Towards understanding learning behavior

Joaquin Vanschoren

With Hendrik Blockeel

K.U.Leuven

September 25, 2006

Outline

Intro: Meta-learning

Limitations

An integrated solution

Conclusion

Outline

Intro: Meta-learning

Limitations

An integrated solution

Conclusion

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,...

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,...

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,...

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,. . .

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,...

Intro

Discovering structure in data:

- **Data preprocessing**: prepare data for learning (algorithm)
- **Algorithm selection**: find a learning model fitting the data

Machine Learning Bias

Learn efficiently: make *assumptions* about data structure *(bias)*

- Good learning performance ⇔ assumptions hold for data.

Types of bias:

- Representation: data model *(language bias)*
- Hypothesis evaluation: search heuristics *(procedural bias)*
- Data configuration: skewness, discretization,. . .

Meta-learning: definition

How to know if ML bias matches the given data?

Meta-Learning

Use experience of previous ML experiments to learn (automatically) how to improve automatic learning.

Goals:

- Gain insight into learning behavior to improve existing algorithms
- Select most promising learning techniques after analysis of new learning tasks

Meta-learning: definition

How to know if ML bias matches the given data?

Meta-Learning

Use experience of previous ML experiments to learn (automatically) how to improve automatic learning.

Goals:

- Gain insight into learning behavior to improve existing algorithms
- Select most promising learning techniques after analysis of new learning tasks

Meta-learning: definition

How to know if ML bias matches the given data?

Meta-Learning

Use experience of previous ML experiments to learn (automatically) how to improve automatic learning.

Goals:

- Gain insight into learning behavior to improve existing algorithms
- Select most promising learning techniques after analysis of new learning tasks

Meta-learning

Algorithm selection: start with looking at given data

- Prior knowledge available about dataset?
- Can we *compute* some data properties?

Approach

- Compute dataset characteristics (size, corr., entropy,. . .)
- Record performance of algorithms on dataset (experiments)
- Predict performance on new datasets (data mining)

Meta-learning

Algorithm selection: start with looking at given data

- Prior knowledge available about dataset?
- Can we *compute* some data properties?

Approach

- Compute dataset characteristics (size, corr., entropy,. . .)
- Record performance of algorithms on dataset (experiments)
- Predict performance on new datasets (data mining)

Meta-learning

Algorithm selection: start with looking at given data

- Prior knowledge available about dataset?
- Can we *compute* some data properties?

Approach

- Compute dataset characteristics (size, corr., entropy,. . .)
- Record performance of algorithms on dataset (experiments)
- Predict performance on new datasets (data mining)

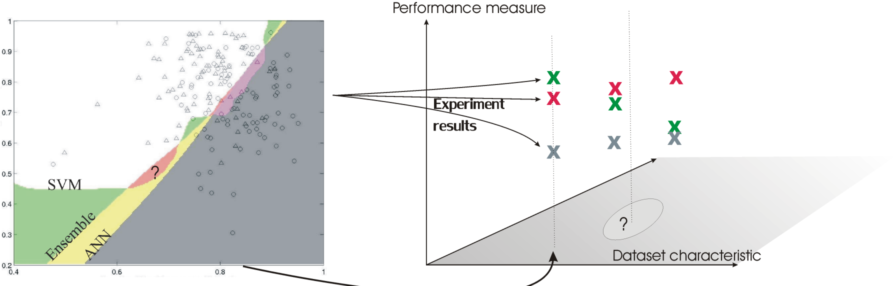
Meta-learning

Algorithm selection: start with looking at given data

- Prior knowledge available about dataset?
- Can we *compute* some data properties?

Approach

- Compute dataset characteristics (size, corr., entropy,. . .)
- Record performance of algorithms on dataset (experiments)
- Predict performance on new datasets (data mining)



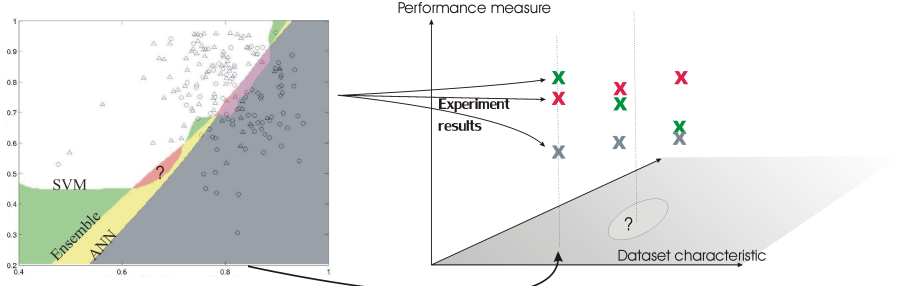
Meta-learning

Algorithm selection: start with looking at given data

- Prior knowledge available about dataset?
- Can we *compute* some data properties?

