ILP 2017
27th International Conference on Inductive Logic Programming
4-6 Sep 2017 Orléans (France)

Automatic Feature Construction for Supervised Classification from Large Scale Multi-Relational Data

Marc Boullé, Orange Labs

September 5, 2017

Orange today

Orange is one of the topmost European and African operators for mobile and broadband internet services as well as a world leader in providing telecommunication services to businesses.

Over 263 millions customers worldwide

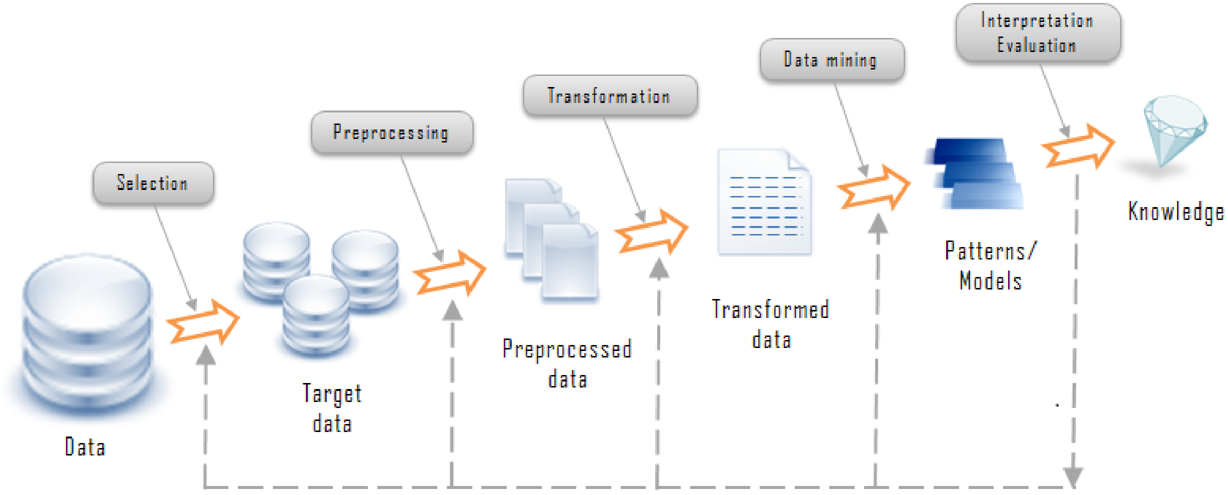
The Group provides services for residential customers in 30 countries and for business customers in 220 countries and territories.

Data Mining in Orange
Example of use case

- Marketing campaigns
  - Objective: scoring
    - churn, appetency, up-selling…
  - Millions of instances
  - Multiple tables source data
    - Customer contracts
    - Call detail records (**billions**)
    - Multi-channel customer support
    - External data
    - …
  - Train sample
    - 100 000 instances
    - 10 000 variables (based on expertise)
    - Heavily unbalanced
    - Missing values
    - Thousands of categorical values
    - …
  - Challenge: industrial scale
    - Hundred of scores every month

Data Mining in Orange

How to efficiently apply data mining techniques in an industrial context?



Schedule

Automatic Feature Construction
for Supervised Classification from
Large Scale Multi-Relational Data

- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
- Conclusion

Data Mining under Limited Resources

- Data Mining in Industrial Context
  - Applicable in a large variety of contexts
  - Vast demand but slow dissemination

- Resource
  - Disk space: fast growth

  - RAM: medium growth
  - CPU: medium growth

  - Skilled data analysts: steady

- The lack of data analysts is the lock to the wide dissemination of the data mining solutions in business

lack of data analysts => Automatize !

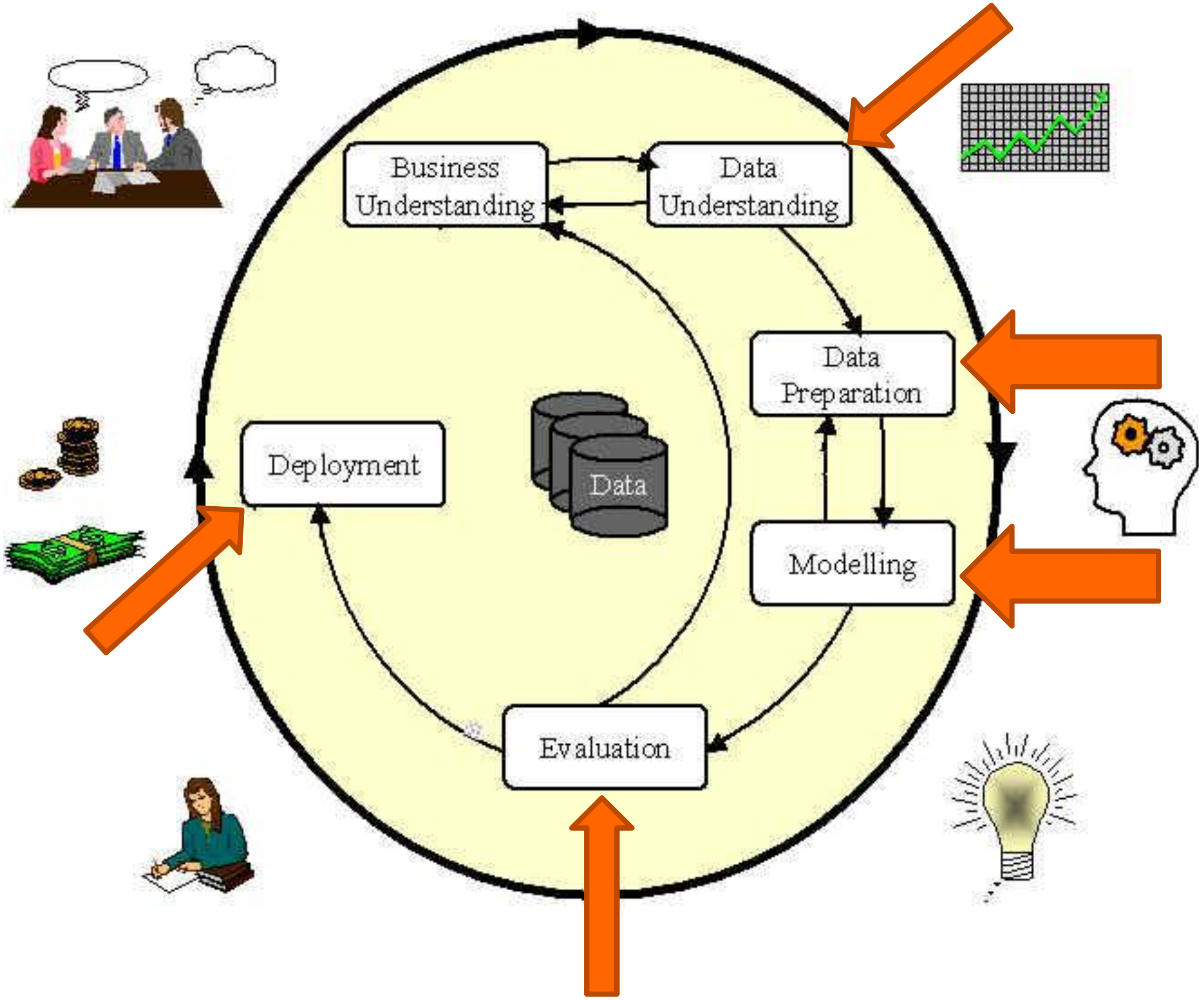
Data Mining in Orange

A wide variety of contexts

- Many domains
  - Marketing
  - Text mining
  - Web mining
  - Traffic classification
  - Sociology
  - Ergonomics
- Many scales
  - Tens to millions of instances
  - Up to billions of secondary records
  - Tens to hundreds of thousands of variables
- Many types of data
  - Numerical
  - Categorical
  - Text
  - Image
  - Relational databases
- Many tasks
  - Data exploration
  - Supervised
  - Unsupervised
- Data constraints
  - Heterogeneous
  - Missing values
  - Categorical data with many values
  - Multiple classes
  - Heavily unbalanced distributions
- Training requirements
  - Fast data preparation and modeling
- Model requirements
  - Reliable
  - Accurate
  - Parsimonious (few variables)
  - Understandable
- Deployment requirement
  - Fast deployment
  - Up to real time classification in network devices
- Business requirement
  - Return of investment for the whole process

very large variety of contexts => Genericity

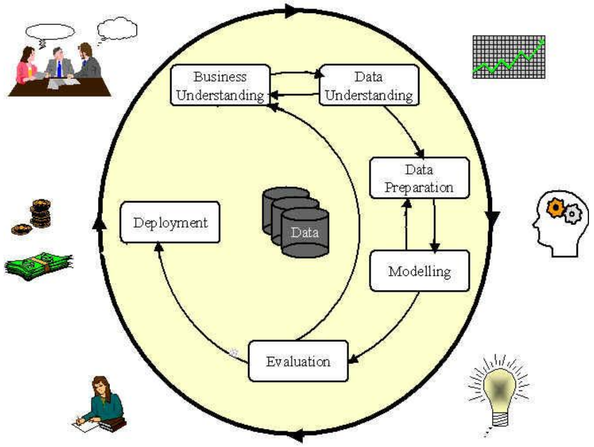
Objective: ease and automatize many tasks in a data-mining project



Objective

- Towards an effective automation of data mining

- Evaluation criterions
  - Genericity
  - No parameter
  - Robustness
  - Accuracy
  - Understandability
  - Efficiency



Lift the brakes to the dissemination
With a high-quality tool

Related work

- Multi-table relational data mining
  - Inductive logic programming
    - uniform representation for examples, background knowledge and hypotheses
    - formal logic rather than database oriented
  - Propositionalisation
    - build a flat instance x variable table from relational data
    - use of a pattern language (aka declarative bias) to limit the expressiveness

- Our approach
  - Closely related to propositionalisation (aka feature construction)
  - Introduction of a probabilistic bias
    - Simple to use by the data analyst → Genericity
    - One single parameter: number of features to construct → No parameter
    - Resilient to over-fitting → Robustness
    - Scalable → Efficiency

- Evaluation criterions
  - Genericity
  - No parameter
  - Robustness
  - Accuracy
  - Understandability
  - Efficiency

Schedule

Automatic Feature Construction for Supervised Classification from Large Scale Multi-Relational Data

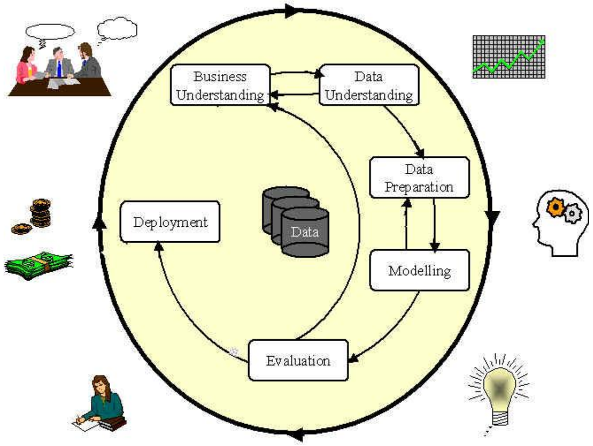
- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
- Conclusion

Context

- Statistical learning
  - Objective: train a model
    - Classification: the output variable is categorical
    - Regression: the output variable is numerical
    - Clustering: no output variable

- Data preparation
  - Variable selection
  - Search for a data representation

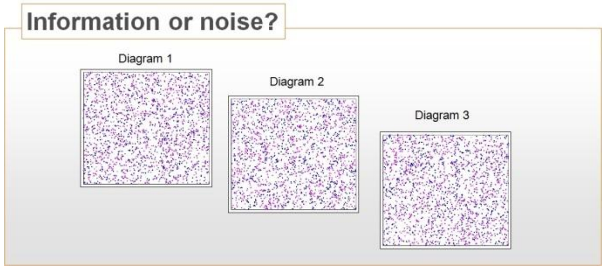
- Data preparation is critical
  - 80% of the process time
  - Requires skilled data analysts



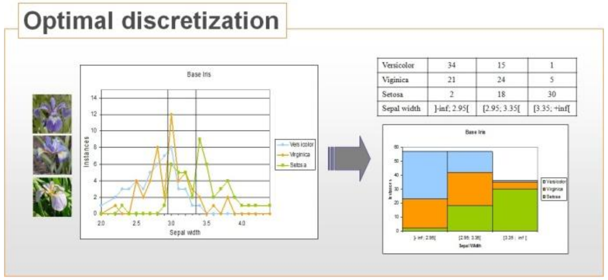
Single-table datasets
instances x variables

Age	Education	Education Num	Marital status	Occupation	Race	Sex	Hours Per week	Native country	…	Class
39	Bachelors	13	Never-married	Adm-clerical	White	Male	40	United-States	…	less
50	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	13	United-States	…	less
38	HS-grad	9	Divorced	Handlers-cleaners	White	Male	40	United-States	…	less
53	11th	7	Married-civ-spouse	Handlers-cleaners	Black	Male	40	United-States	…	less
28	Bachelors	13	Married-civ-spouse	Prof-specialty	Black	Female	40	Cuba	…	less
37	Masters	14	Married-civ-spouse	Exec-managerial	White	Female	40	United-States	…	less
49	9th	5	Married-spouse-absent	Other-service	Black	Female	16	Jamaica	…	less
52	HS-grad	9	Married-civ-spouse	Exec-managerial	White	Male	45	United-States	…	more
31	Masters	14	Never-married	Prof-specialty	White	Female	50	United-States	…	more
42	Bachelors	13	Married-civ-spouse	Exec-managerial	White	Male	40	United-States	…	more
37	Some-college	10	Married-civ-spouse	Exec-managerial	Black	Male	80	United-States	…	more
30	Bachelors	13	Married-civ-spouse	Prof-specialty	Asian	Male	40	India	…	more
23	Bachelors	13	Never-married	Adm-clerical	White	Female	30	United-States	…	less
32	Assoc-acdm	12	Never-married	Sales	Black	Male	50	United-States	…	less
…	…	…	…	…	…	…	…	…	…	…

