Semi-supervised tree learning for SOP

Jurica Levatić, Dragi Kocev, Michelangelo Ceci, Sašo Džeroski

September 4th, 2016, Ohrid, Macedonia

MAESTRA
LEARNING FROM MASSIVE, INCOMPLETELY ANNOTATED, AND STRUCTURED DATA

IJS

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

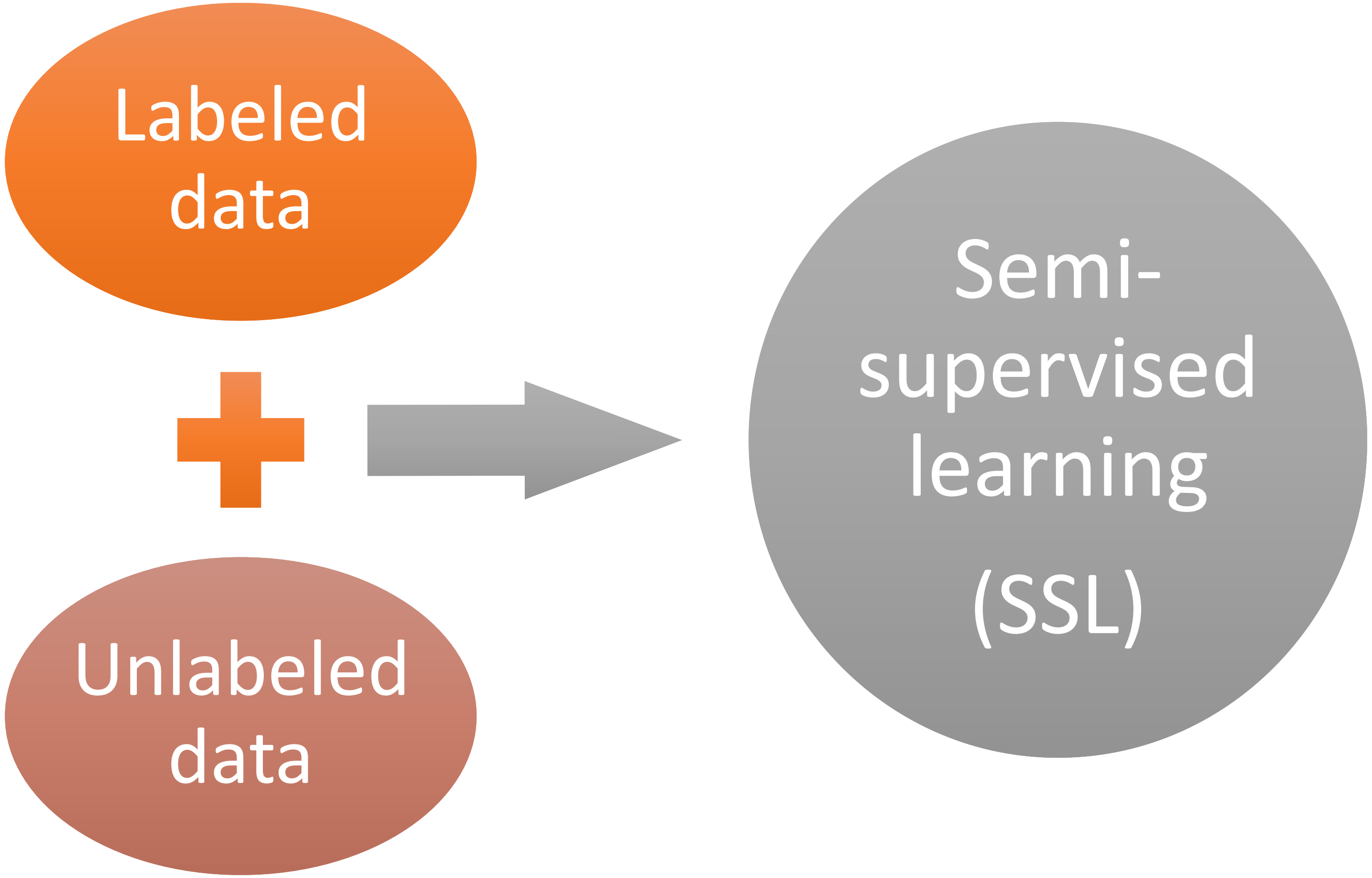
Outline

- Introduction
- Predictive clustering trees
- Distance-based SSL
- Conclusions

Motivation

Supervised learning
- Classification
- Regression

Unsupervised learning
- Clustering
- Dimensionality reduction



Motivation

Supervised learning

- Cla
- Reg

Unsu

- Clu
- Di

Labeled

emi-
ervised
arning
SSL)

- Labeling is **expensive** and **laborious** (especially for structured outputs)
- Unlabeled data are **cheap** and **abundant**.

SSL for SOP has high practical utility!

SSL for classification tasks

	Descriptive space				Target space
Example 1	1	TRUE	0.49	0.69	Yes
Example 2	2	FALSE	0.08	0.07	**?**
Example 3	1	FALSE	0.08	0.07	**?**
Example 4	2	TRUE	0.49	0.69	Yes
Example 5	3	TRUE	0.49	0.69	No
Example 6	4	FALSE	0.08	0.07	**?**
…	…				…

SSL for regression tasks

	Descriptive space				Target space
Example 1	1	TRUE	0.49	0.69	0.84
Example 2	2	FALSE	0.08	0.07	**?**
Example 3	1	FALSE	0.08	0.07	0.11
Example 4	2	TRUE	0.49	0.69	**?**
Example 5	3	TRUE	0.49	0.69	**?**
Example 6	4	FALSE	0.08	0.07	0.78
…	…				…

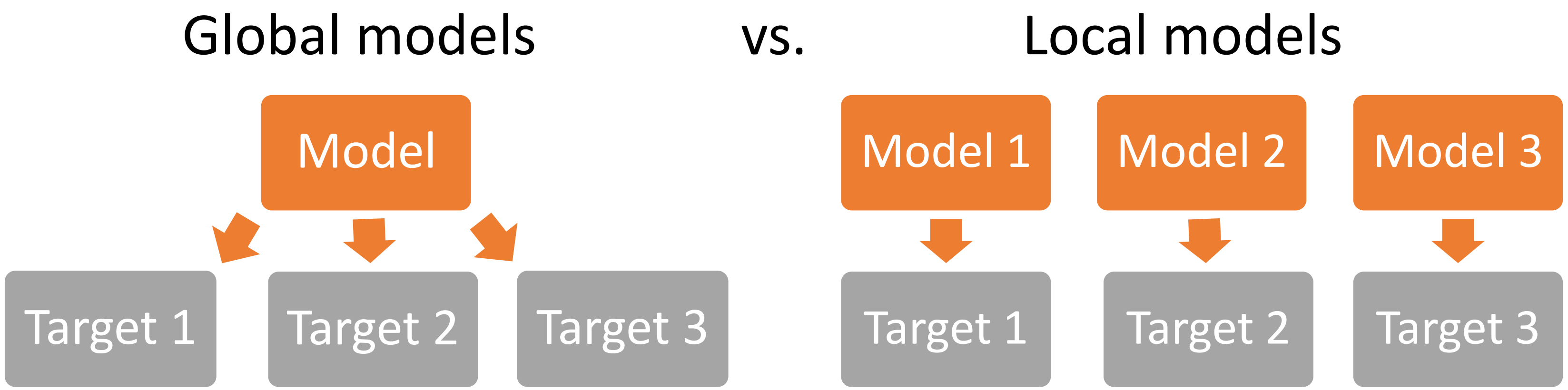
SSL for multi-label classification

	Descriptive space				Target space		
Example 1	1	TRUE	0.49	0.69	?	?	?
Example 2	2	FALSE	0.08	0.07	0	1	1
Example 3	1	FALSE	0.08	0.07	?	?	?
Example 4	2	TRUE	0.49	0.69	1	0	1
Example 5	3	TRUE	0.49	0.69	?	?	?
Example 6	4	FALSE	0.08	0.07	1	0	0
…	…				…	…	…

SSL for multi-target regression

	Descriptive space				Target space		
Example 1	1	TRUE	0.49	0.69	?	?	?
Example 2	2	FALSE	0.08	0.07	0.56	0.99	7.59
Example 3	1	FALSE	0.08	0.07	?	?	?
Example 4	2	TRUE	0.49	0.69	0.08	0.77	8.86
Example 5	3	TRUE	0.49	0.69	?	?	?
Example 6	4	FALSE	0.08	0.07	0.43	2.10	8.09
…	…				…	…	…

