



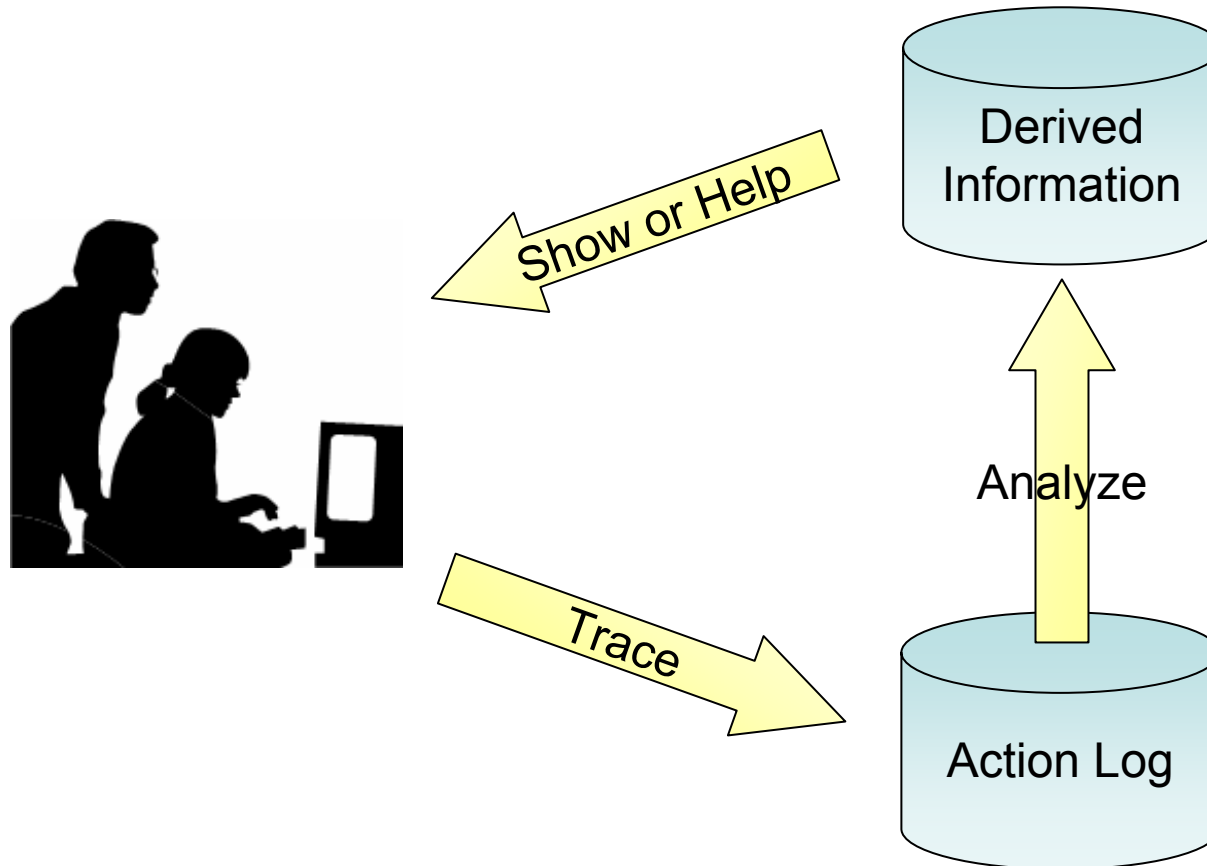
# Action Analysis

Martin.Muehlenbrock@dfki.de

DFKI

German Research Center for Artificial  
Intelligence

# Action Analysis Process





# Goals of Action Analysis

- Make use of available logs of learner actions
- Generate higher-level information
- Provide information or even support for students
- Provide information for teachers
- Provide tools for pedagogical research



# What Makes ITS Data Analyzable?

- Multiple grain sizes
  - duration and frequency of student sessions
  - individual reading of words or sections
- Data must be machine-readable
  - replacing freehand drawing with a limited palette of graphical objects and operations
  - replacing free-form responses with menu selections
- Timing
  - what students see, how long, and how often, e.g.
  - how long they take to read pages, sections, or sentences
  - response time to multiple choice questions to detect guessing
  - time spent on different activities



# Related Fields

- Student Modeling
  - Assessing students' knowledge (learner model) vs. analyzing students' behavior (learner profile)
- Web Server Log Analysis
  - Focus on user navigation, no pedagogical concern
- Data mining
  - Technology for analyzing large databases, no pedagogical concern

# Log Analysis as Data Mining



- Data collection
  - Server-side data collection
  - Client-side data collection
  - Integration of additional data, such as ontological information on learning material and user registration information

# Log Analysis as Data Mining



- Data preparation
  - Fixed transformations for example timestamp conversions, learning material lookup, and extraction of URL parameters
  - User identification using heuristics if data collection does not provide explicit user identification
  - Session identification for example beginning and end of sessions, pauses, etc
  - Flexible transformations for example cumulating information from single clicks to summarize session information
  - Data cleaning for example removing demo sessions, detect user name changes, etc.

# Log Analysis as Data Mining



- Reporting facilities
  - **Access statistics** such as hits, page impressions, peak visit times, duration of sessions, average amount of pages seen
  - **User statistics** such as first time users, returning users, number of sessions per user, average time between user sessions
  - **Session statistics** ranging from number and duration of sessions up to information of referrers, entry points and exit points
  - **More statistics** provide information on effectiveness of hints, click through rates, or failure reports among others

# Log Analysis as Data Mining



- Data mining
  - Association analysis such as analyzing typical navigation paths
  - Sequence analysis e.g. for controlling these typical navigation paths for specific users or user groups over time
  - Cluster analysis e.g. for grouping users according to their behavior and their characteristics
  - Classification analysis for instance in order to try to describe these clusters with classification rules such as decision trees

# Log Analysis as Data Mining



- Result deployment
  - Profile generation such as profiles of specific users
  - Teacher reports such as generation of some high-level report
  - Personalization such as providing personalized links or contents to specific users or user groups

# Example: ActiveMath

active $\text{math}$  Menu | Dictionary | Learner Model | Contact | Help | Eyetracker | Suggestions | Logout

Complete Content of LeAM\_calculus

- Basics
  - On mathematical proofs
  - Fundamentals on straight lines and their slope
  - The binomial formulas
  - Trigonometric formulas
  - Bounded sets
- Sequences, series, and limits
  - Fundamentals on sequences
  - Limits of sequences
  - Bounded sequences
  - Computations with limits
  - Subsequences and rearrangements
  - Cauchy sequences
  - Series
  - Absolutely convergent series
  - Convergency tests for series
  - Power series
- Relations and functions
- Introduction of derivatives
- Derivation rules
- Exercise collections