Proposed approach: data grid models



- Objective
  - Evaluate the informativeness of variables
- Data grid models for non parametric density estimation
  - Discretization of numerical variables
  - Value grouping of categorical variables
  - Data grid are the cross-product of the univariate partitions, with a piecewise constant density estimation in each cell of the grid



- Modeling approach: MODL
  - Bayesian approach for model selection
    - Minimum Description Length
  - Efficient optimization algorithms

Data grid models
for statistical analysis of a data table

- Output variables (*Y*) or input variables (*X*)
- Numerical or categorical variables
- From univariate to multivariate

- Supervised learning: conditional density estimation
- Unsupervised learning: joint density estimation

	Univariate	Bivariate	Multivariate
Classification *Y* categorical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Regression *Y* numerical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Coclustering	–	$P(Y_1, Y_2)$	$P(Y_1, Y_2, \ldots, Y_K)$

Classification
Discretization of numerical variables

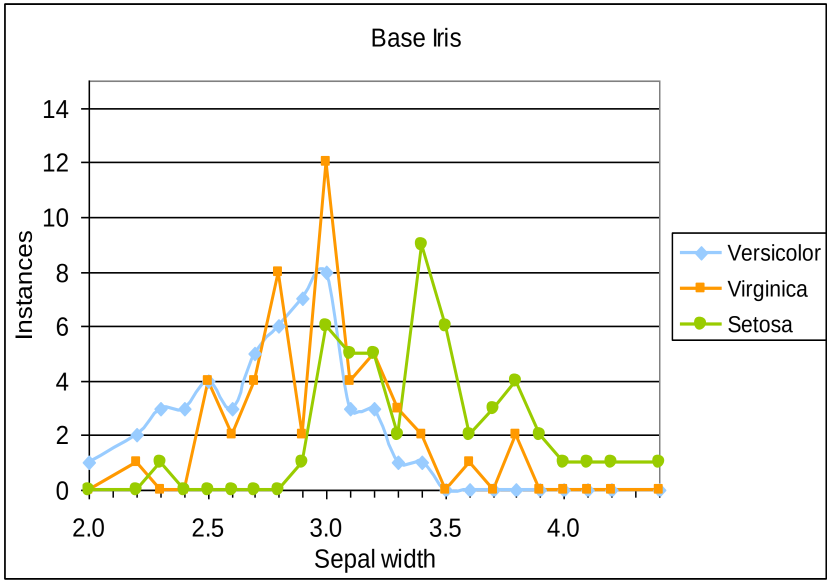
- Univariate analysis
  - Numerical input variable X
  - Categorical output variable Y

- Discretization for univariate conditional density estimation

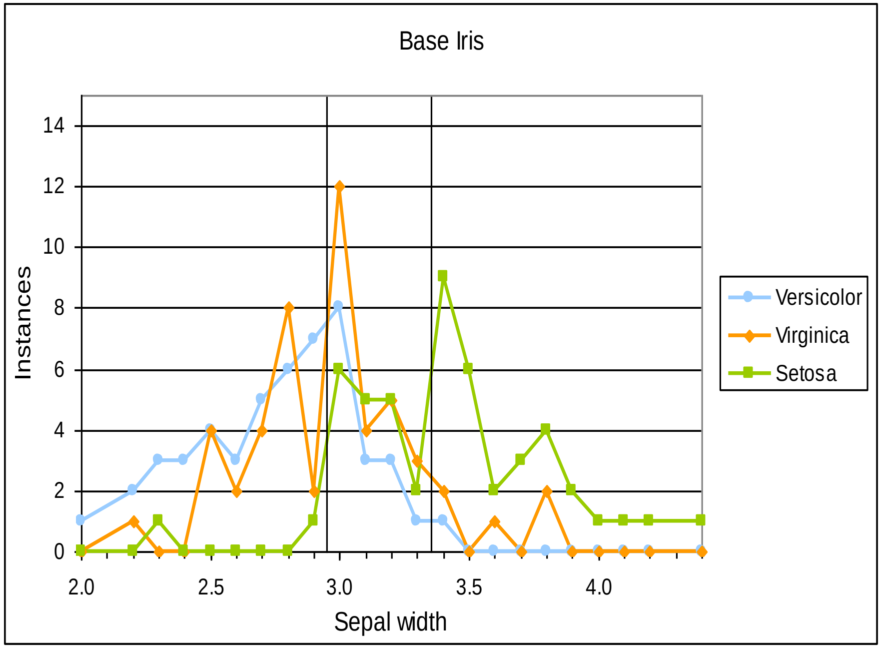
	Univariate	Bivariate	Multivariate
Classification **Y categorical**	**$P(Y \mid X)$**	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \dots, X_K)$
Regression Y numerical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \dots, X_K)$
Clustering	–	$P(Y_1, Y_2)$	$P(Y_1, Y_2, \dots, Y_K)$

Numerical variables
Univariate analysis using supervised discretization

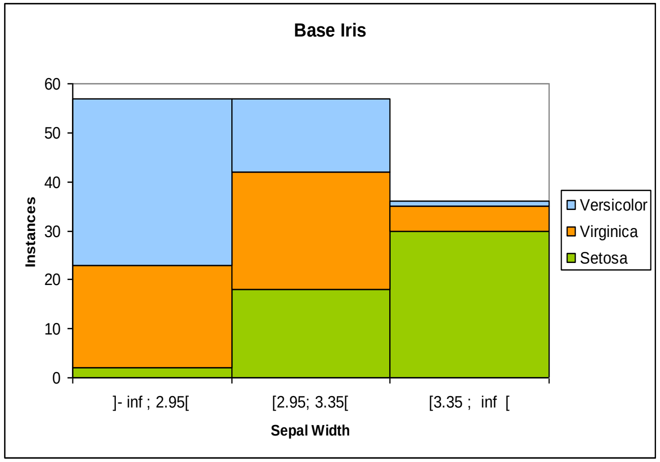
- Discretization:
  - Split of a numerical domain into a set of intervals
- Main issues:
  - Accuracy:
    - Good fit of the data
  - Robustness:
    - Good generalization



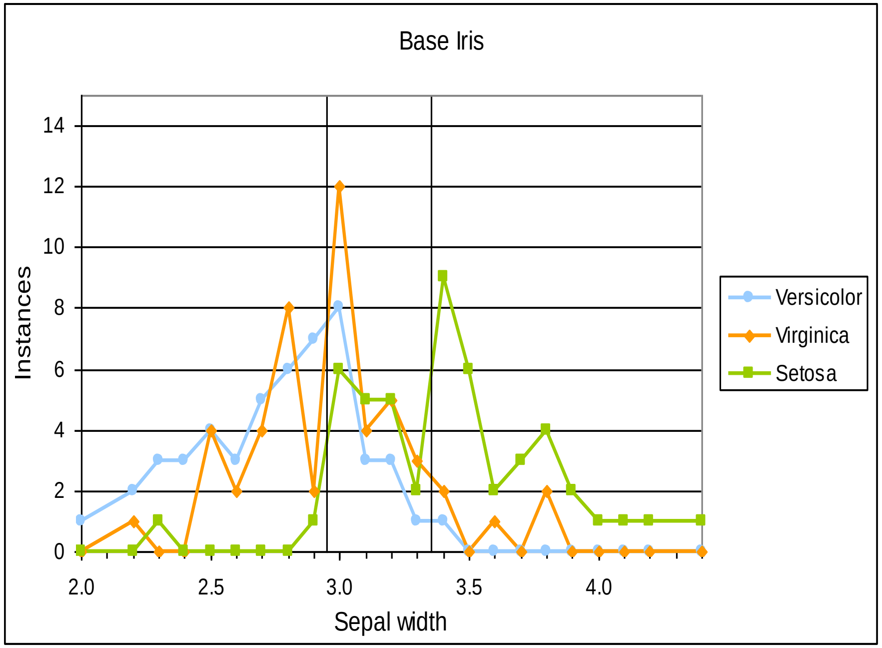
Supervised discretization
Model for conditional density estimation



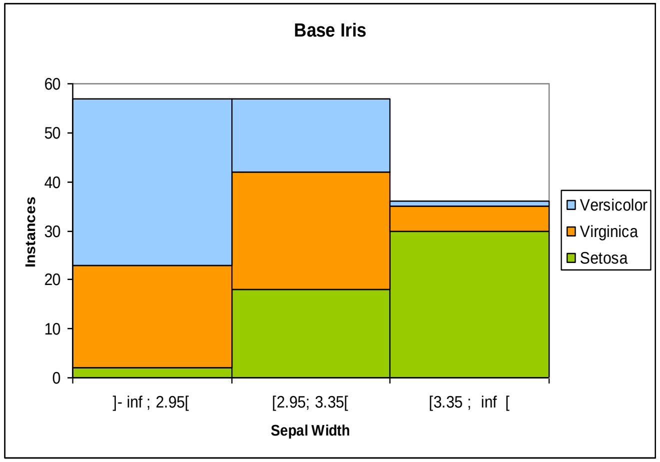
Versicolor	34	15	1
Virginica	21	24	5
Setosa	2	18	30
Sepal width	]- inf ; 2.95[	[2.95; 3.35[	[3.35 ; inf [



Supervised discretization
Model for conditional density estimation



Versicolor	34	15	1
Virginica	21	24	5
Setosa	2	18	30
Sepal width	]- inf ; 2.95[	[2.95; 3.35[	[3.35 ; inf [



How to select the best model?

Formalization

- **Definition:** A discretization model is defined by:
  - the number of input intervals,
  - the partition of the input variable into intervals,
  - the distribution of the output values in each interval.

Formalization

- **Definition:** A discretization model is defined by:
  - the number of input intervals,
  - the partition of the input variable into intervals,
  - the distribution of the output values in each interval.

- **Notations:**
  - N: number of instances
  - J: number of classes
  - I: number of intervals
  - $N_{i.}$: number of instances in the interval i
  - N_{ij}: number of instances in the interval i for class j

Bayesian approach for model selection

- Best model: the most probable model given the data
- Maximize $$P(M \mid D) = \frac{P(M)P(D \mid M)}{P(D)}$$
- Using a decomposition of the model parameters

$$P(M)P(D \mid M) = P(I)P(\{N_i\} \mid I)P(\{N_{ij}\} \mid I, \{N_i\})P(D \mid M)$$

- Assuming independence of the output distributions in each interval

$$P(M)P(D \mid M) = P(I)P(\{N_{i.}\} \mid I)\prod_{i=1}^{I} P(\{N_{ij}\} \mid I, \{N_{i.}\})\prod_{i=1}^{I} P(D_i \mid M)$$

- We now need to evaluate the prior distribution of the model parameters

Prior distribution of the models

- **Definition:** We define the hierarchical prior as follows:
  - the number of intervals is uniformly distributed between 1 et *N*,
  - for a given number of intervals *I*, every set of *I* interval bounds are equiprobable,
  - for a given interval, every distribution of the output values are equiprobable,
  - the distributions of the output values on each input interval are independent from each other.

- Hierarchical prior, uniformly distributed at each stage of the hierarchy

Optimal evaluation criterion MODL

- **Theorem**: A discretization model distributed according the hierarchical prior is Bayes optimal for a given set of instances if the following criterion is minimal:

$$\log(N) + \log\binom{N+I-1}{I-1} + \sum_{i=1}^{I} \log\binom{N_{i.}+J-1}{J-1} + \sum_{i=1}^{I} \log\left(N_{i.}!/N_{i1}!N_{i2}!\ldots N_{iJ}!\right)$$

prior | likelihood

N: number of instances
J: number of classes
I: number of intervals
$N_{i.}$: number of instances in the interval *i*
N_{ij}: number of instances in the interval *i* for class *j*

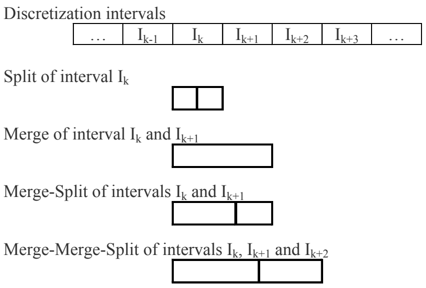
  - 1° term: choice of the number of intervals
  - 2° term: choice of the bounds of the intervals
  - 3° term: choice of the output distribution *Y* in each interval
  - 4° term: likelihood of the data given the model

Discretization algorithm

Quasi-optimal heuristic

- Optimal solution in O(N^3)
  - Based on dynamic programing
  - Usefull to evaluate the quality of optimization heuristics

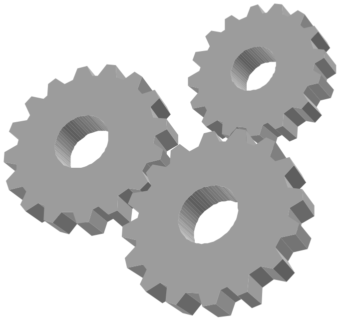
- Approximated solution in O(N log(N))
  - Greedy bottom-up heuristic
  - Post-optimisations to improve the solution
    - split interval, merge interval, move interval boundary



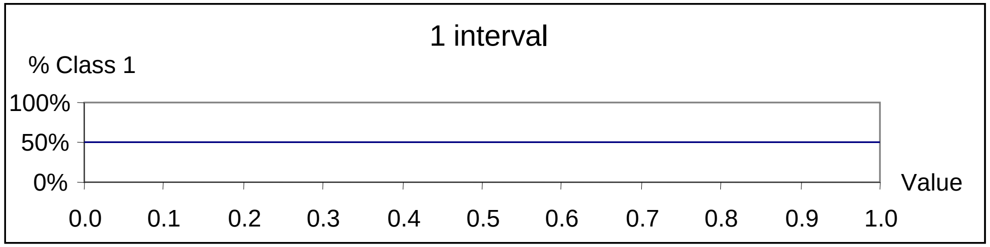
- Evaluation on 2000 discretizations
  - Optimal solution in more than 95% of the cases
  - In the remaining 5%, solution close from the optimal one (diff<0.15%)

