
- 6 categories (Litigious, Negative, Positive, Strong, Uncertainty and Weak)
- Available free of charge via http://www3.nd.edu/~mcdonald/Word_Lists.html

3 Subjectivity Lexicon

- General purpose, contains 3 categories (positive, neutral and negative)
- Available free of charge via http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

4 Diction 5 / 7

- Contains 33 word-categories (e.g. Accomplishment, Aggression, Centrality) and 6 variables based on count ratios in the word categories
- the software is proprietary, see http://www.dictionsoftware.com/

5 Linguistic Inquiry & Word Counts

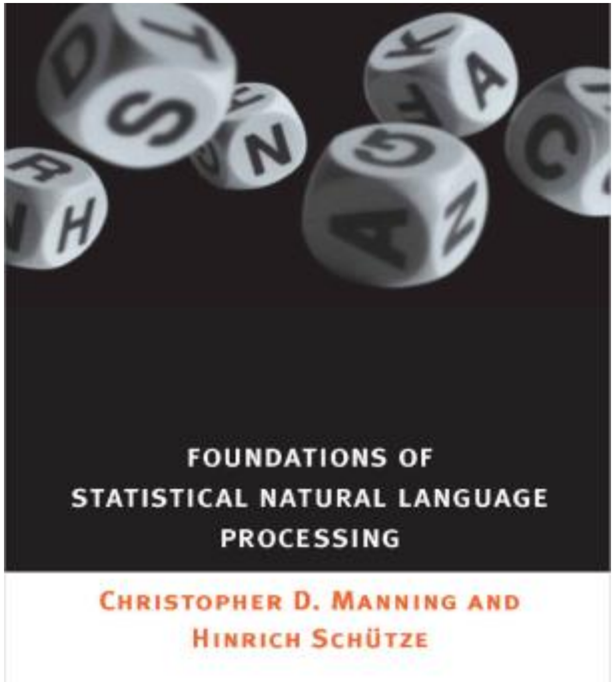
- Social and psychological background, 64 hierarchical word lists and summary statistics
- the software is proprietary, see http://liwc.wpengine.com/

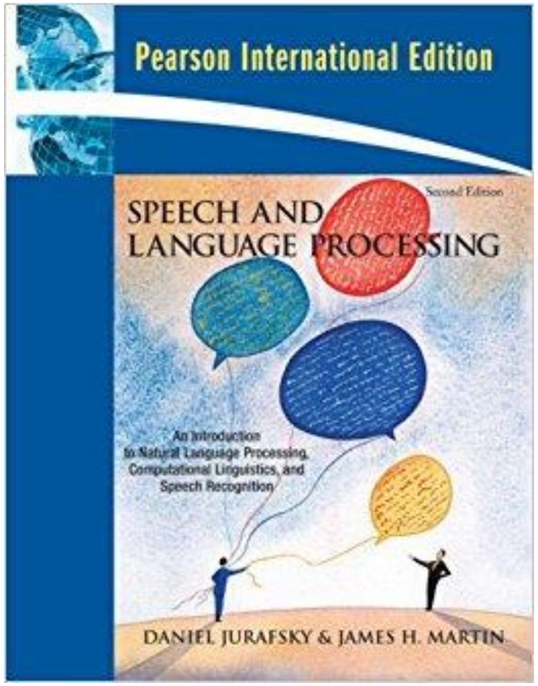
6 Build your own

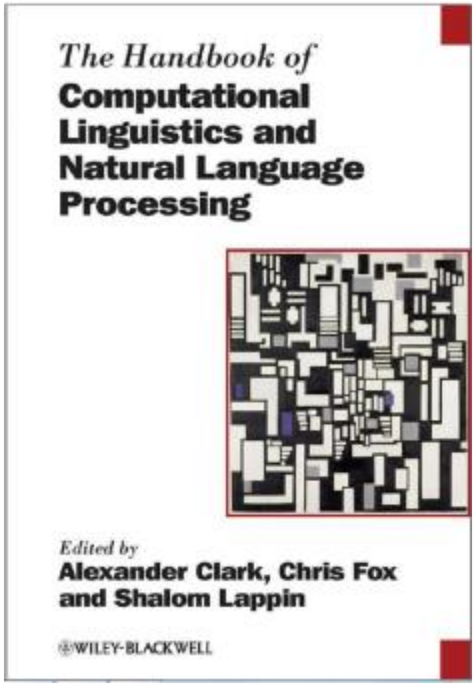
- Based on
  - expert knowledge
  - trainings-set, e.g. find the words with the strongest discriminant power
- Use non-dictionary based classification methods like
  - K-nearest-neighbour
  - Support vector machines
  - Naïve Bayes
  - Maximum entropy

d-fine

For more on text analysis (e.g. machine translation, information retrieval, natural language generation) see







Network analysis in a nutshell

Some general remarks on graph theory and network analysis

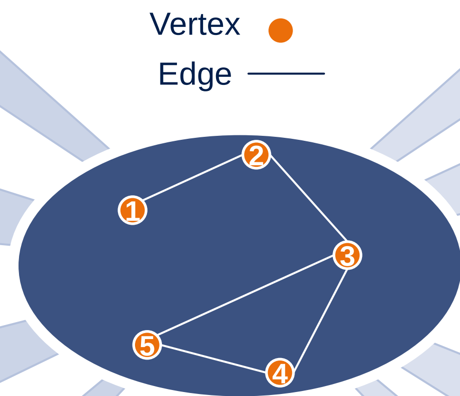
Network properties*

1 Directed vs. non-directed connections

2 Weighted vs. non-weighted connections

3 Self-references, cycle, threes and subgraphs

4 Different kind of edges and vertices



Measures*

a **Degreeness:** Weighted sum over direct connections

b **Closeness:** Average inverse geodic distance

c **Betweenness:** Number of shortest paths a vertex is part of

d **Page-Rank Centrality:** Importance is recursively given by liked vertex

e **Clustering:** Pairs of direct connections, that are directly connected

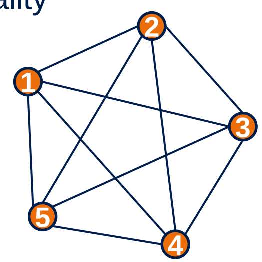
* Not exhaustive

d-fine

Basic structures and properties of graphs

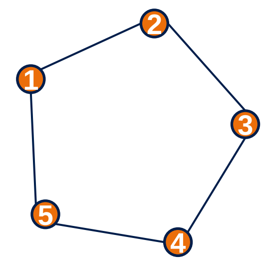
1 Complete Graph

- The complete graph is a graph where all possible edges are present.
- All vertexes are equivalent and have
  - the highest possible degree and closeness centrality
  - but also the highest possible clustering.