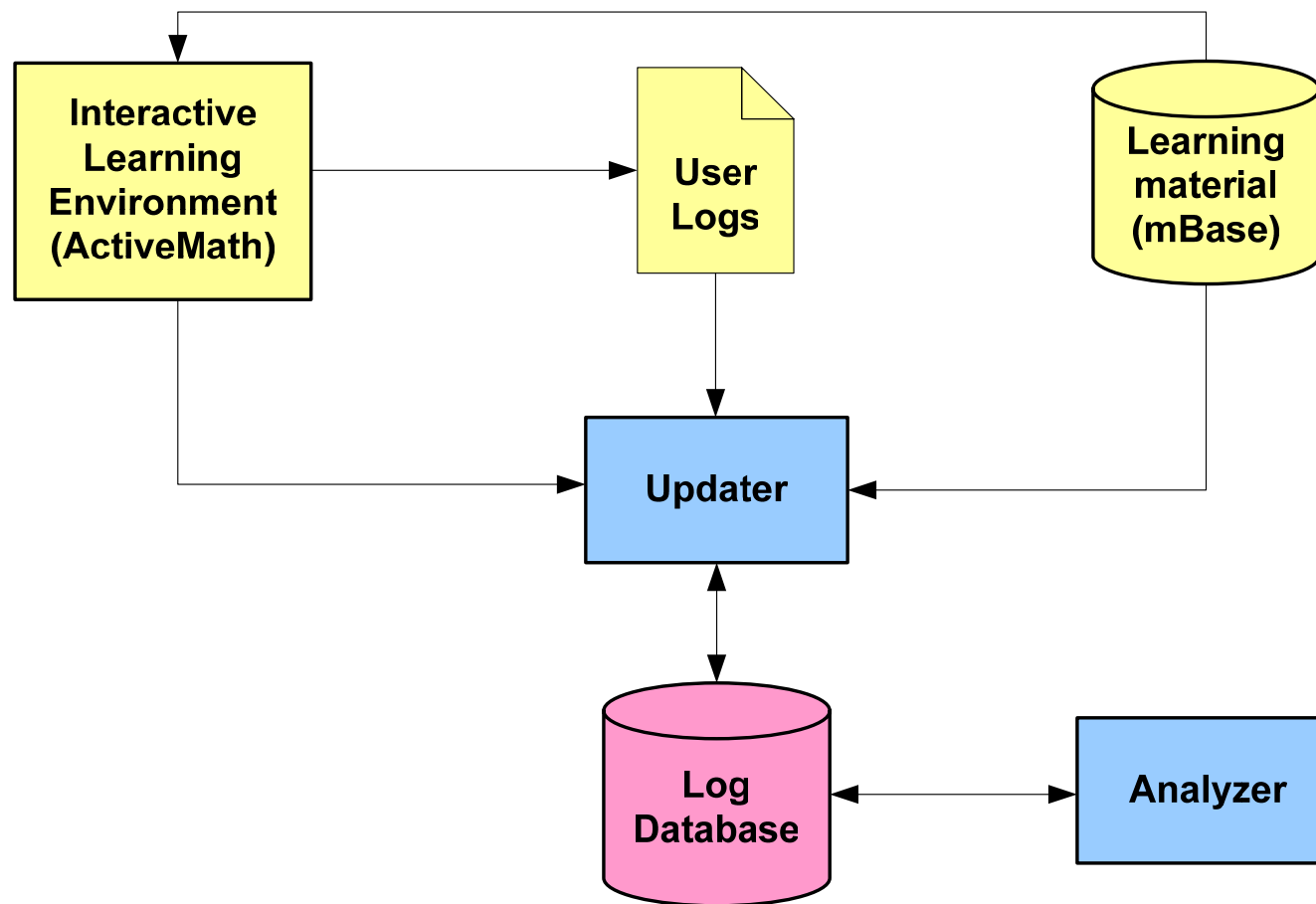
### Ordering of sequences and limits

Let  $x_n$  and  $y_n$  denote convergent real sequences with  $x = \lim_{n \rightarrow \infty} x_n$  and  $y = \lim_{n \rightarrow \infty} y_n$ , as well as  $x_n \leq y_n \in \mathbb{R}$  for all  $n$ . This implies  $x \leq y$ .

The top graph shows two sequences,  $x_n$  (red dots) and  $y_n$  (blue dots), plotted against  $n$  (1 to 30). The  $x_n$  sequence converges to a limit  $x$ , and the  $y_n$  sequence converges to a limit  $y$ . The  $x_n$  sequence is always below the  $y_n$  sequence. The  $x$  and  $y$  limits are shown on the y-axis, with  $x < y$ . The  $x_n$  sequence is always below the  $y_n$  sequence. The  $x$  and  $y$  limits are shown on the y-axis, with  $x < y$ . The  $x_n$  sequence is always below the  $y_n$  sequence. The  $x$  and  $y$  limits are shown on the y-axis, with  $x < y$ .

The bottom graph shows the same sequences, but with  $x_n$  and  $y_n$  swapped. The  $x_n$  sequence (red dots) is now above the  $y_n$  sequence (blue dots). The  $x$  and  $y$  limits are shown on the y-axis, with  $x < y$ . The  $x_n$  sequence is always above the  $y_n$  sequence. The  $x$  and  $y$  limits are shown on the y-axis, with  $x < y$ .

# System Architecture



# ActiveMath Log (1/2)

```
<ActivemathEvent type="UserLoggedIn" ts="1115212152486" source="org.activemath.webapp.controller.Login">  
  <User id="Caroline18"/>  
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>  
  <UserLoggedIn remoteAddr="134.96.236.41" userAgent="Mozilla/4.0 (compatible; MSIE 6.0; Windows NT  
  5.1)"/>  
</ActivemathEvent>
```

```
<ActivemathEvent type="UserPropertyChanged" ts="1115214804505" source="org.activemath.webapp.user.User">  
  <User id="Caroline18"/>  
  <UserPropertyChanged property="language" oldValue="en" newValue="de"/>  
</ActivemathEvent>
```

```
<ActivemathEvent type="PagePresented" ts="1115214852847"  
  source="org.activemath.webapp.controller.ViewBook">  
  <User id="Caroline18"/>  
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>  
  <Book id="LeAM_calculusCompleteRec_auto"/>  
  <PagePresented page="1"/>  
</ActivemathEvent>
```

```
<ActivemathEvent type="ItemPresented" ts="1115214852848"  
  source="org.activemath.webapp.controller.ViewBook">  
  <User id="Caroline18"/>  
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>  
  <Item type="symbol" id="mbase://LeAM_calculus/basics/Q_plus"/>  
  <ItemPresented/>  
</ActivemathEvent>
```

# ActiveMath Log (2/2)

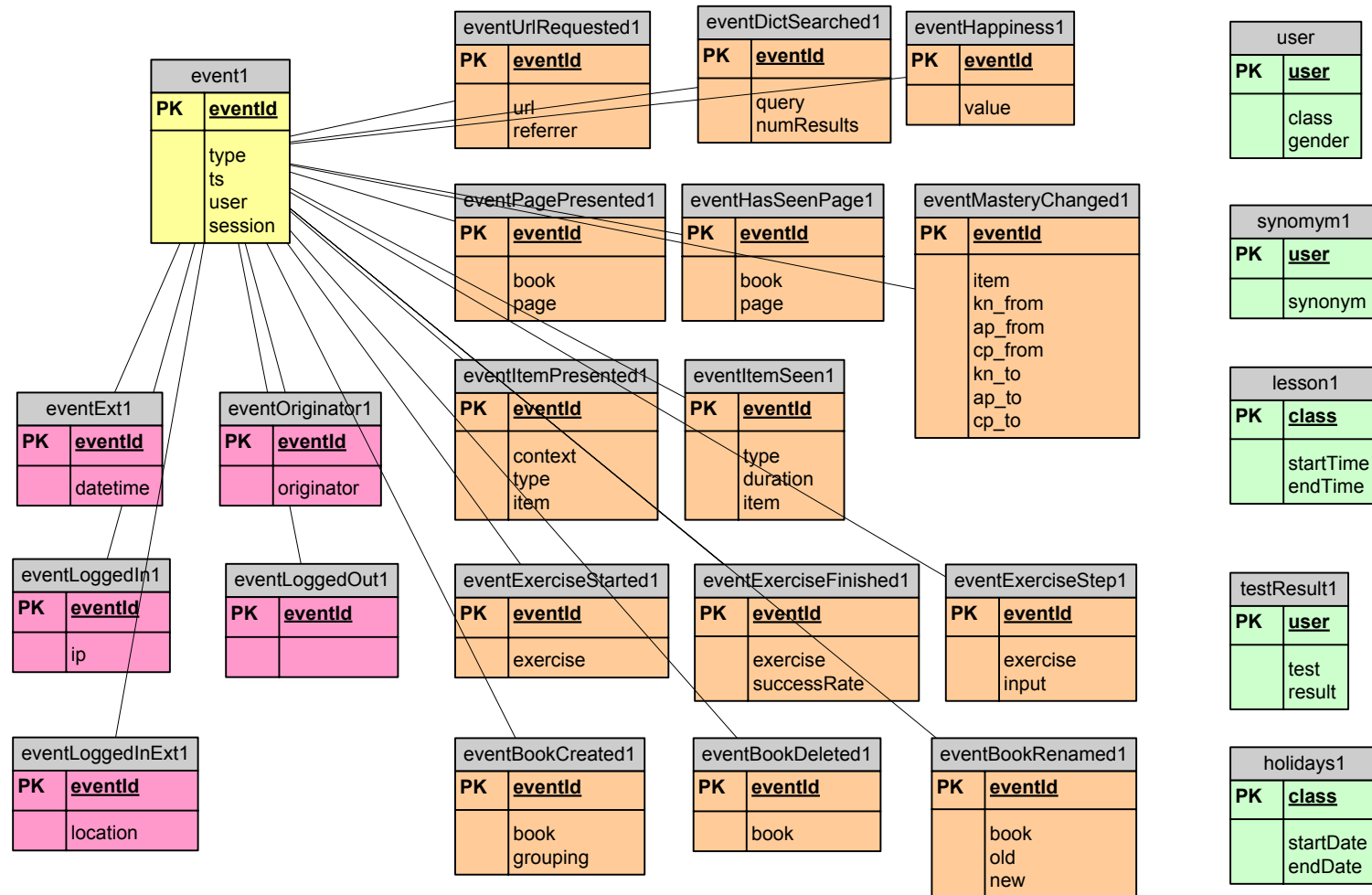
```
<ActivemathEvent type="ExerciseStarted" ts="1115215182332"
  source="org.activemath.webapp.exercises.ExerciseController">
  <User id="Caroline18"/>
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>
  <Item type="exercise" id="mbase://LeAM_calculus/diffquot/exer_slope_all"/>
  <ExerciseStarted/>
</ActivemathEvent>

<ActivemathEvent type="ExerciseStep" ts="1115215224918"
  source="org.activemath.webapp.exercises.ExerciseController">
  <User id="Caroline18"/>
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>
  <Item type="exercise" id="mbase://LeAM_calculus/diffquot/exer_slope_all"/>
  <ExerciseStep input="1/40;/>
</ActivemathEvent>

<ActivemathEvent type="ExerciseFinished" ts="1115215225494"
  source="org.activemath.webapp.exercises.ExerciseController">
  <User id="Caroline18"/>
  <Session id="BDC43AC6CEAE7C1553C6F9EC03E62157"/>
  <Item type="exercise" id="mbase://LeAM_calculus/diffquot/exer_slope_all"/>
  <ExerciseFinished success="1.0"/>
</ActivemathEvent>

<ActivemathEvent type="MasteryChanged" ts="1115215225775" source="">
  <User id="Caroline18"/>
  <Item type="definition.simple" id="mbase://LeAM_calculus/diffquot/def_average_slope"/>
  <MasteryChanged from="{KN=0.02, AP=0.02, CP=0.02}" to="{AP=0.0, KN=0.9, CP=0.0}"/>
</ActivemathEvent>
```