Classification
Discretization of numerical variables

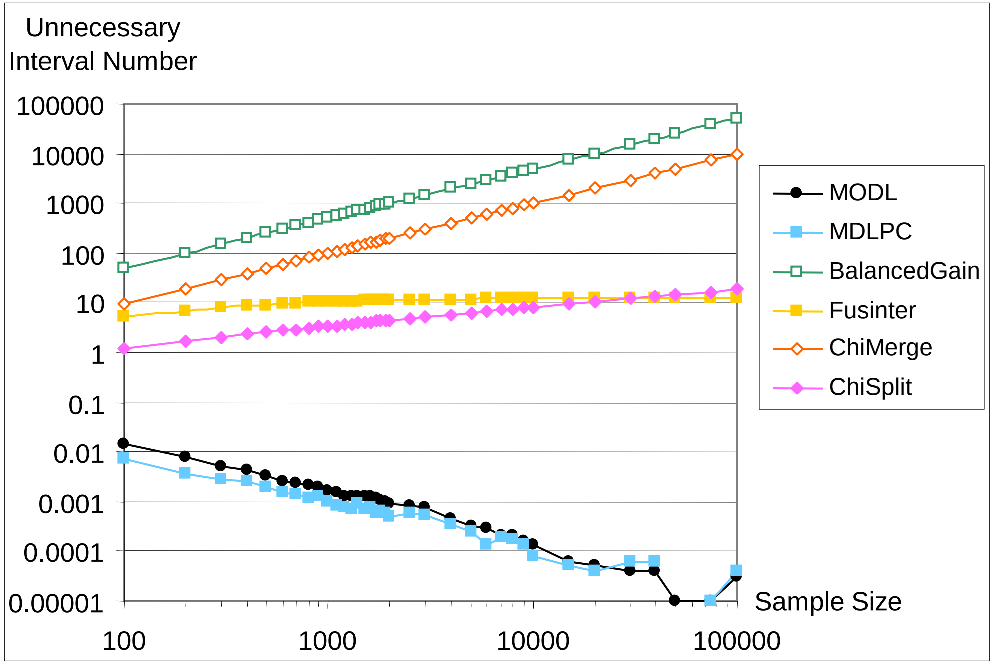
Evaluation



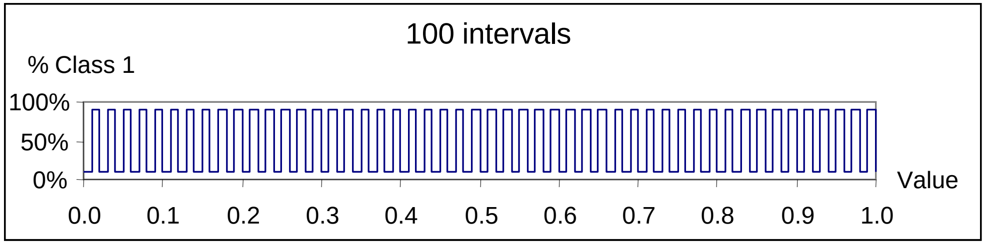
Discretization of a noise pattern



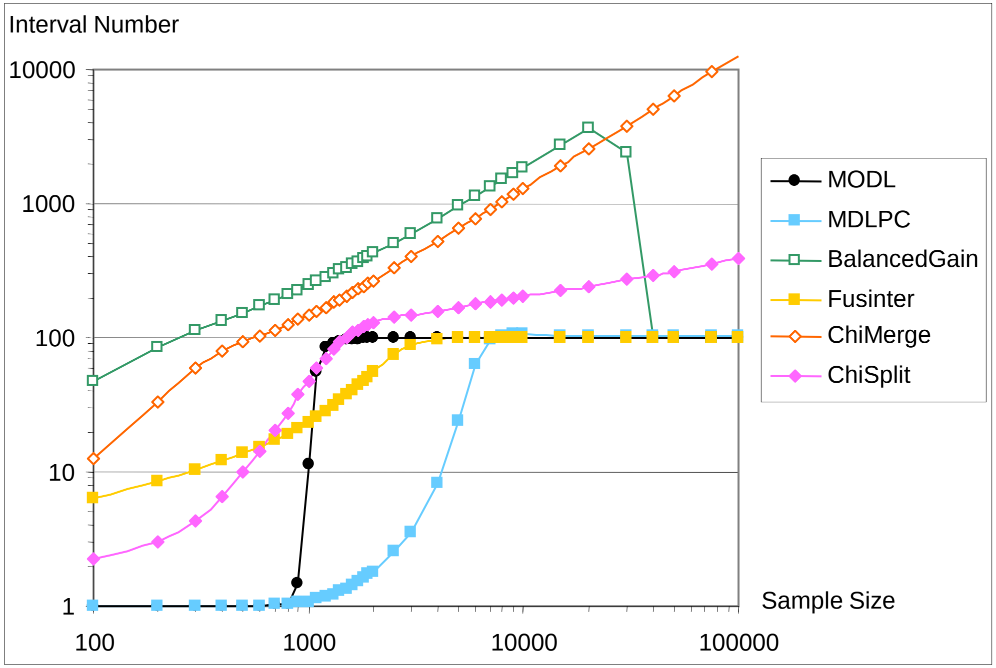
MODL reliably identifies the lack of predictive information



Discretization of a crenel pattern



MODL correctly identifies the relevant information with a minimal number of instances



Classification
Value grouping of categorical variables

- Univariate analysis
  - Categorical input variable X
  - Categorical output variable Y

- Value grouping for univariate conditional density estimation

	Univariate	Bivariate	Multivariate
Classification ***Y* categorical**	**$P(Y \mid X)$**	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Regression *Y* numerical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Coclustering	–	$P(Y_1, Y_2)$	$P(Y_1, Y_2, \ldots, Y_K)$

Categorical variables

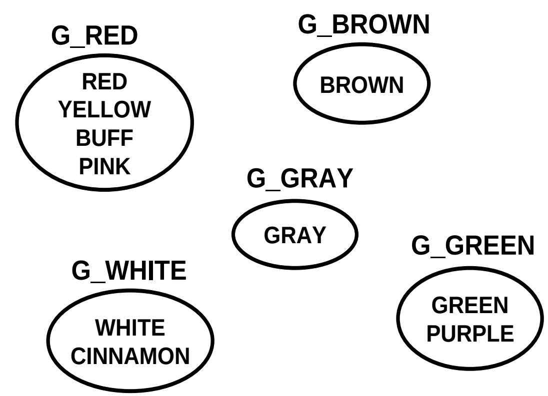
Univariate analysis using value grouping

Cap color	EDIBLE	POISONOUS	Frequency
BROWN	55.2%	44.8%	1610
GRAY	61.2%	38.8%	1458
RED	40.2%	59.8%	1066
YELLOW	38.4%	61.6%	743
WHITE	69.9%	30.1%	711
BUFF	30.3%	69.7%	122
PINK	39.6%	60.4%	101
CINNAMON	71.0%	29.0%	31
GREEN	100.0%	0.0%	13
PURPLE	100.0%	0.0%	10





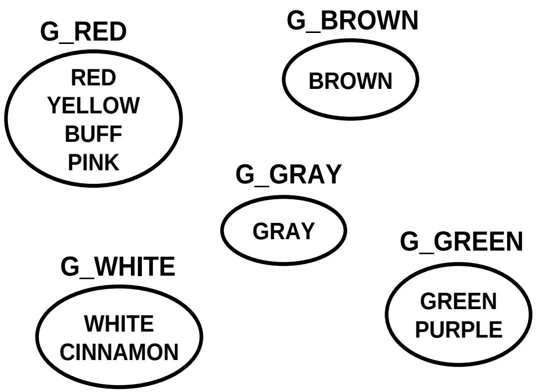
Cap color	EDIBLE	POISONOUS	Frequency
G_RED	38.9%	61.1%	2032
G_BROWN	55.2%	44.8%	1610
G_GRAY	61.2%	38.8%	1458
G_WHITE	69.9%	30.1%	742
G_GREEN	100.0%	0.0%	23



Categorical variables
Univariate analysis using value grouping

Cap color	EDIBLE	POISONOUS	Frequency
BROWN	55.2%	44.8%	1610
GRAY	61.2%	38.8%	1458
RED	40.2%	59.8%	1066
YELLOW	38.4%	61.6%	743
WHITE	69.9%	30.1%	711
BUFF	30.3%	69.7%	122
PINK	39.6%	60.4%	101
CINNAMON	71.0%	29.0%	31
GREEN	100.0%	0.0%	13
PURPLE	100.0%	0.0%	10

Cap color	EDIBLE	POISONOUS	Frequency
G_RED	38.9%	61.1%	2032
G_BROWN	55.2%	44.8%	1610
G_GRAY	61.2%	38.8%	1458
G_WHITE	69.9%	30.1%	742
G_GREEN	100.0%	0.0%	23



How to select the best model?

Value grouping

Same approach as for discretization

- A value grouping model is defined by:
  - the number of groups of inputs values,
  - the partition of the input variable into groups,
  - the distribution of the output values in each group.
- Model selection
  - Bayesian approach for model selection
  - Hierarchical prior for the model parameters
  - Exact analytical criterion to evaluate the models
- Optimization algorithms in O(*N* log(*N*))

Notations:

N: number of instances
V: number of values
J: number of classes
I: number of groups
$N_{i.}$: number of instances in the group *i*
N_{ij}: number of instances in the group *i* for class *j*

$$\underbrace{\log(V) + \log\left(B(V,I)\right) + \sum_{i=1}^{I} \log \binom{N_{i.}+J-1}{J-1}}_{\text{prior}} + \underbrace{\sum_{i=1}^{I} \log\left(N_{i.}! / N_{i1}! N_{i2}! \ldots N_{iJ}!\right)}_{\text{likelihood}}$$

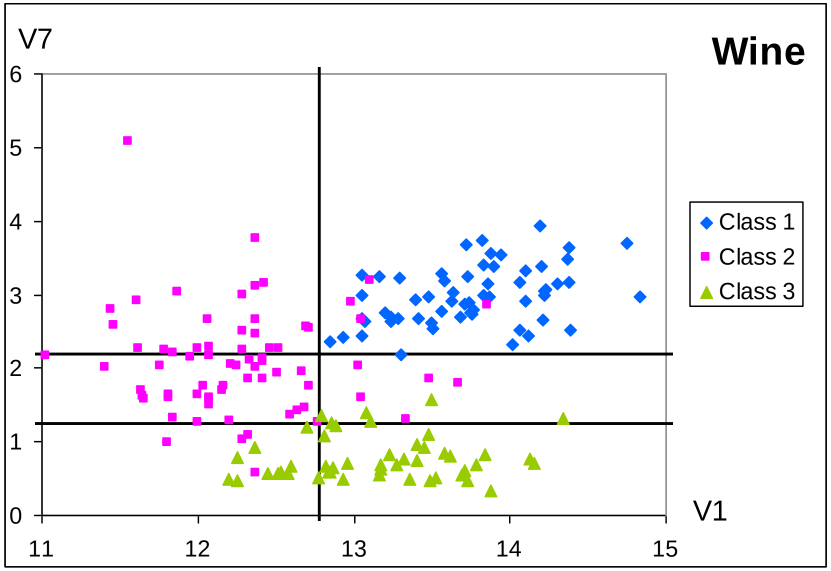
Classification
Bivariate discretization of numerical variables

- Bivariate analysis
  - Numerical input variables X_1 and X_2
  - Categorical output variable Y

- Bivariate discretization for bivariate conditional density estimation

	Univariate	**Bivariate**	Multivariate
Classification ***Y* categorical**	$P(Y \mid X)$	$\mathbf{P(Y \mid X_1, X_2)}$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Regression *Y* numerical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Clustering	–	$P(Y_1, Y_2)$	$P(Y_1, Y_2, \ldots, Y_K)$

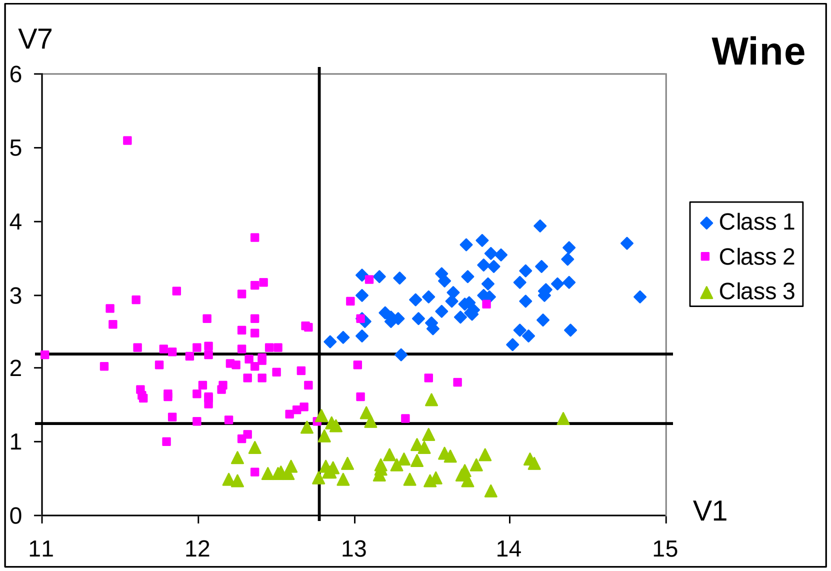
Pair of numerical variables



Pair of numerical variables

Bivariate discretization as a conditional density estimator

- Each input variable is discretized
- We obtain a bivariate data grid



- In each cell, the conditional density is estimated by counting

]2.18;+inf[	(0, 23, 0)	(59, 0, 4)
]1.235;2.18]	(0, 35, 0)	(0, 5, 6)
]-inf;1.235]	(0, 4, 11)	(0, 0, 31)
V7xV1	]-inf;12.78]	]12.78;+inf[

Application of the MODL approach

- Explicit formalization of the model family
  - Definition of the model parameters $I_1, I_2, N_{i_1..}, N_{.i_2.}, N_{i_1 i_2 j}$
- Definition of a prior distribution on the parameters of the bivariate discretization models
  - Hierarchical prior
  - Uniform distribution at each stage of the hierarchy
- We obtain an exact analytical evaluation criterion

prior

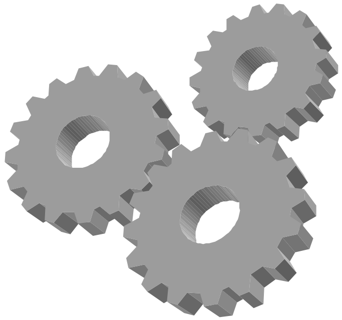
$$\log(N) + \log\binom{N + I_1 - 1}{I_1 - 1} + \log(N) + \log\binom{N + I_2 - 1}{I_2 - 1} + \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\log\binom{N_{i_1 i_2 .} + J - 1}{J - 1} +$$

likelihood

$$\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}\log\left(N_{i_1 i_2 .}! / N_{i_1 i_2 1}! N_{i_1 i_2 2}! \ldots N_{i_1 i_2 J}!\right)$$

Classification
Bivariate discretization of numerical variables

Evaluation



Question: noise or information?