Approach

- Compute dataset characteristics (size, corr., entropy,. . .)
- Record performance of algorithms on dataset (experiments)
- Predict performance on new datasets (data mining)



Meta-knowledge base

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics — Algorithm — Performance measures

⇒ Predict performance on new datasets

- Characteristics of natural datasets
  - General: size, #attributes,. . .
  - Statistical: $\overline{corr(attrX, attrY)}$, skewness, kurtosis,. . .
  - Info-theoretic: $H(class)$, $H(attr)$, $MI(class, attr)$, N/S,. . .
  - Landmarkers, model-based characterisations
- Algorithm
  - Often default parameters, minimal preprocessing
- Performance measures
  - e.g. predictive accuracy and runtime

Meta-knowledge base

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

⇒ Predict performance on new datasets

- Characteristics of natural datasets
  - General: size, #attributes,. . .
  - Statistical: $corr(attrX, attrY)$, skewness, kurtosis,. . .
  - Info-theoretic: $H(class)$, $H(attr)$, $MI(class, attr)$, N/S,. . .
  - Landmarkers, model-based characterisations
- Algorithm
  - Often default parameters, minimal preprocessing
- Performance measures
  - e.g. predictive accuracy and runtime

Meta-knowledge base

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics — size, #attr., ···; Algorithm — algorithm; Performance measures — accuracy, runtime, ···

⇒ Predict performance on new datasets

- Characteristics of natural datasets
  - General: size, #attributes,. . .
  - Statistical: $corr(attrX, attrY)$, skewness, kurtosis,. . .
  - Info-theoretic: $H(class)$, $H(attr)$, $MI(class, attr)$, N/S,. . .
  - Landmarkers, model-based characterisations
- Algorithm
  - Often default parameters, minimal preprocessing
- Performance measures
  - e.g. predictive accuracy and runtime

Meta-knowledge base

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics (size, #attr., ···) · Algorithm (algorithm) · Performance measures (accuracy, runtime, ···)

⇒ Predict performance on new datasets

- Characteristics of natural datasets
  - General: size, #attributes,...
  - Statistical: $corr(attrX, attrY)$, skewness, kurtosis,...
  - Info-theoretic: $H(class)$, $H(attr)$, $MI(class, attr)$, N/S,...
  - Landmarkers, model-based characterisations
- Algorithm
  - Often default parameters, minimal preprocessing
- Performance measures
  - e.g. predictive accuracy and runtime

Outline

Intro: Meta-learning

Limitations

An integrated solution

Conclusion

The curse of dimensionality

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics | Algorithm | Performance measures

Curse of dimensionality:

- Many dataset characterizations: high-dimensional space
- Each instance = result of experiment: new dataset
- Limited number of natural datasets: very sparse evidence
- Low generalisability of results

Many more datasets necessary to make good predictions

The curse of dimensionality

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics   Algorithm   Performance measures

Curse of dimensionality:

- Many dataset characterizations: high-dimensional space
- Each instance = result of experiment: new dataset
- Limited number of natural datasets: very sparse evidence
- Low generalisability of results

Many more datasets necessary to make good predictions

The curse of dimensionality

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics Algorithm Performance measures

Curse of dimensionality:

- Many dataset characterizations: high-dimensional space
- Each instance = result of experiment: new dataset
- Limited number of natural datasets: very sparse evidence
- Low generalisability of results

Many more datasets necessary to make good predictions

The curse of dimensionality

size	#attr.	...	algorithm	accuracy	runtime	...
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

Curse of dimensionality:

- Many dataset characterizations: high-dimensional space
- Each instance = result of experiment: new dataset
- Limited number of natural datasets: very sparse evidence
- Low generalisability of results

Many more datasets necessary to make good predictions

The curse of dimensionality

size	#attr.	...	algorithm	accuracy	runtime	...
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

Curse of dimensionality:

- Many dataset characterizations: high-dimensional space
- Each instance = result of experiment: new dataset
- Limited number of natural datasets: very sparse evidence
- Low generalisability of results

Many more datasets necessary to make good predictions

Generalising over learning methods

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics | Algorithm | Performance measures

Results don't generalise over algorithms:

- What if we change parameter settings?
  - parameters change ML bias (e.g. under/overfitting)
  - Hoste & Daelemans, 2005: significant impact on relative performance
- No link to properties of algorithm (eg. *data fragmentation*)

Algorithm characterization needed to generalise results

Generalising over learning methods

size	#attr.	...	algorithm	accuracy	runtime	...
2300	43		C4.5	.92	43	

Dataset Characteristics — Algorithm — Performance measures

Results don't generalise over algorithms:

- What if we change parameter settings?
  - parameters change ML bias (e.g. under/overfitting)
  - 📄 Hoste & Daelemans, 2005: significant impact on relative performance
- No link to properties of algorithm (eg. *data fragmentation*)

Algorithm characterization needed to generalise results

Generalising over learning methods

size	#attr.	...	algorithm	accuracy	runtime	...
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

Results don't generalise over algorithms:

- What if we change parameter settings?
  - parameters change ML bias (e.g. under/overfitting)
  - Hoste & Daelemans, 2005: significant impact on relative performance
- No link to properties of algorithm (eg. *data fragmentation*)

Algorithm characterization needed to generalise results

Generalising over learning methods

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

Dataset Characteristics — Algorithm — Performance measures

Results don't generalise over algorithms:

- What if we change parameter settings?
    - parameters change ML bias (e.g. under/overfitting)
    - Hoste & Daelemans, 2005: significant impact on relative performance
- No link to properties of algorithm (eg. *data fragmentation*)

Algorithm characterization needed to generalise results

Explaining learning behavior

size	#attr.	⋯	algorithm	accuracy	runtime	⋯
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

⇒ Predict performance on new datasets

We can learn *when* an algorithm fails, but not *why*

- Representation mismatch/ overfitting?
- No explanation in terms of algorithm properties

More thorough investigation needed to diagnose failure/success

Explaining learning behavior

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

⇒ Predict performance on new datasets

We can learn *when* an algorithm fails, but not *why*

- Representation mismatch/ overfitting?
- No explanation in terms of algorithm properties

More thorough investigation needed to diagnose failure/success

Explaining learning behavior

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	

$\underbrace{\qquad\qquad\qquad}_{\text{Dataset Characteristics}}$ $\underbrace{\qquad}_{\text{Algorithm}}$ $\underbrace{\qquad\qquad\qquad}_{\text{Performance measures}}$