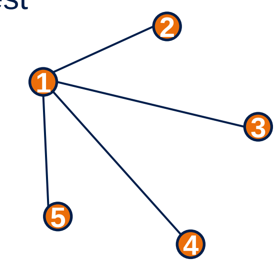
2 Circle

- Each vertex Is connected to exactly two other vertexes
- All vertexes are equivalent having
  - low centrality and
  - low clustering



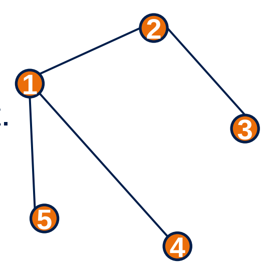
3 Star

- There exists a vertex (the center) such that every edge in the network involves the vertex
- The center has the highest possible centrality but lowest clustering



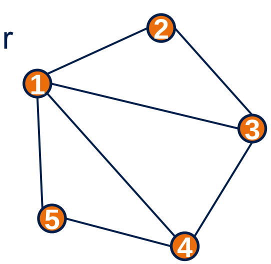
4 Tree and Forest

- A connected graph that that has no cycle. ⇔ its number of edge is smaller that its number of vertexes.
- There is a unique path between any two vertexes
- A graph consisting of more that one tree is called forest.



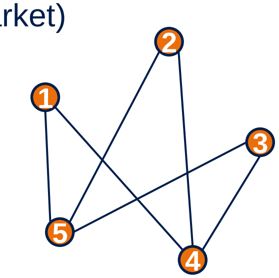
5 Planar

- The graph can be drawn on a piece of paper without having any two edges cross each other (edges only intersect at one of their involved nodes)
- Related to the "four colour theorem"



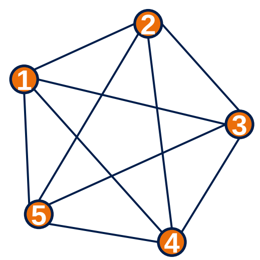
6 Bipartite Graph

- There exists a partition of all vertexes into two groups so that vertexes within a group are not connected
- Related to matching problems (e.g. marriage market)



Each graph can be represented by an quadratic (so called) adjacency matrix

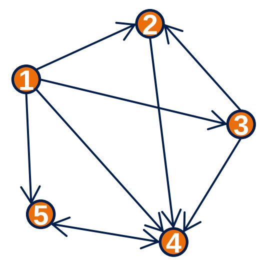
Non-directed & discrete



	1	2	3	4	5
1	0	1	1	1	1
2	1	0	1	1	1
3	1	1	0	1	1
4	1	1	1	0	1
5	1	1	1	1	0

Symmetric & binomial

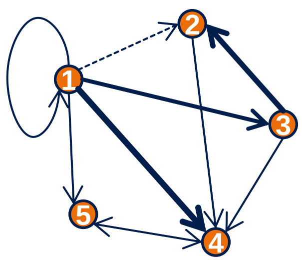
Directed & discrete



	1	2	3	4	5
1	0	1	1	1	1
2	0	0	0	1	0
3	0	1	0	1	0
4	0	0	0	0	1
5	0	0	0	1	0

Binomial

Directed & weighted

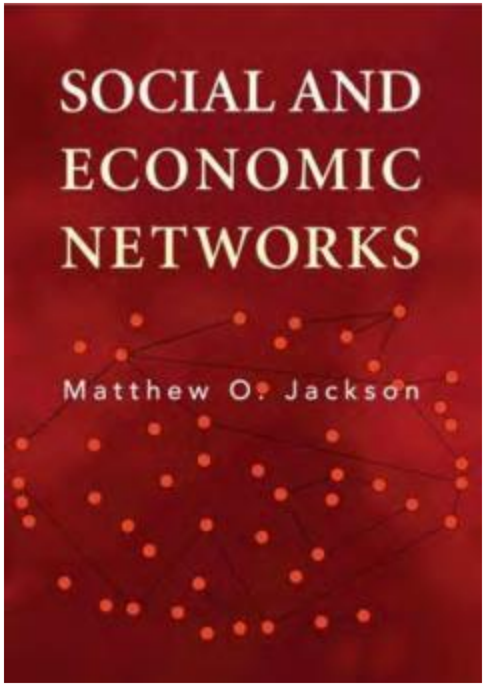


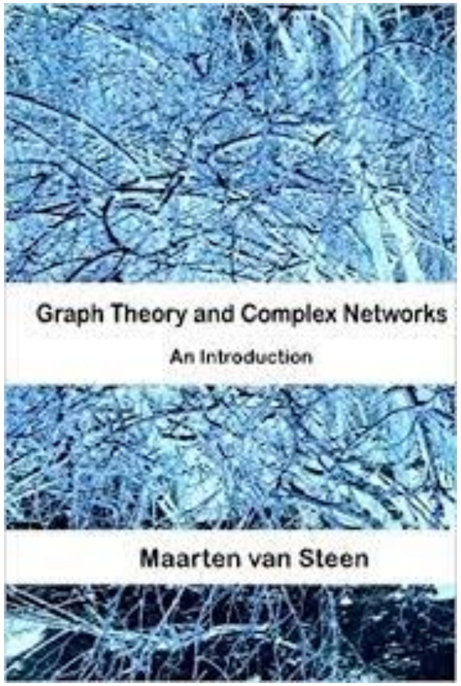
	1	2	3	4	5
1	0,9	1,1	2,1	4,3	1,9
2	0	0	0	0,9	0
3	0	3,2	0	0,8	0
4	0	0	0	0	1,2
5	0	0	0	1,2	0

