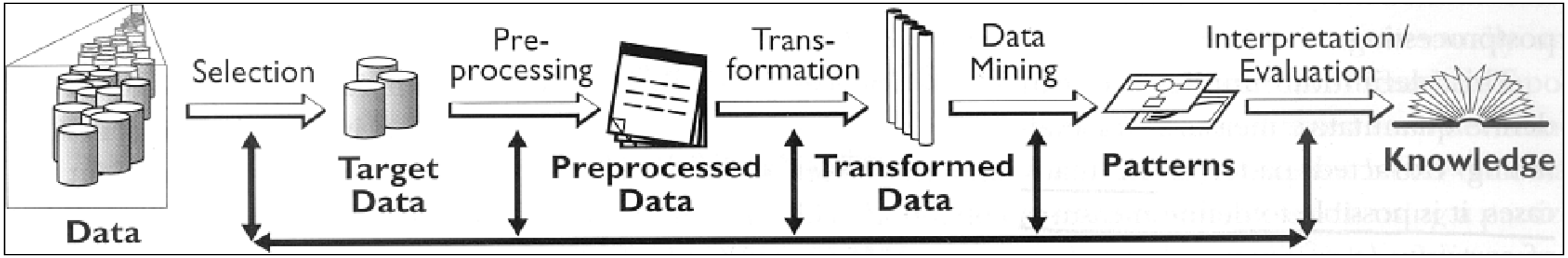
Data Mining and Knowledge Discovery: Practice Notes

Petra Kralj Novak
Petra.Kralj.Novak@ijs.si
19.11.2019

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Keywords



- Data
  - Attribute, example, attribute-value data, target variable, class, discretization
- Algorithms
  - Decision tree induction, entropy, information gain, overfitting, Occam's razor, model pruning, naïve Bayes classifier, KNN, association rules, support, confidence, numeric prediction, regression tree, model tree, heuristics vs. exhaustive search, predictive vs. descriptive DM
- Evaluation
  - Train set, test set, accuracy, confusion matrix, cross validation, true positives, false positives, ROC space, error, precision, recall

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can’t we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Comparison of naïve Bayes and decision trees

- Similarities
  - Classification
  - Same evaluation

- Differences
  - Missing values
  - Numeric attributes
  - Interpretability of the model
  - Model size

Comparison of naïve Bayes and decision trees: Handling missing values

Will the spider catch these two ants?

- Color = white, Time = night ← **missing value for attribute Size**
- Color = black, Size = large, Time = day

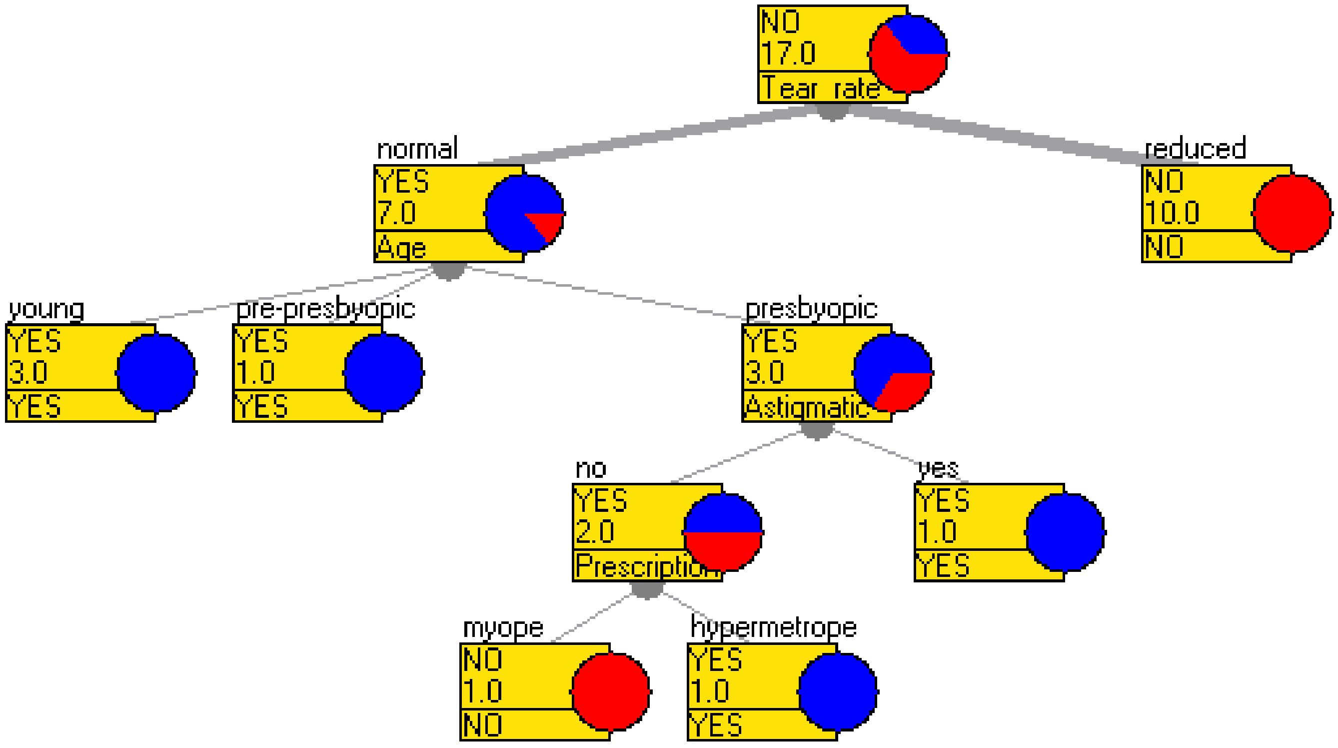
$$
\begin{aligned}
P(C_1|v_1, v_2) &= \\
&= P(\text{YES}|C = w, T = n) \\
&= P(\text{YES}) \cdot P(C = w|\text{YES}) \cdot P(T = n|\text{YES}) \\
&= \frac{1}{2} \cdot \frac{1}{3} \cdot \frac{1}{3} \\
&= \frac{1}{18}
\end{aligned}
$$

$$
\begin{aligned}
P(C_2|v_1, v_2) &= \\
&= P(\text{NO}|C = w, T = n) \\
&= P(\text{NO}) \cdot P(C = w|\text{NO}) \cdot P(T = n|\text{NO}) \\
&= \frac{1}{2} \cdot \frac{1}{3} \cdot 1 \\
&= \frac{1}{6}
\end{aligned}
$$

Naïve Bayes uses all the available information.

