



Prof. Dr. Adrian Ulges

Empolis Workshop “Machine Learning”

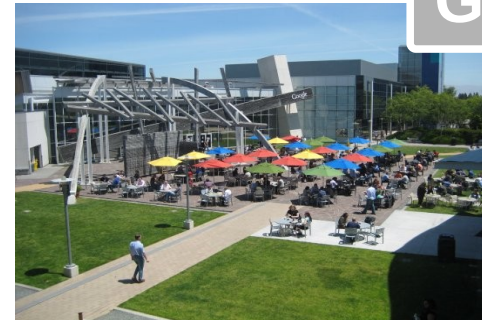
Einführung

Hochschule RheinMain
Department DCSM (*Design, Computer Science, Media*)

DFKI



Google

HSRM
(seit 2013)

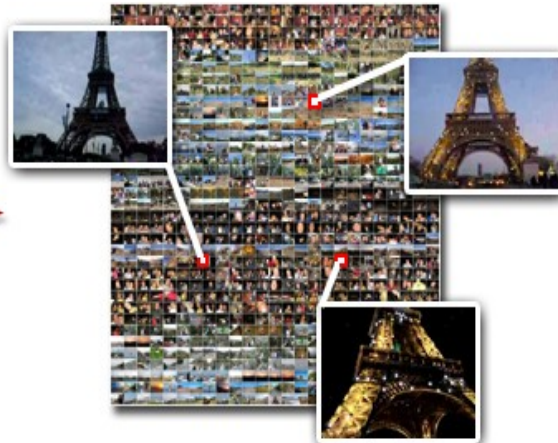
Quintiq



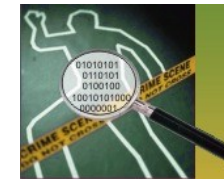
Machine Learning

Multimedia-Analyse

Social Multimedia



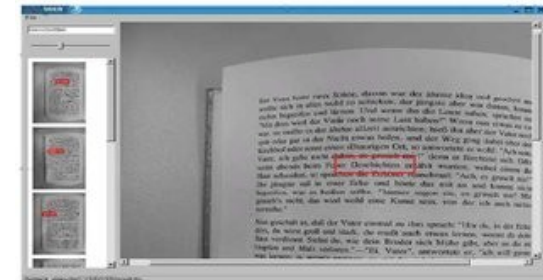
Multimedia-Forensik



MiFor 2011

*Multimedia in Forensics
and Intelligence*

Dokumentbildanalyse



Interaktive Vorlesungsvideos

Hochschule RheinMain
University of Applied Sciences
Wiesbaden / Rüsselsheim

Home Vortragsverwaltung

Analysis
Computer Vision

Motivation May 12, 2015, 9:04 a.m. Regula Falsi May 12, 2015, 9:08 a.m. Bisektion May 12, 2015, 9:13 a.m. Newton's Method May 27, 2015, 9:19 a.m. Taylor Series May 27, 2015, 9:27 a.m.

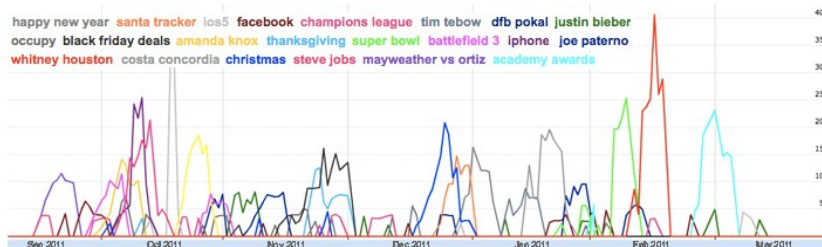
Die Idee des Bisektionsverfahrens

- f sei im Bereich $[a, b]$ stetig.
- Es gelte: $f(a) < 0 < f(b)$ oder $f(a) > 0 > f(b)$

Die Bisektionsverfahren (Pseudo-Code)

1. Bestimme den Mittelpunkt $m = (a+b)/2$.
2. Berechne $f(m)$.
3. Wenn $f(m) = 0$ oder $|f(m)|$ klein genug ist, dann ist m die Nullstelle. Sonst gehe zu Schritt 1.