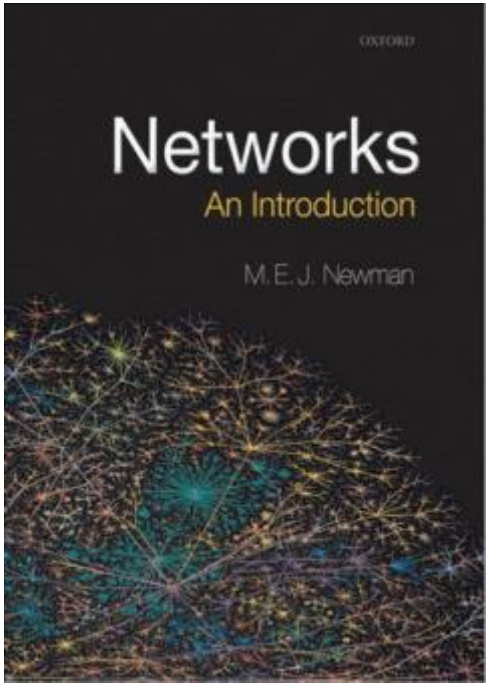
Real valued

Networks may be modelled stochastically (vertexes or edges may be added or removed), simulated and analysed e.g. w.r.t. stability or association.

For more on graph theory (e.g. Hamiltonicity, network models, random graphs and simulation) see







Business cases

Business news in credit risk management

Rating models are intended to measure the probability of default and can divided into three major groups

Default rating

- This rating-type measures the probability of default (usually within the next 12 months after the rating) based on **(internal) default observations**.
- The rating model is usually set-up as logistic regression or decision tree.
- This kind of rating model is often applied to retail and SME portfolios with many observations (incl. defaults).

Shadow rating

- The shadow rating tries to replicate the rating grades of **external rating agencies** (Moody's, S&P, Fitch) in order to measure the credit worthiness of obligors without external rating.
- The rating is usually applied to (large) corporations, financial institutions and sovereigns portfolios with little observations and infrequent defaults.

Cash-flow-rating

- The rating model is based on **Monte-Carlo simulations**, which describe the evolution of the relevant economic derivers for the obligor.
- The probability of default is derived from the number of paths where the credit cannot be repaid and the obligor defaults.
- The rating is applied to project-like obligors (e.g. wind or solar power station, cargo carrier).

Usage of ratings

- Ratings and, hence, the estimated probability of default (PD) are crucial
  - to calculate the interest rate and other credit conditions (e.g. collateral) for the obligor at contract agreement;
  - for the calculation of the regulatory capital requirements (esp. IRBA) and
  - for (regulatory) reporting.
- Other aspects regarding the loss-given-default (LGD), exposure-at-default (EAD), credit-conversion-factor (CCF) or risk mitigation techniques are not covered here.

State-of-the-art platforms for (corporate) ratings are usually modular and cover different areas of information

Risk factors are statistically aggregated to a rating grade

specific

Micro enterprises	Small and mid enterprises	Large corporation
Turnover ≤ 50’’ EUR	Turnover ∈ (50’’ EUR; 1’’’ EUR)	Turnover ≥ 1‘‘‘ EUR

» Statistically optimized combination of module scores
» Manual adjustments due to obligor-specific circumstances if necessary

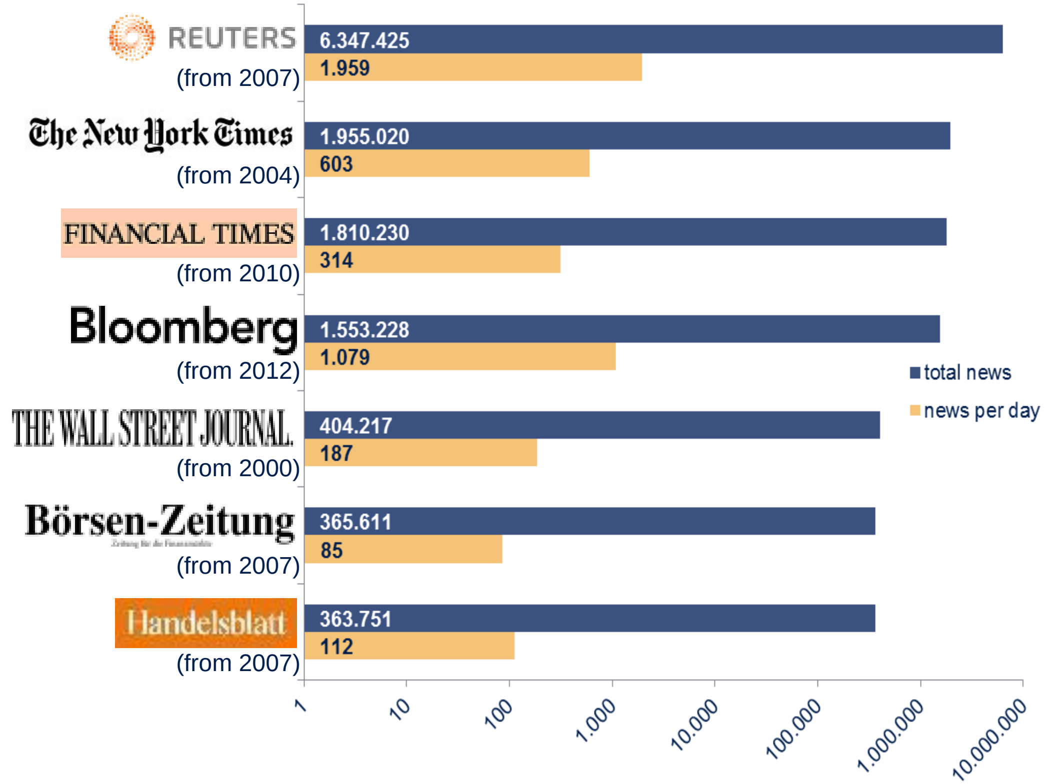
generic

Static obligor information	Payment history	Accounting figures	Qualitative information	Capital market information	Unstructured information
Founding date	Amount past due	Revenue growth	Management quality	Market value of equity	Reuters, BBG, FT Nachrichten
Legal form	Limit usage	Capital structure	Market position	Volatility.	Twitter
Number of employees	Current balance	Net profit ratio.	Business development.	Trading volume	Facebook
…	…	…	…	…	…

The rapidly increasing amount of information is a challenge in banking and forces financial institutions to redefine processes and concepts

On a regular business day 5,000+ news articles were published by the major agencies

- The 7 news provider produced 13,000,000 news articles with about 400 words each. Local news and social media is excluded.
- Printed out and stacked, the news articles would form a 1,3 km high pillar (that is about the same as the worlds highest building (Burj Khalifa) on top of the second highest building (Shanghai Tower)).
- The news articles need 20 gigabyte disk space and each is analysed w.r.t. the corporations listed in a major stock marked index (e.g. S&P500).