Comparison of naïve Bayes and decision trees: Handling missing values

Age	Prescription	Astigmatic	Tear_Rate
?	hypermetrope	no	normal
pre-presbyopic	myope	?	normal



Comparison of naïve Bayes and decision trees: Handling missing values

Algorithm **ID3**: does not handle missing values

Algorithm **C4.5** (J48) deals with two problems:

- Missing values in **train** data:
  - Missing values are not used in gain and entropy calculations
- Missing values in **test** data:
  - A missing **continuous** value is replaced with the median of the training set
  - A missing **categorical** values is replaced
  with the most frequent value

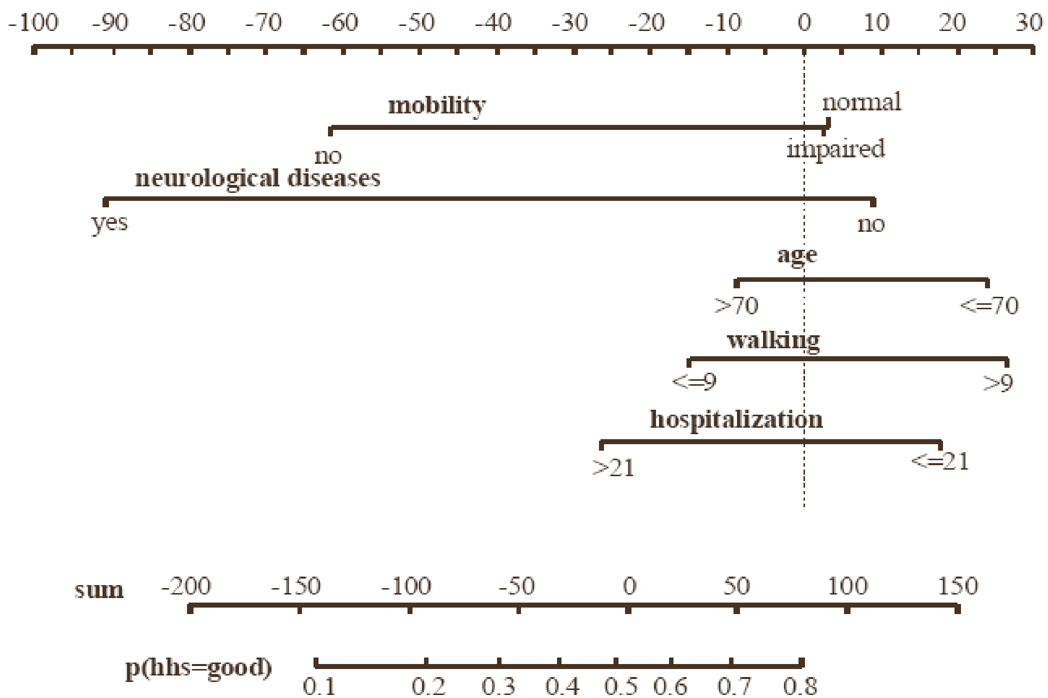
Comparison of naïve Bayes and decision trees: numeric attributes

- Decision trees **ID3** algorithm: does not handle continuous attributes → data need to be discretized
- Decision trees **C4.5** (J48 in Weka) algorithm: deals with continuous attributes as shown earlier
- **Naïve Bayes**: does not handle continuous attributes → data need to be discretized

(some implementations do handle)

Comparison of naïve Bayes and decision trees: Interpretability

- Decision trees are easy to understand and interpret (if they are of moderate size)
- Naïve bayes models are of the "black box type".
- Naïve bayes models have been visualized by nomograms.



DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Comparison of naïve Bayes and decision trees: Model size

- Naïve Bayes model size is low and quite constant with respect to the data
- Trees, especially random forest tend to be very large

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Comparison of cross validation and testing on a separate test set

- Both are methods for evaluating predictive models.
- Testing on a separate test set is simpler since we split the data into two sets: one for training and one for testing. We evaluate the model on the test data.
- Cross validation is more complex: It repeats testing on a separate test *n* times, each time taking 1/n of different data examples as test data. The evaluation measures are averaged over all testing sets therefore the results are more reliable.

(Train – Validation – Test) Set

- **Training** set: a set of examples used **for learning**
- **Validation** set: a set of examples used to **tune the parameters** of a classifier
- **Test** set: a set of examples used **only to assess the performance of a fully-trained classifier**
- Why separate test and validation sets? The error rate estimate of the final model on validation data will be biased (optimistic compared to the true error rate) since the validation set is used to select the final model. After assessing the final model on the test set, YOU MUST NOT tune the model any further!

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Decision tree pruning

- To avoid overfitting
- Reduce size of a model and therefore increase understandability.

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Discretization

- A good choice of intervals for discretizing your continuous feature is key to improving the predictive performance of your model.
- Hand-picked intervals – good knowledge about the data
- Equal-width intervals probably won't give good results
- Find the right intervals using existing data:
  - Equal frequency intervals
  - If you have labeled data, another common technique is to find the intervals which maximize the information gain
  - **Caution: The decision about the intervals should be done based on training data only**
- Global (before building the model) and local (during model construction on a subset at hand) discretization

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Why can't we always achieve 100% accuracy on the training set?

- Two examples have the same attribute values but different classes (noisy data)
- Run out of attributes

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can’t we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Relative frequency vs. Laplace estimate

Relative frequency

- **P(c) = n(c) /N**
- A disadvantage of using relative frequencies for probability estimation arises with small sample sizes, especially if they are either very close to zero, or very close to one.
- In our spider example:
  P(Time=day|caught=NO) =
  = 0/3 = 0

n(c) … number of examples where c is true
N … number of all examples
k … number of possible events

Laplace estimate

- Assumes uniform prior distribution over the probabilities for each possible event
- **P(c) = (n(c) + 1) / (N + k)**
- In our spider example: P(Time=day|caught=NO) = (0+1)/(3+2) = 1/5
- With lots of evidence approximates relative frequency
- If there were 300 cases when the spider didn't catch ants at night: P(Time=day|caught=NO) = (0+1)/(300+2) = 1/302 = 0.003
- With Laplace estimate probabilities can never be 0.

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

Why does Naïve Bayes work well?

$$\text{classification} = \operatorname{argmax}_{c_i} P(c_i) \prod_{j=1}^{n} P(v_j | c_i)$$

Because classification doesn't require accurate probability estimates as long as maximum probability is assigned to the correct class.

Discussion

1. Compare naïve Bayes and decision trees (similarities and differences) .
2. Compare cross validation and testing on a separate test set.
3. Why do we prune decision trees?
4. What is discretization.
5. Why can't we always achieve 100% accuracy on the training set?
6. Compare Laplace estimate with relative frequency.
7. Why does Naïve Bayes work well (even if independence assumption is clearly violated)?
8. What are the benefits of using Laplace estimate instead of relative frequency for probability estimation in Naïve Bayes?