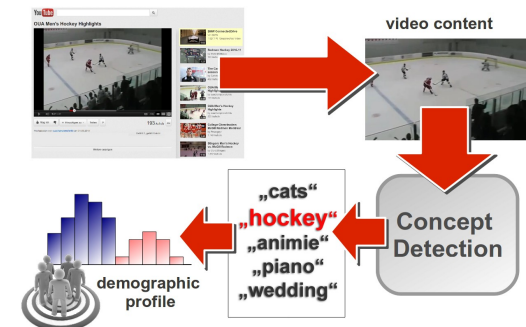
Trending Topics



Skalierbare Bildsuche

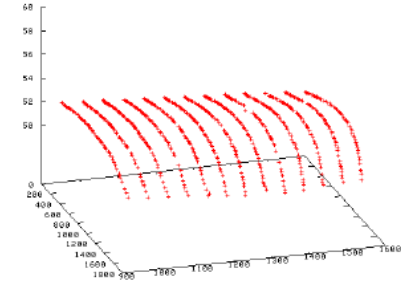
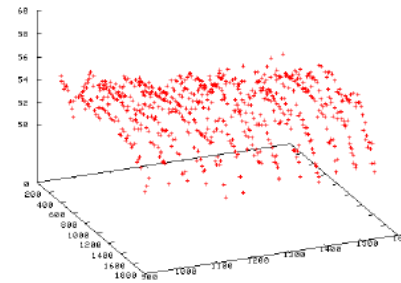


Demographics Prediction

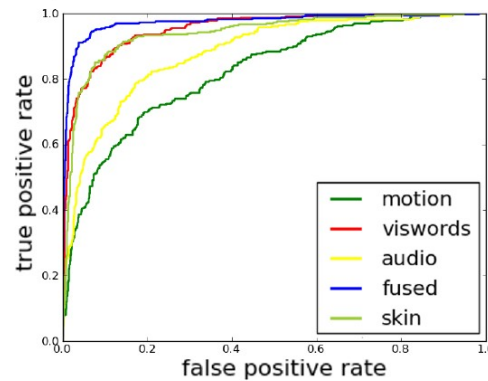
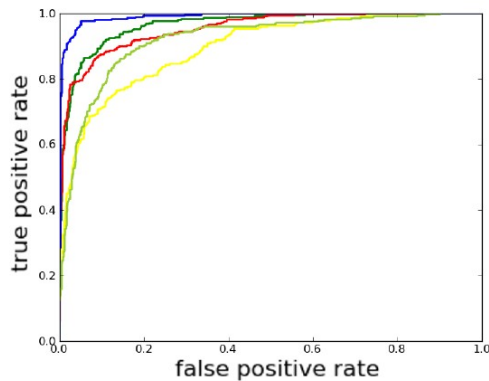


Kamera-basierte Dokumentenerfassung

by my good master, Mr.
Pannel, commander; with



Detektion von CSA



Software Repository Mining

▶ Machine Learning 101

- Grundlagen, Begriffsbildung
- Design-Cycle, Benchmarking, Cross-Validation
- Python-Beispiel: Titanic

▶ Klassifikation

- Probabilistische Klassifikation (Naive Bayes)
- Maximum Entropy
- Entscheidungsbäume und Random Forests
- Feature Engineering
- Python-Beispiel: News-Klassifikation

▶ Clustering

- Clustering im Vektorraum: K-Means, EM
- Model Selection (→ Anzahl der Cluster)
- Agglomeratives Clustering, Topic Modeling
- Python-Beispiel: News-Clustering

▶ Machine Learning 201

- Active Learning
- Anomalie-Erkennung
- Python-Beispiel: Active Learning für News-Daten

- ▶ **R**
 - mächtiges Data Science - Werkzeug
 - Schwerpunkt: explorative Statistik
 - eher für Stand-alone-Analyse