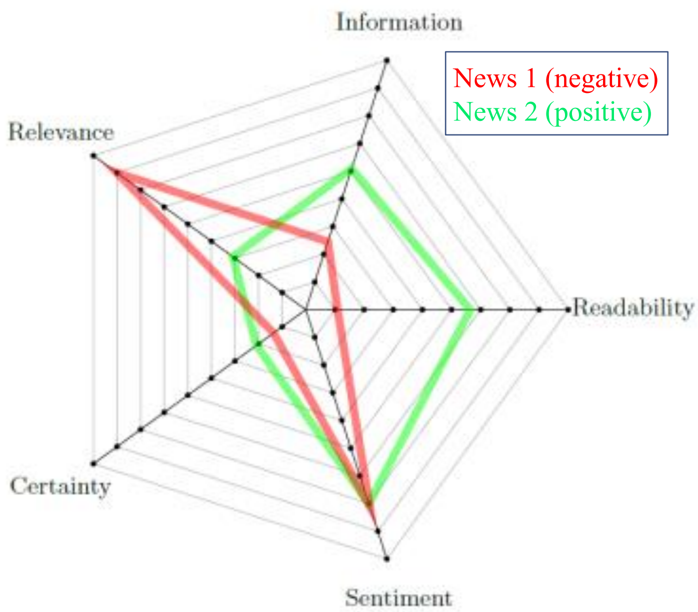
The evaluation of a representative set of business news for a portfolio is computationally demanding.

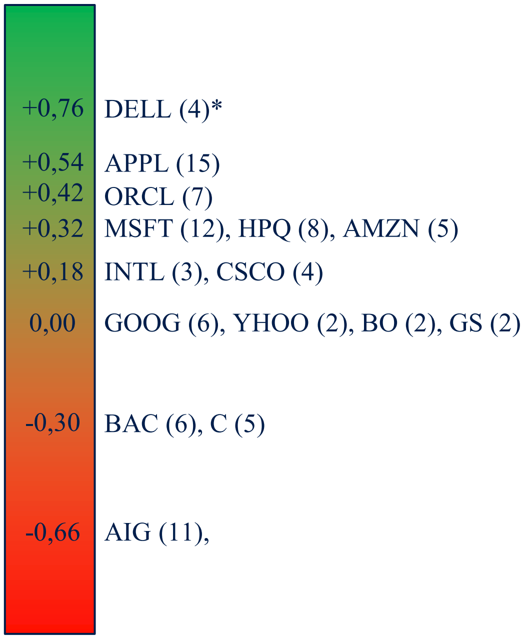
d-fine

For each business news and each corporation five independent indicators are considered aggregated over time to one news signal per company

Each indicator controls for a specific semantic dimension which may be considered by the market

- **Information:** Comparison of news with news published before in order to recognize recurring news.
- **Relevance:** Measures to which degree the news is focused on the considered company.
- **Sentiment:** Measures the degree of the content's positivity / negativity for the considered company.
- **Commitment:** Measure if the article contains final and/or certain information vs. guessing.
- **Readability:** Measures the complexity of the language.





d-fine

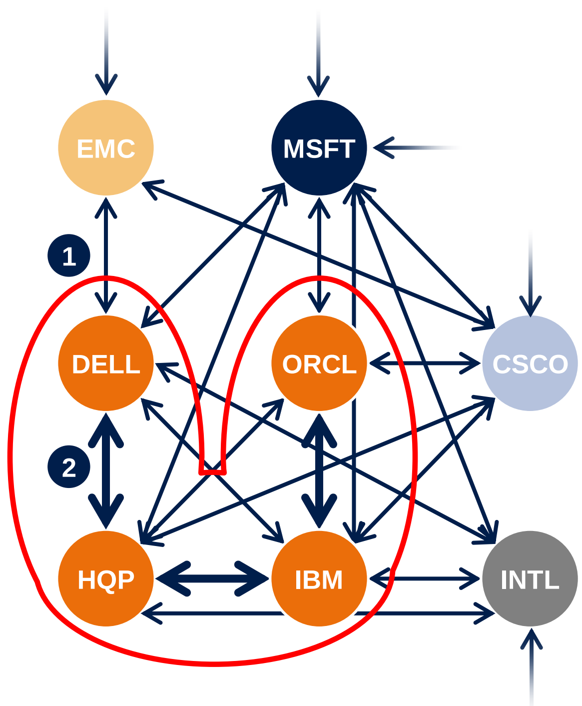
Business news allow to derive a network for corporations, identify groups and to assign an importance measure to each corporation (1/2)

The news-based network may be an easy approximation for the economy

- Corporations that are mentioned together in business news frequently are likely to be related in the business world

- We can establish connections between corporations based on their co-occurrence in business news (see e.g. 1).
- Connections may be of individual strength (see e.g. 2).
- Based on the connection strength, corporation-groups can be identified (see).
- Each corporation may have an individual importance for the economy, which corresponds to its connections and its position in the network.