⇒ Predict performance on new datasets

We can learn *when* an algorithm fails, but not *why*

- Representation mismatch/ overfitting?
- No explanation in terms of algorithm properties

More thorough investigation needed to diagnose failure/success

Explaining learning behavior

size	#attr.	···	algorithm	accuracy	runtime	···
2300	43		C4.5	.92	43	
Dataset Characteristics			Algorithm	Performance measures		

⇒ Predict performance on new datasets

We can learn *when* an algorithm fails, but not *why*

- Representation mismatch/ overfitting?
- No explanation in terms of algorithm properties

More thorough investigation needed to diagnose failure/success

Data transformation

No link to preprocessing techniques

- Preprocessing has large impact on algorithm performance
- Hoste & Daelemans, 2005: significant impact on relative performance

Practical advice should include preprocessing steps

Data transformation

No link to preprocessing techniques

- Preprocessing has large impact on algorithm performance
- 📄 Hoste & Daelemans, 2005: significant impact on relative performance

Practical advice should include preprocessing steps

Data transformation

No link to preprocessing techniques

- Preprocessing has large impact on algorithm performance
- Hoste & Daelemans, 2005: significant impact on relative performance

Practical advice should include preprocessing steps

Outline

Intro: Meta-learning

Limitations

An integrated solution

Conclusion

Descriptive meta-learning

- Goal: Descriptive (vs. comparative) meta-learning
- Investigate specific questions
  - “What would be the effect of increasing parameter X on runtime?”
  - “Would an algorithm able to model fine-grained concepts perform better (or does it overfit)?”
- Explain reasons behind success/failure
  - Gain insights into why an algorithm behaves a certain way
  - For algorithm selection of future algorithm design

Descriptive meta-learning

- Goal: Descriptive (vs. comparative) meta-learning
- Investigate specific questions
  - “What would be the effect of increasing parameter X on runtime?”
  - “Would an algorithm able to model fine-grained concepts perform better (or does it overfit)?”
- Explain reasons behind success/failure
  - Gain insights into why an algorithm behaves a certain way
  - For algorithm selection of future algorithm design

Descriptive meta-learning

- Goal: Descriptive (vs. comparative) meta-learning
- Investigate specific questions
    - "What would be the effect of increasing parameter X on runtime?"
    - "Would an algorithm able to model fine-grained concepts perform better (or does it overfit)?"
- Explain reasons behind success/failure
    - Gain insights into why an algorithm behaves a certain way
    - For algorithm selection of future algorithm design

Experiment databases

C4.5 v.1

MLS	heur.	···	Dataset	TP	FP	···
2	gain		DS1	945	84	

Algorithm parameters

Performance measures

- Blockeel, 2005: improve interpretability of ML experiments
  - Also see Perlich, 2003: ML results ↔ dataset size
- Build database of large number of experiments, such that results are:
  - Generalisable: use large variety of (synthetic) datasets
  - Reusable: store all parameters and measurements (may prove useful later)
  - Reproducible: log all experiment settings (for further tests)
- Online, experimentation in background (cluster)

Experiment databases

C4.5 v.1

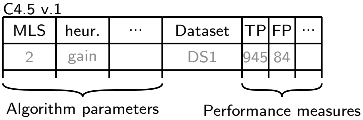
MLS	heur.	⋯	Dataset	TP	FP	⋯
2	gain		DS1	945	84	

Algorithm parameters

Performance measures

- Blockeel, 2005: improve interpretability of ML experiments
  - Also see Perlich, 2003: ML results ↔ dataset size
- Build database of large number of experiments, such that results are:
  - Generalisable: use large variety of (synthetic) datasets
  - Reusable: store all parameters and measurements (may prove useful later)
  - Reproducible: log all experiment settings (for further tests)
- Online, experimentation in background (cluster)

Experiment databases



C4.5 v.1

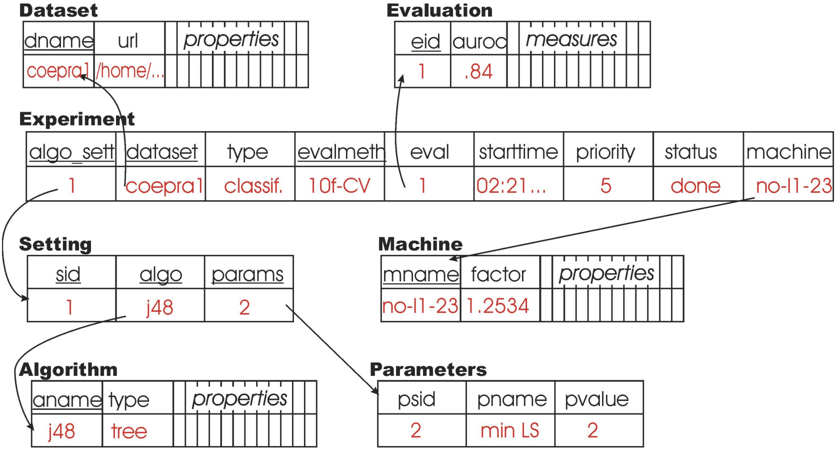
MLS	heur.	···	Dataset	TP	FP	···
2	gain		DS1	945	84	

Algorithm parameters

Performance measures

- Blockeel, 2005: improve interpretability of ML experiments
  - Also see Perlich, 2003: ML results ↔ dataset size
- Build database of large number of experiments, such that results are:
  - Generalisable: use large variety of (synthetic) datasets
  - Reusable: store all parameters and measurements (may prove useful later)
  - Reproducible: log all experiment settings (for further tests)
- Online, experimentation in background (cluster)

ExpDB design



Dataset

dname	url	properties
coepra1	/home/...	

