Data

- **Columns**: Variables, feature, attribute
  - Input variable
  - Output variable
- **Rows**: instances, patterns.
- **Variables / features**:
  - **Numerical**: integral, real
    - Signed, unsigned
    - Bounded, unbounded
  - **Categorical**
    - Sorted: Low, medium, high
      : Baby, toddler, child.
    - Unsorted: Color, country, gender.
  - **Textual**
    - Identification of pattern (name, ID).
    - Free text: report, description.
  - **Complex attributes**
    - images, sound, time series.

Problems →

1) Prediction
2) Classification
3) Clustering
4) Dimensionality Reduction
5) Optimization.
6) Encoding
7) Prototyping
8) Visualization
9) Familiarity
10) Feature mapping

Prediction

- Multilinear Regression (MLR)
- Polynomial approximation
- Multilayer Neural Network (NN)

- Non-linear dependencies
- Highly dimensionality $m \gg 1$

Classification

- Binary
- Non-Binary
  - ↳ Tree

- Linear classifiers
- K-nearest neighbous
- Multi-layer NN
- Support Vector Machine
- Decision tree
- Random forest

Clustering

- find Partition belong to
- Similiar pattern

- K-means
- Agglomerative Hierachy clustering (AHC)
- Density models
- Vector Quantization (VQ)
- Self-Organizing Maps (SOM)

Dimensionality Reduction

- Principal Component Analysis (PCA)
- Self learning (supervised) Auto encoder
- t-SNE
- Self-Organizing Maps (SOM)

Optimization

minimize if energy/error func

MAX if fitness function.

- Classical optimization heuristics
  - Gradient Descent
  - Simulated Annealing
  - Tabu Search
  - External Optimization
- Evolutionary computation & Swarm intelligence
  - Genetic Algorithms
  - Particle Swarm optimization
  - Ant Colony Systems
  - Artificial Bee colony
  - Grey Wolf Optimizer
  - Memetic algorithms.

Standard approach

↳ Use optimization heuristics

⬇

→ Approximate Solution ←

Data Preparation

- Instance Selection
- Outlier detection
- Missing values
- Data representation
- Data normalization
- Feature selection
- Feature construction

Instance Selection

- Objective
  - Select patterns to be used, not all patterns
  - Desire pattern could seems too noise
- Approaches
  - Noise filters
  - Condensation / wrapper algorithms
  - Protoype selection

Outlier Detection

- Objective
  - identify patterns with abnormal values compared to the rest of the patterns
- Approaches
  - Identifies the outliers
    - □ Analysis of distributⁿ value of each variable
    - □ Use multivariate outlier detectⁿ method.
  - Action (Similar with missing value)
    - □ ignore pattern
    - □ Assign new value
- Variables by variable
  - If normally distributed, using Z-score.
  - In general, using box & whiskers plot.
- Multivariable analysis.

Missing Value

- Objective
  - Decide how to proceed with patterns having missing value.
- Approaches
  - Ignore value/pattern
  - Assign value to the missing value
    - □ Average
    - □ mode
    - □ new value.
    - □ Median
    - □ Regression

Data Representation

→ Objective
↳ Algorithm need number, no categories or free text.
↳ Subtitute non-numerical value by appropriate numerical representation.

→ Approaches
↳ Finding suitable numerical representat^n for categorical variables.
↳ Apply NLP tech. to free txt to identify relevant information.

→ Categorical variables.
→ Unsorted : Unary representation
→ Sorted : Numerical representation

Data Normalization

→ Objective
→ Make all features equally imp.

→ Approaches.
→ Each unrelated feature is normalized independently
→ For each related var. sets a common normalization could be meaingful
→ Two method ⇒

→ Normalization
- linear transformation
- De-normalization

→ Standardization
- linear transformation
- De-standardization.

Feature Selection

- Objective
  - leave only relevant variables.
- Approaches
- Removal of variables
  - Reductant
  - Unrelated to objective of analysis
  - Low quality:- missing value, outliers, noisy
- Selection of variables
  - Identify the relevant var.
- Usually time consuming.