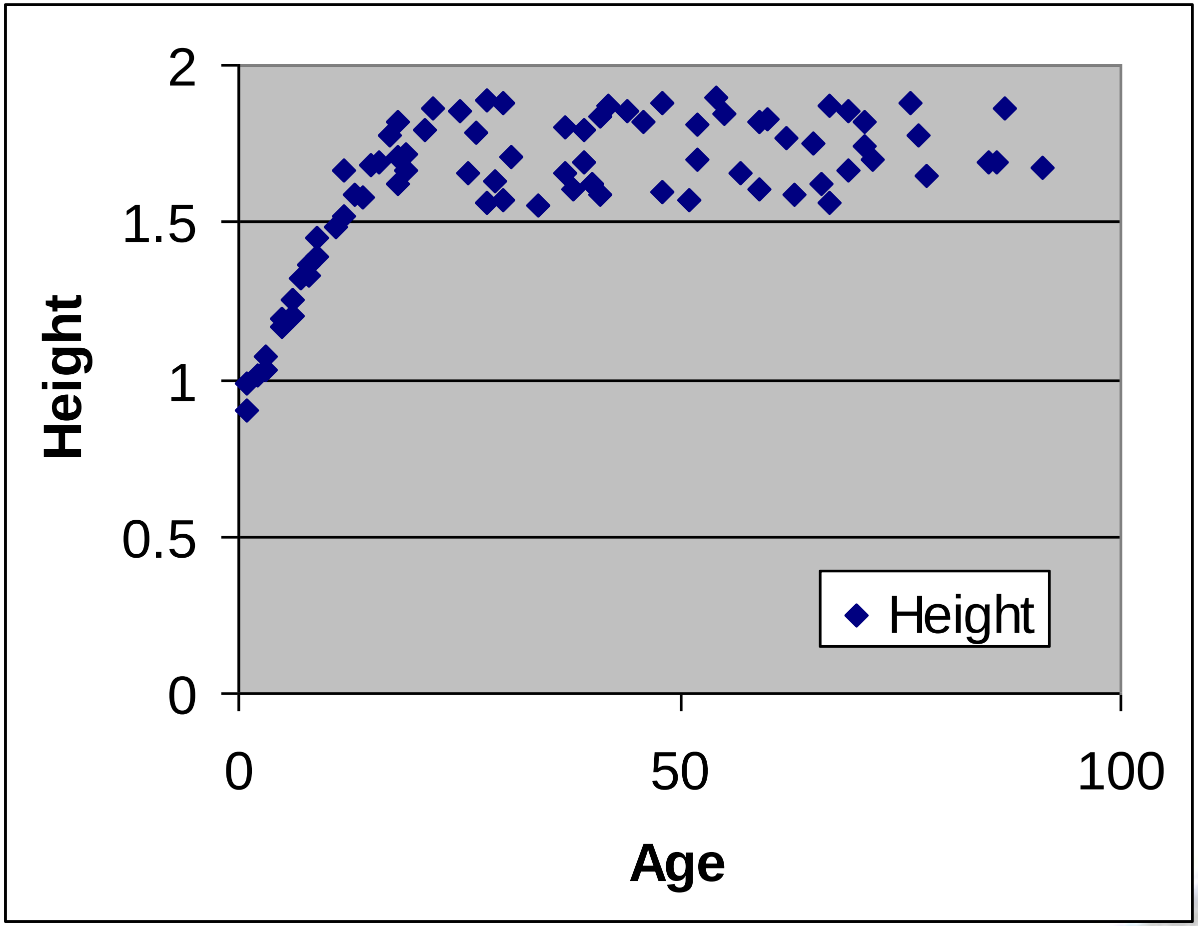
Benefits of Laplace estimate

- With Laplace estimate we avoid assigning a probability of 0, as it denotes an impossible event
- Instead we assume uniform prior distribution of k classes

Numeric prediction

Example

- data about 80 people: Age and Height



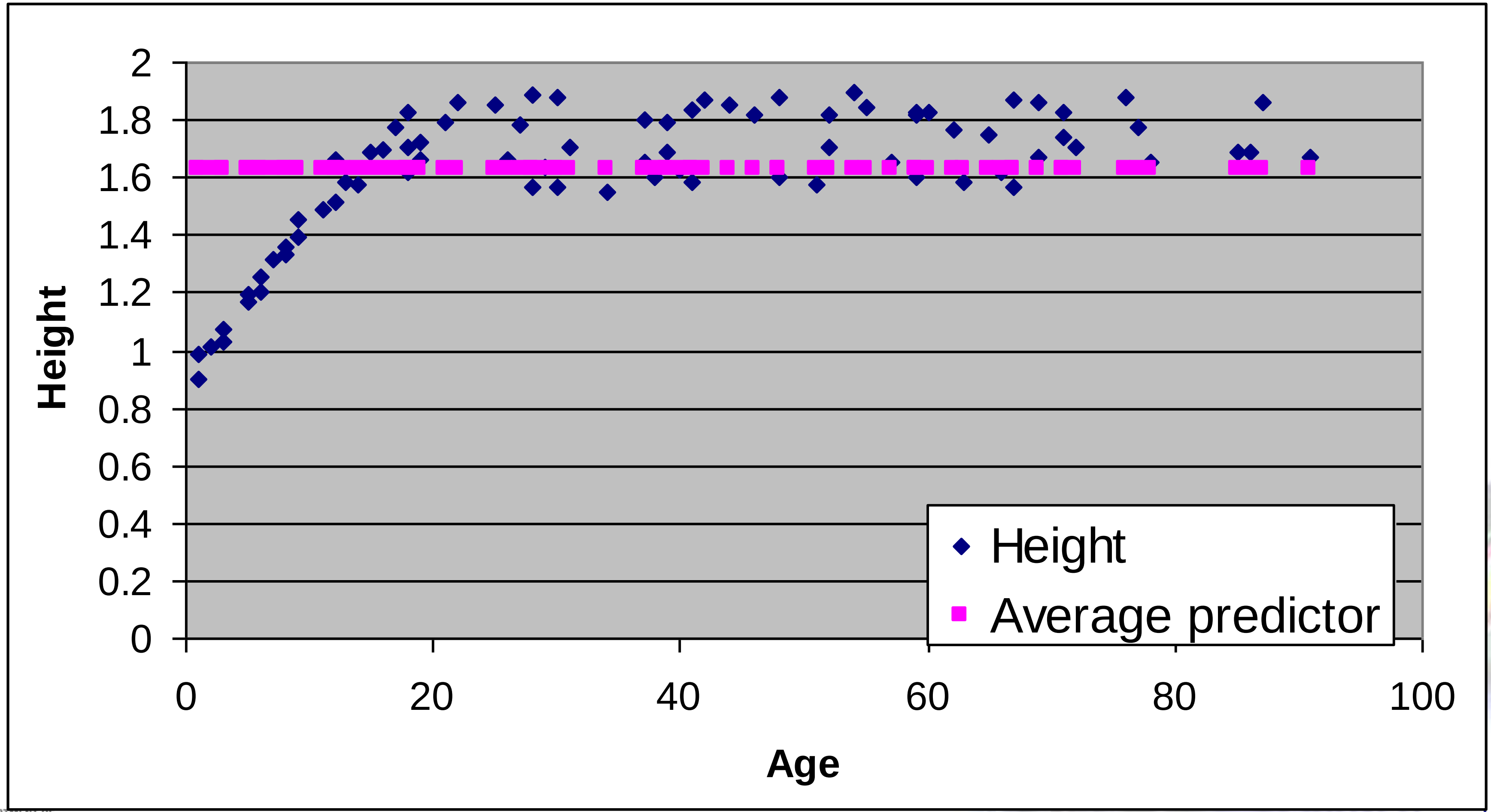
Age	Height
3	1.03
5	1.19
6	1.26
9	1.39
15	1.69
19	1.67
22	1.86
25	1.85
41	1.59
48	1.60
54	1.90
71	1.82
...	...