Motivation



Global models

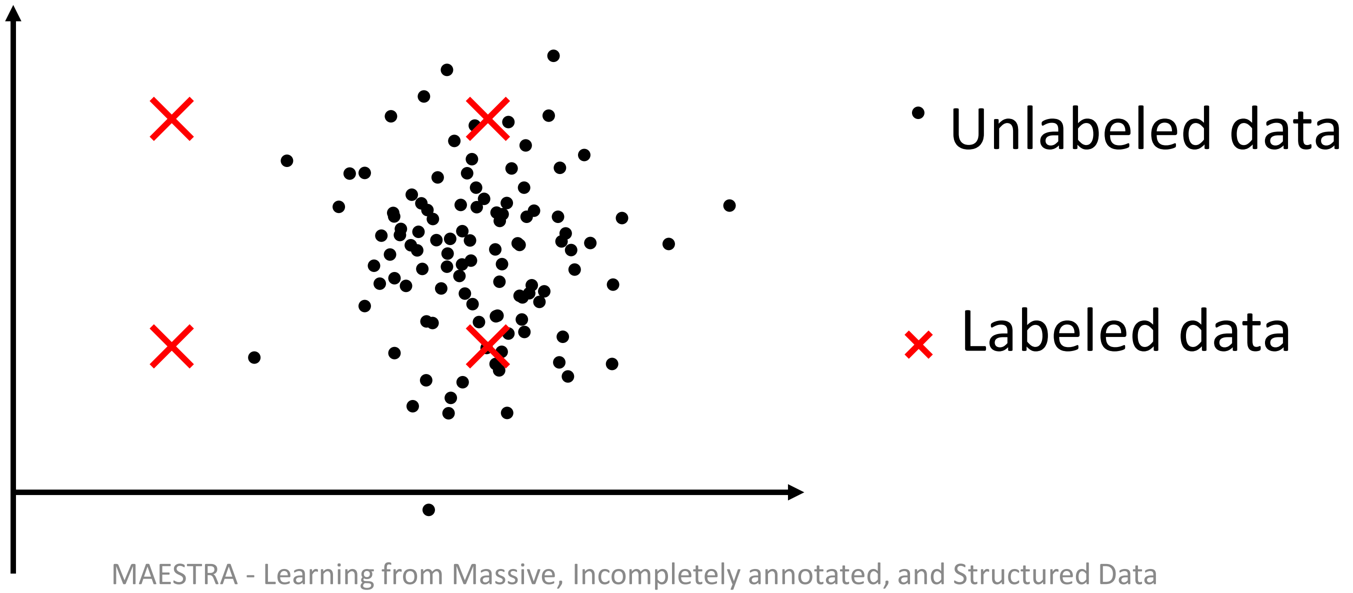
- Can catch dependencies among the target variables
  - Can “smooth” prediction function
  - Avoiding large discontinuities of the prediction function (Torgo, 1999; Quinlan, 1992)
- Better predictive performance, computationally more efficient, produce simpler models, and overfit less

SSL underlying mechanism

Unlabeled data is a complementary way to “smooth” the prediction function

Semi-supervised smoothness assumption

„If two points x_i and x_j in a high density region are close, then also their outputs y_i and y_j should be close“



Existing approaches

Semi-supervised learning

- Majority of work is dealing with classical (unstructured) classification tasks
- SSL for SOP largely unexplored

Kernel based approaches (semi-supervised support vector machines):

- *(Xu and Schuurmans, 2005; Altun et al., 2006; Zien et al., 2007; Brefeld et al., 2008)*

Graph based SSL methods for SOP:

- *(Zha et al., 2009; Subramanya et al., 2010)*

Other approaches: Hidden Markov Models with Latent Dirichlet Allocation *(Li and McCallum, 2005),* Conditional Random Fields *(Jiao et al., 2006; Wang et al., 2009)*, Hybrid generative/discriminative approach for sequence labeling *(Dhillon et al., 2011)*, Weight space based graph regularization *(Dhillon et al., 2012)*

Existing approaches: Issues

1) **High computational complexity**
   - Not applicable to large datasets and problems with large-size outputs
2) **Un-interpretable models**
   - Majority are kernel based methods
3) **Focus only on a specific type of structured output**
   - e.g., sequence learning
4) **Methods are applied and evaluated only on specific domains**
   - Text-mining and related domains

MAESTRA: Extend PCT framework towards SSL

Predictive Clustering framework

- Efficiently solving various SOP tasks: multi-target prediction, hierarchical multi-label classification, and time-series prediction
- Several possibilities for extension towards semi-supervised learning

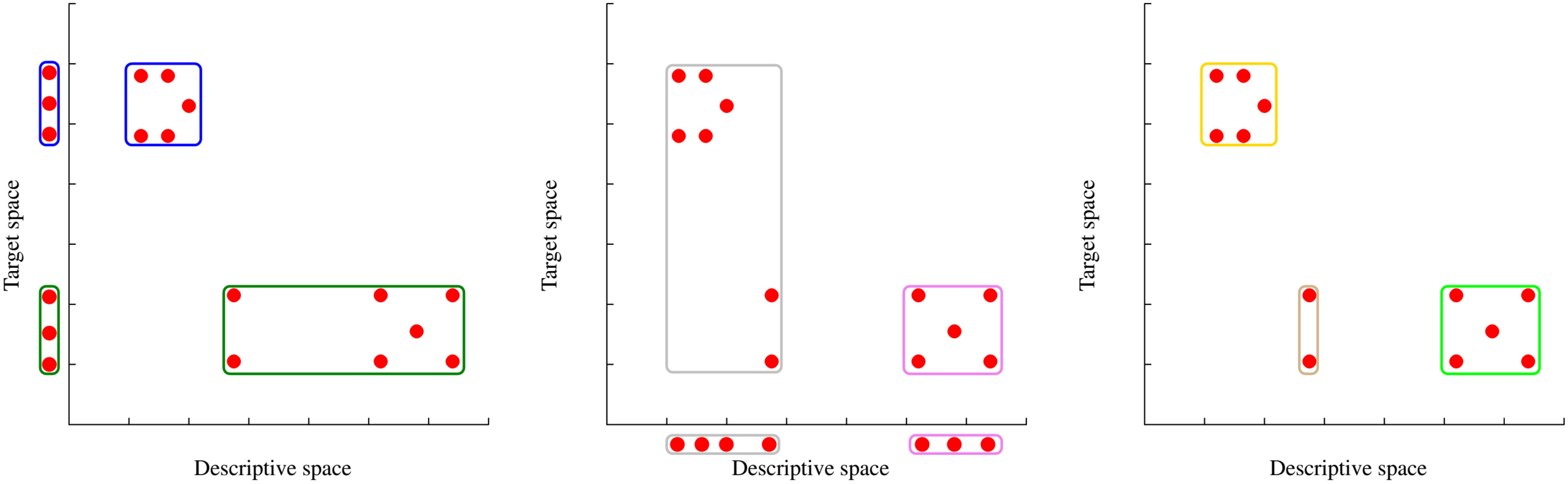
Goals:

1) Develop SSL methods within the PC framework **efficient** in terms of **predictive power** and **computational complexity**
2) Retain current **interpretability** of models in the PC framework
3) Develop SSL methods that can handle **various types of SOP** tasks
4) Evaluate methods in **various domains** (eco. modeling, comp. sys. Biology, chemoinformatics, etc.)

Outline

- Introduction
- Predictive clustering trees
- Distance-based SSL
- Conclusions

Predictive clustering



Predictive modelling

Clustering

Predictive Clustering

Predictive clustering trees

- Implemented in the CLUS system (KU Leuven & JSI)
- The tree is a hierarchy of clusters
- Heuristic score: minimize intra-cluster variance
- Instantiation of the variance for different tasks

procedure PCT(E) **returns** tree
$(t^*, h^*, \mathcal{P}^*) = \mathrm{BestTest}(E)$
if $t^* \neq none$ **then**
    for each $E_i \in \mathcal{P}^*$ **do**
        $tree_i = \mathrm{PCT}(E_i)$
    return $\mathrm{node}(t^*, \bigcup_i \{tree_i\})$
else
    return $\mathrm{leaf}(\mathrm{Prototype}(E))$

procedure BestTest(E)
$(t^*, h^*, \mathcal{P}^*) = (none, 0, \emptyset)$
for each possible test t **do**
    $\mathcal{P}$ = partition induced by t on E
    $h = Var(E) - \sum_{E_i \in \mathcal{P}} \frac{|E_i|}{|E|} Var(E_i)$
    if $(h > h^*) \wedge \mathrm{Acceptable}(t, \mathcal{P})$ **then**
        $(t^*, h^*, \mathcal{P}^*) = (t, h, \mathcal{P})$
return $(t^*, h^*, \mathcal{P}^*)$

Predictive clustering trees

- Implemented in the CLUS system (KU Leuven & JSI)
- The tree is a hierarchy of clusters
- Heuristic score: minimize intra-cluster variance
- Instantiation of the variance for different tasks

procedure PCT(E) **returns** tree
$(t^*, h^*, \mathcal{P}^*) = \mathrm{BestTest}(E)$
if $t^* \neq none$ **then**
  for each $E_i \in \mathcal{P}^*$ **do**
    $tree_i = \mathrm{PCT}(E_i)$
  return $\mathrm{node}(t^*, \bigcup_i \{tree_i\})$
else
  return $\mathrm{leaf}(\mathrm{Prototype}(E))$

procedure BestTest(E)
$(t^*, h^*, \mathcal{P}^*) = (none, 0, \emptyset)$
for each possible test t **do**
  $\mathcal{P}$ = partition induced by t on E
  $h = Var(E) - \sum_{E_i \in \mathcal{P}} \frac{|E_i|}{|E|} Var(E_i)$
  if $(h > h^*) \wedge \mathrm{Acceptable}(t, \mathcal{P})$ **then**
    $(t^*, h^*, \mathcal{P}^*) = (t, h, \mathcal{P})$
return $(t^*, h^*, \mathcal{P}^*)$