Diagram 1

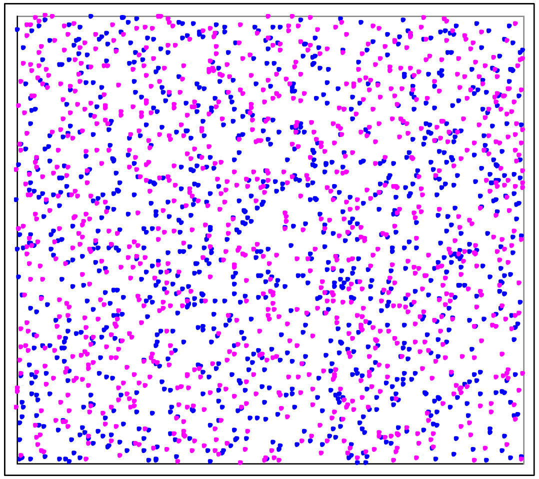


Diagram 2

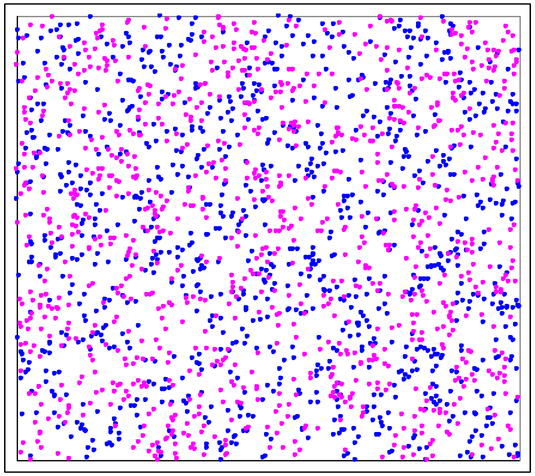


Diagram 3

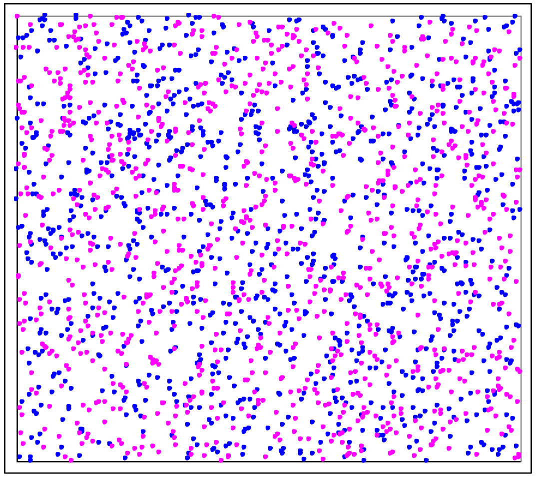
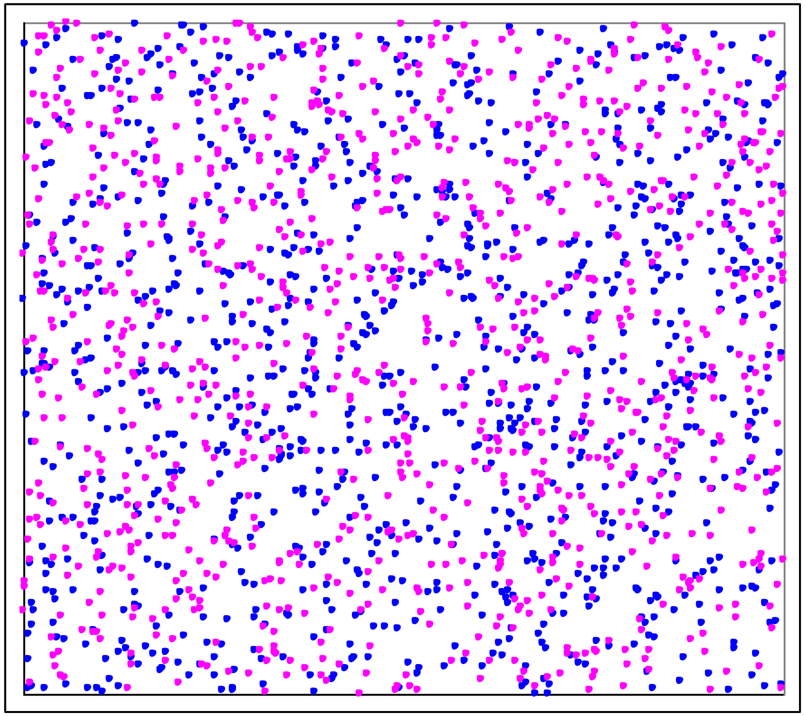
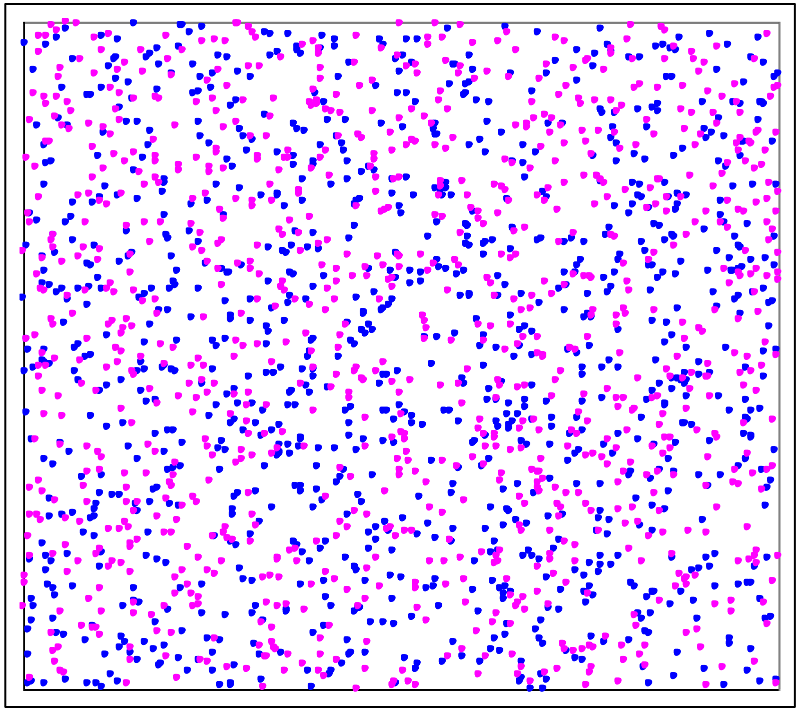


Diagram 1
noise

Diagram 1 (2000 points)



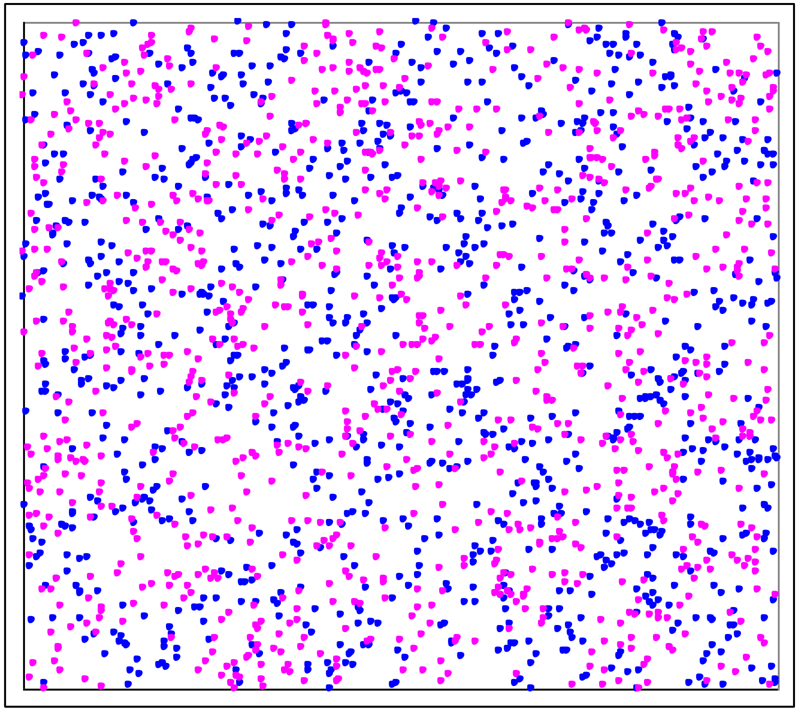
Data grid 1 x 1 (1 cell)



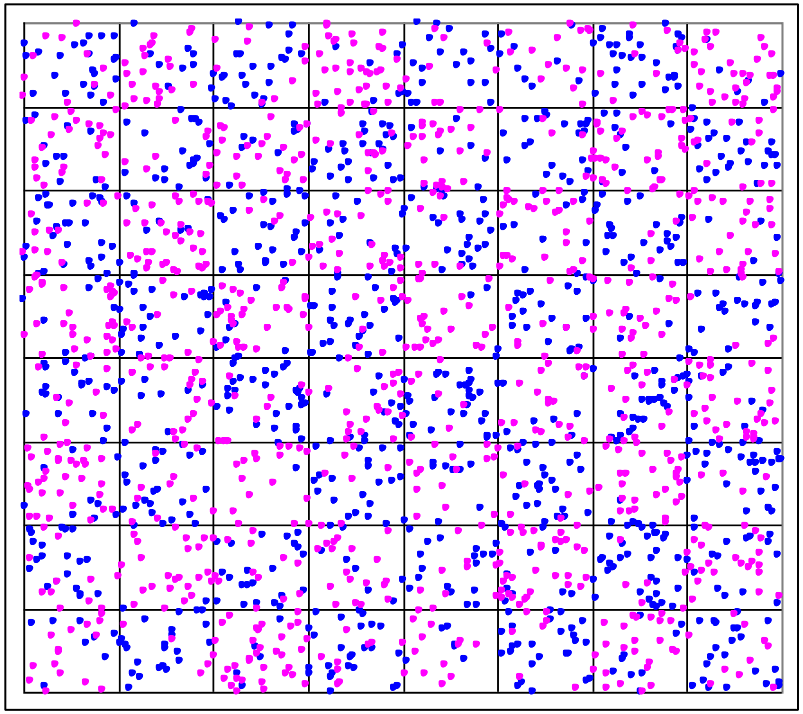
Value of criterion = 2075

Diagram 2
chessboard 8 x 8 with 25% noise

Diagram 2 (2000 points)



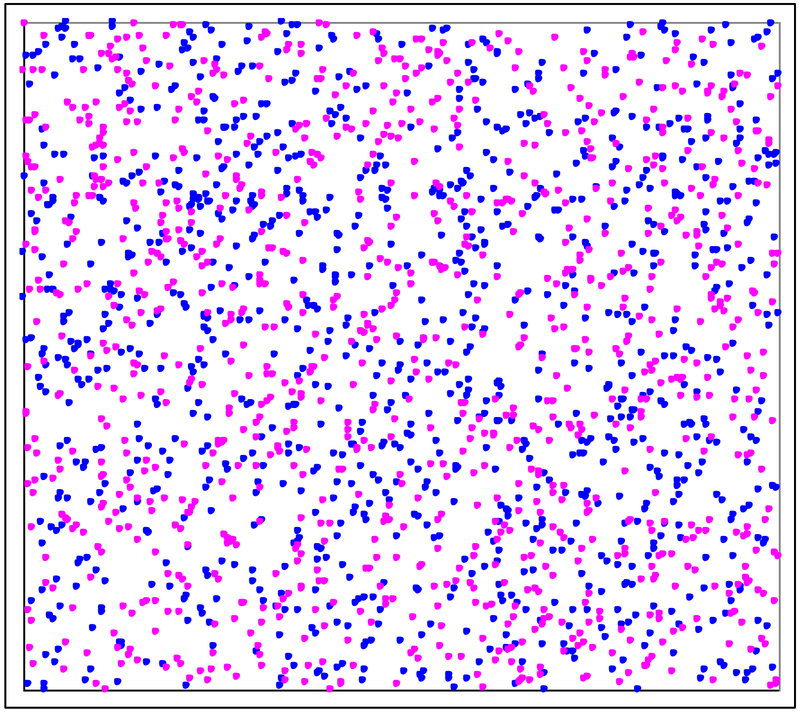
Data grid 8 x 8 (64 cells)