# Log Database Schema



# Log Database Schema

The screenshot displays the MySQL Administrator interface. The title bar reads "MySQL Administrator - muehlenb@amath-one.ags.uni-sb.de: 3306". The menu bar includes "File", "Edit", "View", "Tools", and "Help". On the left, there is a navigation pane with icons for "Server Information", "Service Control", "Startup Variables", "User Administration", "Server Connections", "Health", "Server Logs", "Replication Status", "Backup", "Restore", and "Catalogs". Below this is a "Schemata" section with a search box and a list of databases: "eduTechUserLogs", "hibernatetest", "information\_schema", "mysql", "project\_user\_logs", "schulversuch\_2005\_06\_01", "test", "universuch\_mh\_2005\_06\_01", and "universuch\_paul\_2005\_06\_01". The main area shows the "Schema Tables" view for the "eduTechUserLogs" schema. It contains a table with the following columns: "Table Name", "Engine", "Rows", "Data length", "Index length", and "Update time". The table lists 21 tables, all using the MyISAM engine. At the bottom of the main area, there are summary statistics: "Num. of Tables: 21", "Rows: 53,588", "Data Len: 4,3 MB", and "Index Len: 1 MB". Below these statistics are four buttons: "Details >>", "Create Table", "Edit Table", "Maintenance", and "Refresh".

Table Name	Engine	Rows	Data length	Index length	Update time
event	MyISAM	24019	2,7 MB	291 kB	2005-05-30 14:37:20
eventDictSearched	MyISAM	29	0,9 kB	3 kB	2005-05-30 14:37:14
eventExerciseFinished	MyISAM	49	3,3 kB	3 kB	2005-05-30 14:37:15
eventExerciseStarted	MyISAM	108	6,6 kB	7 kB	2005-05-30 14:37:15
eventExerciseStep	MyISAM	176	1,5 kB	9 kB	2005-05-30 14:37:15
eventFocusChanged	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
eventHappiness	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
eventItemPresented	MyISAM	22235	1,4 MB	539 kB	2005-05-30 14:37:20
eventItemSeen	MyISAM	17	1,1 kB	3 kB	2005-05-30 14:36:29
eventMasteryChanged	MyISAM	925	55,5 kB	25 kB	2005-05-30 14:37:15
eventPagePresented	MyISAM	309	11,9 kB	11 kB	2005-05-30 14:37:15
eventUserBookDeleted	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
eventUserBookPlanned	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
eventUserBookRenamed	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
eventUserCreated	MyISAM	20	464 B	3 kB	2005-05-30 14:37:15
eventUserLoggedIn	MyISAM	64	6,2 kB	3 kB	2005-05-30 14:37:17
eventUserLoggedOut	MyISAM	61	549 B	3 kB	2005-05-30 14:37:15
eventUserPropertyChanged	MyISAM	26	1,1 kB	3 kB	2005-05-30 14:37:14
goals	MyISAM	0	0 B	1 kB	2005-05-30 14:36:05
newAssessment	MyISAM	2775	65 kB	59 kB	2005-05-30 14:37:15
oldAssessment	MyISAM	2775	59,7 kB	59 kB	2005-05-30 14:37:15



# Sample Query (1/3)

```
select
  hour(datetime) as hour,
  count(*) as hourcount
from eventext
group by hour
```

How much user activity is  
at what time of the day?

- select: start of query
- hour: extracts hour information
- as: define new field name
- from: specify table from which to select the information
- group by: make a bin for each different value
- count(\*): count the number of entries in each bin

# Result Set

The screenshot displays a MySQL database interface. The main window shows a result set table with two columns: 'hour' and 'hourcount'. The data is as follows:

hour	hourcount
7	59
8	4959
9	16366
10	10664
11	10826
12	132925
13	50168
14	4742
15	6093
16	4801
17	1057
18	281
19	461
20	30
22	95

The interface also includes a 'Schemata' panel on the right, showing a tree view of database objects. The 'eventtext' schema is expanded, showing fields like 'eventid' (datetime) and 'eventhasseenpage'. The 'Syntax' panel at the bottom right lists categories like 'Data Manipulation', 'Data Definition', 'MySQL Utility', and 'Transactional and Locking'. The status bar at the bottom indicates 'ress 0000000C'.



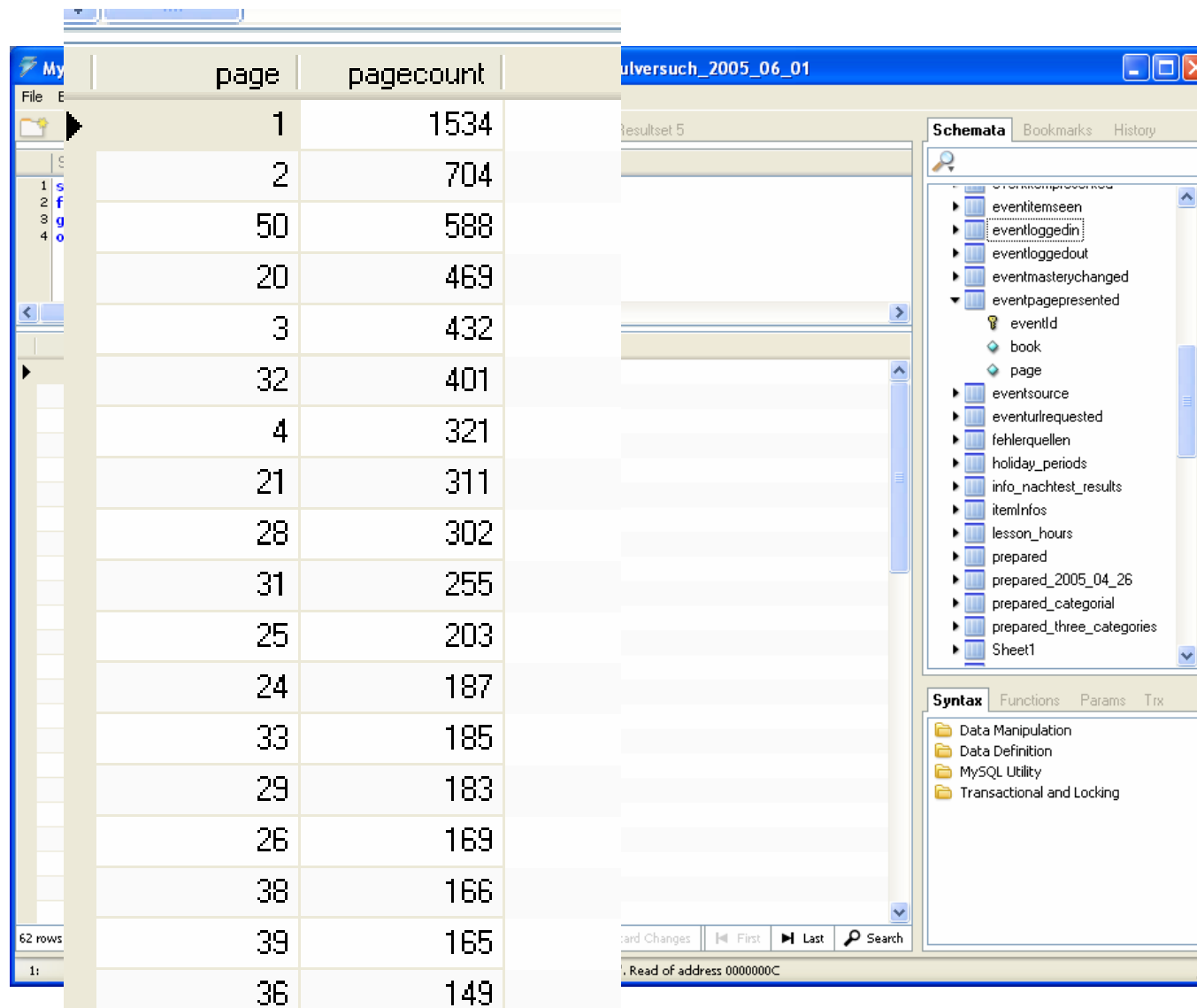
## Sample Query (2/3)

```
select page, count(*) as  
    pagecount  
from eventpagepresented  
group by page  
order by pagecount desc
```

- order by: list entries in ascending order
- desc: descending order instead of ascending order

What are the most popular pages?

# Result Set



The screenshot shows a MySQL database client window titled 'ulversuch\_2005\_06\_01'. The main area displays a result set with two columns: 'page' and 'pagecount'. The data is as follows:

page	pagecount
1	1534
2	704
50	588
20	469
3	432
32	401
4	321
21	311
28	302
31	255
25	203
24	187
33	185
29	183
26	169
38	166
39	165
36	149

The interface includes a 'Schemata' panel on the right showing a tree view of database schemas, with 'eventloggedin' selected. Below it is a 'Syntax' panel with categories like 'Data Manipulation' and 'Data Definition'. The status bar at the bottom indicates 'Read of address 0000000C'.