Test set

Age	Height
2	0.85
10	1.4
35	1.7
70	1.6

Baseline numeric predictor

- Average of the target variable



DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

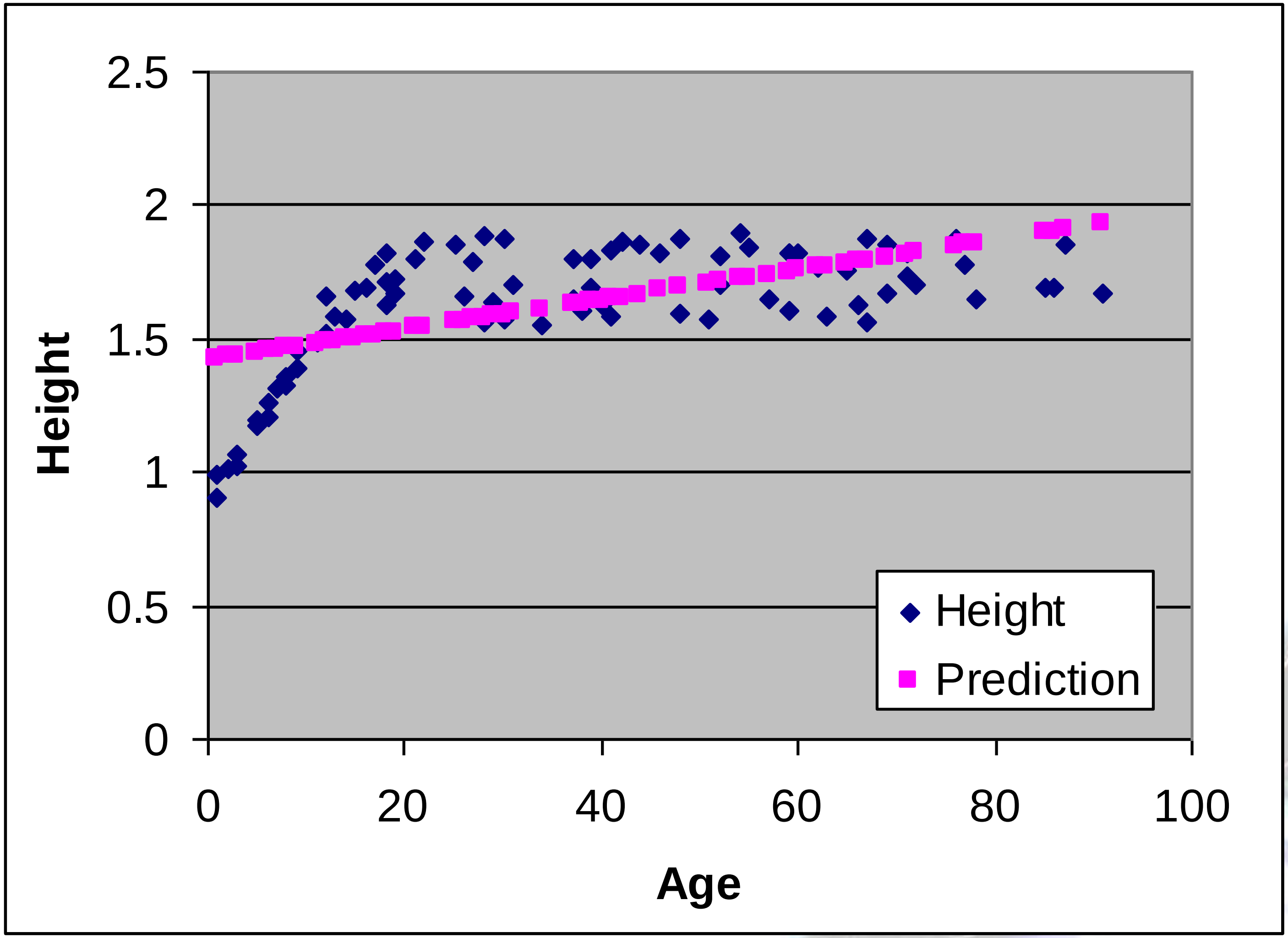
Baseline predictor: prediction

Average of the target variable is 1.63

Age	Height	Baseline
2	0.85	
10	1.4	
35	1.7	
70	1.6	

Linear Regression Model

Height = 0.0056 * Age + 1.4181