▶ Python

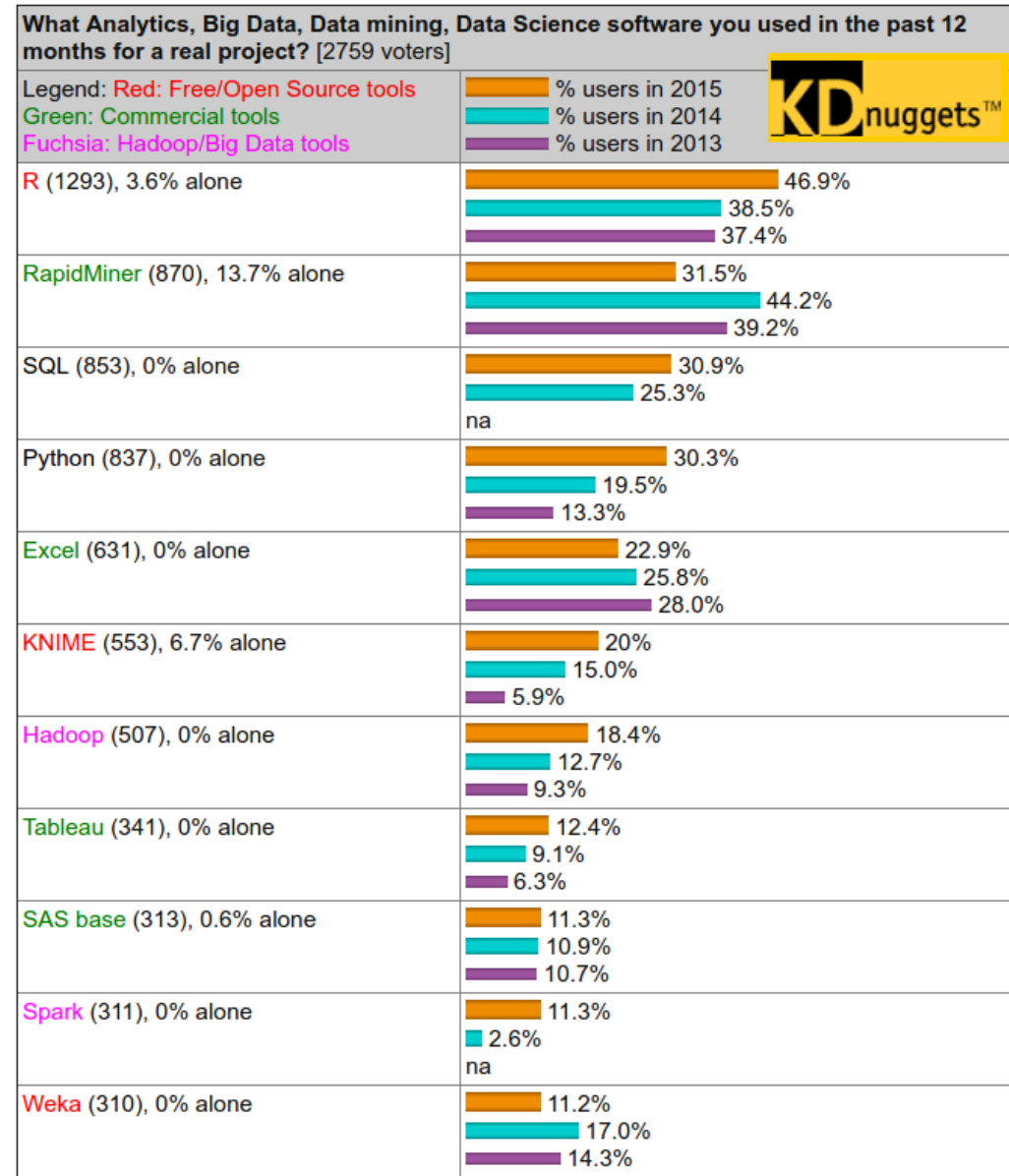
- Funktionalität ähnlich zu R
- Volle OO Skriptsprache (#5 im Tiobe-Ranking)
- optimal für Integration mit Produktivsystemen