PCTs instantiations

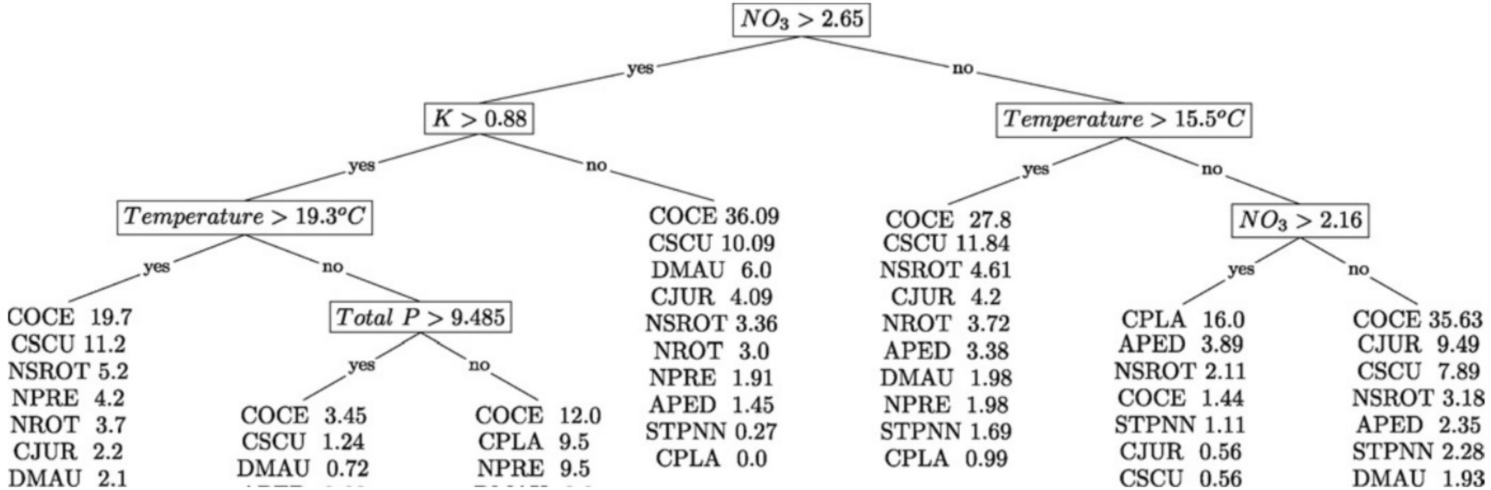
- Multi-target regression
  - Prototype: Average
  - Variance: $$Var(E) = \sum_{i=1}^{T} Var(Y_i)$$
- Multi-target classification/Multi-label classification
  - Prototype: Probability distribution and Majority vote
  - Variance: $$Var(E) = \sum_{i=1}^{T} Gini(E, Y_i) \text{ or } Var(E) = \sum_{i=1}^{T} Entropy(E, Y_i)$$
- Hierarchical multi-label classification
  - Prototype: Average with a threshold for class membership
  - Hierarchy type: tree or DAG
  - Variance: $$Var(E) = \frac{1}{|E|} \cdot \sum_{E_i \in E} d(L_i, \overline{L})^2,$$

$$d(L_1, L_2) = \sqrt{\sum_{i=1}^{|L|} \omega(c_i) \cdot (L_{1,i} - L_{2,i})^2}, \ \omega(c_i) = \omega_0 \cdot \omega(par(c_i))$$

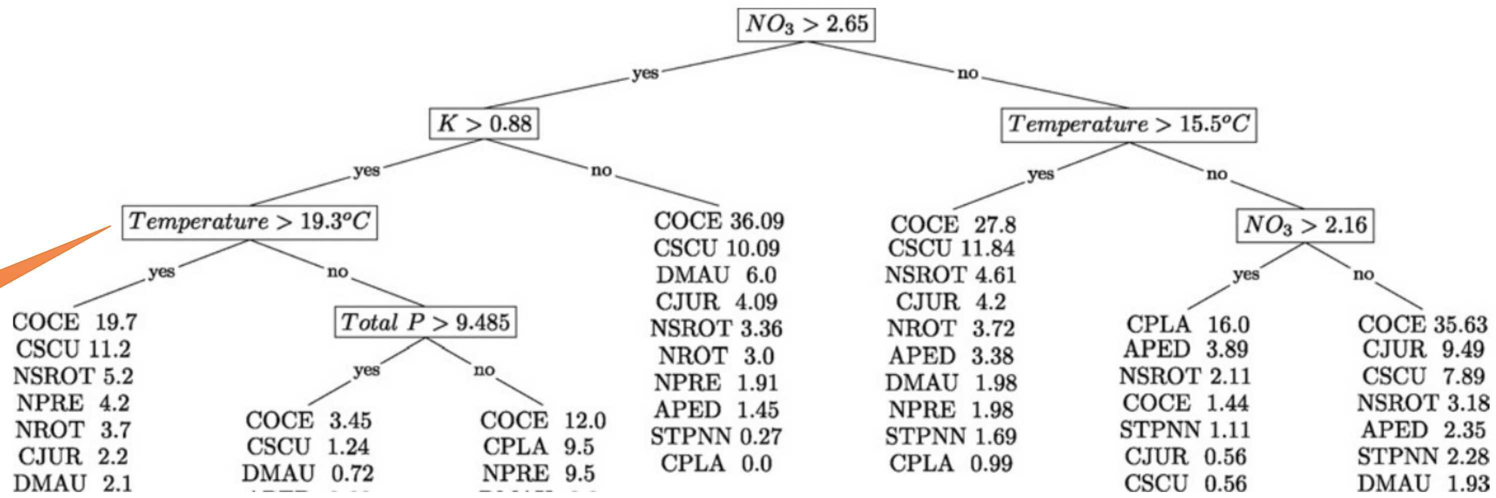
Outline

- Introduction
- Predictive clustering trees
- Distance-based SSL
- Conclusions

Supervised PCTs



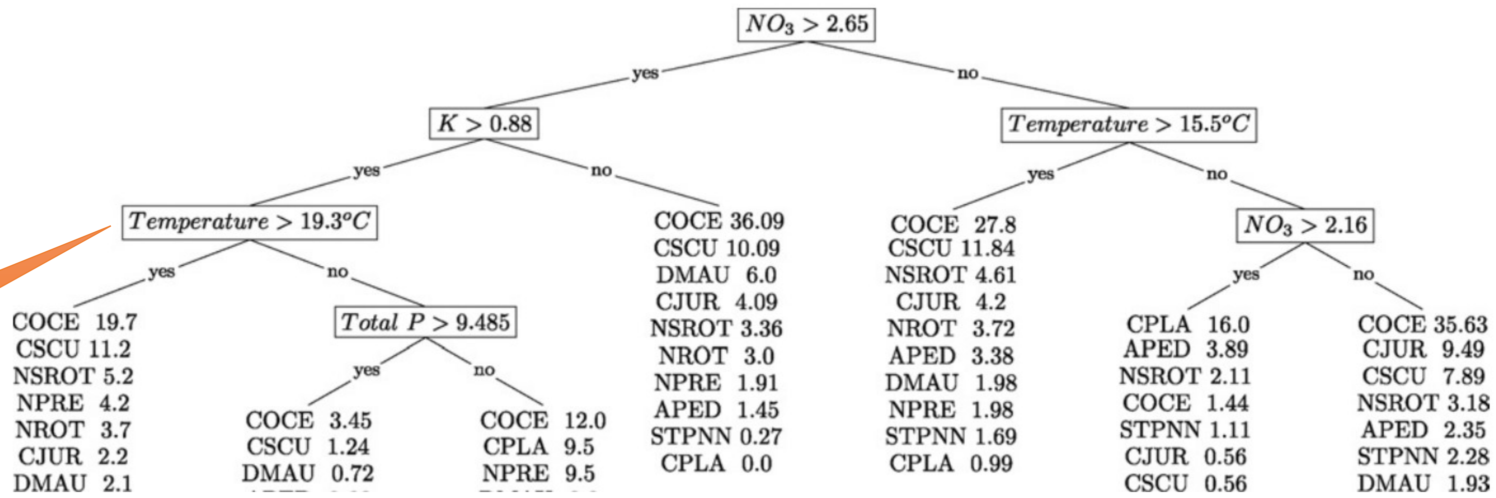
Supervised PCTs



Maximize variance reduction

$$h = \overbrace{Var(E)}^{\text{Variance of a parent node}} - \overbrace{\sum_{E_i \in \mathcal{P}} \frac{|E_i|}{|E|} Var(E_i)}^{\text{Sum of variances of child nodes}}$$

Supervised PCTs



Maximize variance reduction

$$h = \overbrace{Var(E)}^{\text{Variance of a parent node}} - \overbrace{\sum_{E_i \in \mathcal{P}} \frac{|E_i|}{|E|} Var(E_i)}^{\text{Sum of variances of child nodes}}$$

$$Var(E) = \frac{1}{T} \cdot \sum_{i=1}^{T} Var(Y_i)$$

Average of the variances of the ***target variables***

Semi-supervised PCTs

Variance function of semi-supervised PCTs:

$$Var(E) = \frac{1}{T + D} \cdot \left(w \cdot \sum_{i=1}^{T} Var(Y_i) + (1 - w) \cdot \sum_{j=1}^{D} Var(X_j) \right)$$