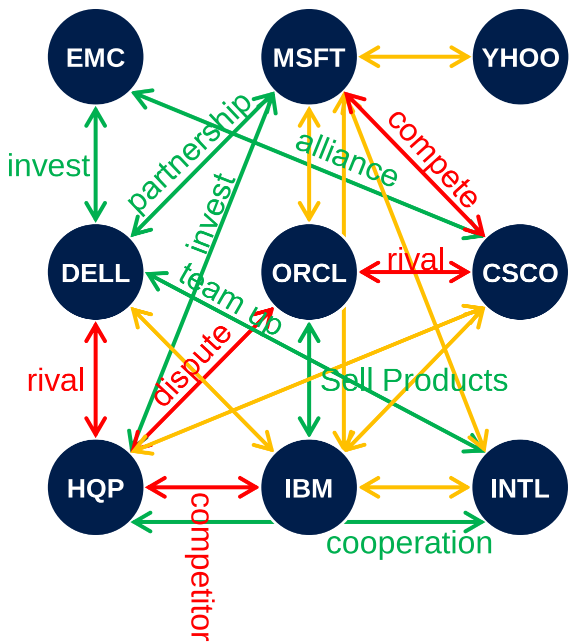
- Furthermore, a connection may be interpreted with respect to
  - direction and impact (positive / negative)
  - the node (e.g. financial institution, sovereign or person)

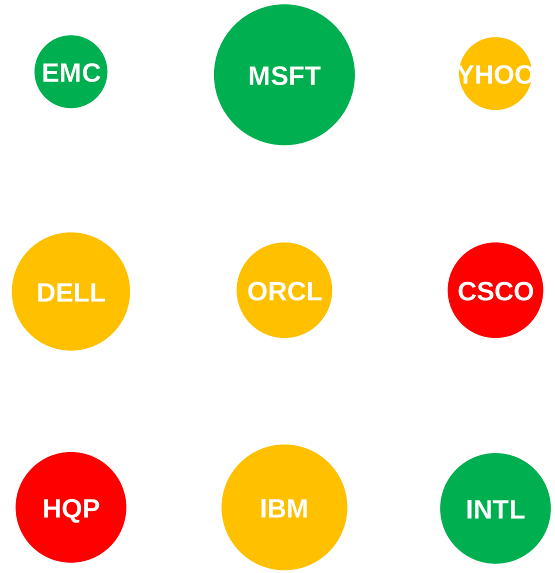


d-fine

Business news allow to derive a network for corporations, identify groups and to assign an importance measure to each corporation (2/2)

Based on the words in the business news, the connection may be classified





- Conflicting connections dominate
- Supporting connections dominate
- Connections are unclear

The radius indicates the importance (i.e. centrality) of a corporation

Corporations are interpreted a part of the economy and analyst can easily follow-up on the near neighbors of a corporation.

d-fine

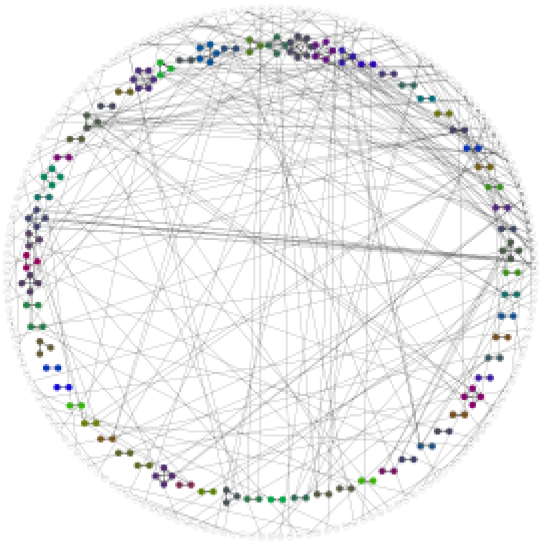
The network analysis and the news signal can directly be integrated in rating- and early-warning-system

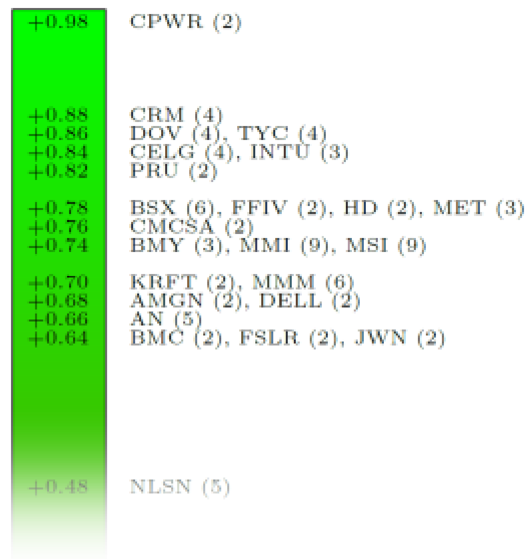
- » We consider a shadow rating model based on external ratings from the 3 major rating agencies.
- » The rating grades were numbered in sequence from good to bad. After this a Box-Cox transformation was applied to account for the risk-increasing nature of most rating scales.
- » We test the relationship between the creditworthiness of a company and its
  - › network properties, quantified by 4 centrality measures and the clustering, and
  - › medial situation, measured by the news signal and attention (i.e. log of number of news).
- » Control variables cover all traditional financial indicators (e.g. RoE, Debt to Equity) and market indicators (stock return, stock return volatility).
- » The results indicate that both kind of information add significant explanatory power.