# Sample Query (3/3)

```
select e1.user,  
       from_unixtime(left(e1.ts,  
                          10)) as logintime,  
       (e2.ts - e1.ts)/60000 as  
       duration  
from event e1  
left join event e2  
  using(session)  
where e1.type = 'LoggedIn'  
and e2.type = 'LoggedOut'  
and e2.ts - e1.ts > 0  
order by duration
```

- left join: combine two tables by concatenating entries
- using: only concatenate entries where this field is identical
- where: define condition on selecting entries
- and: define further condition
- from\_unixtime(left): date conversion
- -, /, =, >: arithmetic or logical operations

# Result Set

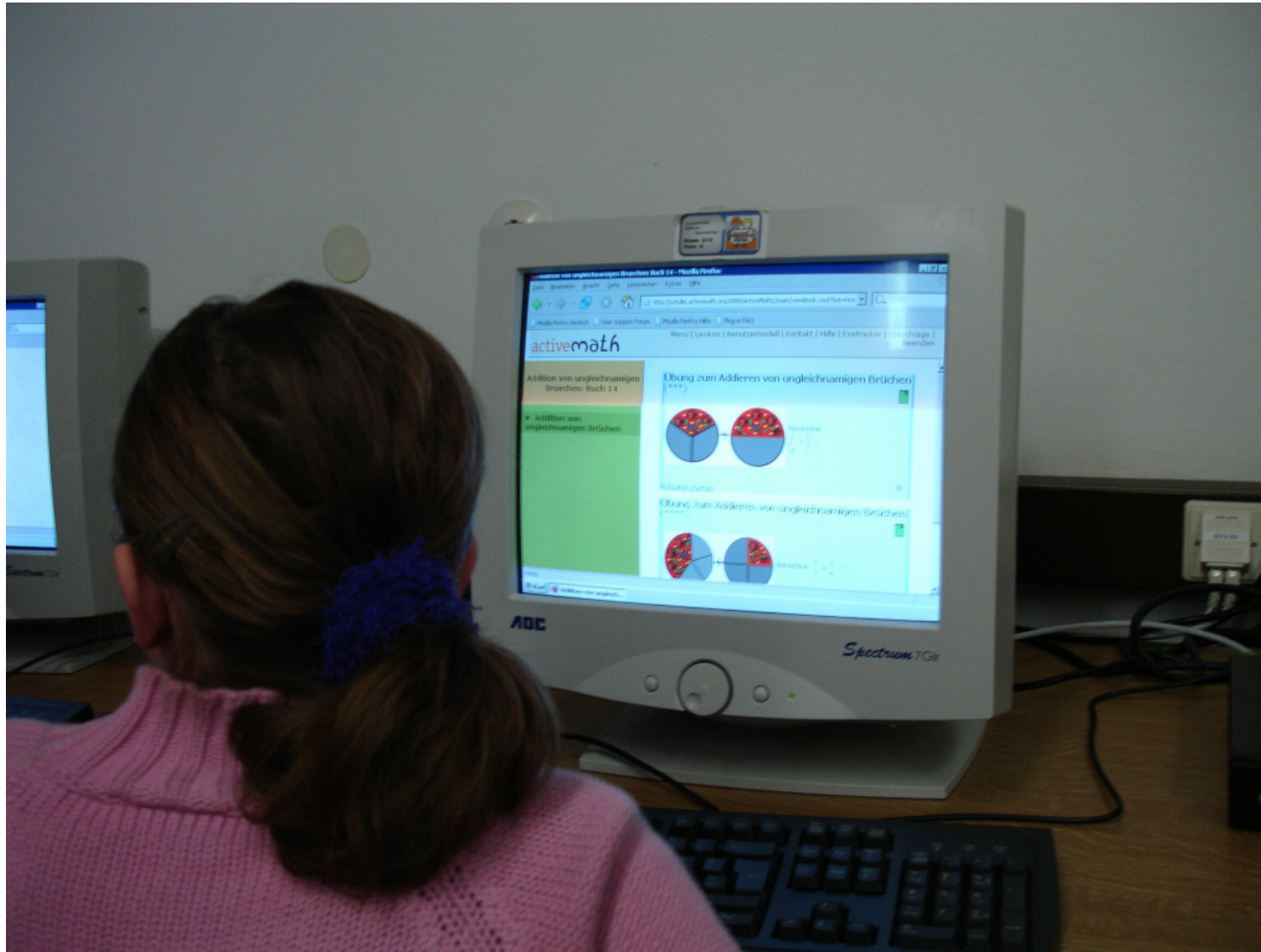
MySQL Query Browser - muehlenb@amath-one.ags.uni-sb.de:3306 / schulversuch\_2005\_06\_01

user	logintime	duration
ou-ganis	2005-03-04 12:17:33	27.63370
6b-Baerchen1	2004-11-12 12:19:21	27.63488
Ricki	2005-01-27 11:43:20	27.64673
asde	2004-10-07 11:04:28	27.65200
otto	2004-12-08 08:33:03	27.65482
edgar	2005-01-28 12:28:15	27.66843
6b-pluto1	2004-11-26 12:23:44	27.69603
6d-neptun	2005-03-11 12:59:17	27.74727
6a-magic-m-m	2005-01-10 12:19:40	27.78980
6a-blume	2005-02-21 12:17:02	27.87835
6b-Drossel	2005-01-14 12:21:12	27.94512
otto	2004-12-07 10:33:21	27.95210

3320 rows fetched in 0.1579s (5,7489s)

1: 1 Access violation at address 0056A867 in module 'MySQLQueryBrowser.exe'. Read of address 0000000C