T = #target attributes, D = #descriptive attributes, w = weight parameter

Variance is calculated on both **target** and **descriptive** side

Semi-supervised PCTs: mixed attributes

Extended variance function:

$$Var(E) = \frac{w}{T_{nu} + T_{no}} \cdot \left(\sum_{i=1}^{T_{nu}} Var(Y_i) + \sum_{i=1}^{T_{no}} gini(Y_i) \right)$$
$$+ \frac{(1-w)}{D_{nu} + D_{no}} \cdot \left(\sum_{j=1}^{D_{nu}} Var(X_j) + \sum_{j=1}^{D_{no}} gini(X_j) \right)$$

T_{nu} = #numerical target attributes, T_{no} = #nominal target attributes
D_{nu} = #numerical descriptive attributes, D_{no} = #nominal descriptive attributes

Semi-supervised PCTs: mixed attributes

Extended variance function:

$$
\left.
\begin{aligned}
Var(E) = \; & \frac{w}{T_{nu} + T_{no}} \cdot \left(\sum_{i=1}^{T_{nu}} Var(Y_i) + \sum_{i=1}^{T_{no}} gini(Y_i) \right) \\
& + \frac{(1-w)}{D_{nu} + D_{no}} \cdot \left(\sum_{j=1}^{D_{nu}} Var(X_j) + \sum_{j=1}^{D_{no}} gini(X_j) \right)
\end{aligned}
\right\}
$$

We are (potentially) mixing apples and oranges

T_{nu} = #numerical target attributes, T_{no} = #nominal target attributes
D_{nu} = #numerical descriptive attributes, D_{no} = #nominal descriptive attributes

Semi-supervised PCTs for SOP

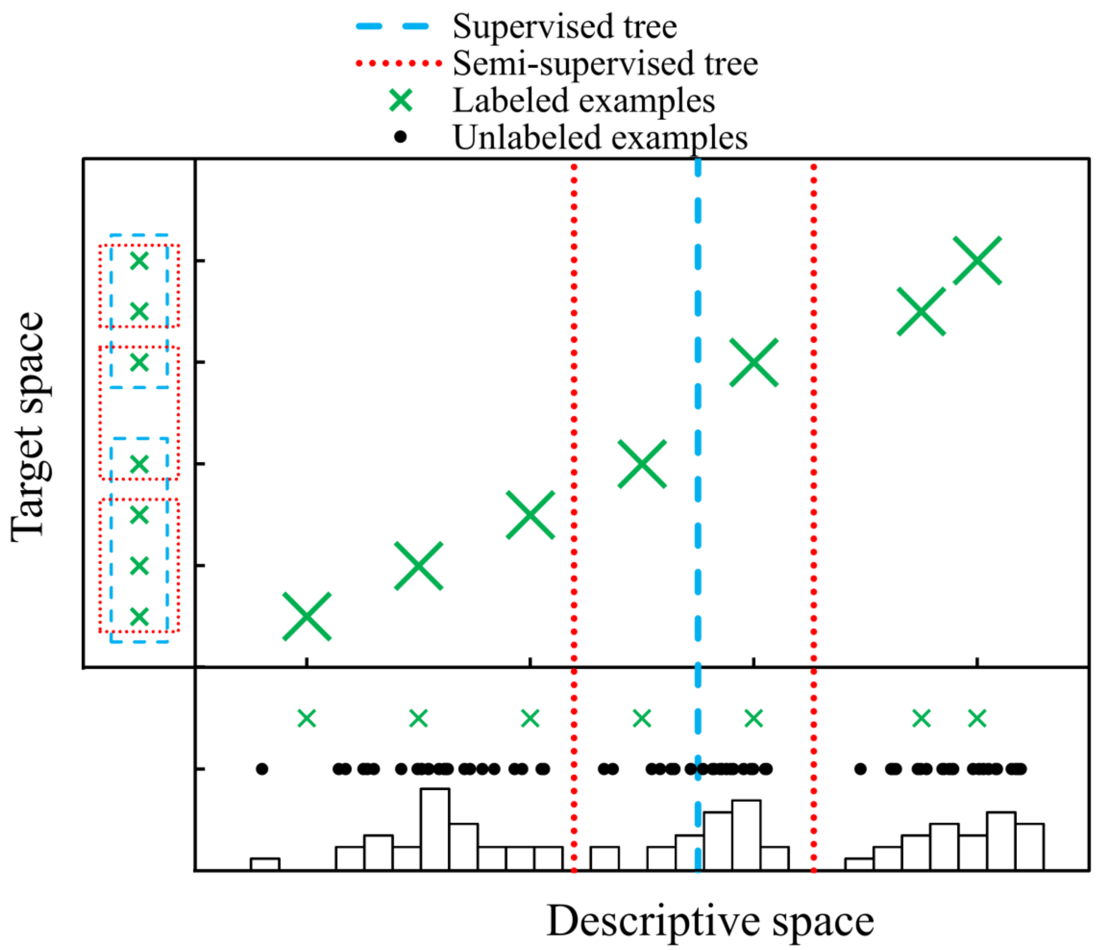
Extended variance function:

$$Var(E) = \frac{w}{T_{nu} + T_{no}} \cdot \left(\sum_{i=1}^{T_{nu}} \frac{Var(Y_i)}{Var_{train}(Y_i)} + \sum_{i=1}^{T_{no}} \frac{gini(Y_i)}{gini_{\text{train}}(Y_i)} \right)$$

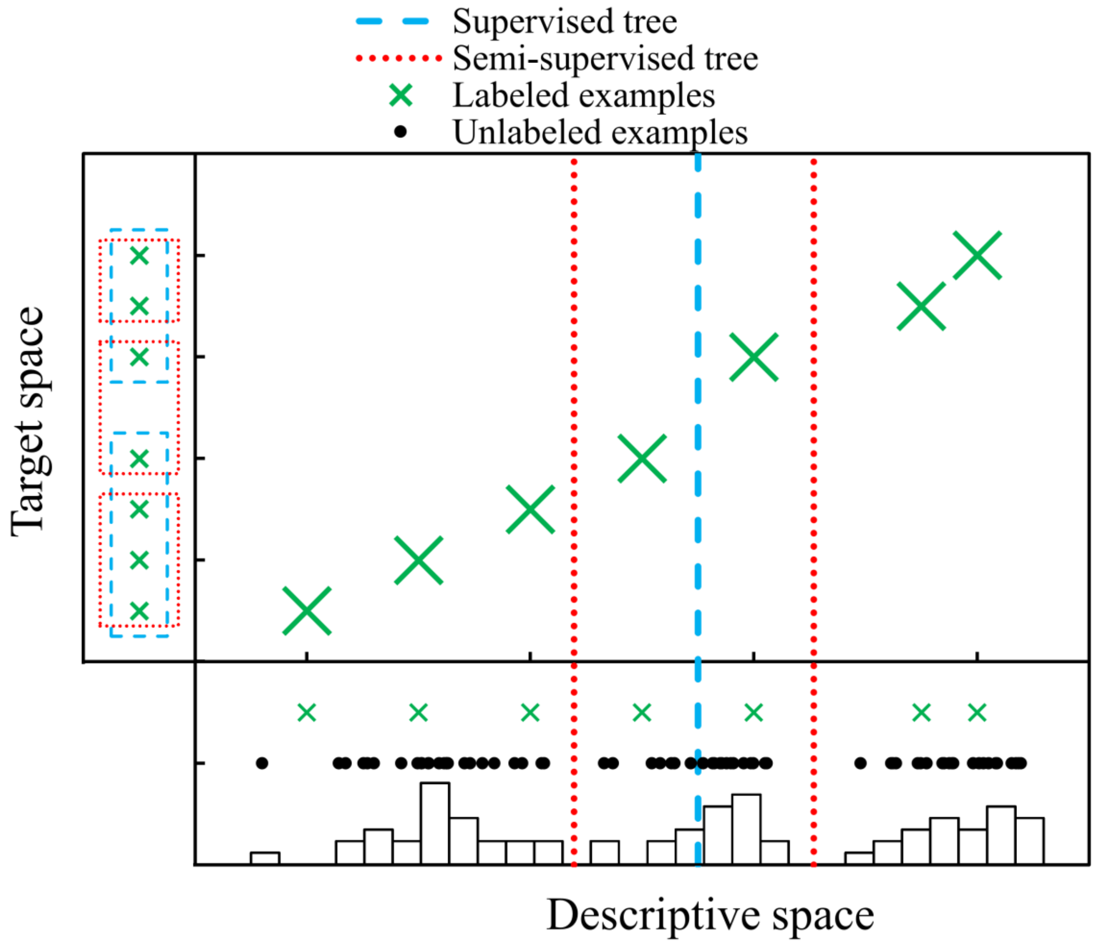
$$+ \frac{(1-w)}{D_{nu} + D_{no}} \cdot \left(\sum_{j=1}^{D_{nu}} \frac{Var(X_j)}{Var_{\text{train}}(X_j)} + \sum_{j=1}^{D_{no}} \frac{gini(X_j)}{gini_{\text{train}}(X_j)} \right)$$