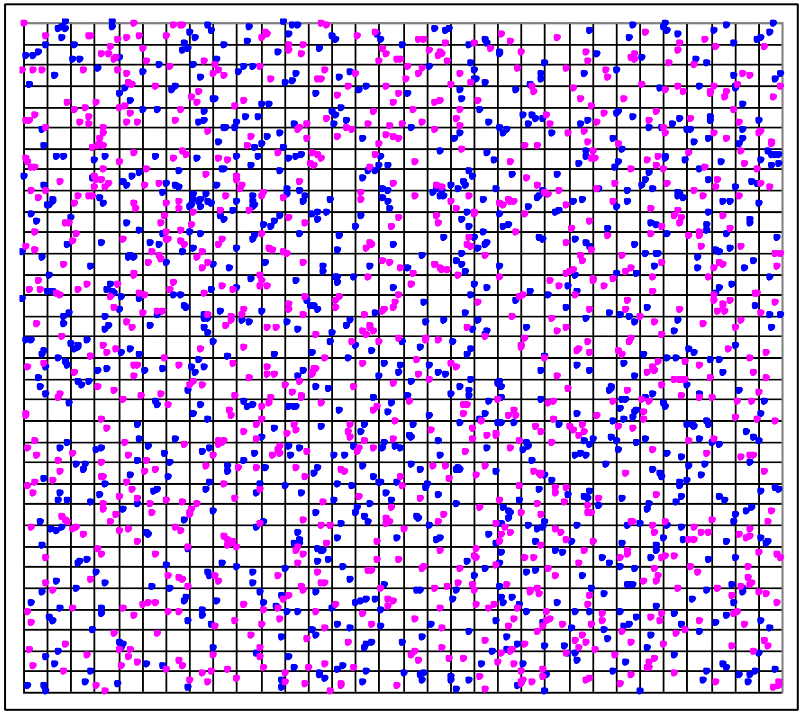
Value of criterion = 1900

Diagram 3
chessboard 32 x 32, without noise

Diagram 3 (2000 points)



Data grid 32 x 32 (1024 cells)



Value of criterion = 1928

Genericity of the data grid models

	Univariate	Bivariate	Multivariate
Classification Y categorical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Regression Y numerical	$P(Y \mid X)$	$P(Y \mid X_1, X_2)$	$P(Y \mid X_1, X_2, \ldots, X_K)$
Clustering	–	$P(Y_1, Y_2)$	$P(Y_1, Y_2, \ldots, Y_K)$

MODL approach

- Density estimation using data grids
  - Discretization of numerical variables
  - Value grouping of categorical variables
  - Density estimation based on data grid models, with piecewise constant density per cell
  - Strong expressiveness
- Model selection
  - Bayesian approach for model selection
  - Hierarchical prior for the model parameters
  - Exact analytical criterion
- Optimization algorithm
  - Combinatorial algorithms
  - Heuristic exploiting the sparseness of the data grids and the additivity of the criterion
  - Efficient implementation

MODL approach
Towards an automatisation of data preparation

- Data preparation
  - Recoding
  - Evaluation of conditional or joint density
  - Variable selection
    - Variables can be sorted by decreasing informativeness

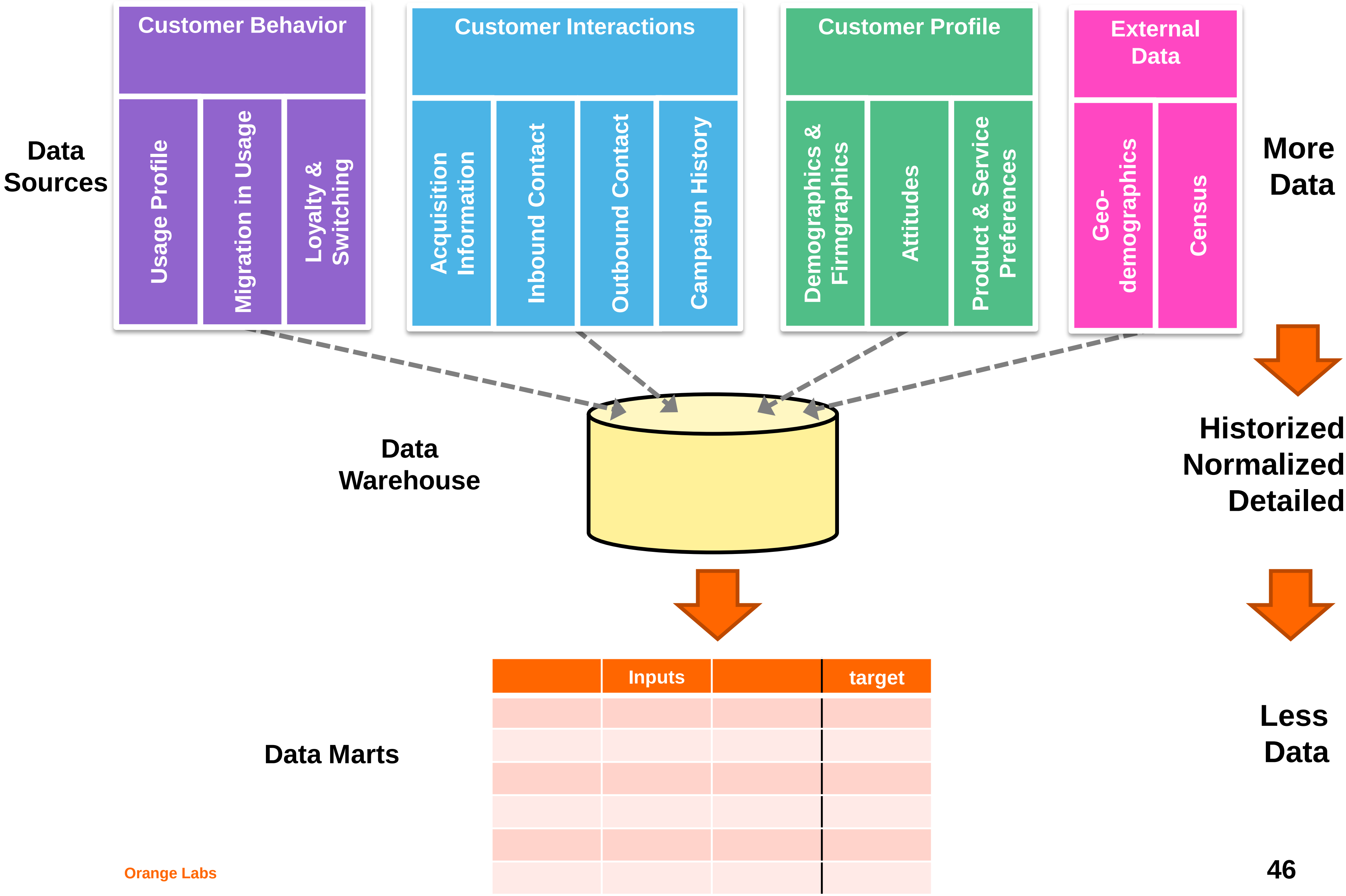
- Advantages of the MODL approach
  - Genericity
  - Parameter-free
  - Reliability
  - Accuracy
  - Interpretability
  - Efficiency

Schedule

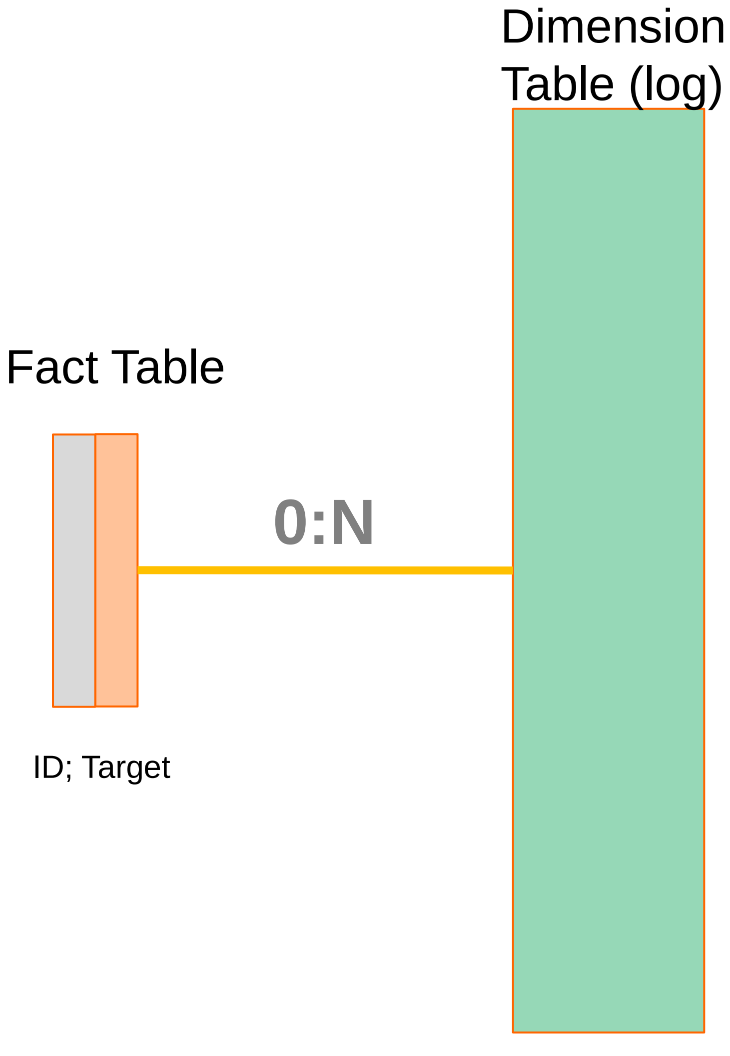
Automatic Feature Construction
for Supervised Classification from
Large Scale Multi-Relational Data

- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
- Conclusion

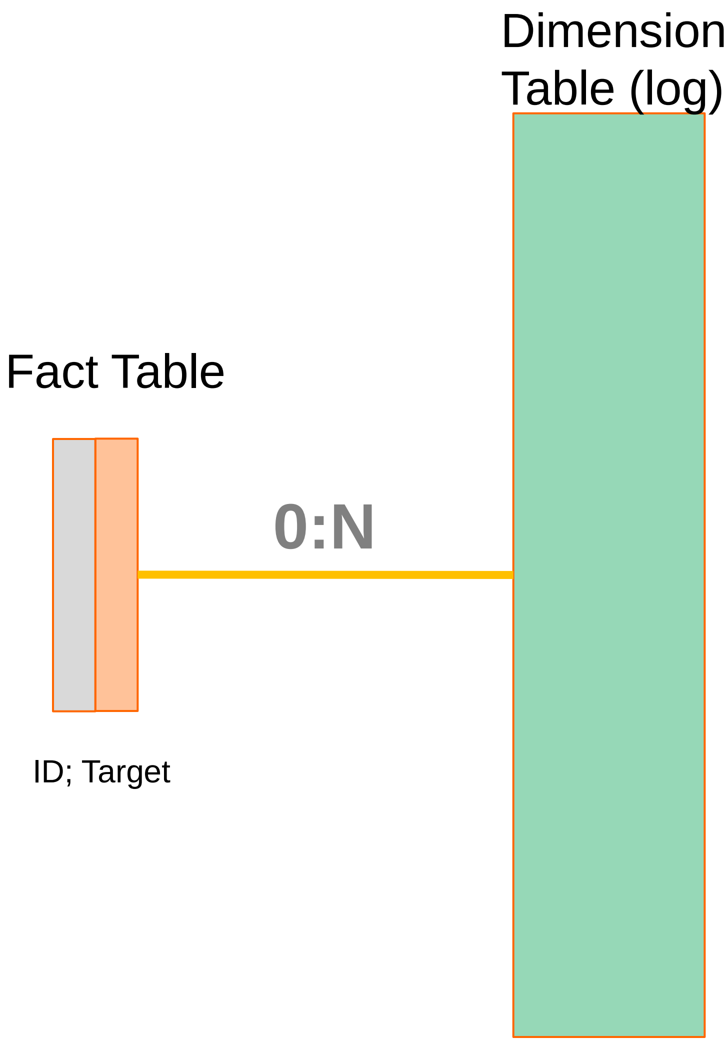
Where does data come from?



Big Data = relational data!



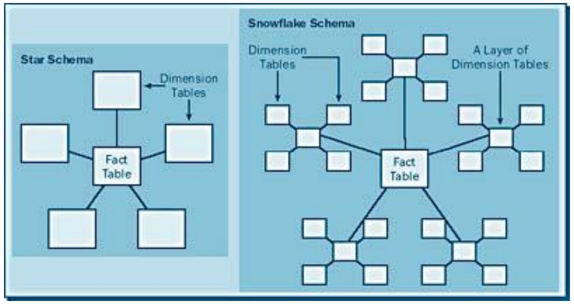
Big Data = relational data!