DEPARTMENT OF KNOWLEDGE TECHNOLOGIES Jožef Stefan Institute

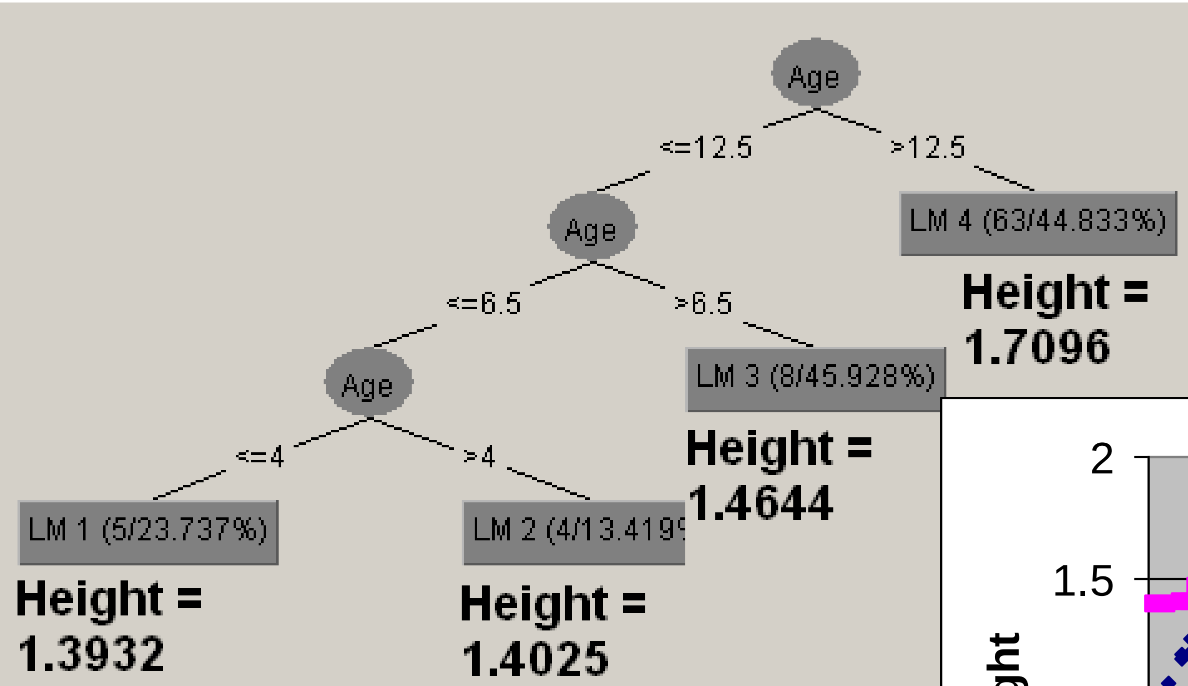
Linear Regression: prediction

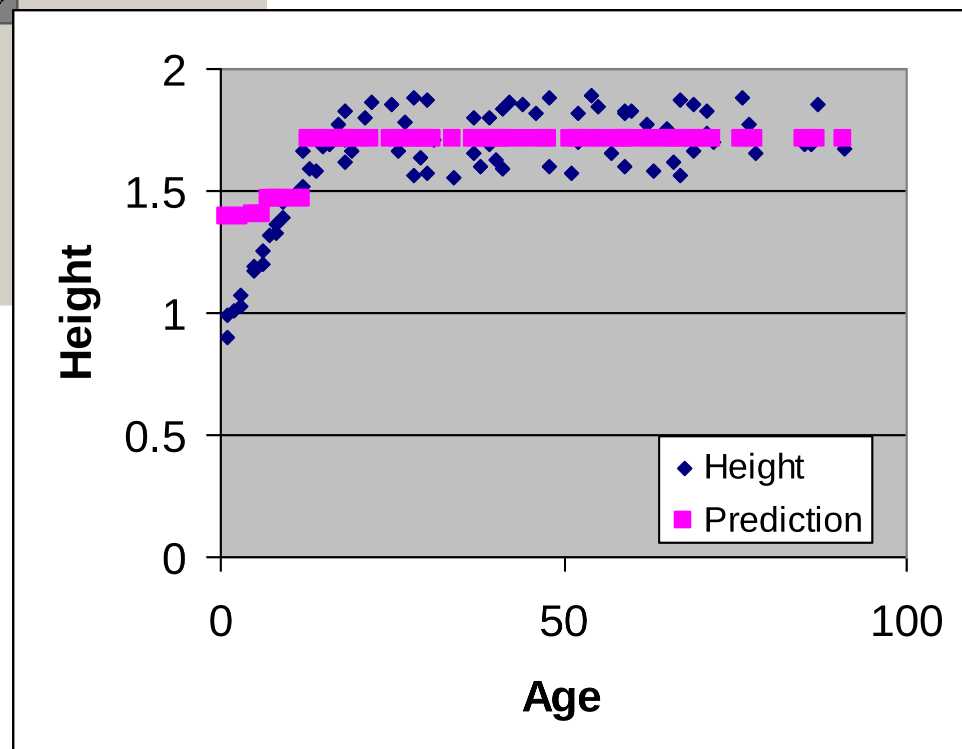
Height = 0.0056 * Age + 1.4181

Age	Height	Linear regression
2	0.85	
10	1.4	
35	1.7	
70	1.6	

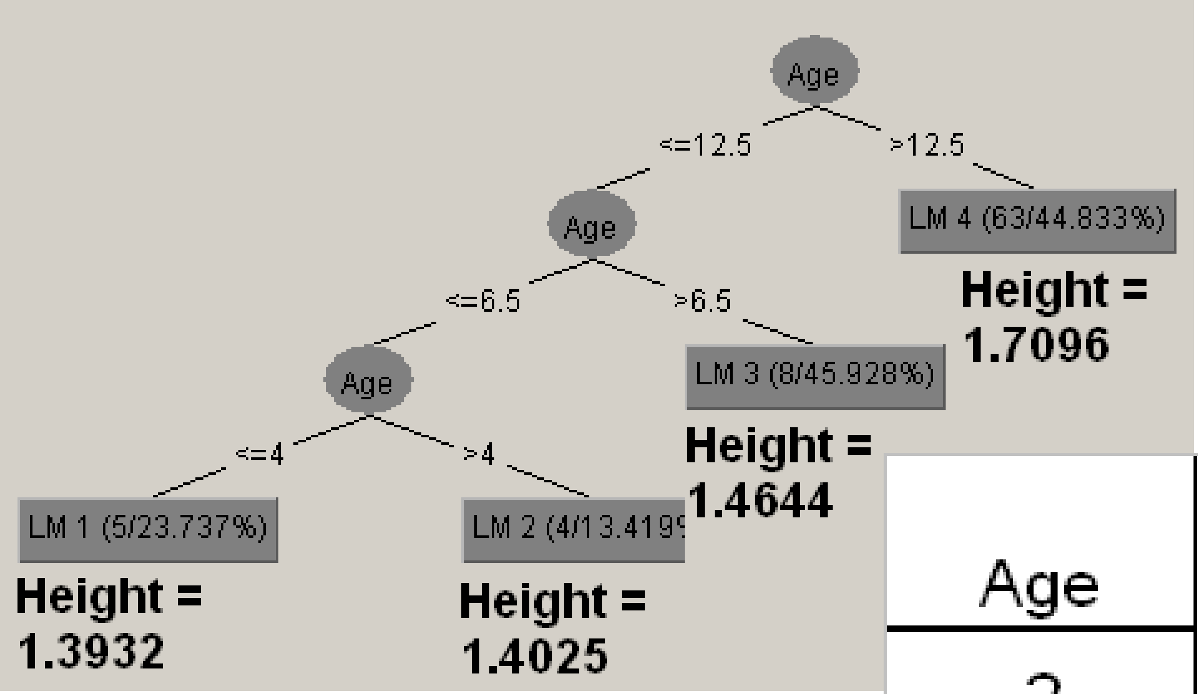
DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Regression tree