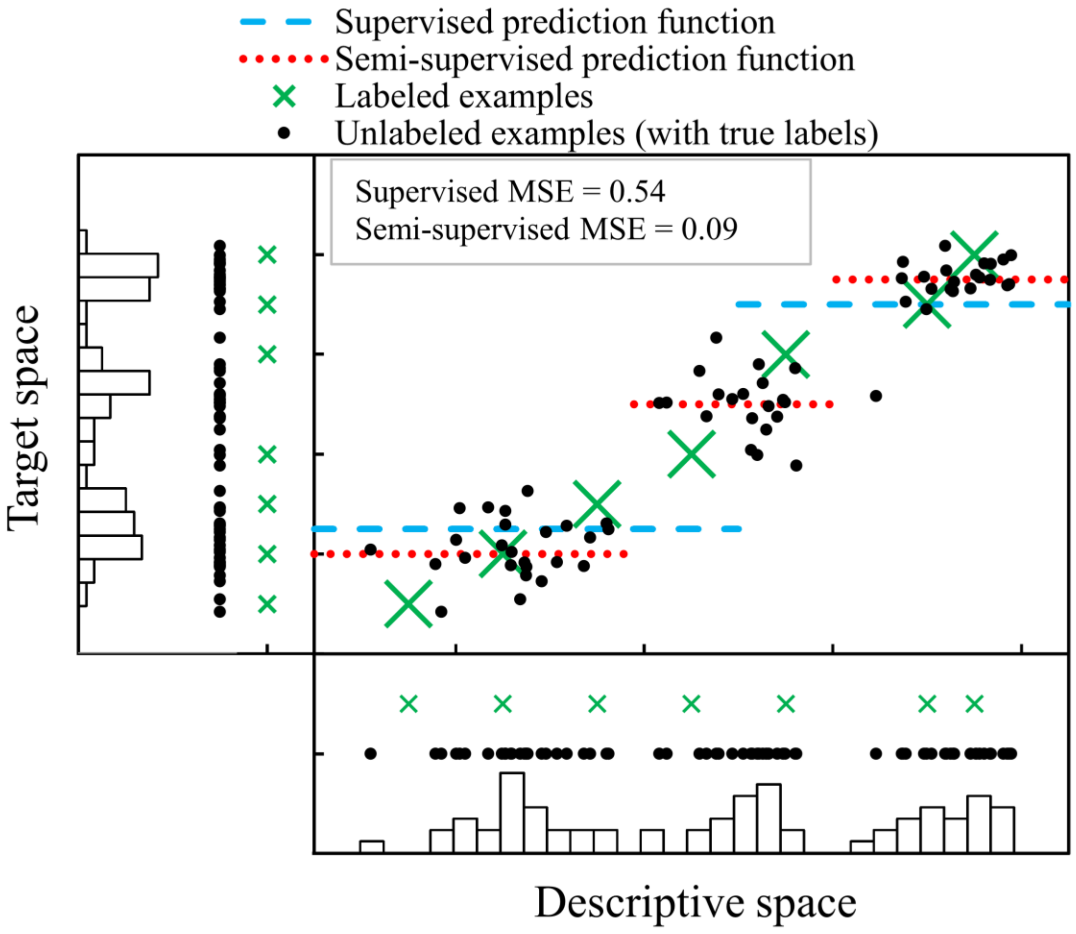
T_{nu} = #numerical target attributes, T_{no} = #nominal target attributes
D_{nu} = #numerical descriptive attributes, D_{no} = #nominal descriptive attributes

SSL PCTs: Smoothness in the target space



SSL PCTs: Smoothness in the target space





SSL PCTs extensions

- Multi-target regression
- Binary classification
- Multi-class classification
- Multi-label classification

Semi-supervised PCTs for MTR

Variances of individual target (Y_i) and descriptive (X_i) attributes:

$$Var(Y_i) = \frac{\frac{N-1}{K_i-1} \cdot \sum_{j=1}^{N} \left(y_{i\,j}\right)^2 - N \cdot \left(\frac{1}{K_i} \cdot \sum_{j=1}^{N} y_{i\,j}\right)^2}{N}$$

N = number of examples, K_i = number of examples with ***non missing values***

Semi-supervised PCTs for MTR

Variances of individual target (Y_i) and descriptive (X_i) attributes:

$$Var(Y_i) = \frac{\frac{N-1}{K_i-1} \cdot \sum_{j=1}^{N} \left(y_{i_j}\right)^2 - N \cdot \left(\frac{1}{K_i} \cdot \sum_{j=1}^{N} y_{i_j}\right)^2}{N}$$

$N =$ number of examples, $K_i =$ number of examples with ***non missing values***

Extreme cases (K= 0):

(1) leafs of the decision tree may contain only unlabeled examples

(2) the calculation of variance for attribute which has only missing values

How we handle extreme cases?

I. estimation of variance with variance of the parent node

   Moderate penalization → medium sized trees

II. estimation of variance with variance on the entire training set

   Maximal penalization → small trees

III. ignoring such attributes

   No penalization → large trees

$$Var(E) = \frac{1}{\hat{T} + \hat{D}} \cdot \left(w \cdot \sum_{i=1}^{\hat{T}} Var(Y_i) + (1 - w) \cdot \sum_{j=1}^{\hat{D}} Var(X_j) \right)$$

$\hat{T}, \hat{D}$ = #target/descriptive attributes with $K_i > 1$

Feature weighted semi-supervised PCTs

Problem: Irrelevant descriptive attributes may hurt performance!

Solution: Weight them by feature ranks

$$Var(E) = \frac{1}{T+D}\left(w \cdot \sum_{i=1}^{T} Var(Y_i) + (1-w) \cdot \sum_{j=1}^{D} \sigma_j \cdot Var(X_j)\right)$$

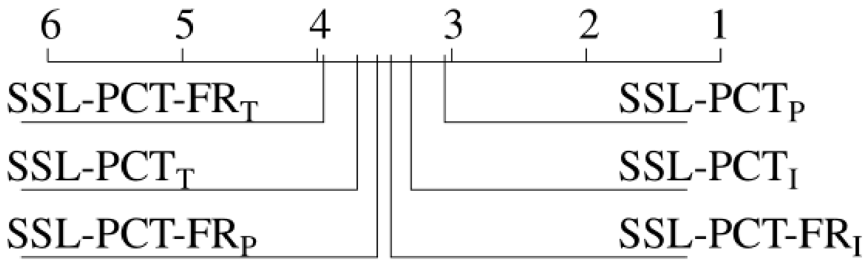
σ_j = normalized feature importance (e.g.,Random forest feature ranking)

Experimental design

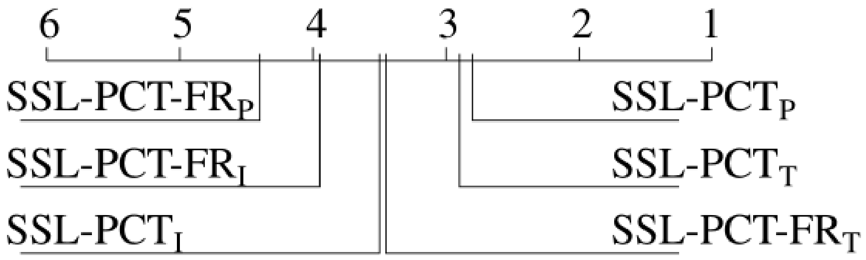
- 6 variants of semi-supervised PCTs
  - SSL-PCT$_P$, SSL-PCT$_T$, SSL-PCT$_I$
  - SSL-PCT-FR$_P$, SSL-PCT-FR$_T$, SSL-PCT-FR$_I$
- Comparison to two baselines on 10 MTR datasets
  - Standard supervised PCTs (Base-PCT)
  - Supervised counterpart of SSL-PCTs (SL-PCT, SL-PCT-FR)
- We explore the influence of the amount of labeled data
  - 25, 50, 100, 200 labeled examples
  - 1%, 5%, 10%, 30% labeled examples
- Transductive evaluation scenario: unlabeled examples = test examples
- 10 runs with different random initialization

Results: Statistical analysis

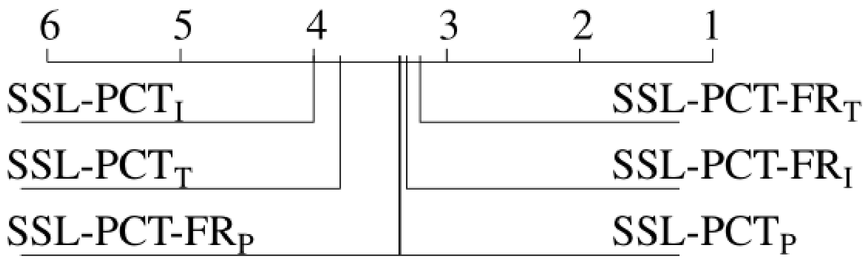
Average ranks diagrams, absolute number of labeled examples