Evaluation

eid	auroc	measures
1	.84	

Experiment

algo_sett	dataset	type	evalmeth	eval	starttime	priority	status	machine
1	coepra1	classif.	10f-CV	1	02:21...	5	done	no-l1-23

Setting

sid	algo	params
1	j48	2

Machine

mname	factor	properties
no-l1-23	1.2534	

Algorithm

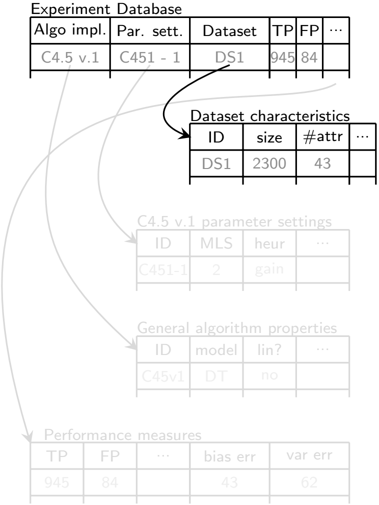
aname	type	properties
j48	tree	

Parameters

psid	pname	pvalue
2	min LS	2

Http://www.cs.kuleuven.be/~joaquin/expdb/expdb.php

Experiment databases



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

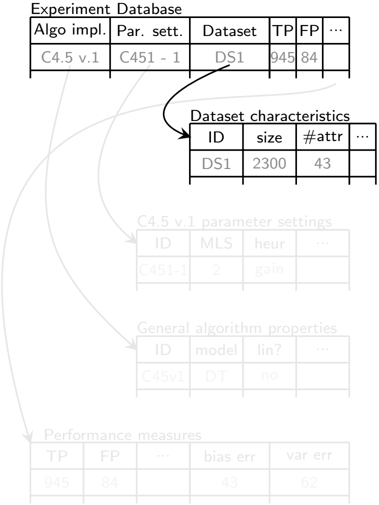
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Experiments not focused on one hypothesis, but to learn about algorithm
- Allows thorough investigation:
    - Test hypothesis by querying expDB
        - "What is the effect of parameter X on runtime for large datasets?"
    - Find patterns by data mining expDB
        - Rules, decision trees, association rules,. . .
        - Prediction of algorithm performance (e.g. kNN)

Experiment databases



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

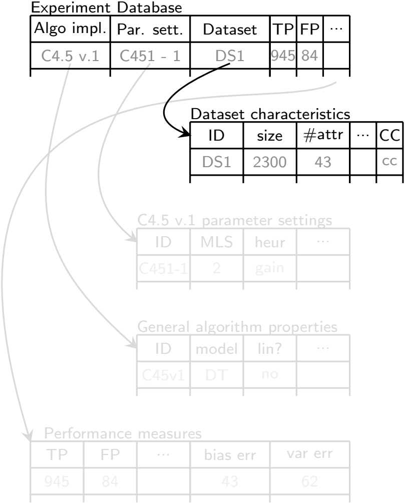
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Experiments not focused on one hypothesis, but to learn about algorithm
- Allows thorough investigation:
  - Test hypothesis by querying expDB
    - "What is the effect of parameter X on runtime for large datasets?"
  - Find patterns by data mining expDB
    - Rules, decision trees, association rules,. . .
    - Prediction of algorithm performance (e.g. kNN)

Synthetic datasets



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···	CC
DS1	2300	43		cc

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

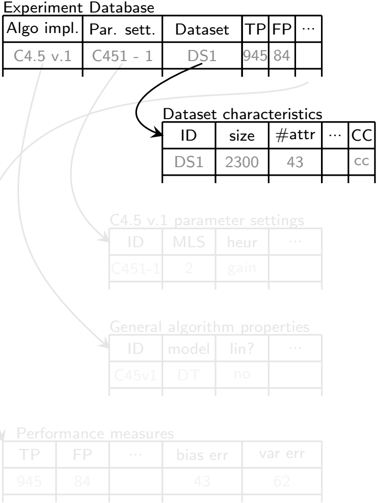
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Maintain validity of meta-learning experiments
- Unbiased: hide large range of different concepts + characterize concept
  - model characteristics
  - concept variation
  - example cohesion,. . .
- “Natural”: approximate characteristics of natural datasets
  - complex attribute relations
  - complex value distributions
  - noise, missing values,. . .
- Coverage: control characteristics to cover meta-feature space
  - experiment design

Synthetic datasets



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	⋯
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	⋯	CC
DS1	2300	43		cc

C4.5 v.1 parameter settings

ID	MLS	heur	⋯
C451-1	2	gain	

General algorithm properties

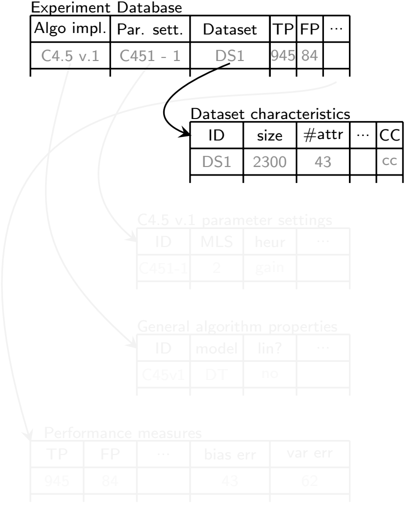
ID	model	lin?	⋯
C45v1	DT	no	

Performance measures

TP	FP	⋯	bias err	var err
945	84		43	62

- Maintain validity of meta-learning experiments
- Unbiased: hide large range of different concepts + characterize concept
  - model characteristics
  - concept variation
  - example cohesion,. . .
- “Natural”: approximate characteristics of natural datasets
  - complex attribute relations
  - complex value distributions
  - noise, missing values,. . .
- Coverage: control characteristics to cover meta-feature space
  - experiment design