	Factor	Coef. (P-val)	Coef. (P-val)	Coef. (P-val)	Coef. (P-val)	Coef. (P-val)	Coef. (P-val)
Network	Degreeness	--	--	1,7789 (0,0000)	--	--	--
	Closeness	--	--	--	1,9010 (0,0003)	--	--
	Betweenness	--	--	--	--	1,9428 (0,0000)	--
	Page-Rank	--	--	--	--	--	1,9253 (0,0000)
	Clustering	--	--	-1,4441 (0,0000)	-2,2675 (0,0000)	-1,0238 (0,0013)	-1,5405 (0,0000)
News	News Signal	--	0,7074 (0,0333)	0,9968 (0,0028)	0,9627 (0,0030)	0,9194 (0,0042)	1,0054 (0,0025)
	Attention	--	1,3173 (0,0000)	0,4186 (0,1071)	0,9363 (0,0004)	0,8554 (0,0005)	0,4066 (0,1195)
Controls	Stock return	0,0737 (0,7503)	0,0789 (0,7281)	0,1388 (0,5202)	0,0818 (0,7115)	0,2116 (0,3318)	0,1361 (0,5262)
	Stock return volatility	-0,3806 (0,0000)	-0,4005 (0,0000)	-0,3715 (0,0001)	-0,3746 (0,0001)	-0,3475 (0,0001)	-0,3690 (0,0001)
	Return on equity	0,0213 (0,0000)	0,0189 (0,0000)	0,0179 (0,0000)	0,0176 (0,0000)	0,0183 (0,0000)	0,0181 (0,0000)
	Free cash flow to sales	0,0287 (0,0000)	0,0232 (0,0000)	0,0233 (0,0000)	0,0249 (0,0000)	0,0246 (0,0000)	0,0239 (0,0000)
	Debt to equity	-0,5410 (0,0000)	-0,5163 (0,0000)	-0,4834 (0,0000)	-0,4870 (0,0000)	-0,5090 (0,0000)	-0,4881 (0,0000)
	Short term to total debt	0,0527 (0,0076)	0,0441 (0,0186)	0,0426 (0,0154)	0,0440 (0,0133)	0,0462 (0,0058)	0,0429 (0,0144)
	3 year revenue growth	0,0199 (0,0031)	0,0174 (0,0061)	0,0175 (0,0043)	0,0162 (0,0099)	0,0161 (0,0076)	0,0173 (0,0047)
	Adj. R2	0,3184	0,3612	0,3958	0,3851	0,4156	0,3970

Traditional rating systems can be improved significantly by news and network information.

User-friendly and interactive visualizations allow an efficient monitoring of many communication channels





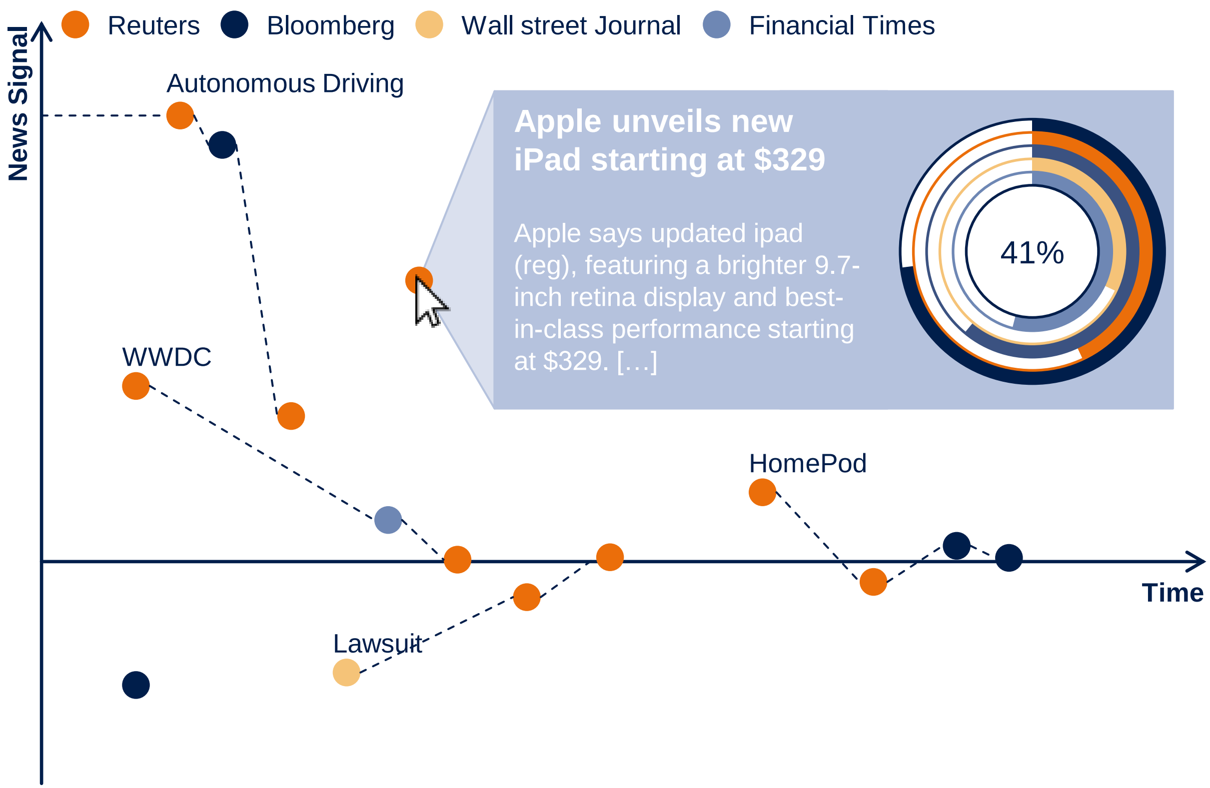
low — level of detail — high

Overview	Analysis	Content
The graph shows all corporations, and all connections between them. corporations with strong connections build groups which are coloured identically. A connection between two entities is established if both entities appear together in a significant number of news. Subgraphs can be selected.	All corporations are listed and compared based on **aggregated text-analytics results** for the business news.	Given a corporation, the content of all corresponding news is shown e.g. with a word-cloud or word highlighting. It enables the analyst to **understand and validate** the text-analytics result, and to give **feedback** on it (supervised learning). The feedback is automatically incorporated by subsequent text-evaluations.

d-fine

Ergonomic visualization of news give a comprehensive overview and allow to zoom in if needed

News are presented so that analysts can review then, understand the method and adjust it if necessary