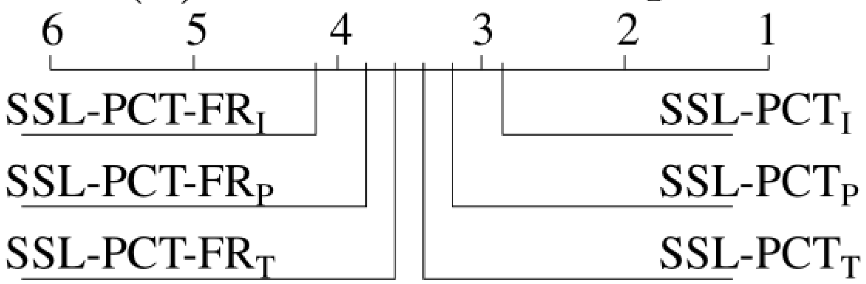
(a) 25 labeled examples



(b) 50 labeled examples



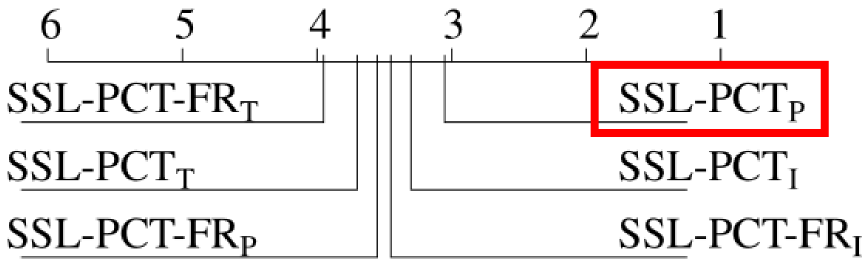
(c) 100 labeled examples



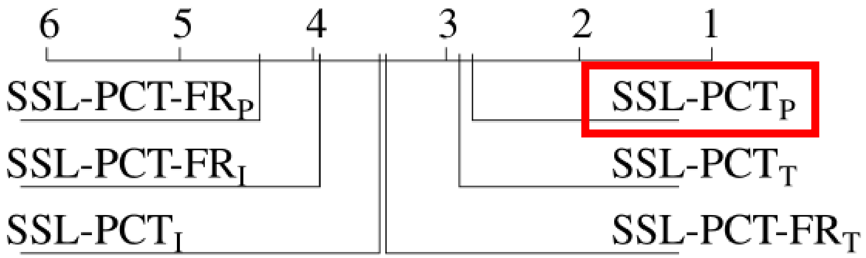
(d) 200 labeled examples

Results: Statistical analysis

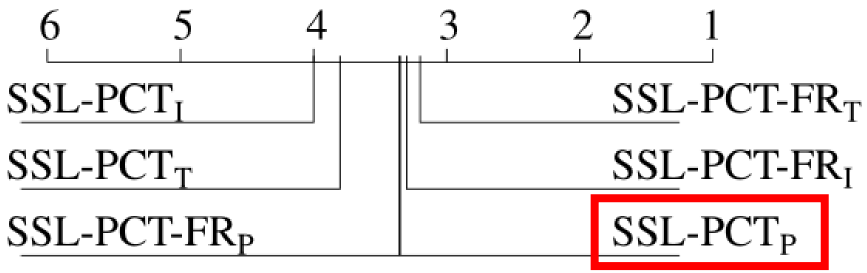
Average ranks diagrams, absolute number of labeled examples