Synthetic datasets



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···	CC
DS1	2300	43		cc

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

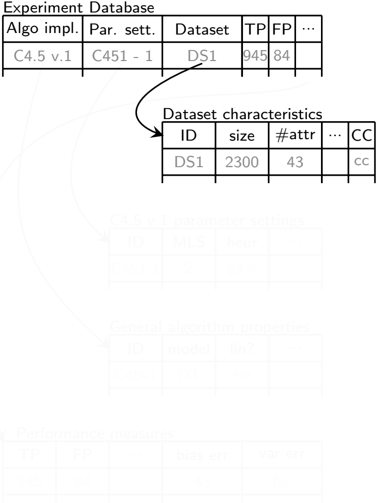
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Maintain validity of meta-learning experiments
- Unbiased: hide large range of different concepts + characterize concept
  - model characteristics
  - concept variation
  - example cohesion,. . .
- “Natural”: approximate characteristics of natural datasets
  - complex attribute relations
  - complex value distributions
  - noise, missing values,. . .
- Coverage: control characteristics to cover meta-feature space
  - experiment design

Synthetic datasets



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···	CC
DS1	2300	43		cc

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Maintain validity of meta-learning experiments
- Unbiased: hide large range of different concepts + characterize concept
  - model characteristics
  - concept variation
  - example cohesion,. . .
- “Natural”: approximate characteristics of natural datasets
  - complex attribute relations
  - complex value distributions
  - noise, missing values,. . .
- Coverage: control characteristics to cover meta-feature space
  - experiment design

Dataset generator

- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
    - could be combined
- Example generation: multi-tier approach
    - Low-level description
        - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
        - Can be nested
    - High-level description
        - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

Dataset generator

- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
  - could be combined
- Example generation: multi-tier approach
  - Low-level description
    - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
    - Can be nested
  - High-level description
    - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

Dataset generator

- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
  - could be combined
- Example generation: multi-tier approach
  - Low-level description
    - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
    - Can be nested
  - High-level description
    - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

Dataset generator

- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
    - could be combined
- Example generation: multi-tier approach
    - Low-level description
        - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
        - Can be nested
    - High-level description
        - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

Dataset generator

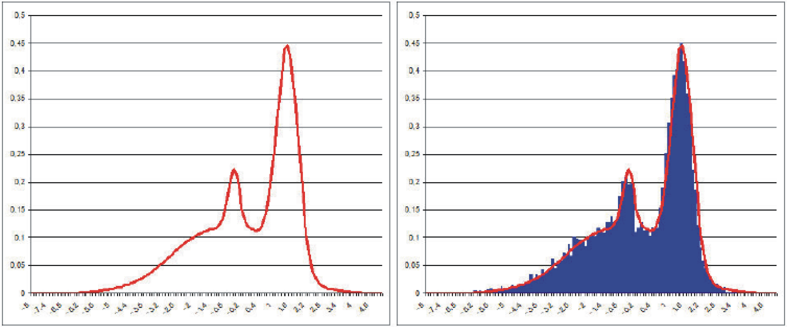
- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
  - could be combined
- Example generation: multi-tier approach
  - Low-level description
    - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
    - Can be nested
  - High-level description
    - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

Dataset generator

- Ongoing work
- Underlying concepts: several modules (DT, NN,. . .)
    - could be combined
- Example generation: multi-tier approach
    - Low-level description
        - initializes *attribute generators* for imposing dependencies, value distributions, noise,. . .
        - Can be nested
    - High-level description
        - based on dependency model (eg. Bayesian net) and high-level parameters
- Built on WEKA

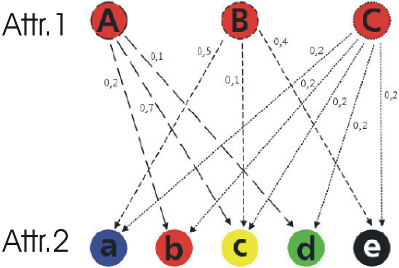
Attribute generator: value distributions

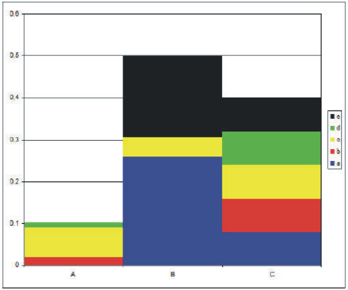
```
<attgen attname="att1" type="combi">
  <attgen probability=".15" type="normal" mean="0" stddev="1" />
  <attgen probability=".1" type="normal" mean="-2" stddev="1" />
  <attgen probability=".4" type="normal" mean="1.5" stddev=".4" />
  <attgen probability=".05" type="normal" mean="-.5" stddev=".2" />
  <attgen probability=".3" type="normal" mean="-1" stddev="2" />
</attgen>
```