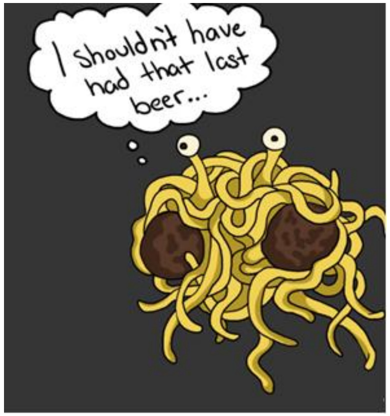
Generalization

Star Schema

Snowflake Schema

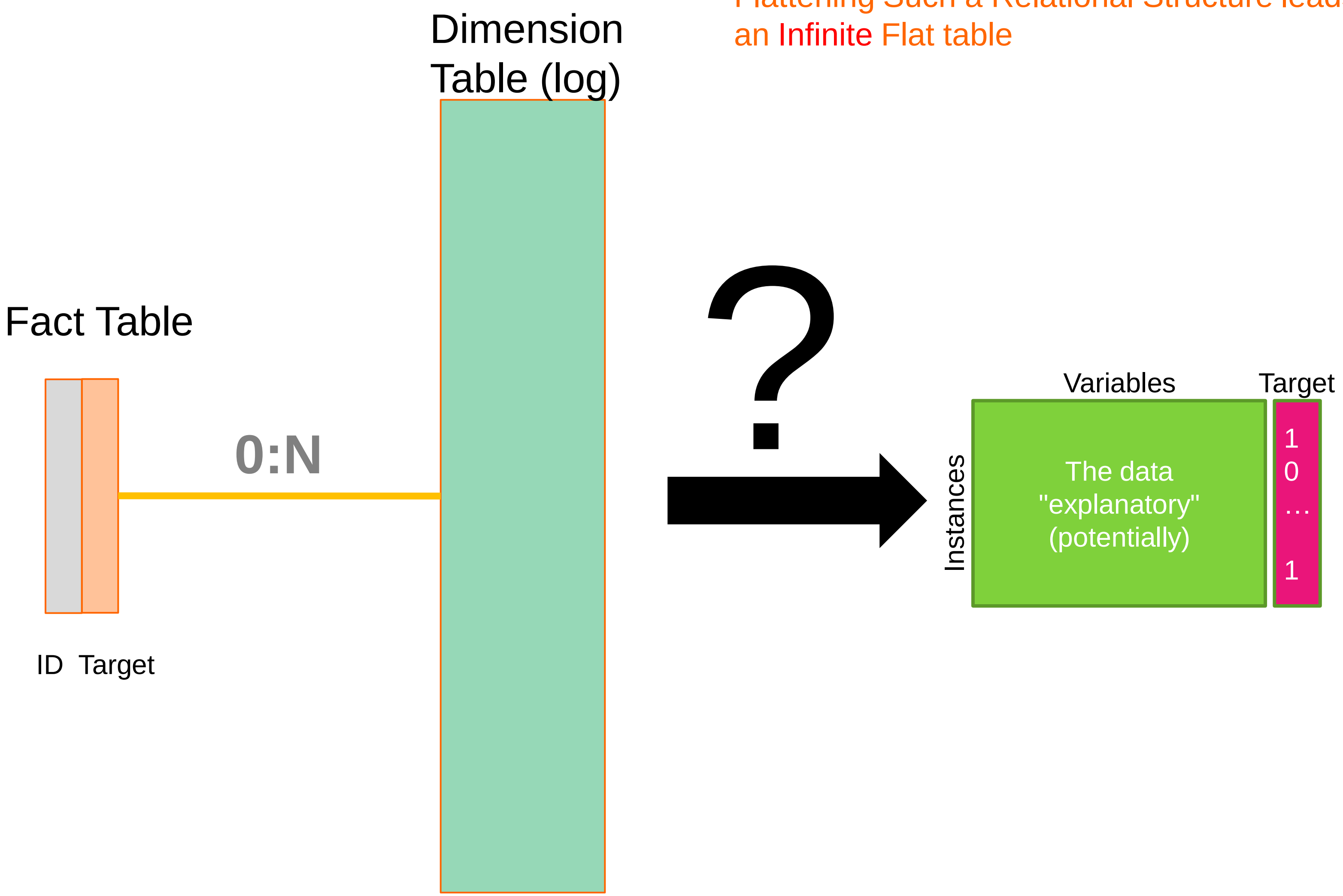


More complex structures are not considered (Yet):

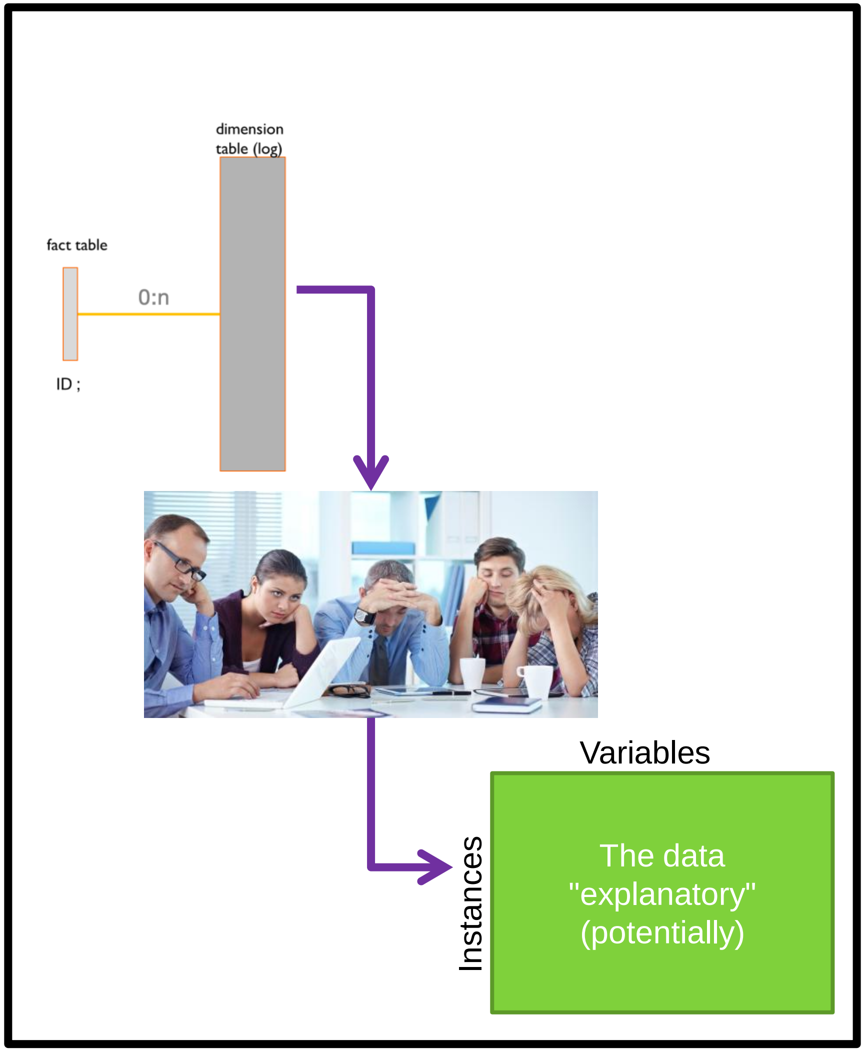


Big Data = relational data!

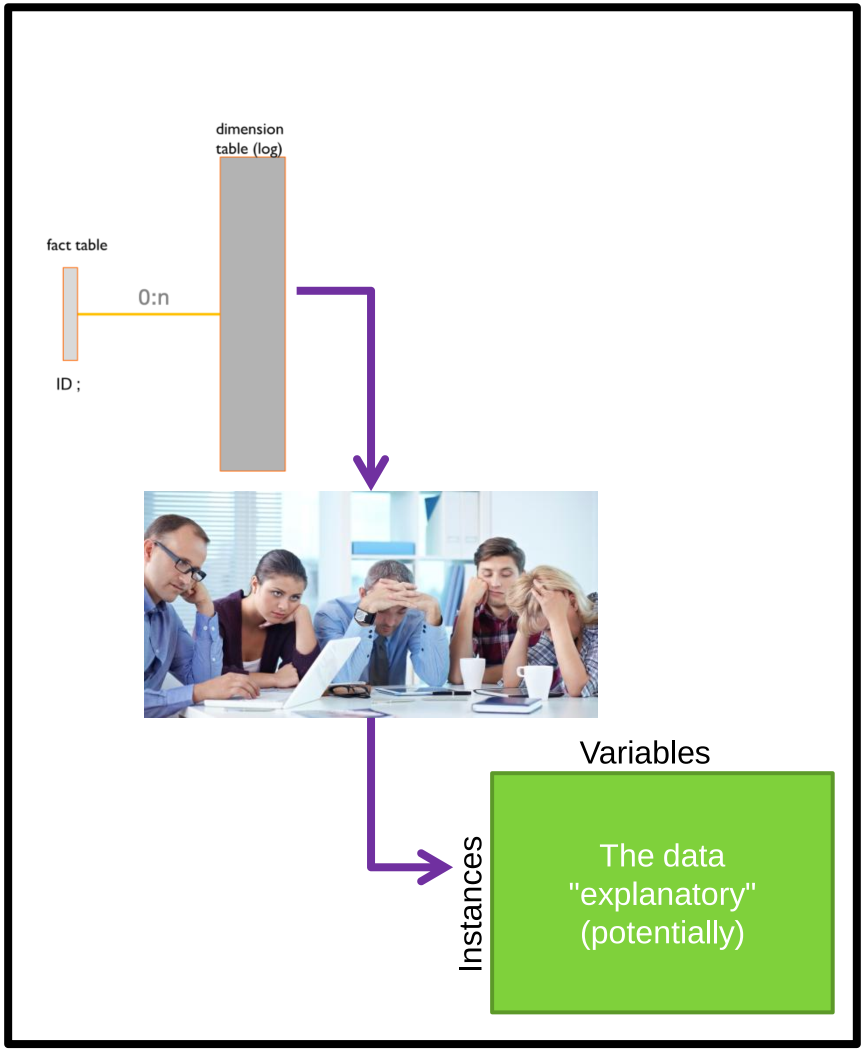
Flattening Such a Relational Structure leads to an Infinite Flat table



Creation of aggregates



Creation of aggregates



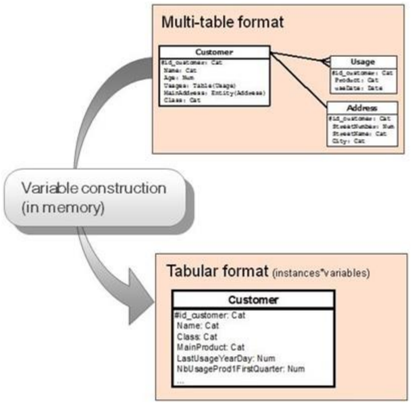
- Long
  - Time expensive process to get a flat table usable for data analysis
- Costly
  - Expert knowledge necessary to constructed new variables
- Risky
  - Risk of missing informative variables
  - Risk of constructing and selecting irrelevant variables

- Data-mart specified once for all from business knowledge from a History …
- … and it is hoped valid for a whole range of Future problems

- (a little caricature, the specification of the data mart evolves in the course of the time but always a posteriori)

Automatic variable construction

- Search for an efficient data representation
  - Context: supervised analysis
    - especially, in the multi-tables settings
  - Data preparation:
    - automatic variable selection
    - next step: automatic variable construction (propositionalisation)



- Objective:
  - Explore numerous data representations using variable construction
  - Select the best representation

- Challenges
  - The number of constructed variables is infinite
    - it is a subset of all computer programs
  - How to specify domain knowledge in order to control the space of constructed variables?
  - How to efficiently exploit this domain knowledge in order to reach the objective?
    - Explore a very large search space
    - Prevent the risk of over-fitting

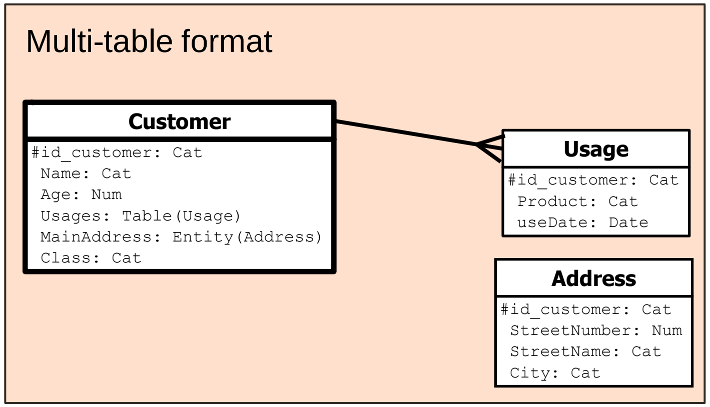
Schedule

Automatic Feature Construction
for Supervised Classification from
Large Scale Multi-Relational Data

- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
  - Specification of domain knowledge
  - Evaluation of constructed variables
  - Sampling a subset of constructed variables
  - Experiments
- Conclusion

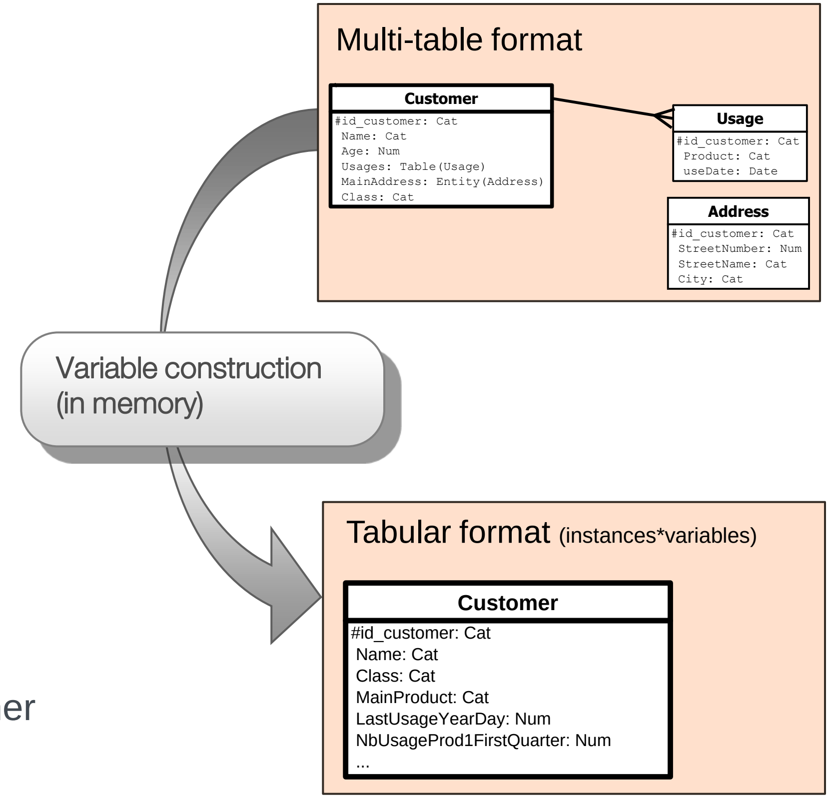
Specification of data format

- Table
  - Two kinds of tables
    - Root table: statistical unit of the studied problem
    - Secondary table: sub-part of the statistical unit
  - Variables of simple type
    - Numerical (Num)
    - Categorical (Cat)
  - Variables of advanced type
    - Date, Time, Timestamp…
  - Variables of relation type
    - Simple composition: sub-entity with 0-1 relation (Entity)
    - Multiple composition: sub-entity with 0-n relation (Table)