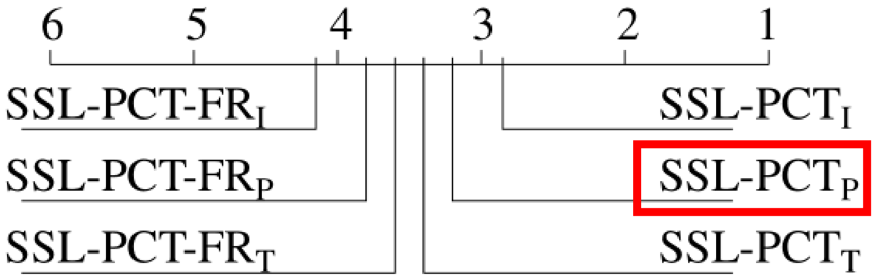
(a) 25 labeled examples



(b) 50 labeled examples



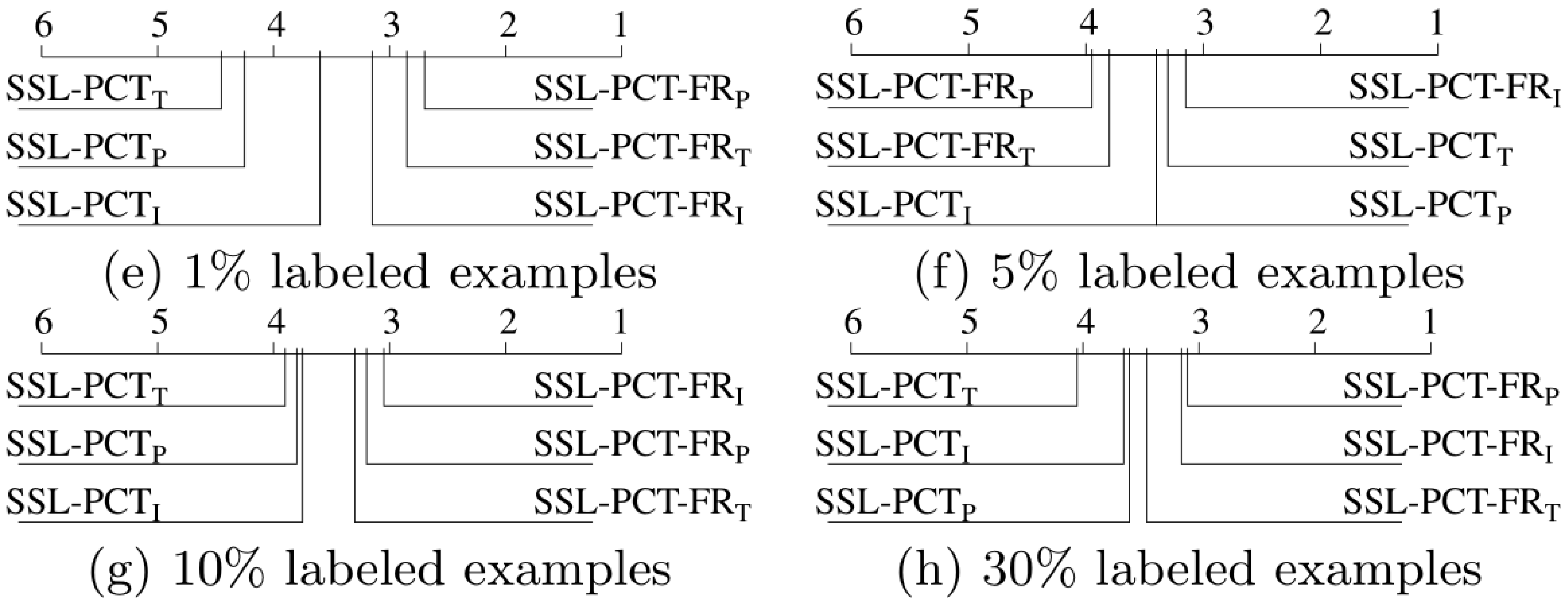
(c) 100 labeled examples



(d) 200 labeled examples

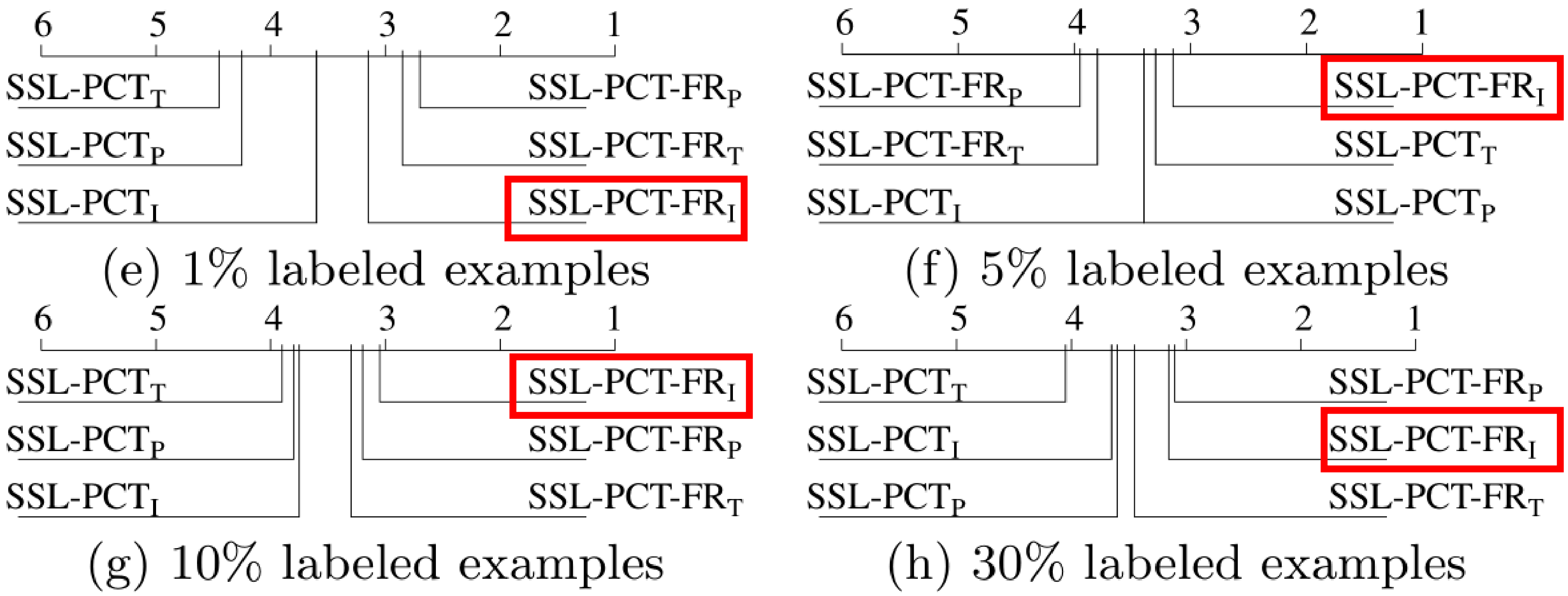
Results: Statistical analysis

Average ranks diagrams, relative number of labeled examples



Results: Statistical analysis

Average ranks diagrams, relative number of labeled examples



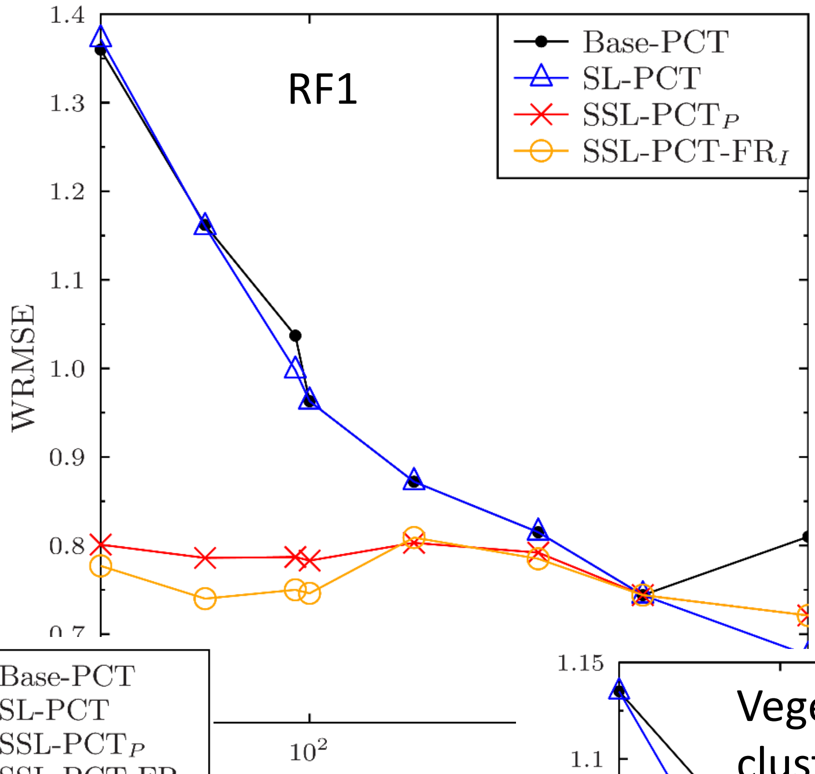
(e) 1% labeled examples

(f) 5% labeled examples

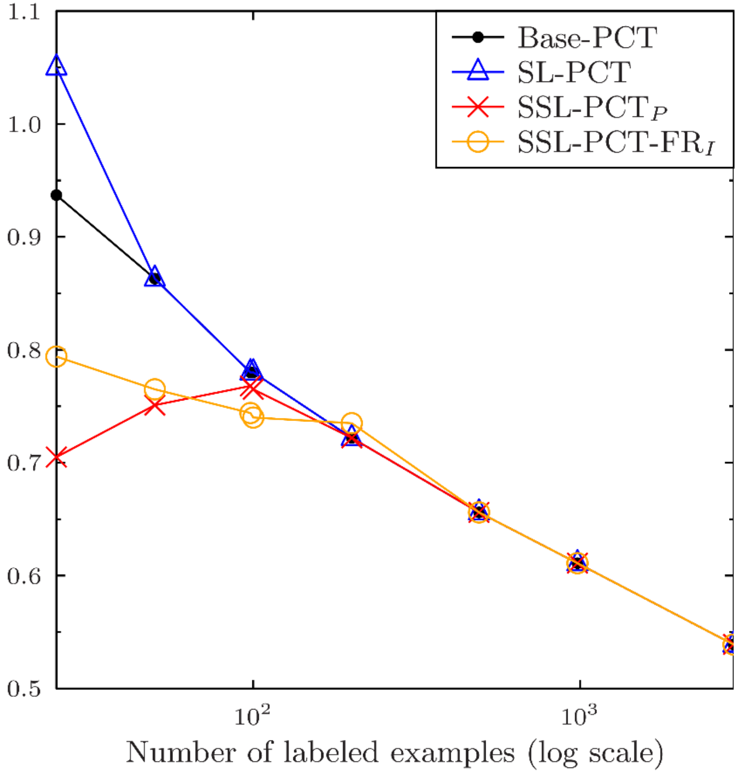