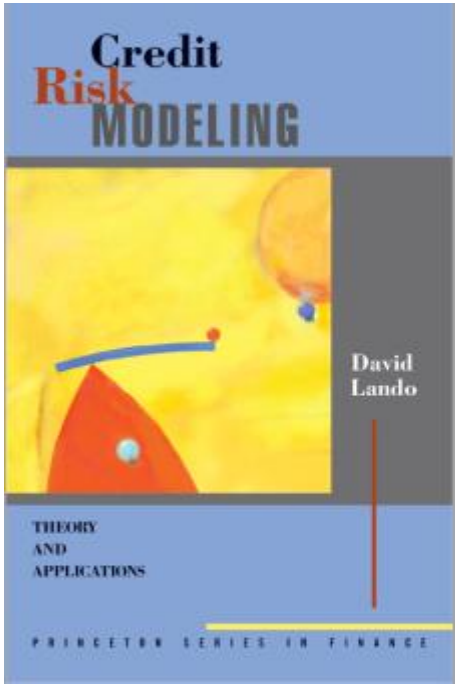


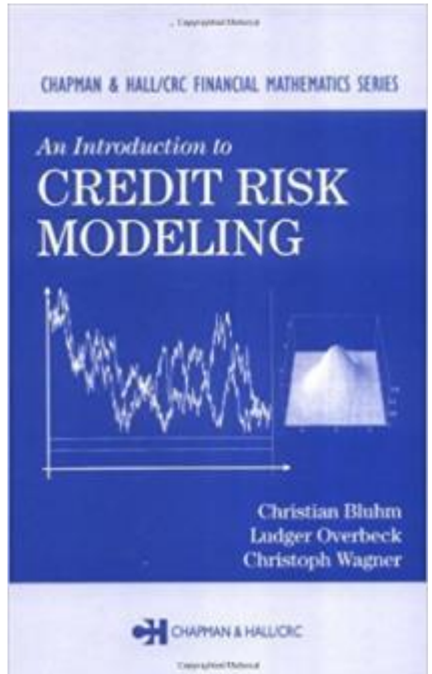
- **A** Model predictions
- **B** Review by a human analyst
- **C** Feedback / Supervised Learning

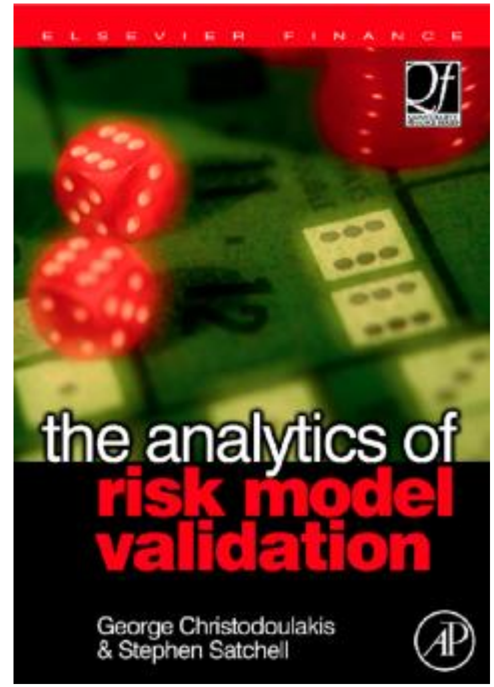
Analysts can actively train the methods so that confidence arises and more processes can be automated.

d-fine

For more on credit risk modelling (e.g. portfolio models, securitization, coherent risk measures) see







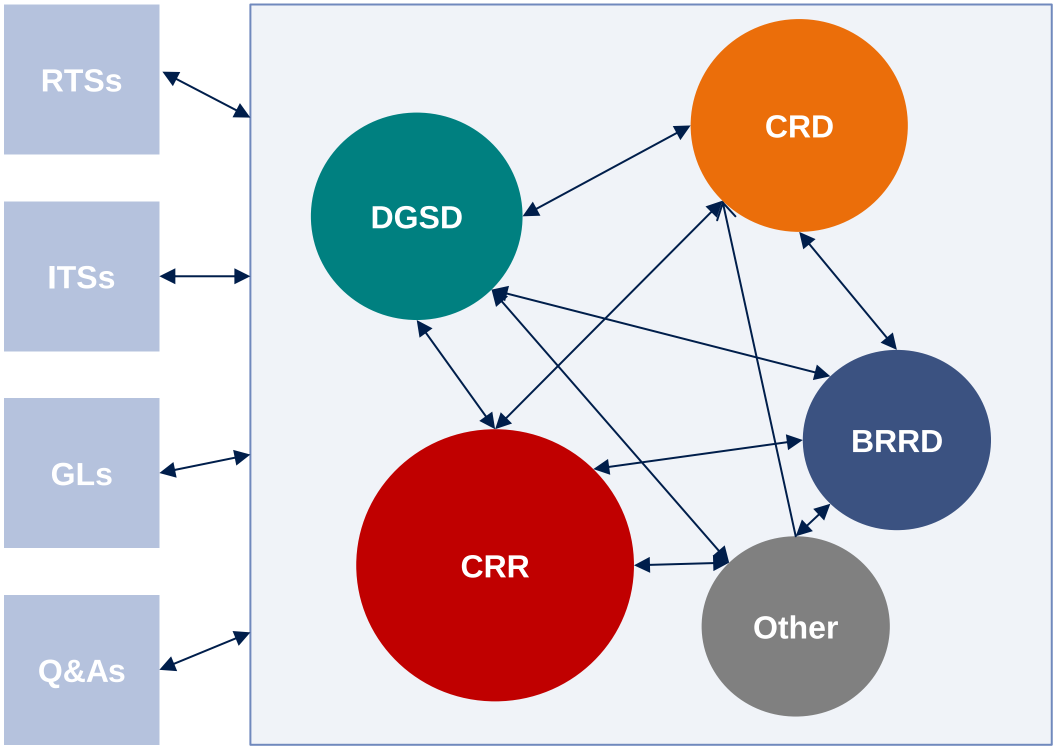
Single rulebook in banking

The complexity of bank regulation increased over the past years giving challenges to familiarize and stay up-to-date

- "The Single Rulebook aims to provide a single set of harmonised prudential rules which [financial] institutions throughout the EU must respect. […] This will ensure uniform application of Basel III in all Member States. […]" (see http://www.eba.europa.eu)
- The European Banking Authority (**EBA**) plays a key role in building up of the Single Rulebook in banking.
- The key elements of the Single Rulebook are:
  - The Capital Requirements Regulation (**CRR**) and the Capital Requirements Directive IV (**CRD IV**)
  - The Bank Recovery and Resolution Directive (**BRRD**) and Deposit Guarantee Schemes Directive (**DGSD**)
  - 90+ supporting documents (and still counting), e.g. Regulatory Technical Standards (**RTS**), Implementing Technical Standards (**ITS**) and Guidelines (**GL**)
- Articles and documents are strongly interlinked and are frequently updated, making it hard to get a general idea on some topic within reasonable time.
- Moreover, there is no possibility for advanced search options since all documents are separately published as pdf or html.