Specification of a variable construction language

- Construction rule
  - Program function
    - Input: one or several values
    - Output: one value
  - Type of values
    - Simple: Numerical, Categorical
    - Advanced: Date, Time, Timestamp…
    - Relation: Entity or Table
- Constructed variable
  - Output of a construction rule
  - Rule operands
    - Value
    - Variable
    - Output of another rule
- Examples:
  - New variables constructed in table Customer
    - *MainProduct = Mode(Usages, Product)*
    - *LastUsageYearDay = Max(Usages, YearDay(useDate))*
    - *NbUsageProd1FirstQuarter = Count(Selection(Usages, YearDay(useDate) in [1 ;90] and Product = "Prod1"))*
    - *…*



Variable construction language
List of construction rules

Name	Return type	Operands	Label
Count	Num	Table	Number of records in a table
CountDistinct	Num	Table, Cat	Number of distinct values
Mode	Cat	Table, Cat	Most frequent value
Mean	Num	Table, Num	Mean value
StdDev	Num	Table, Num	Standard deviation
Median	Num	Table, Num	Median value
Min	Num	Table, Num	Min value
Max	Num	Table, Num	Max value
Sum	Num	Table, Num	Sum of values
Selection	Table	Table, (Cat, Num…)	Selection from a table given a selection criterion
YearDay	Num	Date	Day in year
WeekDay	Num	Date	Day in week
DecimalTime	Num	Time	Decimal hour in day
…	…	…	…

Schedule

Towards automatic variable construction, data preparation and modeling for large scale multi-tables datasets

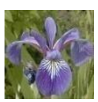
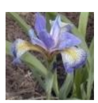
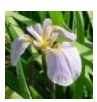
- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
  - Specification of domain knowledge
  - Evaluation of constructed variables
  - Sampling a subset of constructed variables
  - Experiments
- Conclusion

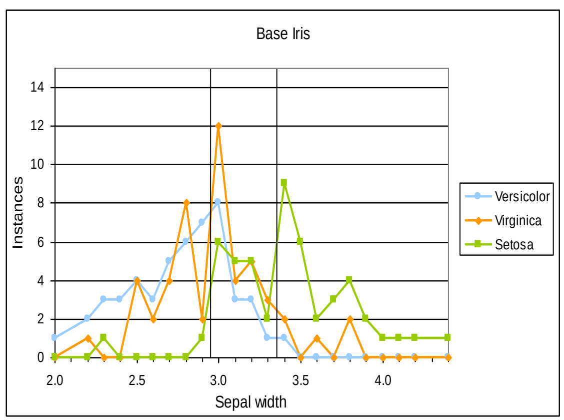
Preliminary: MODL supervised preprocessing

Minimum Optimized Description Length

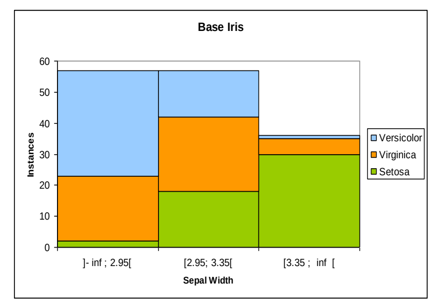
- Evaluation of the informativeness of a variable

- Preprocessing models M_P of conditional density estimation p(Y|X)
  - Partition of numerical variables into intervals and categorical variables into groups of values
  - Conditional density estimation per interval/group
    - Multinomial distribution of class values in each interval/group
    - Piecewise constant estimation



Versicolor	34	15	1
Virginica	21	24	5
Setosa	2	18	30
Sepal width	]- inf ; 2.95[	[2.95; 3.35[	[3.35 ; inf [



Which model is the best one?

MODL approach: evaluation of one variable

Posterior probability of a preprocessing model

- Prior distribution of parameters of model M_P
  - Bayesian approach MAP (maximum a posteriori)
    - Hierarchical prior
    - Uniform at each stage of the parameter hierarchy

$$p\left(M_P(X)\right) * p\left(D_Y \mid M_P(X), D_X\right)$$

  - Crude MDL approach
    - Negative log of the prior probability and of the likelihood
    - Basic coding based of counting the number of possible parameterizations

- Evaluation criterion
  - Exact analytical formula

$$c(X) = L\left(M_P(X)\right) + L\left(D_Y \mid M_P(X), D_X\right)$$

  - Regularized conditional entropy estimator

$$c(X) \approx N\, Ent(Y \mid X)$$

- Null model and variable filtering
  - Null model: coding the target variable directly

$$c(\varnothing) \approx N\, Ent(Y)$$

  - Variables with cost beyond the null cost are filtered to prevent over-fitting
  - Evaluation of a variable: compression rate

$$Level(X) = 1 - c(X)/c(\varnothing)$$

Penalization of complex preprocessing models

MODL approach: construction of one variable

- Definition of modeling space M_C of constructed variables
  - Exploit the domain knowledge
  - Exploit the multi-table format of the input data
  - A constructed variable X is a formula
    - it is a « small » computer program

- Definition of a prior distribution on all constructed variables

$$L(M_C(X)) = -\log p(M_C(X))$$

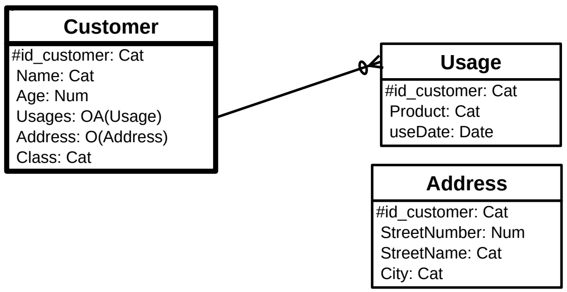
- Evaluation criterion of a constructed variable

$$c(X) = \boxed{L(M_C(X))} + L(M_P(X)) + L(D_Y \mid M_P(X), D_X)$$

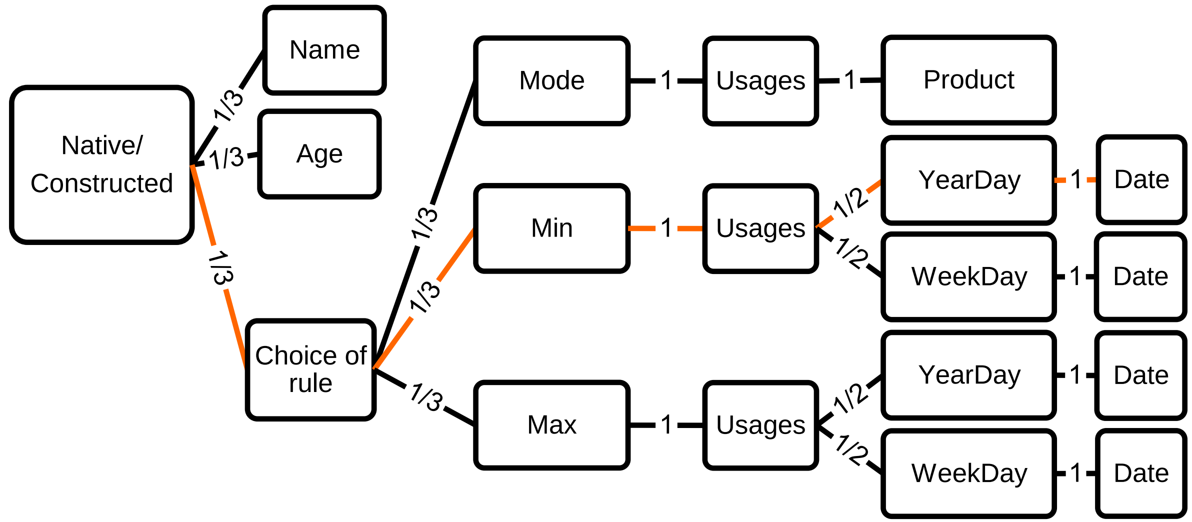
Penalization of complex constructed variables

Prior distribution on all constructed variables

Example



- Rules
  - YearDay
  - Weekday
  - Mode
  - Min
  - Max



Hierarchy of Multinomial Distributions with potentially Infinite Depth (HMDID) prior

- Cost of Name $L(M_C(X)) = \log(3)$
  - Choice of variable : log(3)

- Cost of Min(Usages, YearDay(Date)) $L(M_C(X)) = \log(3)+\log(3)+\log(1)+\log(1)+\log(2)+\log(1)$
  - Choice of constructing a variable: log(3)
  - Choice of rule Min: log(3)
  - Choice of first operand (Usages) of Min: log(1)
  - Choice of constructing a variable for second operand of Min: log(1)
  - Choice of rule YearDay: log(2)
  - Choice of operand of YearDay (Date): log(1)

Schedule

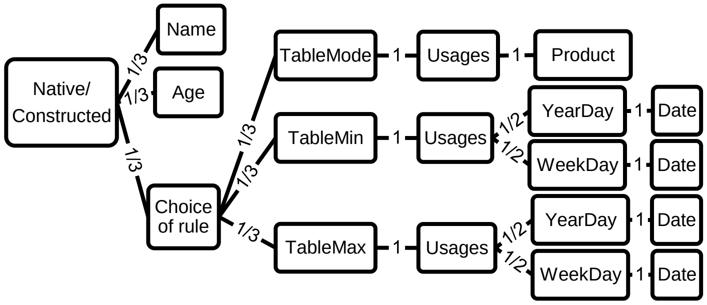
- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
  - Specification of domain knowledge
  - Evaluation of constructed variables
  - Sampling a subset of constructed variables
  - Experiments
- Conclusion

Exploitation of domain knowledge
How to draw a sample from the space of variable construction?

- Objective: draw a sample of *K* variables
  - At this step, the problem of selecting the informative variables is ignored

- Principle
  - Draw the variables one by one according to the HMDID prior

- Naive algorithm: successive random draws
  - Input: *K* {Number of draws}
  - Sortie: $\mathbb{X}$={*X*} ,|$\mathbb{X}$|≤*K* {Sample of constructed variables}
    - 1: $\mathbb{X}$=Ø
    - 2: **for** k = 1 to *K* **do**
    - 3:   Draw *X* according to HMDID prior
    - 4:   Add *X* into $\mathbb{X}$
    - 5: **end for**



Exploitation of domain knowledge

The naive algorithm is neither efficient not computable

- The naive algorithm is not efficient
  - Most draws do not produce new variables
  - Few constructed variables are drawn in case of numerous native variables
- The naive algorithm is not computable
  - Example:
    - Variable v de type Num, rule f(Num, Num) -> Num
    - Example: f = Sum(., .)
    - Family of constructed variables

Size	Example	Coding	Coding length	Prior	Number of variables
1	x	0	1	2^{-1}	1
2	f(x,x)	100	3	2^{-3}	1
3	f(f(x,x), x)	11000	5	2^{-5}	2
4	f(f(x,f(x,x)), x)	1101000	7	2^{-7}	5
5	f(f(x,f(x,x)), f(x,x))	110100100	9	2^{-9}	14
...					
n			2n-1	$2^{-(2n-1)}$	C(n-1)

  - Catalan number C_n
    - C_n is the number of different ways n + 1 factors can be completely parenthesized
    - C_n is also the number of full binary trees with n+1 leaves
  - Expectation of the size of formula: infinite

$$E\left(s\left(X\right)\right)=\sum_{n=1}^{\infty} n 2^{-(2n-1)} C_{n-1} = \infty$$

Exploitation of domain knowledge

Draw many constructed variables simultaneously

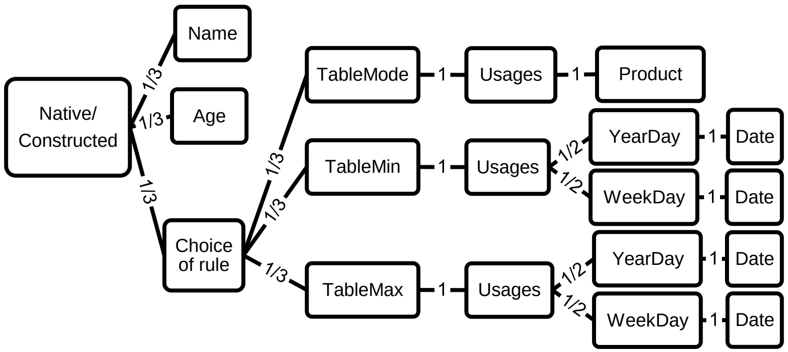
- Principle
  - Draw directly a sample of variables according to prior HMDID
  - Exploit the multinomial maximum likelihood of the whole sample

$$p(D) = \frac{n!}{n_1! n_2! \dots n_K!} p_1^{n_1} p_2^{n_2} \dots p_K^{n_K}$$

M L reached with frequencies $n_k = p_k n$

- Whole sample algorithm: simultaneous random draws
  - Input: K {Number of draws}
  - Output: $\mathbb{X}=\{X\}$,$|\mathbb{X}| \leq K$ {Sample of constructed variables}
    - 1: $\mathbb{X}=\varnothing$
    - 2: Start from root node of hierarchy of HMDID prior
    - 3: Compute number of draws K_i per child node of the prior (native variable, rule, operand...)
    - 4: **for all** child node in current node of the prior **do**
    - 5: **if** leaf node of the prior (constructed variable with complete formula) **then**
    - 6: Add X into $\mathbb{X}$
    - 7: **else**
    - 8: Propagate construction recursively by distributing the K_i draws on each child node according to the multinomial distribution
    - 9: **end if**
    - 10: **end for**