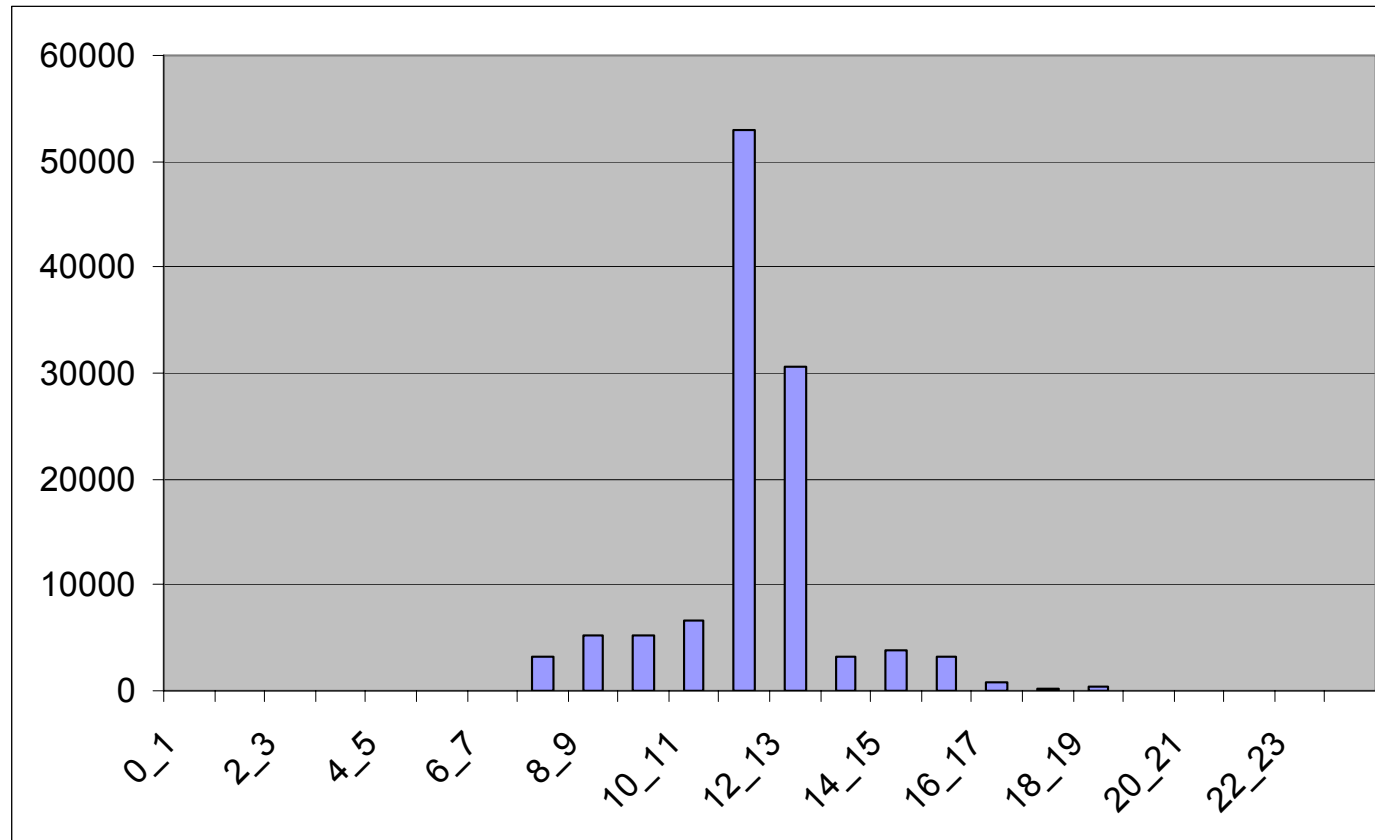
# School Experiment



# School Experiment



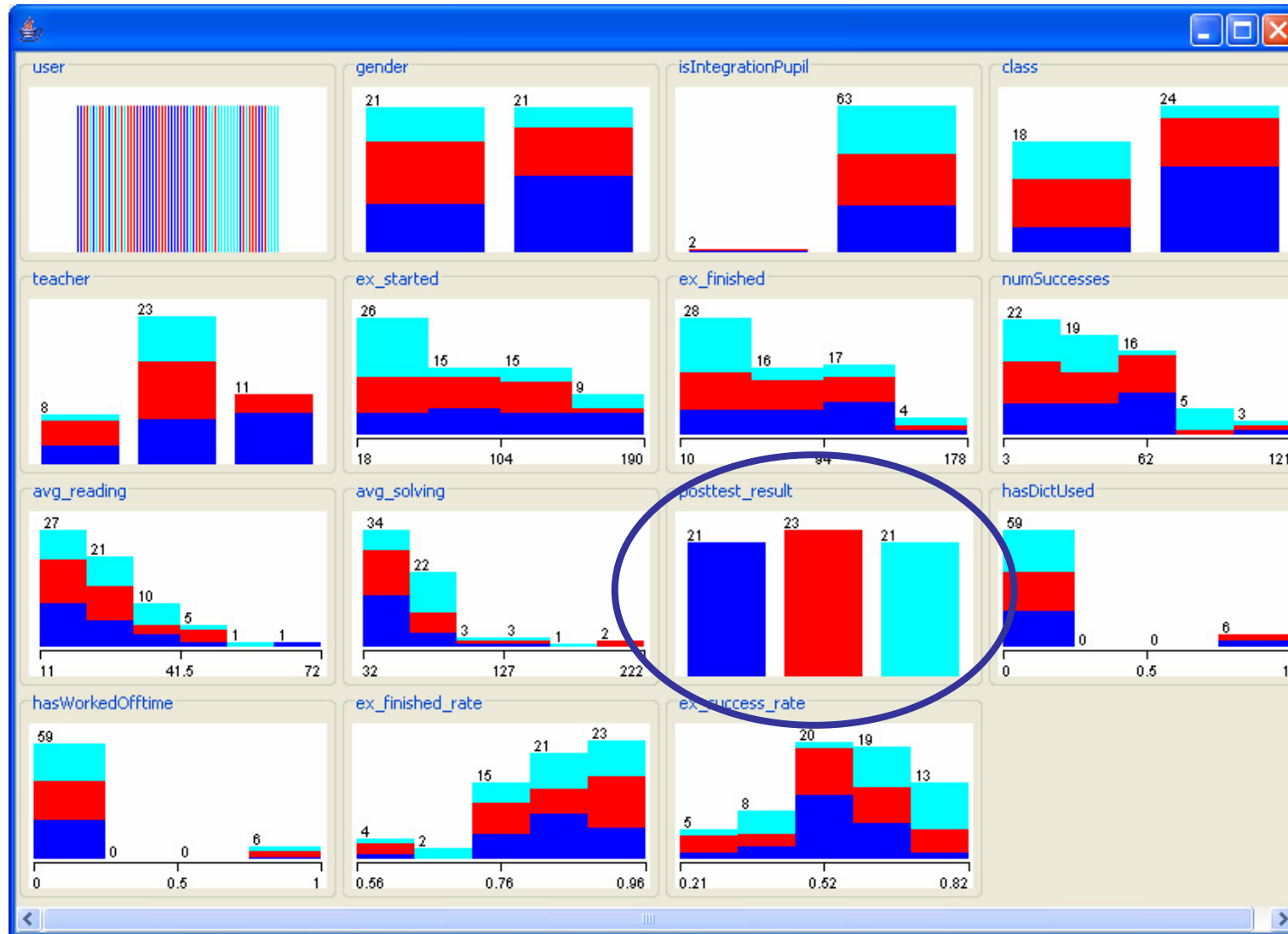
# Visualization of Query Results



# Attributes for Further Analysis

Attribute	Generation	Comment
User	Manual	User name (not used for the decision tree learning)
Class	Manual	Course, each comprised of about 20 students (not used for the decision tree learning)
Teacher	Manual	Each class has been split into two subgroups, with each being taught by another teacher (not used for the decision tree learning)
Gender	Manual	Male or female
Integration pupil	Manual	Whether the student is handicapped
Post test result	Manual	Results in the post test done in writing (binned into low, medium, and high for the decision tree learning)
Ex_started	Automatic	Number of exercises started
Ex_finished	Automatic	Number of exercises finished
Num_successes	Automatic	Number of successful exercises
Avg_reading	Automatic	Average number of reading actions in a session
Avg_solving	Automatic	Average number of exercise solving actions in a session
DictUsed	Automatic	Whether the student used the dictionary for searching information
WorkedOffTime	Automatic	Whether the student accessed the learning environment beyond lesson hours, e.g. from home or during free periods
Ex_finished_rate	Automatic	Rate of finished exercises to all started exercises
Ex_success_rate	Automatic	Rate of successful exercises to all finished exercises