▶ Rapidminer

- Java
- basic edition open-source, professional closed-source

▶ matlab

- kommerziell, proprietär
- offene Alternative: Octave



► Python

- Objektorientierte Skriptsprache (“Perl by a sane person”)
- einfach erlernbar (“pseudo-code that runs”)
- interaktiv, open-source
- ideal für **rapid prototyping**
- plattformunabhängig, interpretiert, Bytecode (JIT mit pypy)

► Features

- **Dynamische Typisierung**
- Garbage Collection
- eingebaute **Datenstrukturen** (Listen, Strings, Dictionaries)
- hohe Ausdrucksmächtigkeit (**imperativ, OO, funktional**)

► Unterstützung für große Projekte

- Unterstützung für große Projekte (Module, Ausnahmen, ...)
- Integration (C, C++, Java, XMLRPC, REST, SOAP)
- **Bibliotheken** (Statistik, Web, Email, RegExp, Textverarbeitung, Bildverarbeitung, ...)



Python für ML

► numpy

- Support für Vektoren und Matrizen
- Kompakter-Code durch Array-Operationen

► scikit-learn (sklearn)

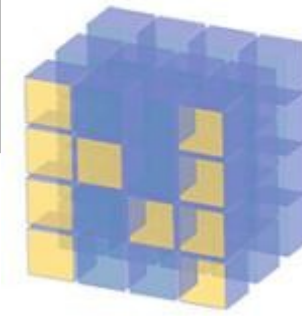
- Machine Learning – Algorithmen
- Klassifikation (SVMs, NN, Logistic Regression, Decision Trees, ...)
- Clustering
- Dimensionsreduktion
- Feature Engineering
- Neuronale Netze

► matplotlib

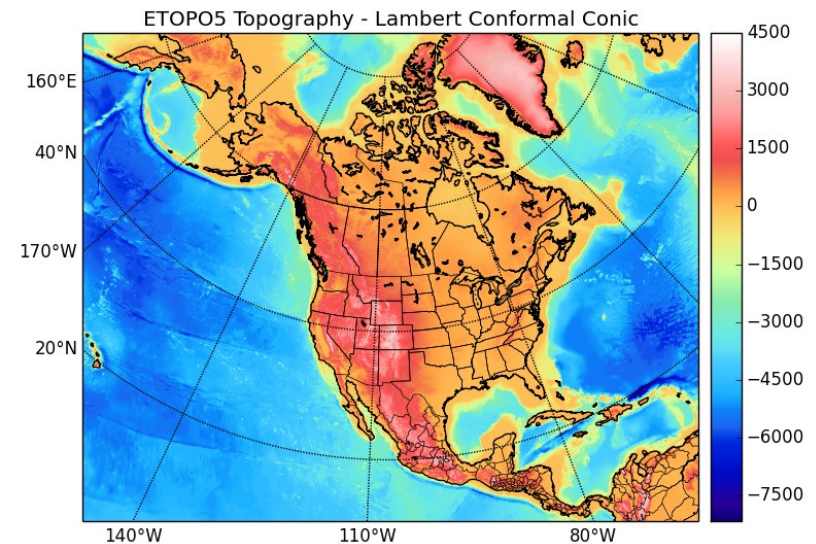
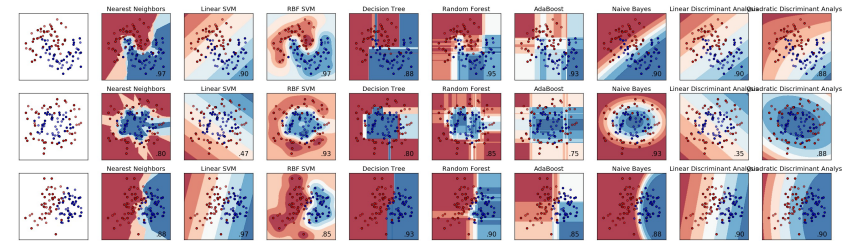
- 2D-Plotting und Visualisierung

► pandas

- Datenanalyse und -Inspektion



NumPy



pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$