There is definitely need for supporting the work with regulatory text

Banking regulation is governed by a manifold of regulatory texts



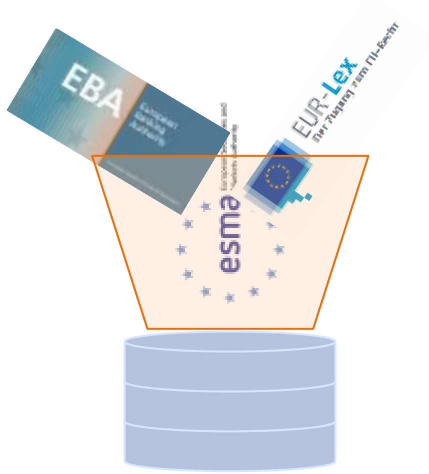
- » Which of these 10 references contains the information I need?
- » How can I jump quickly to a certain Article?
- » Are there any relevant Articles referencing ***to*** this Article?
- » Or is there a RTS/Q&A/… related to this Article?
- » Which Articles are the most important ones from a regulatory framework?
- » Can I search within a certain scope this regulation?
- » …

How can text analytics help us with these issues when reading legal texts?

Building a single rulebook for banking regulation

1 Crawl

- Crawl relevant legal texts from various sources (EU lex, EBA homepage, …)
- Manual effort required to select relevant sources and adapt crawler
- Automatic download of all text from given source

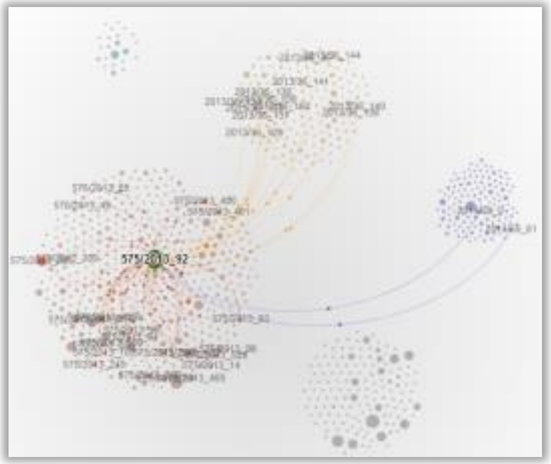


2 Parse

- Parse texts for references to other articles
- Build database of all relevant texts including these references

3 Display

- Provide a user-friendly reading application
- Easy navigation through all relevant texts
- Show relevant additional information to each Article (e.g. RTS or Q&A)

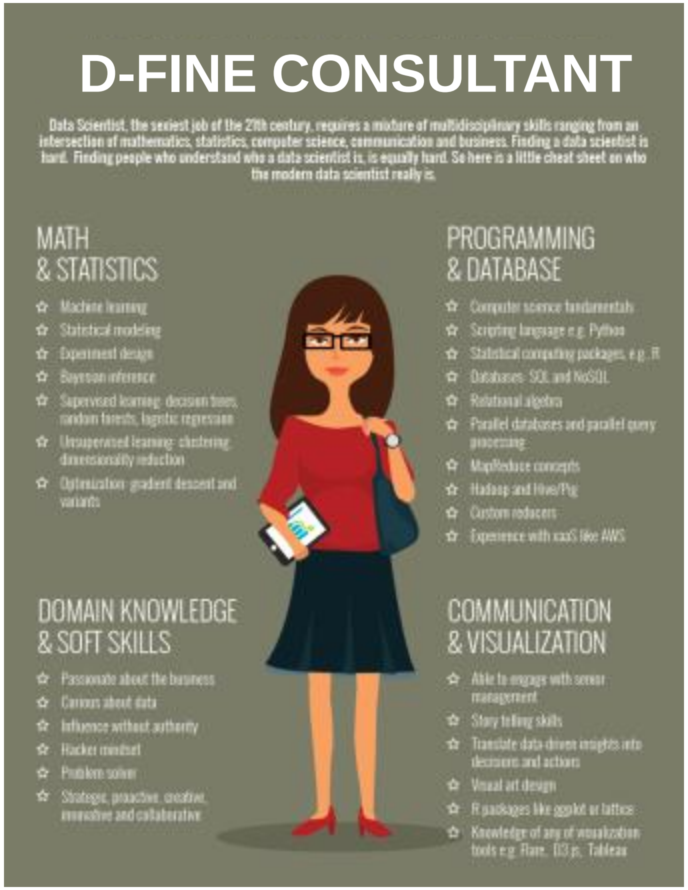


d-fine

For more on banking supervision and regulation see

Concluding remark

Data-centric project are one of d-fine's core competencies and will gain importance in the near future



Fraud detection and prevention*

By using big-data solutions, typical customer behaviour patterns can be identified and evaluated. In this way suspicious account activities can be detected and prevented early on.

Compliance and reporting*

The regulatory requirements for banks and insurance companies are higher than ever before. "Big Data" allows the recording and control of trading activities and helps banks meet their reporting requirements.

Customer segmentation*

Big Data helps companies to segment their customer base more accurately. By analysing existing sales and marketing activities, these can be made more targeted and effective.

Personalized products*

Financial service providers can benefit from the increasing digitalization of their business. In this way, product offers can be personalized and intelligently priced through the real-time analysis of clickstream and geo-location data.

Fine-grained risk management* and trading

Banks and other financial service providers are exposed to a multitude of risks that must be mastered. The inclusion of large amounts of data improves the results of scenario simulations and thus facilitates companies to recognize risks and react quickly to market developments.

* Projects are already in the pipeline or pitched

Dr Ferdinand Graf
Manager
Mobile +49 162 2630080
E-Mail Ferdinand.Graf@d-fine.de

Dr Ulf Menzler
Senior Consultant
Mobile +49 152 57975056
E-Mail ulf.menzler@d-fine.de

d-fine

Frankfurt
Munich
London
Vienna
Zurich

Headquarters

d-fine GmbH
An der Hauptwache 7
60313 Frankfurt/Main
Germany

Tel +49 69 90737-0
Fax +49 69 90737-200

www.d-fine.com