# Visualization of Dependencies

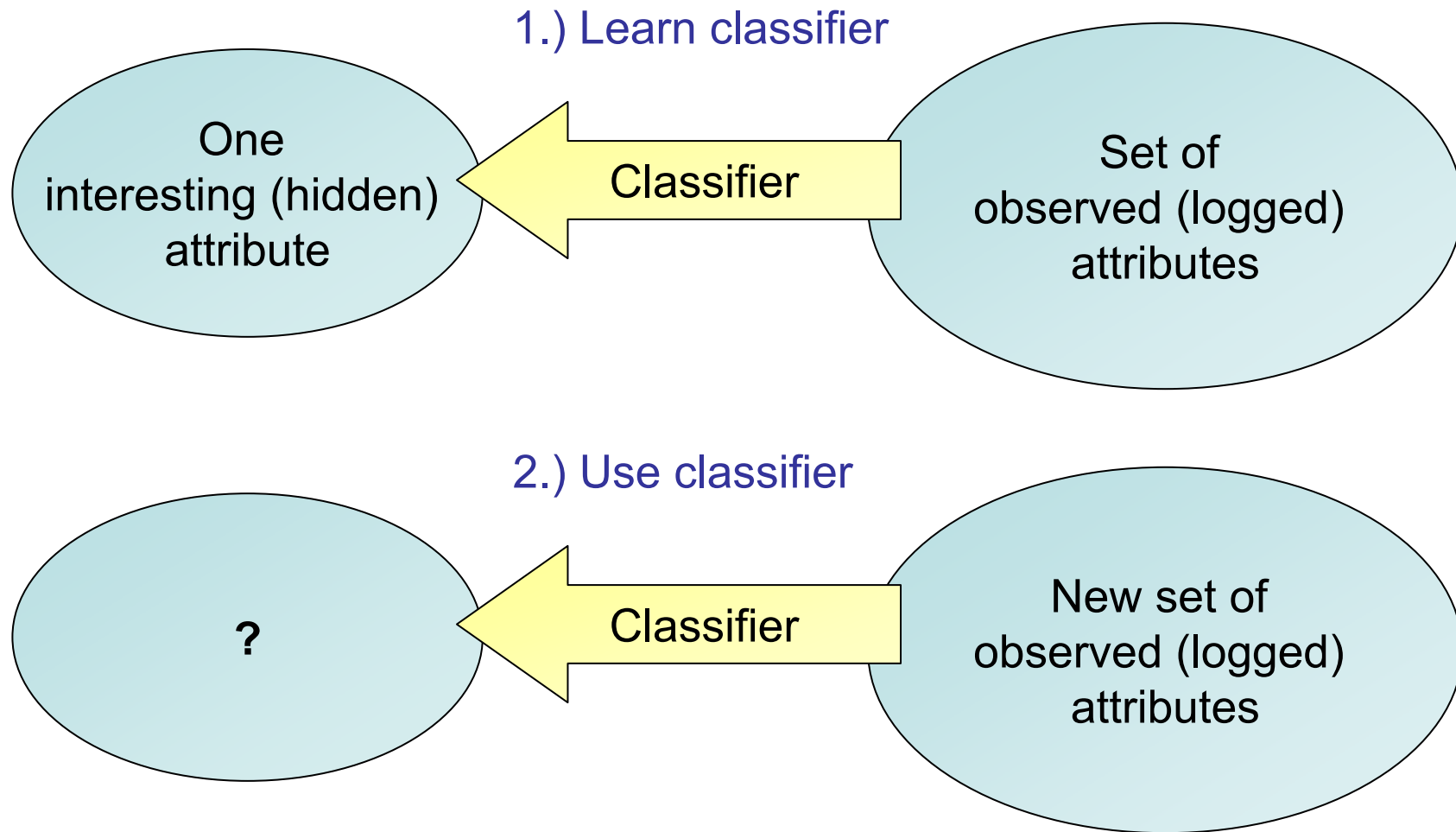




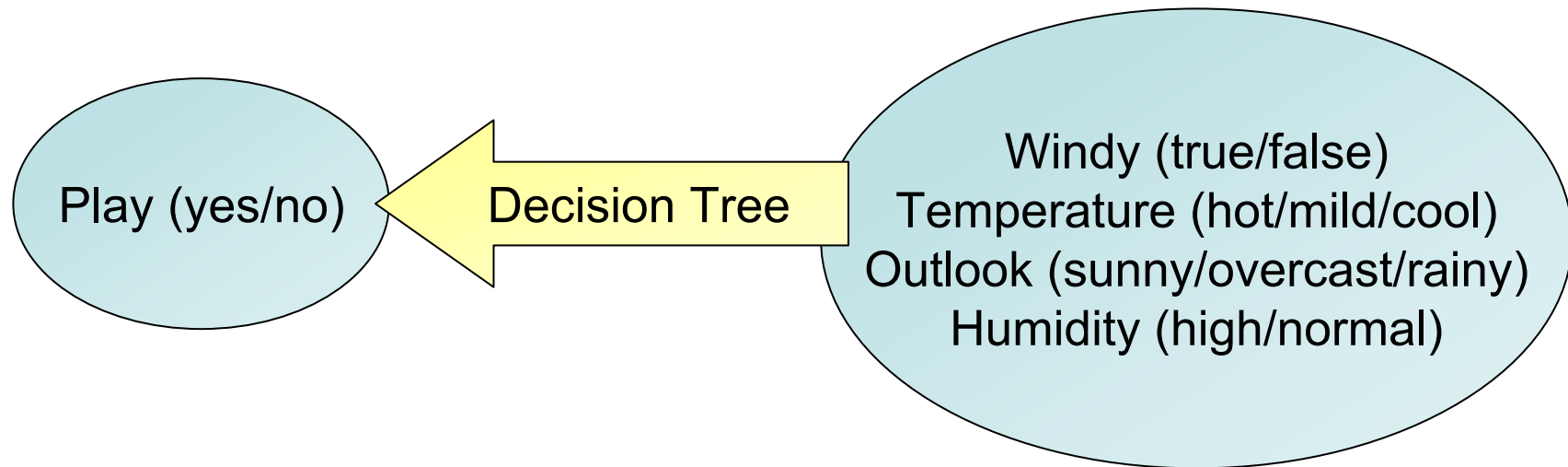
# Classification Analysis

- Goals
  - Represent interrelations and dependencies in the data (**characterize**)
  - Often: Provide explicit and intelligible description
  - Classify new data (**prediction**)
- Main ingredients
  - **Training set**: Data used for machine learning
  - **Classifier**: Indicator for class membership
  - **Test set**: Data used for evaluating the quality of the classifier
  - **(Positive/Negative) Example**: Element of the training or test set characterized by the (binary) classifier

# Classification Analysis



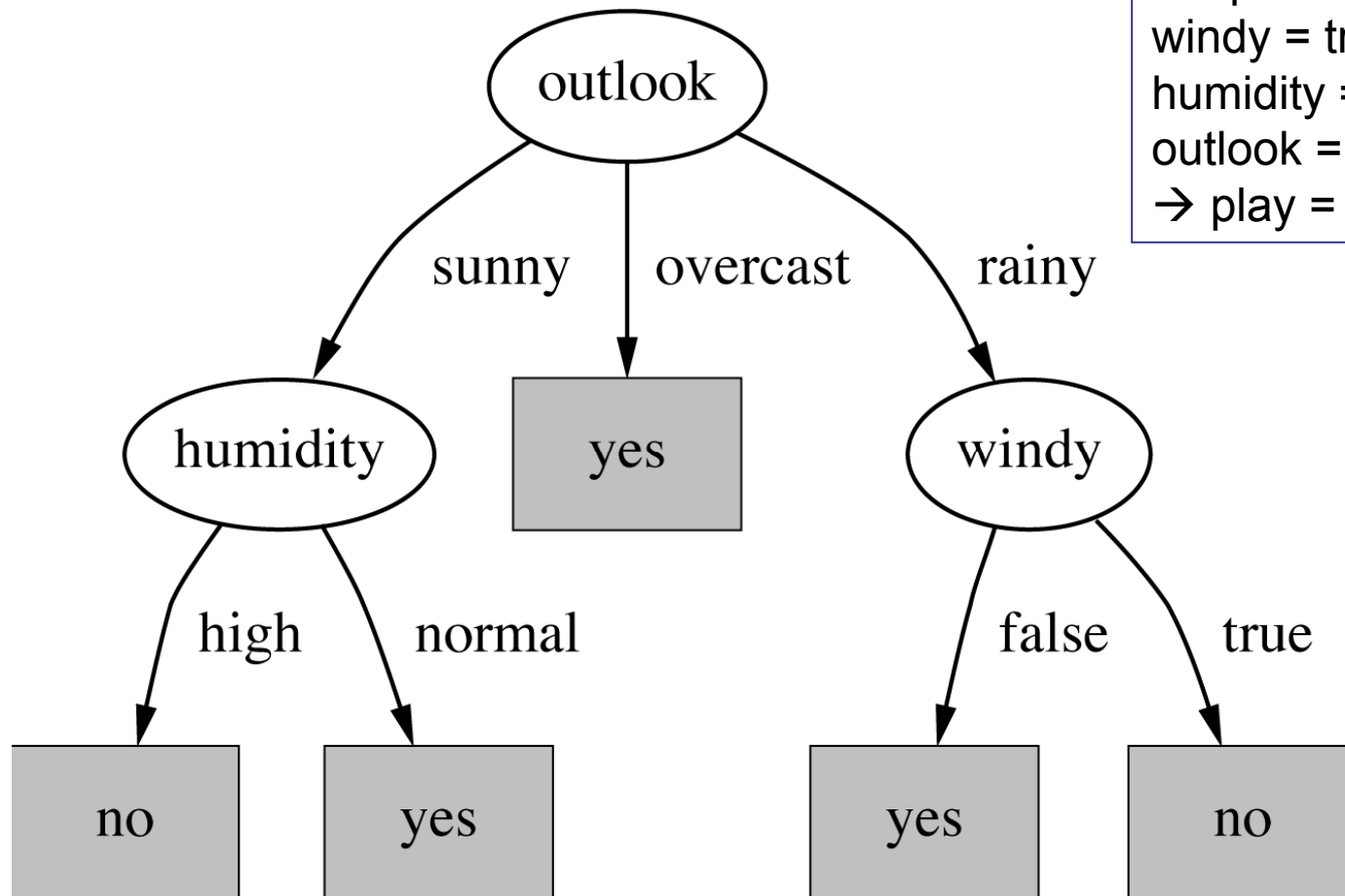
# Classifier: Decision Trees



Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	Normal	False	Yes
...	...	...	...	...

example taken from (Witten & Frank, 2000)

# Example: Decision Trees



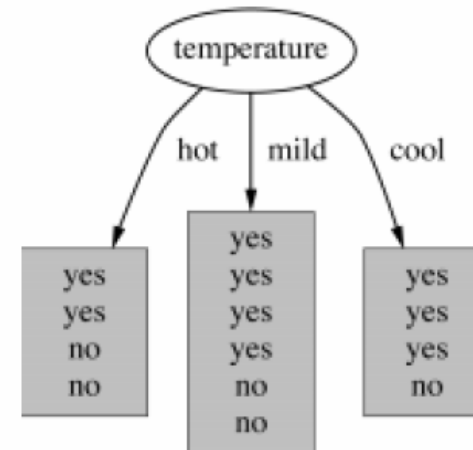
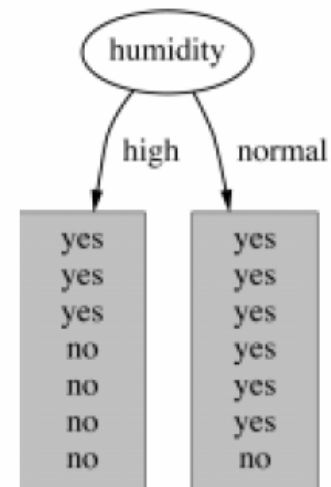
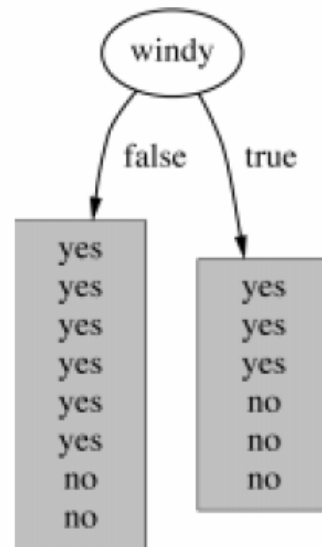
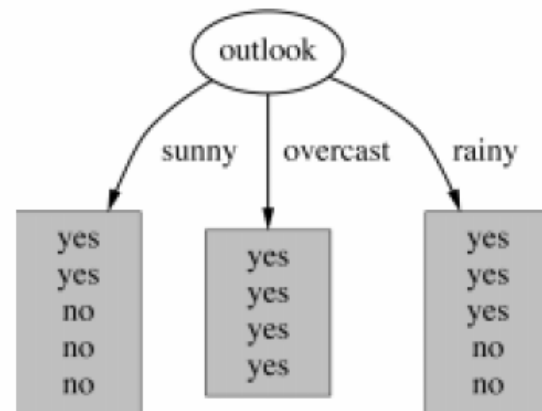
temperature = hot &  
windy = true &  
humidity = normal &  
outlook = sunny  
→ play = ?