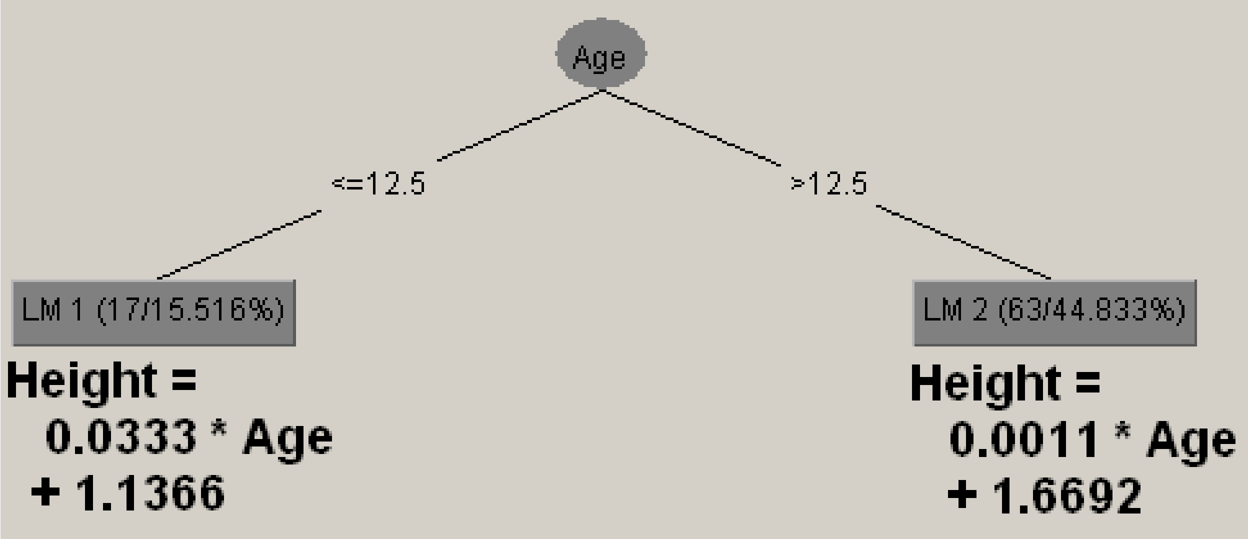


Regression tree: prediction

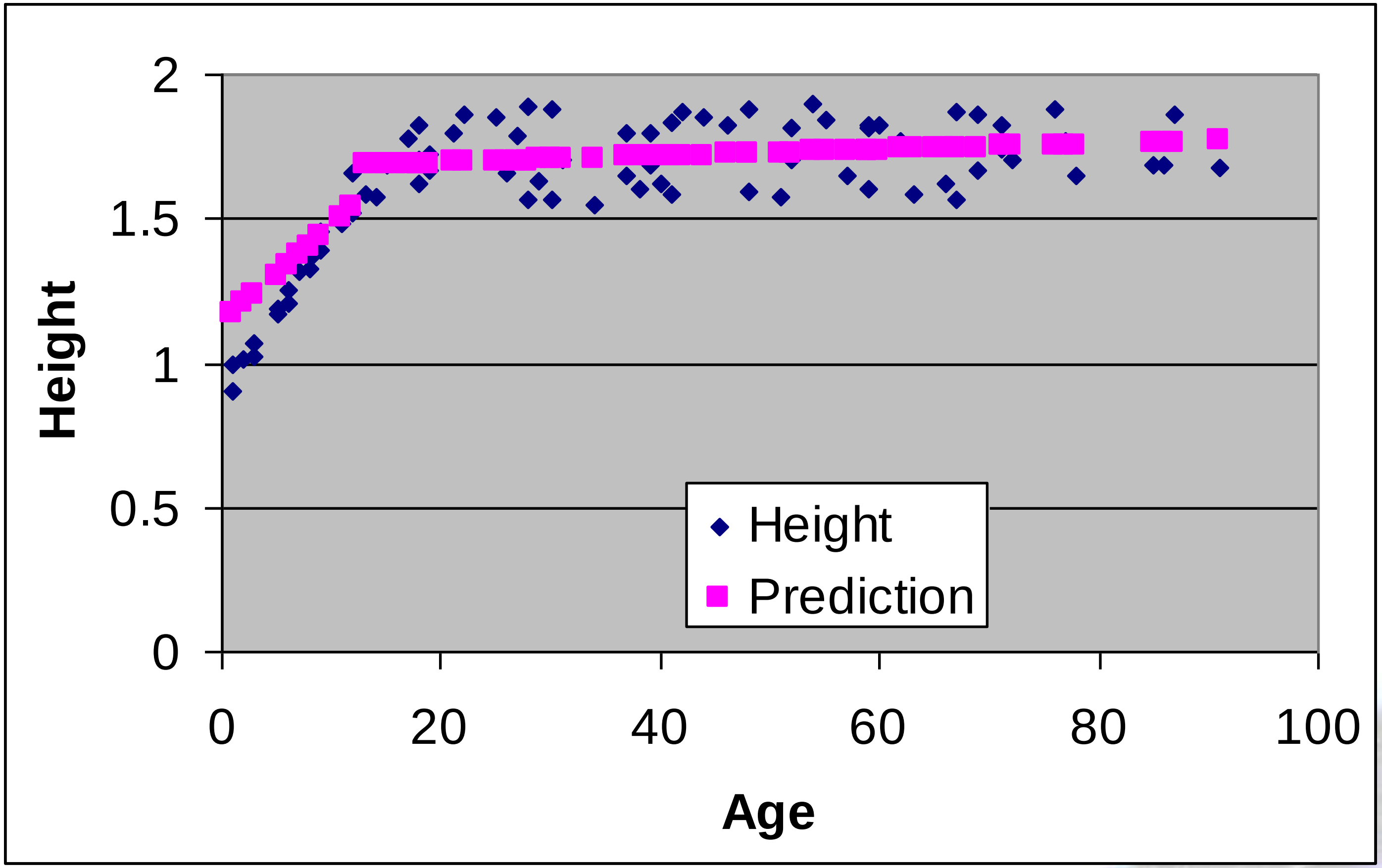


Age	Height	Regression tree
2	0.85	
10	1.4	
35	1.7	
70	1.6	

Model tree



Age
<=12.5
>12.5
LM 1 (17/15.516%)
LM 2 (63/44.833%)
Height =
0.0333 * Age
+ 1.1366
Height =
0.0011 * Age
+ 1.6692