- The whole sample algorithm is both efficient and computable



Schedule

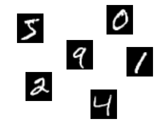
Automatic Feature Construction
for Supervised Classification from
Large Scale Multi-Relational Data

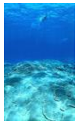
- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
  - Specification of domain knowledge
  - Evaluation of constructed variables
  - Sampling a subset of constructed variables
  - Experiments
- Conclusion

Benchmark

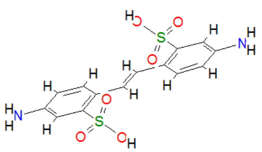
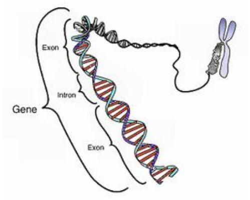
Datasets

- 14 benchmark multi-tables datasets
  - Various domains
    - Handwritten digit
    - Pen tip trajectory character
    - Australian sign language
    - Image
    - Speaker recognition
    - Molecular chemistry
    - Genomics
    - …
  - Various sizes and complexity
    - 100 to 5000 instances
    - 500 to 5000000 records in secondary tables
    - Numerical and categorical variables
    - 2 to 96 classes
    - Unbalanced class distribution





Dataset	Instances	Records	Cat. var	Num. var	Classes	Maj.
Auslan	2565	146949	1	23	96	0.011
CharacterTrajectories	2858	487277	1	4	20	0.065
Diterpenes	1503	30060	2	1	23	0.298
JapaneseVowels	640	9961	1	13	9	0.184
MimlDesert	2000	18000	1	15	2	0.796
MimlMountains	2000	18000	1	15	2	0.771
MimlSea	2000	18000	1	15	2	0.71
MimlSunset	2000	18000	1	15	2	0.768
MimlTrees	2000	18000	1	15	2	0.72
Musk1	92	476	1	166	2	0.511
Musk2	102	6598	1	166	2	0.618
Mutagenesis	188	10136	3	4	2	0.665
OptDigits	5620	5754880	1	3	10	0.102
SpliceJunction	3178	191400	2	1	3	0.521

Benchmark

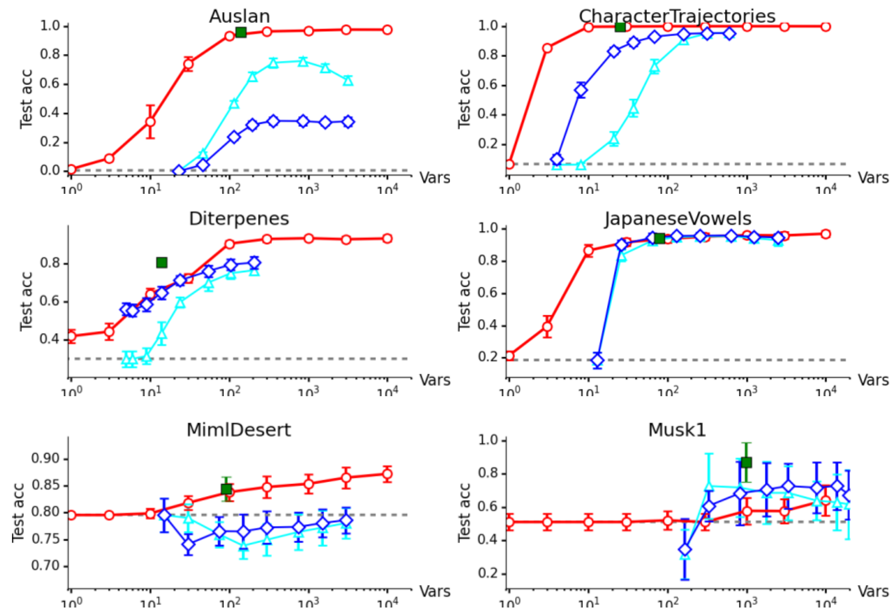
Evaluation protocol

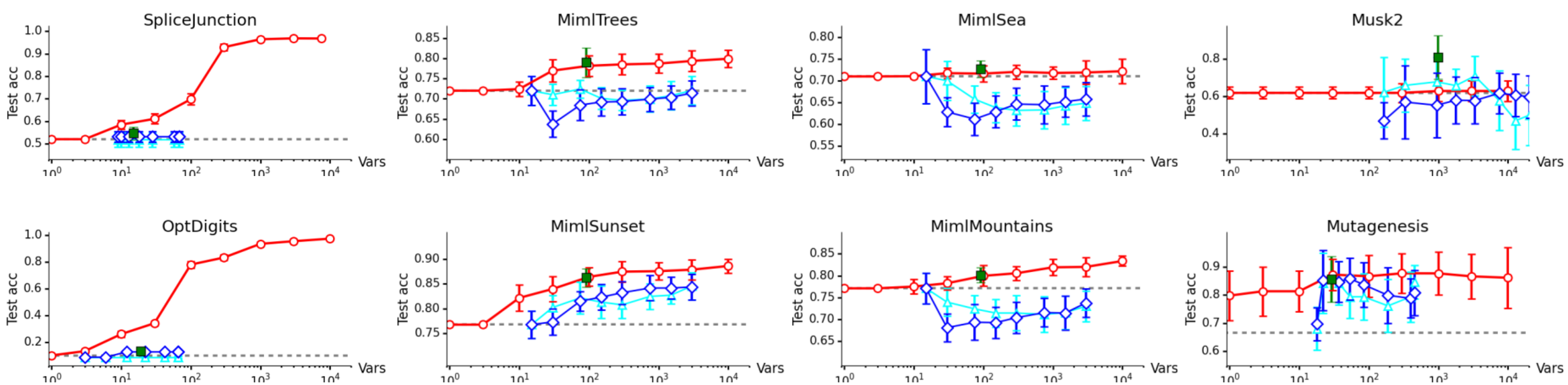
- Compared methods
  - MODL: our method
    - Construction rules: Selection, Count, Mode, CountDistinct, Mean, Median, Min, Max StdDev, Sum
    - Preprocessing: supervised discretisation and value grouping
    - Classifier: Selective Naive Bayes (variable selection and model averaging)
    - Number of variables to construct: 1, 3, 10, 30, 10, 300, 1000, 3000, 10000
  - RELAGGS: (Krogel et al, 2001)
    - Construction rules: same as MODL (except Selection), plus Count per categorical value
    - Preprocessing and classifier: same as MODL
  - 1BC: (Lachiche et al, 1999)
    - first-order Bayesian classifier with prepositionalisation
    - Preprocessing: equal frequency discretization with 1, 2, 5, 10, 20, 50, 100, 200 bins
  - 1BC2: (Lachiche et al, 2002)
    - Successor of 1BC
    - True first order classifier
- Evaluation protocol
  - Stratified 10-fold cross validation
  - Collected results: number of constructed variables and test accuracy

Benchmark results

Control of variable construction

- RELAGGS, 1BC, 1BC2:
  - No control on the number of constructed variables
- MODL
  - Exactly the requested number of constructed variables

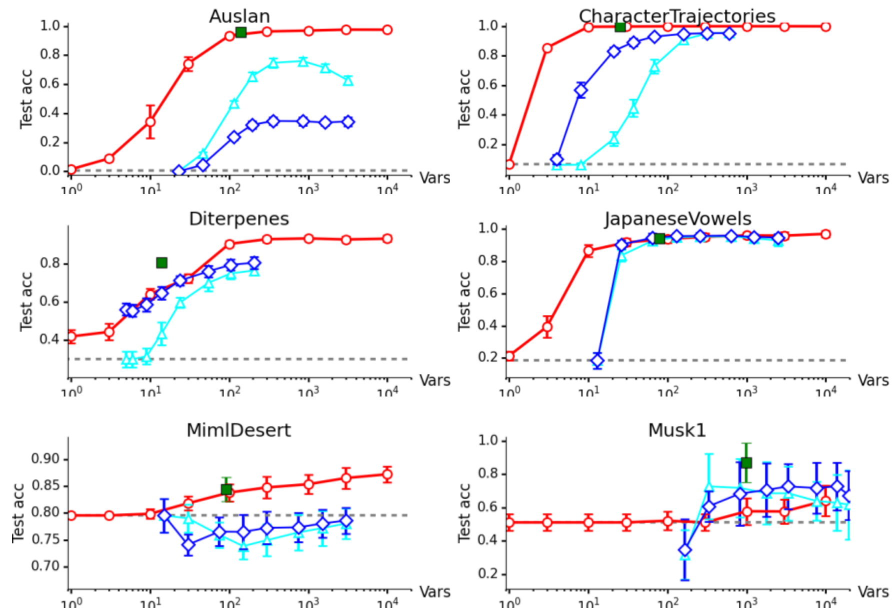


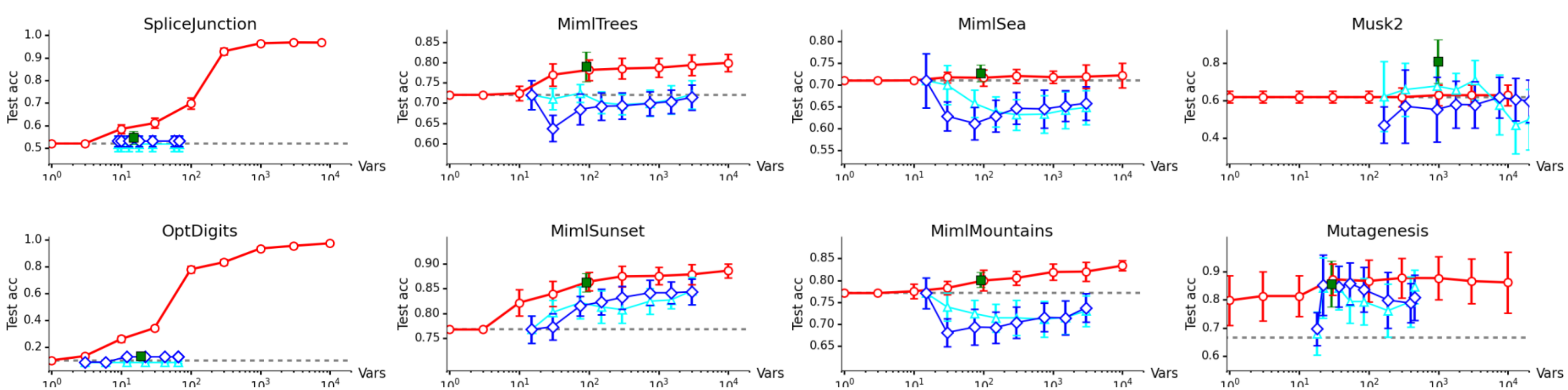


Benchmark results

Test accuracy

- 1BC, 1BC2:
  - Similar performance
- RELAGGS:
  - Better than 1BC and 1BC2
- MODL
  - Underfit in tiny datasets (Musk)
  - Performance increases with the number of variables
  - Best accuracy overall





Benchmark: robustness

- Protocol
  - Random shuffle of class values in each dataset
  - Experiments repeated in 10 cross-validation
    - 10000 constructed variables per dataset in each fold
    - 1.4 million of variables evaluated overall
- Results
  - With construction regularization
    - Not one single wrongly selected variable, among the 1.4 million
    - Highly robust approach

Use cases in Orange

- Experiments on large datasets
  - 100 000 customers
    - up to millions in main table
  - 50 millions call detail records
    - up to billions in secondary tables
    - up to hundreds of GB
  - Up to 100 000 automatically constructed variables

- Results
  - Genericiy
  - Parameter-free
    - Rely on domain knowledge description: multi-table specification and choice of construction rules
  - Reliability
  - Accuracy
  - Interpretability:
    - Constructed variables may be numerous, redundant and some of them complex
  - Efficicency

- Use cases and methodology: needs to be invented
  - Automatic evaluation of additional data sources
  - Fast automatic solution to many data mining problems
  - Help to suggest new variables to construct
  - …

Schedule

Towards automatic variable construction, data preparation and modeling for large scale multi-tables datasets

- Introduction
- Automatic data preparation (single-table dataset)
- Automatic variable construction (multi-tables dataset)
- Conclusion

Summary

- Variable selection using data grid models
  - Discretization/value grouping
  - Conditional/joint density estimation
- Specification of domain knowledge
  - Multi-table format, advanced data types (Date, Time…)
  - Construction variable language
- Specification of a prior distribution on the space of variable construction
  - Hierarchy of Multinomial Distributions with potentially Infinite Depth
- Sampling algorithm on this infinite variable construction space
  - Concept of maximum likelihood of a whole sample of variables
- Experiments with promising results, on many multi-tables datasets
  - Now widely used on large Orange datasets: effective automation of variable construction

tool available at www.khiops.com or www.predicsis.com (commercial use)

Future work

- Future work: numerous open problems
  - Design of more parsimonious prior
  - Extension of the specification of domain knowledge
  - Large scale parallelization for exploration of the space of variable construction
  - Sampling constructed variable according to their posterior (*vs*. prior) distribution
  - Any time variable construction, jointly with multivariate classifier training
  - …

thank you for your attention!

References

- Data preparation
  - M. Boullé. A Bayes optimal approach for partitioning the values of categorical attributes. Journal of Machine Learning Research, 6:1431-1452, 2005
  - M. Boullé. MODL: a Bayes optimal discretization method for continuous attributes. Machine Learning, 65(1):131-165, 2006
  - M. Boullé. Data grid models for preparation and modeling in supervised learning. In Hands-On Pattern Recognition: Challenges in Machine Learning, volume 1, I. Guyon, G. Cawley, G. Dror, A. Saffari (eds.), pp. 99-130, Microtome Publishing, 2011

- Modeling
  - M. Boullé. Compression-Based Averaging of Selective Naive Bayes Classifiers. Journal of Machine Learning Research, 8:1659-1685, 2007

- Feature construction
  - M. Boullé. Towards Automatic Feature Construction for Supervised Classification. In ECML/PKDD 2014, Pages 181-196, 2014