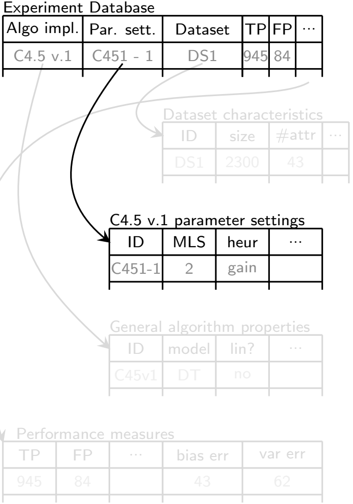
Attribute generator: dependencies

```
<attgen attname="att1" type="fixedprob"
                                     values="A;B;C" probabilities=".1;.5;.4"/>
<attgen type="transition" parent="att1" values="a;b;c;d;e" >
 <attgen type="fixedprob" values="b;c;d" probs=".2;.7;.1"/>
 <attgen type="fixedprob" values="a;c;e" probs=".5;.1;.4"/>
 <attgen type="fixedprob"/>
</attgen>
```





Algorithm characterization



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

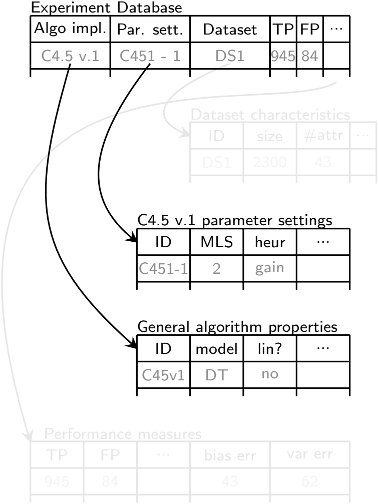
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Algorithm parameters settings
  - Stored as parameter name-value pairs
- General algorithm properties
  - representation model
  - dependency on linear separability, conditional independency,. . .
  - use of data fragmentation, attribute summation,. . .
  - ability to handle fine-grained concepts, local relevance,. . .

Algorithm characterization



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

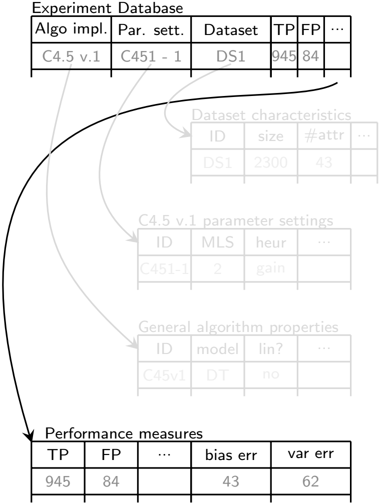
ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Algorithm parameters settings
  - Stored as parameter name-value pairs
- General algorithm properties
  - representation model
  - dependency on linear separability, conditional independency,. . .
  - use of data fragmentation, attribute summation,. . .
  - ability to handle fine-grained concepts, local relevance,. . .

Investigating inductive performance



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

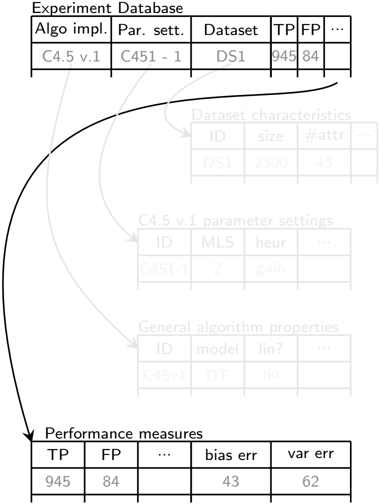
TP	FP	···	bias err	var err
945	84		43	62

- Misclassification error can be decomposed into :
  - bias error: systematic error: algorithm underfits target concept
  - variance error: variation on different samples (overfitting)

Rep.Bias	Comp.Bias	Bias err	Var. err
appr.	too strong	high	low
appr.	ok	low	low
appr.	too weak	low	high
inappr.	too strong	high	low
inappr.	ok	high	avg
inappr.	too weak	high	high

- Diagnose bad performance and link to dataset/algorithm characteristics:
  - bias ↗: bad representation model
  - variance ↗: bad parameter settings

Investigating inductive performance



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

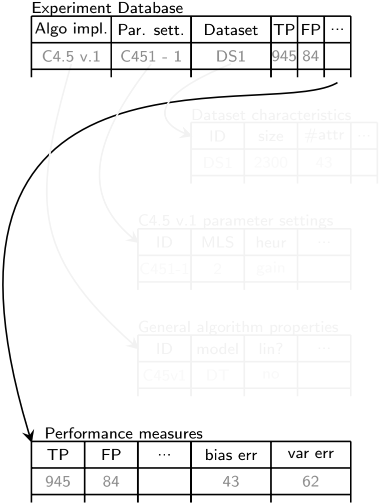
TP	FP	···	bias err	var err
945	84		43	62

- Misclassification error can be decomposed into :
  - bias error: systematic error: algorithm underfits target concept
  - variance error: variation on different samples (overfitting)

Rep.Bias	Comp.Bias	Bias err	Var. err
appr.	too strong	high	low
appr.	ok	low	low
appr.	too weak	low	high
inappr.	too strong	high	low
inappr.	ok	high	avg
inappr.	too weak	high	high

- Diagnose bad performance and link to dataset/algorithm characteristics:
  - bias ↗: bad representation model
  - variance ↗: bad parameter settings

Investigating inductive performance



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

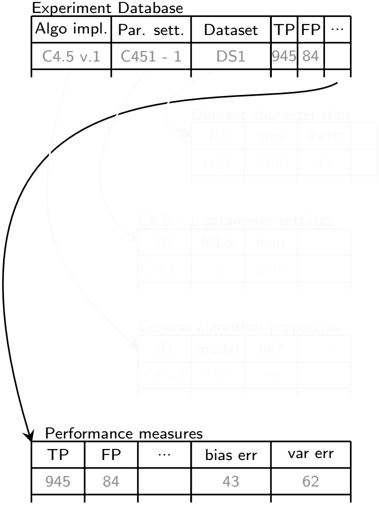
TP	FP	···	bias err	var err
945	84		43	62