2
1.5
1
0.5
0
Height
0
20
40
60
80
100
Age
Height
Prediction

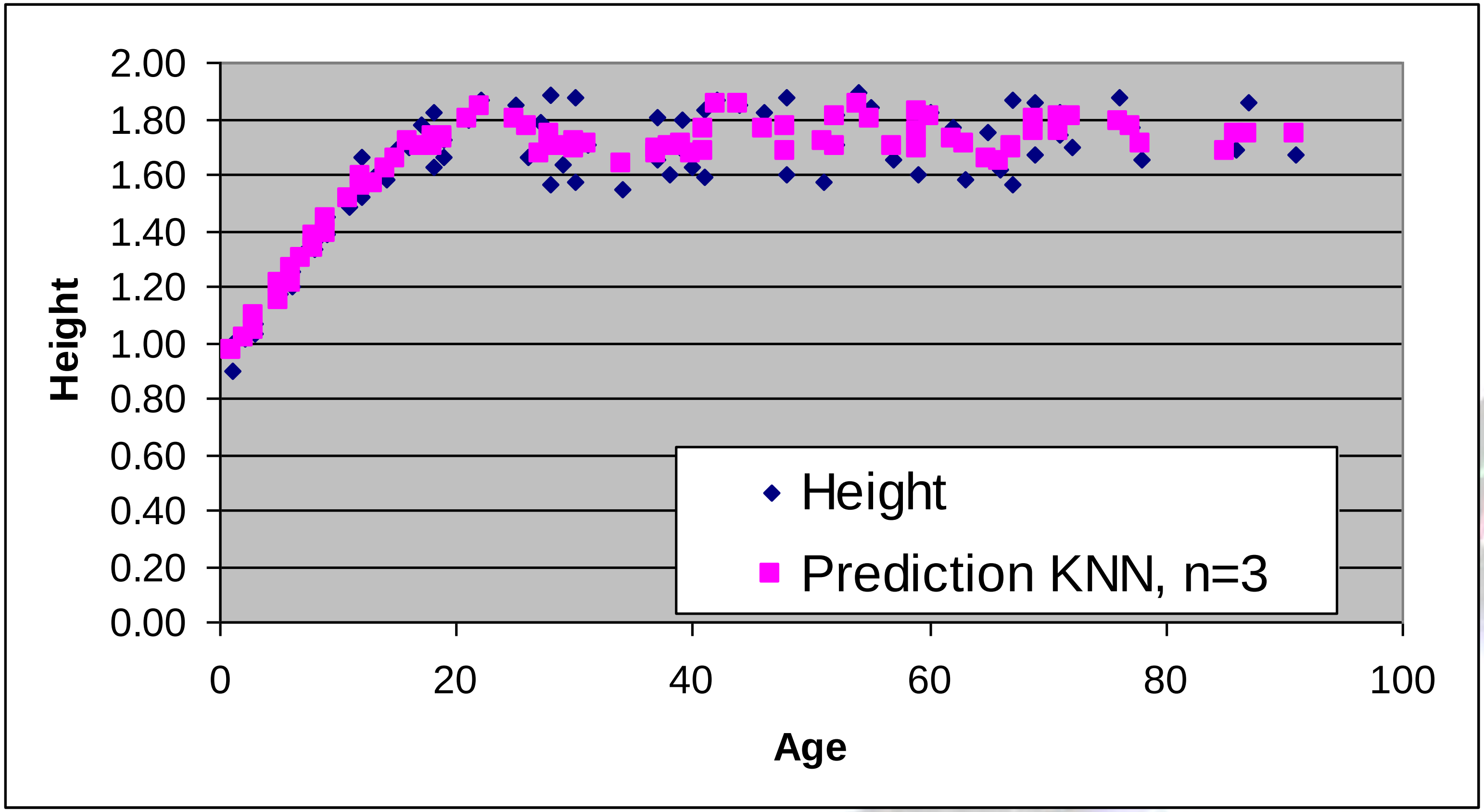
Model tree: prediction

Age	Height	Model tree
2	0.85	
10	1.4	
35	1.7	
70	1.6	

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES Jožef Stefan Institute

KNN – K nearest neighbors

- Looks at K closest examples (by non-target attributes) and predicts the average of their target variable
- In this example, K=3



DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

KNN prediction

Age	Height
1	0.90
1	0.99
2	1.01
3	1.03
3	1.07
5	1.19
5	1.17

Age	Height	kNN
2	0.85	
10	1.4	
35	1.7	
70	1.6	

KNN prediction

Age	Height
8	1.36
8	1.33
9	1.45
9	1.39
11	1.49
12	1.66
12	1.52
13	1.59
14	1.58

Age	Height	kNN
2	0.85	
10	1.4	
35	1.7	
70	1.6	

KNN prediction

Age	Height
30	1.57
30	1.88
31	1.71
34	1.55
37	1.65
37	1.80
38	1.60
39	1.69
39	1.80

Age	Height	kNN
2	0.85	
10	1.4	
35	1.7	
70	1.6	

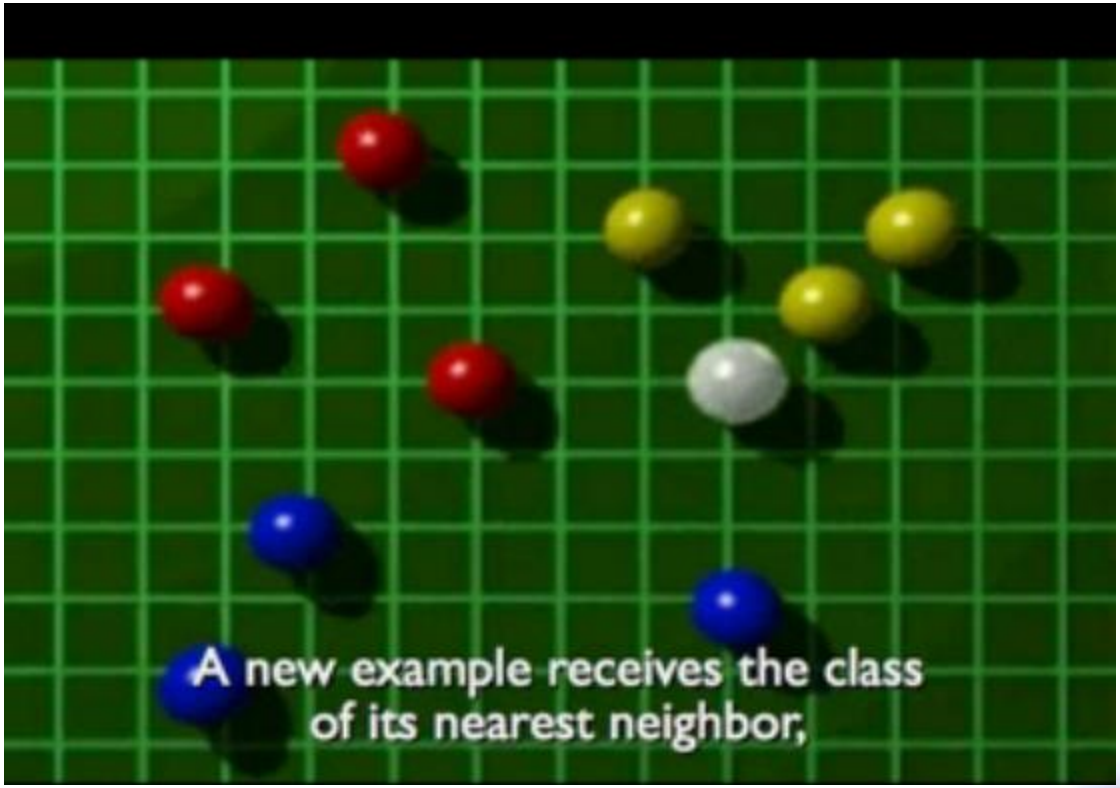
KNN prediction

Age	Height
67	1.56
67	1.87
69	1.67
69	1.86
71	1.74
71	1.82
72	1.70
76	1.88

Age	Height	kNN
2	0.85	
10	1.4	
35	1.7	
70	1.6	

KNN video

- http://videolectures.net/aaai07_bosch_knnc



Which predictor is the best?

Age	Height	Baseline	Linear regression	Regression tree	Model tree	kNN
2	0.85	1.63	1.43	1.39	1.20	1.00
10	1.4	1.63	1.47	1.46	1.47	1.44
35	1.7	1.63	1.61	1.71	1.71	1.67
70	1.6	1.63	1.81	1.71	1.75	1.77

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Evaluating numeric prediction

Performance measure	Formula
mean-squared error	$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}$
root mean-squared error	$\sqrt{\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}}$
mean absolute error	$\frac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{n}$
relative squared error	$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}$, where $\bar{a}=\frac{1}{n}\sum_i a_i$
root relative squared error	$\sqrt{\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(a_1-\bar{a})^2+\ldots+(a_n-\bar{a})^2}}$
relative absolute error	$\frac{\lvert p_1-a_1\rvert+\ldots+\lvert p_n-a_n\rvert}{\lvert a_1-\bar{a}\rvert+\ldots+\lvert a_n-\bar{a}\rvert}$
correlation coefficient	$\frac{S_{PA}}{\sqrt{S_P S_A}}$, where $S_{PA}=\frac{\sum_i(p_i-\bar{p})(a_i-\bar{a})}{n-1}$, $S_p=\frac{\sum_i(p_i-\bar{p})^2}{n-1}$, and $S_A=\frac{\sum_i(a_i-\bar{a})^2}{n-1}$