Feature Construction

- Objective
  - Add new synthetic var. which could faciliate the analysis
- Approaches
  - Build a new var. from existing one
    - linear or non-linear combination
    - Change of reference (e.g. from cartesian to Polar coordinate)

Artificial Neural Network

- Neuron or unit Processing
- Weight (w_{ij})
- Threshold (or bias) θ_i, b_i
- Field of unit i
- Activation function
  - Heaviside (or threshold)
  - Sign
  - sigmoidal
  - Hyperbolic tengent
  - linear
  - Rectified linear unit (ReLU)

Supervised Learning

→ **Prediction**
- → give the best prediction to approximate data
- → Multilinear regression (MLR)

→ **Classification**
- → K-nearest neighbors (K-NN)
- → Logistic regression, naive bayes. (based on Probabilities)

Multilinear Regression (MLR)

- → Unidimensional output
- → minimize quardratic error over training patterns
- → Results

Logistic Regression

- → Statistical model for binary classifict[ion] from set of input data (continous/discr[ete])
- → It is based on a logistic function.
- → To fit model (identify best co-efficient) we use Maxium Likelihood Estimation (MLE)

Naive Bayes Classifer

⇒ Algo. for classification based on Probabilites & Prior information.

K-Nearest Neighbors (KNN)

⇒ Dataset

Predict **X**, based on it's K-nearest neighbors. in dataset

Evaluation of Predictions

→ If applied to whole dataset
↳ They measure quality of fitting not the quality of Prediction

→ Solution
- Division of dataset

Training set
- To fit f

Validation Set
- To evaluate Parameters

Test set
- To evaluate Predictions

→ Cross-Validation (multiple validation)
- 3-fold cross validation

→ Validation is nesessary to avoid overfitting

→ Prediction evaluation measures
↳ Mean squared error
↳ Mean absolute error
↳ Relative absolute error.

→ Evaluation of Prediction: Classification of evaluation measures.

- Sensitivity ✓
- Speceificity ✓
- Fall-out ✓
- Precision ✓ (or Rand index)
- Accuracy ✓ (or thread index)
- Jaccard ✓
- F1 score ✓
- ROC-curve → Area under the curve (AUC) ✓
- Confusion matrix (or contingency table) ✓

Back-Propogation

Application of gradient descent to minimize the quadractic error over the training set of a MLR

Observations

↳ gradient calculation (chain rule of derivatives)

↳ It is needed a differentiable activation function (Sigmoid, hyberbolic tangent, ReLU)

Error Back-Propogation

→ Output-layer

→ Rest of the layer (in backwards order)

→ Amount of weights & thresholds update

→ Includes
- → Learning rate η
- → Momentum α

Derivative of activation function

→ Sigmoid

→ ReLU

Comparison

- Batched back-Propogation
  - ↳ Correct gradient descent
  - ↳ Too slow
- Online back-Propogation
  - ↳ Stochastic gradient descent
  - ↳ Fast but with fluctuations
- Partial batched back-Propogation
  - ↳ Intermediate b/w online & batched
  - ↳ fast & stable
  - ↳ Efficient use of hardware.

Multilayer Neural Network

→ Architecture
- input layer
- Output layer
- One or more hidden layers

→ Feed forward Propogation

→ Prediction Problem

→ Training - find weight & threshold of multilayer N.N that minimize quardratic error over the training patterns

- Minimize
- Problem → E has too many local minima
  → Cannot be solve by derivates =0

Optimization By Gradient Decent

→ Idea.
- → Start in random Position
- → Try to move down in small step

→ formalization (minus the gradient $-\nabla E$)
- → Update weights & threshold to steepest descent.
- → To ensure small step, learning rate η is used.
- → To aviod oscillation, include INERTIA to the movement.
  - Movementum term.