- Misclassification error can be decomposed into :
  - bias error: systematic error: algorithm underfits target concept
  - variance error: variation on different samples (overfitting)

Rep.Bias	Comp.Bias	Bias err	Var. err
appr.	too strong	high	low
appr.	ok	low	low
appr.	too weak	low	high
inappr.	too strong	high	low
inappr.	ok	high	avg
inappr.	too weak	high	high

- Diagnose bad performance and link to dataset/algorithm characteristics:
  - bias↗: bad representation model
  - variance↗: bad parameter settings

Investigating inductive performance



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

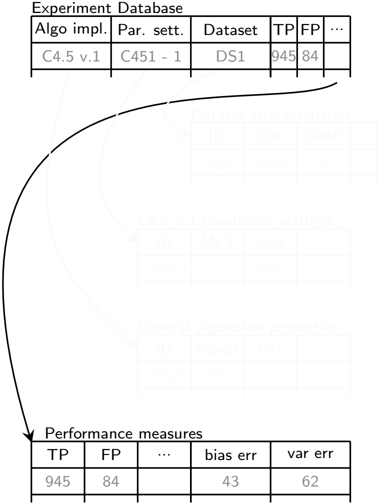
TP	FP	···	bias err	var err
945	84		43	62

- Misclassification error can be decomposed into :
  - bias error: systematic error: algorithm underfits target concept
  - variance error: variation on different samples (overfitting)

Rep.Bias	Comp.Bias	Bias err	Var. err
appr.	too strong	high	low
appr.	ok	low	low
appr.	too weak	low	high
inappr.	too strong	high	low
inappr.	ok	high	avg
inappr.	too weak	high	high

- Diagnose bad performance and link to dataset/algorithm characteristics:
  - bias↗: bad representation model
  - variance↗: bad parameter settings

Investigating inductive performance



Experiment Database

Algo impl.	Par. sett.	Dataset	TP	FP	···
C4.5 v.1	C451 - 1	DS1	945	84	

Dataset characteristics

ID	size	#attr	···
DS1	2300	43	

C4.5 v.1 parameter settings

ID	MLS	heur	···
C451-1	2	gain	

General algorithm properties

ID	model	lin?	···
C45v1	DT	no	

Performance measures

TP	FP	···	bias err	var err
945	84		43	62

- Misclassification error can be decomposed into :
  - bias error: systematic error: algorithm underfits target concept
  - variance error: variation on different samples (overfitting)

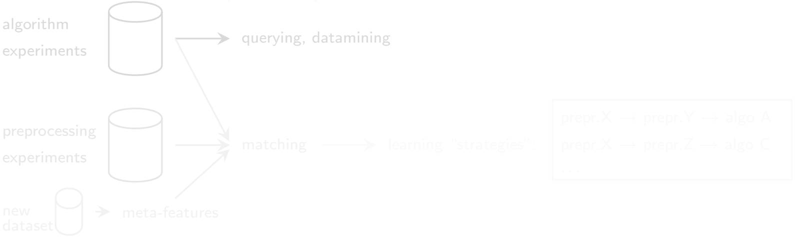
Rep.Bias	Comp.Bias	Bias err	Var. err
appr.	too strong	high	low
appr.	ok	low	low
appr.	too weak	low	high
inappr.	too strong	high	low
inappr.	ok	high	avg
inappr.	too weak	high	high

- Diagnose bad performance and link to dataset/algorithm characteristics:
  - bias ↗: bad representation model
  - variance ↗: bad parameter settings

Preprocessing steps

Data preprocessing has very large effect on inductive performance

- Experiment database: effect of dataset char. on performance
- Separate database: effect of preprocessing on dataset char.
- For new dataset characteristics:
  - Predict how preprocessing changes characteristics
  - Predict algorithm performance on projected dataset char.
- Propose (ranked) list of machine learning "strategies"



algorithm experiments → querying, datamining

preprocessing experiments → matching → learning "strategies":

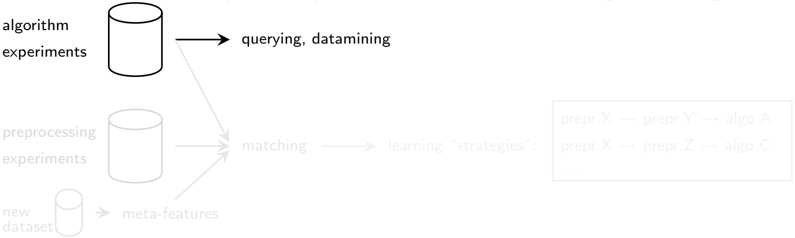
prepr.X → prepr.Y → algo A
prepr.X → prepr.Z → algo C
. . .

new dataset → meta-features

Preprocessing steps

Data preprocessing has very large effect on inductive performance

- Experiment database: effect of dataset char. on performance
- Separate database: effect of preprocessing on dataset char.
- For new dataset characteristics:
  - Predict how preprocessing changes characteristics
  - Predict algorithm performance on projected dataset char.
- Propose (ranked) list of machine learning "strategies"



algorithm experiments → querying, datamining

preprocessing experiments

new dataset → meta-features

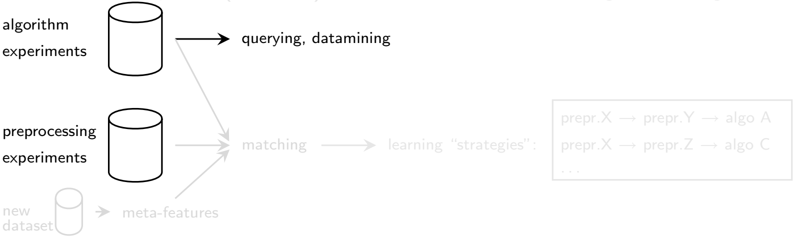
matching ⟶ learning "strategies":

prepr.X ⟶ prepr.Y ⟶ algo A
prepr.X ⟶ prepr.Z ⟶ algo C
. . .