# Constructing decision trees

- Normal procedure: top down in recursive *divide-and-conquer* fashion
  - ◆ First: attribute is selected for root node and branch is created for each possible attribute value
  - ◆ Then: the instances are split into subsets (one for each branch extending from the node)
  - ◆ Finally: procedure is repeated recursively for each branch, using only instances that reach the branch
- Process stops if all instances have the same class

# Which attribute to select?





# A criterion for attribute selection

- Which is the best attribute?
  - ◆ The one which will result in the smallest tree
  - ◆ Heuristic: choose the attribute that produces the “purest” nodes
- Popular *impurity criterion: information gain*
  - ◆ Information gain increases with the average purity of the subsets that an attribute produces
- Strategy: choose attribute that results in greatest information gain



# Computing information

- Information is measured in *bits*
  - ◆ Given a probability distribution, the info required to predict an event is the distribution's *entropy*
  - ◆ Entropy gives the information required in bits (this can involve fractions of bits!)
- Formula for computing the entropy:

$$\text{entropy}(p_1, p_2, \dots, p_n) = -p_1 \log p_1 - p_2 \log p_2 \dots - p_n \log p_n$$

# Example: attribute "Outlook"

- "Outlook" = "Sunny":

$$\text{info}([2,3]) = \text{entropy}(2/5,3/5) = -2/5 \log(2/5) - 3/5 \log(3/5) = 0.971 \text{ bits}$$

- "Outlook" = "Overcast":

$$\text{info}([4,0]) = \text{entropy}(1,0) = -1 \log(1) - 0 \log(0) = 0 \text{ bits}$$

*Note: this is normally not defined.*

- "Outlook" = "Rainy":

$$\text{info}([3,2]) = \text{entropy}(3/5,2/5) = -3/5 \log(3/5) - 2/5 \log(2/5) = 0.971 \text{ bits}$$

- Expected information for attribute:

$$\begin{aligned} \text{info}([3,2],[4,0],[3,2]) &= (5/14) \times 0.971 + (4/14) \times 0 + (5/14) \times 0.971 \\ &= 0.693 \text{ bits} \end{aligned}$$

# Computing the information gain

- Information gain: information before splitting – information after splitting

$$\begin{aligned} \text{gain("Outlook")} &= \text{info}([9,5]) - \text{info}([2,3],[4,0],[3,2]) = 0.940 - 0.693 \\ &= 0.247 \text{ bits} \end{aligned}$$

- Information gain for attributes from weather data:

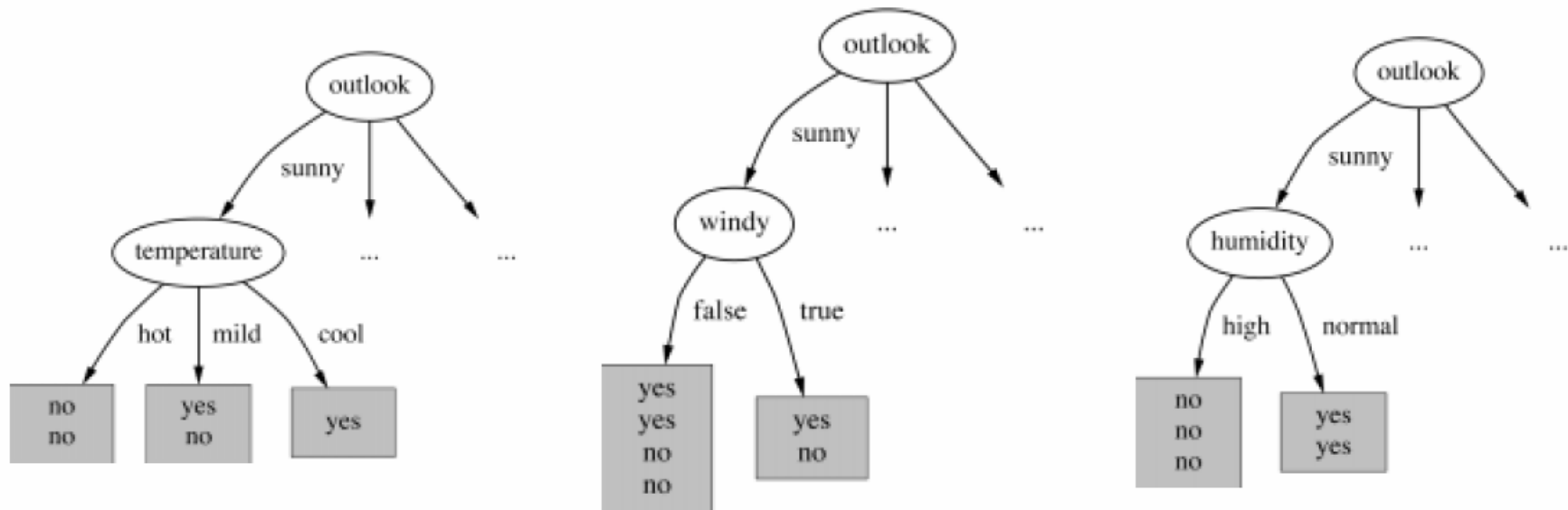
$$\text{gain("Outlook")} = 0.247 \text{ bits}$$

$$\text{gain("Temperature")} = 0.029 \text{ bits}$$

$$\text{gain("Humidity")} = 0.152 \text{ bits}$$

$$\text{gain("Windy")} = 0.048 \text{ bits}$$

# Continuing to split

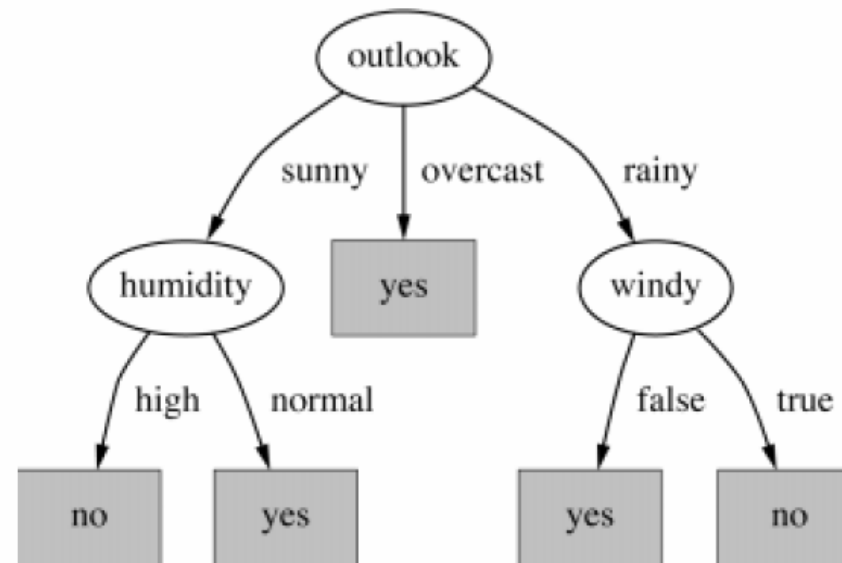


$\text{gain}(\text{"Temperature"}) = 0.571 \text{ bits}$

$\text{gain}(\text{"Humidity"}) = 0.971 \text{ bits}$

$\text{gain}(\text{"Windy"}) = 0.020 \text{ bits}$

# The final decision tree



- Note: not all leaves need to be pure; sometimes identical instances have different classes  
⇒ Splitting stops when data can't be split any further