Numeric prediction	**Classification**
Data: attribute-value description	
Target variable: Continuous	**Target variable**: Categorical (nominal)
Evaluation: cross validation, separate test set, …	
Error: MSE, MAE, RMSE, …	**Error**: 1-accuracy
Algorithms: Linear regression, regression trees,…	**Algorithms**: Decision trees, Naïve Bayes, …
Baseline predictor: Mean of the target variable	**Baseline predictor**: Majority class

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Association Rules

Association rules

- Rules **X → Y**, X, Y conjunction of items
- Task: Find **all** association rules that satisfy minimum support and minimum confidence constraints

- **Support**:

  $Sup(X \rightarrow Y) = \#XY/\#D \cong p(XY)$

- **Confidence**:

  $Conf(X \rightarrow Y) = \#XY/\#X \cong p(XY)/p(X) = p(Y|X)$

Association rules - algorithm

1. Generate frequent itemsets with a minimum support constraint
2. Generate rules from frequent itemsets with a minimum confidence constraint

* Data are in a transaction database

Association rules – transaction database

Items: **A**=apple, **B**=banana,
C=coca-cola, **D**=doughnut

- Client 1 bought: A, B, C, D
- Client 2 bought: B, C
- Client 3 bought: B, D
- Client 4 bought: A, C
- Client 5 bought: A, B, D
- Client 6 bought: A, B, C

Frequent itemsets

- Generate frequent itemsets with support at least 2/6

A	B	C	D
1	1	1	1
	1	1	
	1		1
1		1	
1	1		1
1	1	1	

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

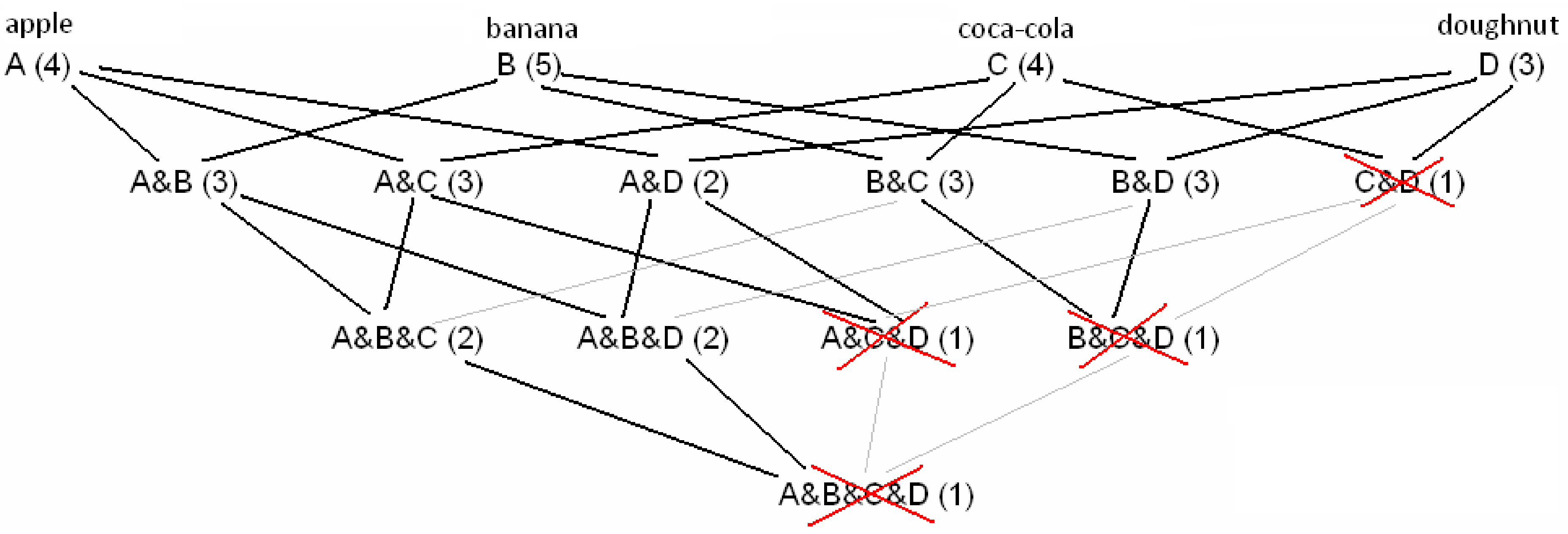
Frequent itemsets algorithm

Items in an itemset should be **sorted** alphabetically.

1. Generate all 1-itemsets with the given minimum support.
2. Use 1-itemsets to generate 2-itemsets with the given minimum support.
3. From 2-itemsets generate 3-itemsets with the given minimum support as unions of 2-itemsets with the same item at the beginning.
4. …
5. From n-itemsets generate (n+1)-itemsets as unions of n-itemsets with the same (n-1) items at the beginning.

- To generate itemsets at level n+1 items from level n are used with a constraint: itemsets have to start with the same n-1 items.

Frequent itemsets lattice



Frequent itemsets:

- A&B, A&C, A&D, B&C, B&D
- A&B&C, A&B&D

Rules from itemsets

- A&B is a frequent itemset with support 3/6
- Two possible rules
  - A→B confidence = #(A&B)/#A = 3/4
  - B→A confidence = #(A&B)/#B = 3/5
- All the counts are in the itemset lattice!

Quality of association rules

Support(X) = #X / #D .... P(X)
Support(X→Y) = Support (XY) = #XY / #D .... P(XY)
Confidence(X→Y) = #XY / #X .... P(Y|X)

Lift(X→Y) = Support(X→Y) / (Support (X)*Support(Y))

Leverage(X→Y) = Support(X→Y) – Support(X)*Support(Y)

Conviction(X → Y) = 1-Support(Y)/(1-Confidence(X→Y))

Next week

- Hands on Scikit-learn
- **Scikit-learn** is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

- Install:
  - Python 3
  - NumPy
  - Pandas
  - Scikit-learn

  - Recommended IDE: PyCharm Community

Discussion – part 1

1. Can KNN be used for classification tasks?
2. Compare KNN and Naïve Bayes.
3. Compare decision trees and regression trees.
4. Consider a dataset with a target variable with five possible values:
    1. non sufficient
    2. sufficient
    3. good
    4. very good
    5. excellent

    1. Is this a classification or a numeric prediction problem?
    2. What if such a variable is an attribute, is it nominal or numeric?

Discussion – part 2

- Transformation of an attribute-value dataset to a transaction dataset.
- What are the benefits of a transaction dataset?
- What would be the association rules for a dataset with two items A and B, each of them with support 80% and appearing in the same transactions as rarely as possible?
  - minSupport = 50%, min conf = 70%
  - minSupport = 20%, min conf = 70%
- What if we had 4 items: A, ¬A, B, ¬ B
- Compare decision trees and association rules regarding handling an attribute like “PersonID”. What about attributes that have many values (eg. Month of year)