Preprocessing steps

Data preprocessing has very large effect on inductive performance

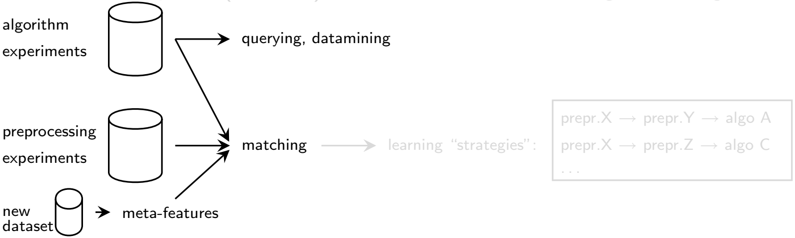
- Experiment database: effect of dataset char. on performance
- Separate database: effect of preprocessing on dataset char.
- For new dataset characteristics:
    - Predict how preprocessing changes characteristics
    - Predict algorithm performance on projected dataset char.
- Propose (ranked) list of machine learning "strategies"



Preprocessing steps

Data preprocessing has very large effect on inductive performance

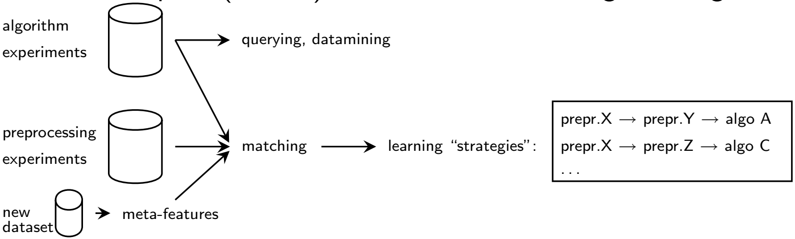
- Experiment database: effect of dataset char. on performance
- Separate database: effect of preprocessing on dataset char.
- For new dataset characteristics:
  - Predict how preprocessing changes characteristics
  - Predict algorithm performance on projected dataset char.
- Propose (ranked) list of machine learning "strategies"



Preprocessing steps

Data preprocessing has very large effect on inductive performance

- Experiment database: effect of dataset char. on performance
- Separate database: effect of preprocessing on dataset char.
- For new dataset characteristics:
  - Predict how preprocessing changes characteristics
  - Predict algorithm performance on projected dataset char.
- Propose (ranked) list of machine learning "strategies"



Preprocessing steps

Strong link between preprocessing steps and bias/variance error:

- Feature construction and transformation
  - reduces bias error by changing data representation
  - e.g. removing attribute correlations
- Feature selection
  - reduces variance error by removing irrelevant attributes
  - e.g. less "noise", less chance of overfitting

We can use bias/variance error to predict when preprocessing step may improve algorithm performance

Preprocessing steps

Strong link between preprocessing steps and bias/variance error:

- Feature construction and transformation
  - reduces bias error by changing data representation
  - e.g. removing attribute correlations
- Feature selection
  - reduces variance error by removing irrelevant attributes
  - e.g. less “noise”, less chance of overfitting

We can use bias/variance error to predict when preprocessing step may improve algorithm performance

Preprocessing steps

Strong link between preprocessing steps and bias/variance error:

- Feature construction and transformation
  - reduces bias error by changing data representation
  - e.g. removing attribute correlations
- Feature selection
  - reduces variance error by removing irrelevant attributes
  - e.g. less “noise”, less chance of overfitting

We can use bias/variance error to predict when preprocessing step may improve algorithm performance

Outline

Intro: Meta-learning

Limitations

An integrated solution

Conclusion

Conclusion

- Ideas for a descriptive form of meta-learning
  - thorough investigation of algorithm behavior
  - explain behavior in terms of their properties
- Experiment databases: efficient experimentation
  - synthetic datasets: unbiased, "natural", covering
  - generalization over algorithms
    - parameter settings
    - general algorithm properties
  - bias/variance error decomposition
- Idem for effect of preprocessing techniques
  - learn when preprocessing useful
  - propose machine learning "strategies"

Conclusion

- Ideas for a descriptive form of meta-learning
  - thorough investigation of algorithm behavior
  - explain behavior in terms of their properties
- Experiment databases: efficient experimentation
  - synthetic datasets: unbiased, "natural", covering
  - generalization over algorithms
    - parameter settings
    - general algorithm properties
  - bias/variance error decomposition
- Idem for effect of preprocessing techniques
  - learn when preprocessing useful
  - propose machine learning "strategies"

Conclusion

- Ideas for a descriptive form of meta-learning
    - thorough investigation of algorithm behavior
    - explain behavior in terms of their properties
- Experiment databases: efficient experimentation
    - synthetic datasets: unbiased, “natural”, covering
    - generalization over algorithms
        - parameter settings
        - general algorithm properties
    - bias/variance error decomposition
- Idem for effect of preprocessing techniques
    - learn when preprocessing useful
    - propose machine learning “strategies”

Questions?