# Attributes for Further Analysis

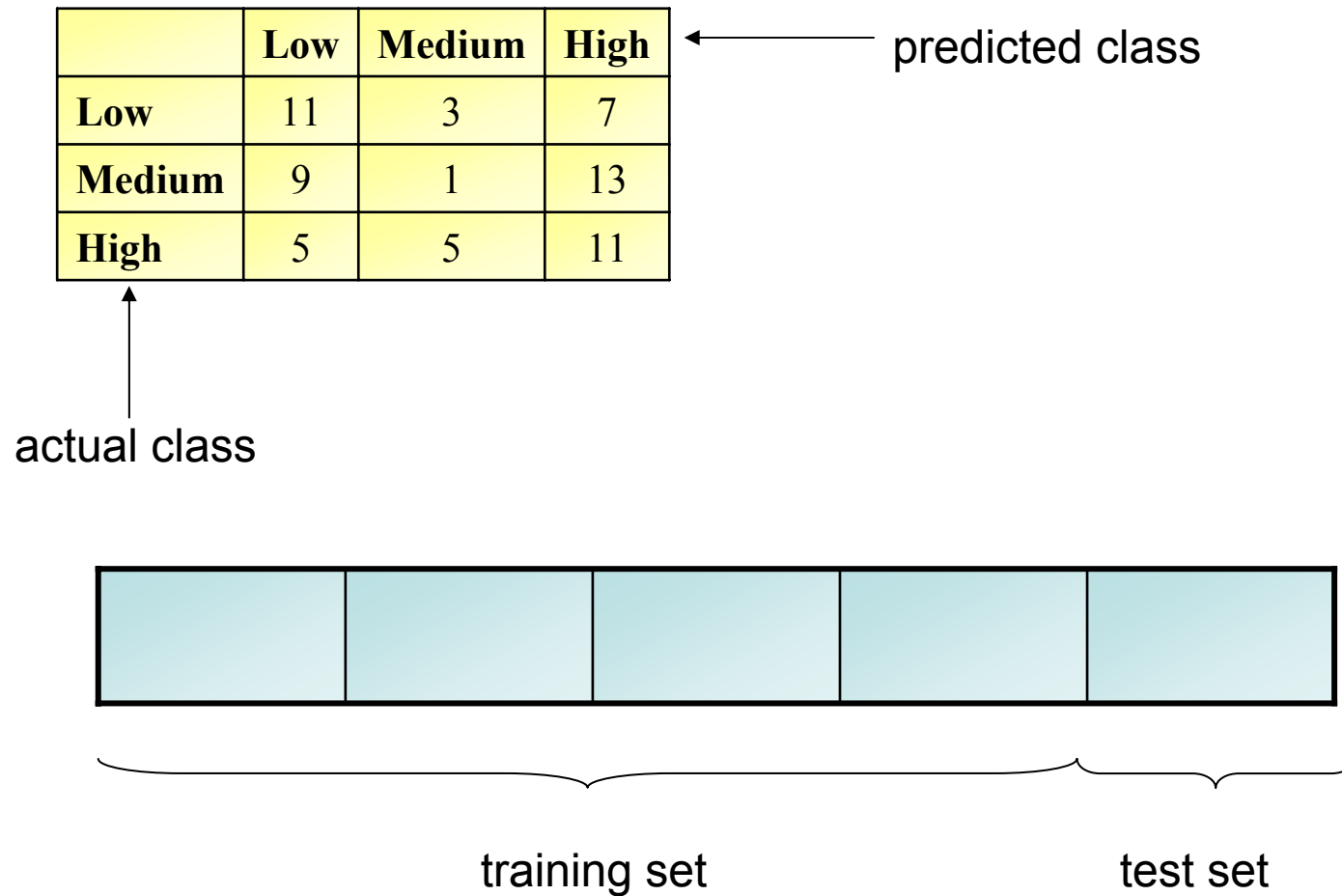
Attribute	Generation	Comment
User	Manual	User name (not used for the decision tree learning)
Class	Manual	Course, each comprised of about 20 students (not used for the decision tree learning)
Teacher	Manual	Each class has been split into two subgroups, with each being taught by another teacher (not used for the decision tree learning)
Gender	Manual	Male or female
Integration pupil	Manual	Whether the student is handicapped
Post test result	Manual	Results in the post test done in writing (binned into low, medium, and high for the decision tree learning)
Ex_started	Automatic	Number of exercises started
Ex_finished	Automatic	Number of exercises finished
Num_successes	Automatic	Number of successful exercises
Avg_reading	Automatic	Average number of reading actions in a session
Avg_solving	Automatic	Average number of exercise solving actions in a session
DictUsed	Automatic	Whether the student used the dictionary for searching information
WorkedOffTime	Automatic	Whether the student accessed the learning environment beyond lesson hours, e.g. from home or during free periods
Ex_finished_rate	Automatic	Rate of finished exercises to all started exercises
Ex_success_rate	Automatic	Rate of successful exercises to all finished exercises

# Decision Tree for Post Test Result

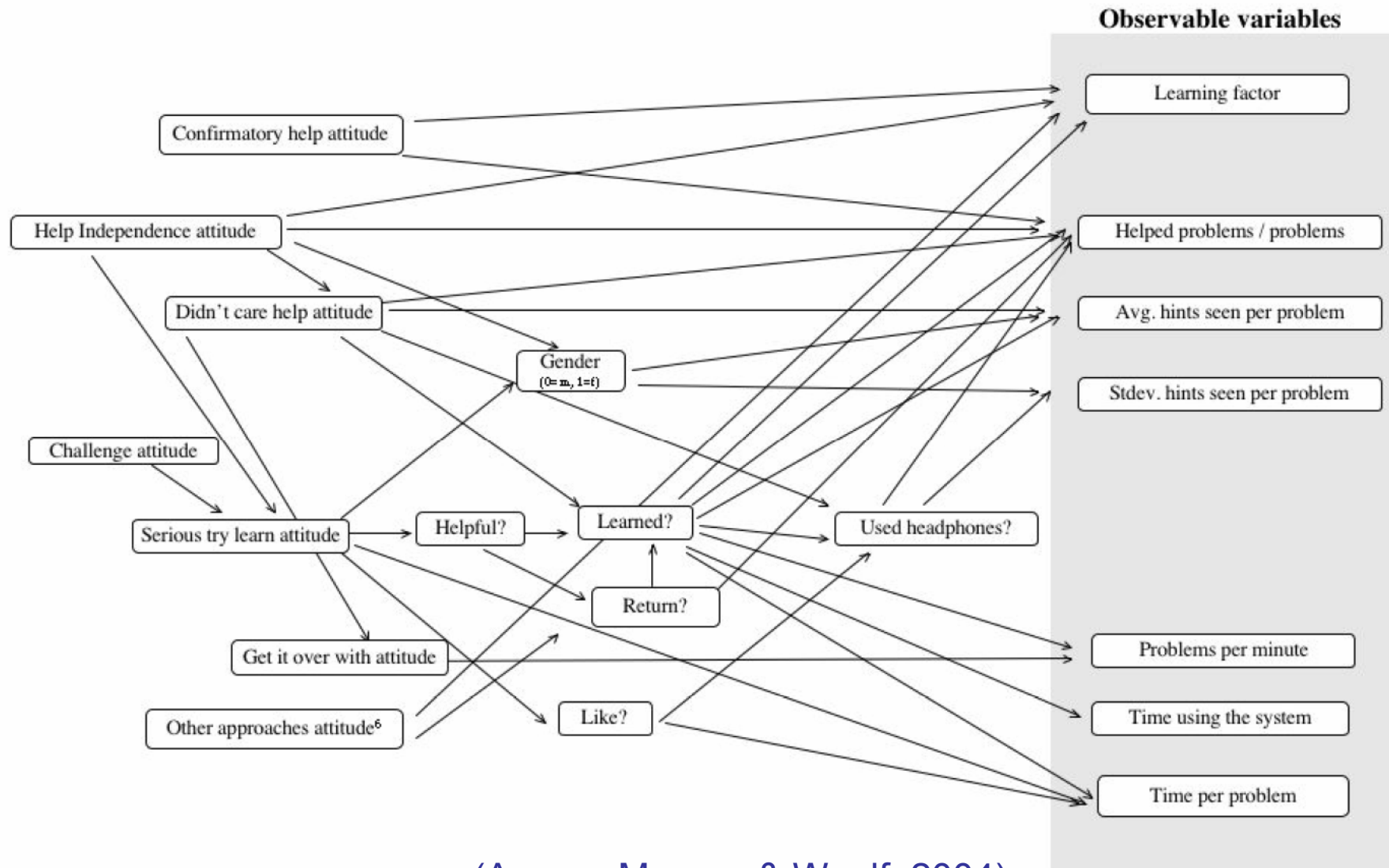


`ex_success_rate = 0,76` &  
`ex_finished_rate = 0,93` &  
`avg_reading = 56`  
→ post test result = ?

# Confusion Matrix and Cross Validation



# Bayesian Approach



(Arroyo, Murray, & Woolf, 2004)

# Conditional Probability Table

Serious Attitude	Liked	Learned	Time per Problem	Cases	Probability
T	F	F	Low	1	<i>0.33</i>
			High	2	<i>0.67</i>
		T	Low	3	<i>0.8</i>
			High	0	<i>0.2*</i>
	T	F	Low	1	<i>0.5</i>
			High	1	<i>0.5</i>
		T	Low	24	<i>0.45</i>
			High	29	<i>0.55</i>

# Sample Inference

Observed "leaf" nodes	Observed Value
Learning factor	High
Helped probs/total probs	Low
Hints per problem	High
Std. Hints per problem	High
Problems per minute	Low
Total time in system	High
Time per problem	High

=>

Hidden nodes	Inferred P(TRUE)
Learned?	0.9352
Didn't care help	0.0935
Headphones on?	0.5503
Helpful?	0.6186
Gender=female	0.6285
Serious?	0.7062
Like?	0.9071
Challenge Attitude	0.594
Help Independence	0.6495
Confirmatory Help	0.0824
Return?	0.7827
Other Approaches	0.2869
Get Over With	0.0396



# Further Work

- Project group on
  - motivation and ActiveMath usage
  - possibly learning style and ActiveMath usage