Linear Support Vector Machine (SVM)

- Linearly separated Data
- Best separation
- Maximize marigin
- Support vectors
- Separating hyperplane
- Auxilliary hyperplane
- Margin.
- Optimization Objective
  - Quardrati Programing Problem with linear inequality constraints
  - Optimization with Lagrange multipliers & Karush-Kuhn-Tucker (KKT) conditions
- Primal lagrangian
- Dual Lagrangian ⇒ Subjected to KKT
  - Condition for KKT
  - All patterns correctly classified.
  - Only support vector (SV) contribute to result
  - Separating hyperlane only depends on SV
- Solving the quardratic Programing Problem with linear inequality constraints
  - it is a convex problem & can be solved in polynomial time
  - many algorithms exists
    - Interior point method
    - Active set method
    - conjugate gradient method
    - Augumented Lagrangian method.

Linear Support Vector Machine (SVM)

- What if data is non-linearly separable?
  - ↳ Due to non-linear separation ⇒ Non-linear SVM

⇒ Linear SVM with slack variables →

- → Slack variables $\{\xi^{\mu} \geq 0\}$ account for error in classificat^n
- → Optimization objective
- → Primal lagrangian
- → Dual lagrangian
- → Separating hyperlane
- → Prediction

Non-Linear SVM

→ Kernel trick
→ Make non-linear transf. of data & apply LSVM
→ Dual lagrangian
→ Separating hyperlane
→ Prediction

- → How to find non-linear transformation Φ?
- → Observation [Dot. Product]
- → Kernel trick - Insted of $\Phi(x)$, choose Kernel $K(x, y)$
- → Dual lagrangian with Kernel
- → Prediction
- → Kernel
- → common Kernel
  - → Polynomial (homo + inhomogenous)
  - → Gaussian / Radial Basis function (RBF)
  - → Sigmoidal
  - → Linear
  - → 2^{nd} Poly. 3^{rd} Polynomial

Unsupervised Learning Approaches

- Dimensionality Reduction
- Visualization
- Clustering
- Prototyping
- Encoding
- feature mapping.

Dimensionality Reduction / Visualization

Methods
- Principal Component Analysis (PCA)
- t-distributed Stochastic Network Embedding (t-SNE)
- Hebbian learning
- Self-supervised learning / Autoencoders
- Self-organizing Maps (SOM)

Clustering

Methods
- Centroid-based clustering
  - K-mean, K-median, K-mediod, fuzzy C-means.
- Connectivity-based clustering
  - Agglomerative Hierarchical clustering (AHC)
- Density-based clustering
  - DBSCAN
- Distribution-based clustering
  - Gaussian Mixture model
- Competitive clustering
  - Vector Quantization (VQ), Self-organizing Maps (SOM)

Prototyping

- it define a clustering $\{C_1 \dots C_k\}$ of the original data.
- Some clustering algo. are based on Prototyping
- Methods
  - Centroid - based clustering
    - K-means, K-means, K-medoids, fuzzy C-mean
  - Competitive clustering
    - Vector Quantization (VQ), Self-organ--izing maps (SOM)

Encoding

- find lower-dimensional Prototypes & the encoding & decoding function (with loss function)
- Methods
  - Combination of dimensionality reduction & prototyping algorithms

Feature Mapping

- find prototyping that preserve a certain geometric arrangment
- Methods
  - Self-organizing-Maps (SOM)

Classical Models

- Dimensionality Reduction
  - Principal Component Analaysis (PCA)
  - t-distributed Stochastic Network Embedding (t-SNE)
- Clustering
  - K-means
  - Agglomerative Hierarchical clustering (AHC)

Unsupervised Learning with Neural Network

Hebbian learning
↳ for dimensionality reduction & visualization

Self-Supervised learning / Autoencoder
↳ for dimensionality reduction & visualization

Competitive learning

Unsupervised competitive learning
↳ Vector Quantization (VQ)
- for clustering & Protyping.
↳ Self-organizing maps (SOM)
- for clustering, Protyping & feature mapping

Supervised competitive learning
ART (Adaptive Resonance Theory)
↳ Adaptive competitive learning
- learning Vector Quantization (LVQ)
- ARTMAP