(g) 10% labeled examples

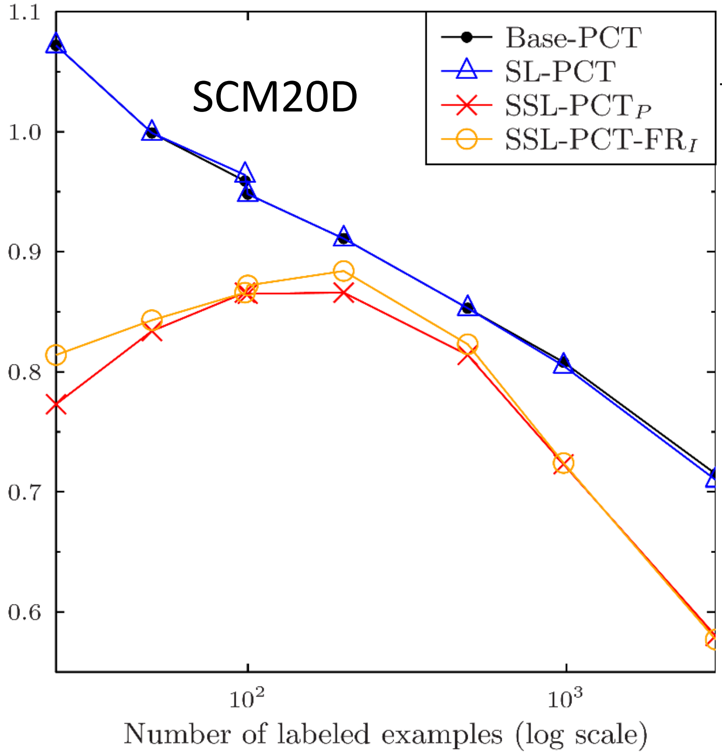
(h) 30% labeled examples

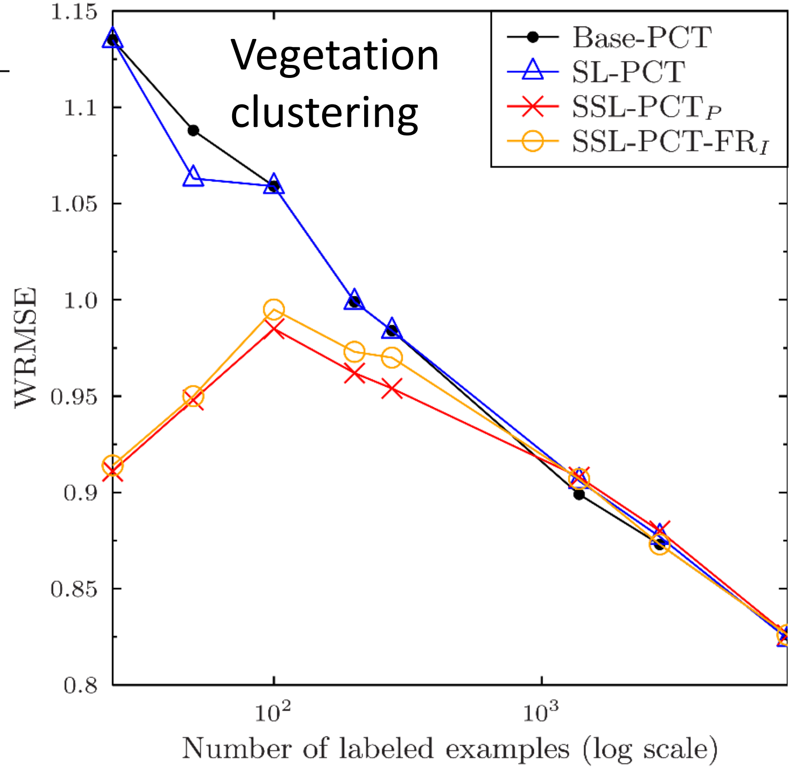
Results: Per-dataset



1) SSL improves a lot

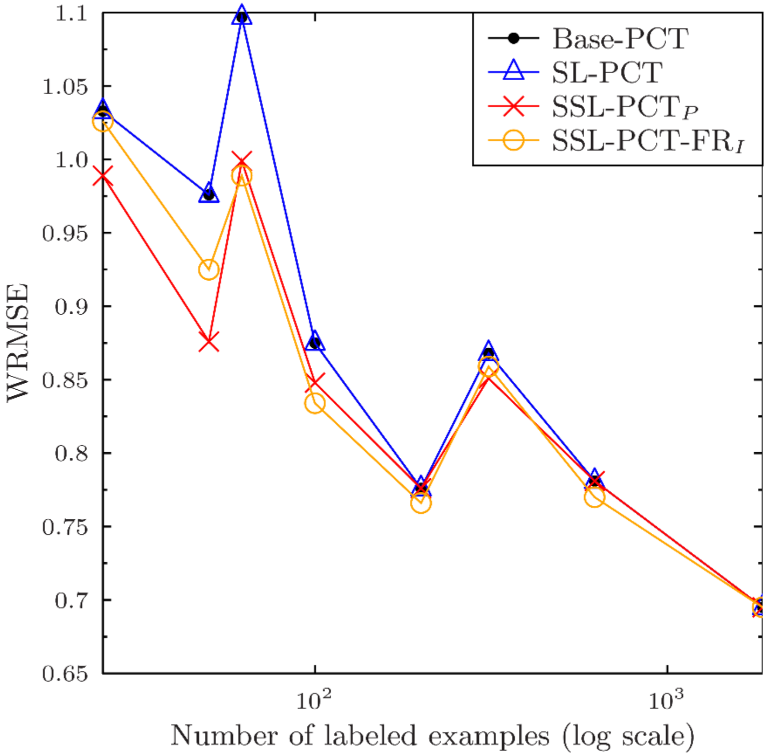




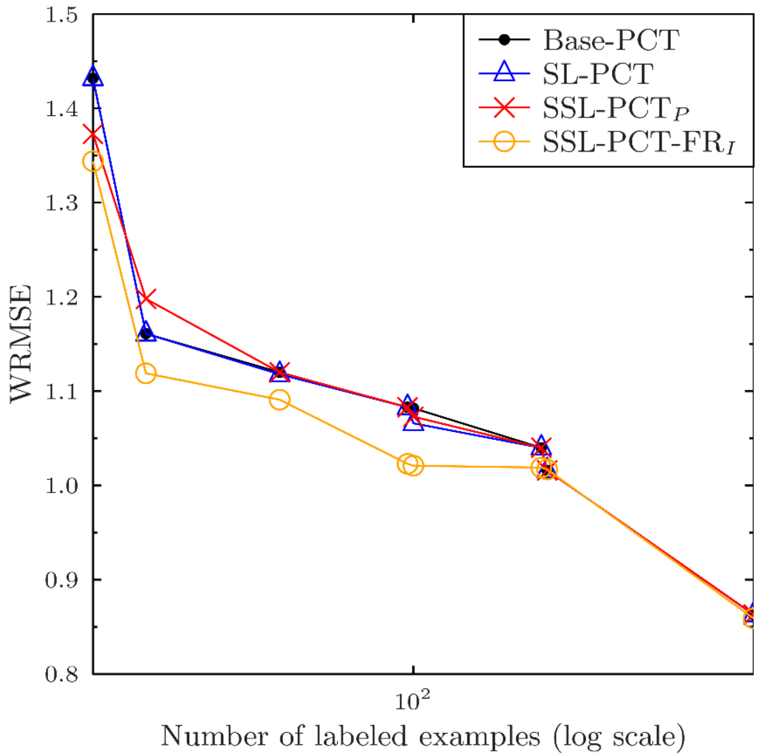


Results: Per-dataset

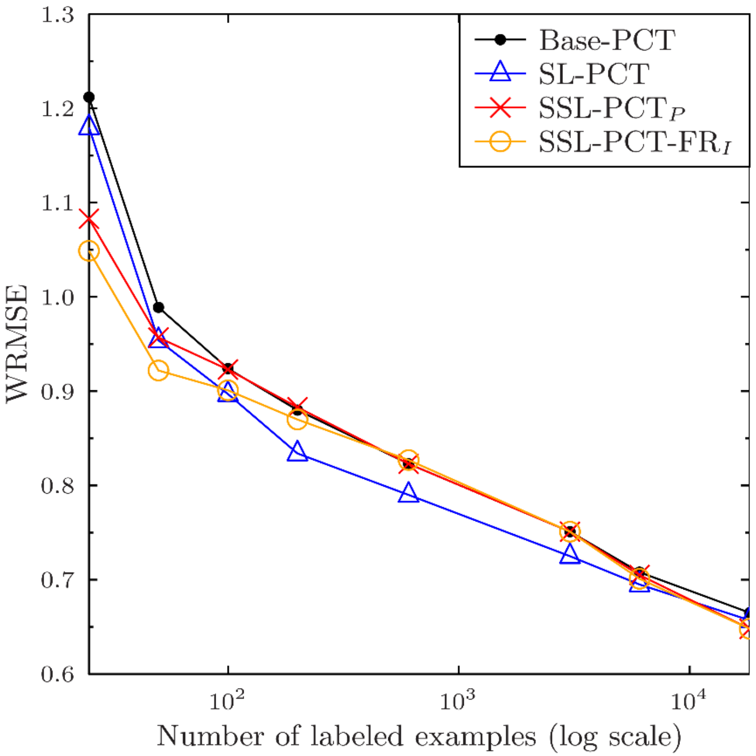
2) SSL improves moderately



Forestry LIDAR LandSat



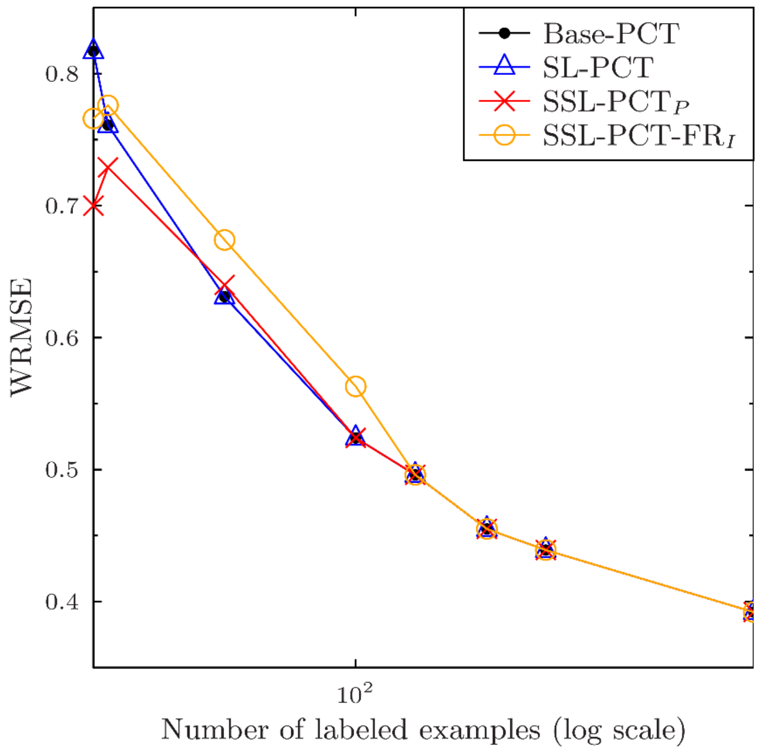
Soil quality



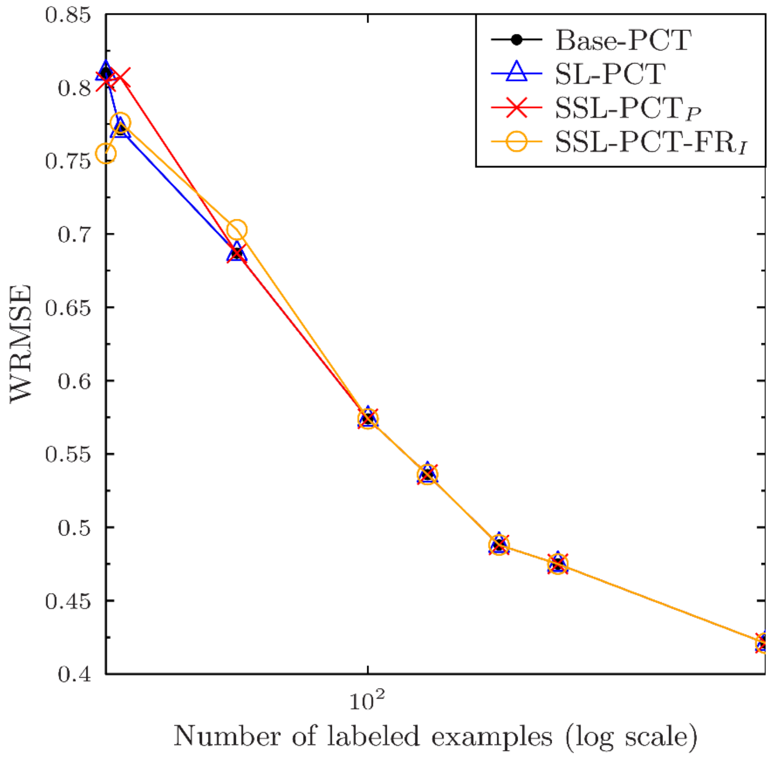
Forestry Kras

Results: Per-dataset

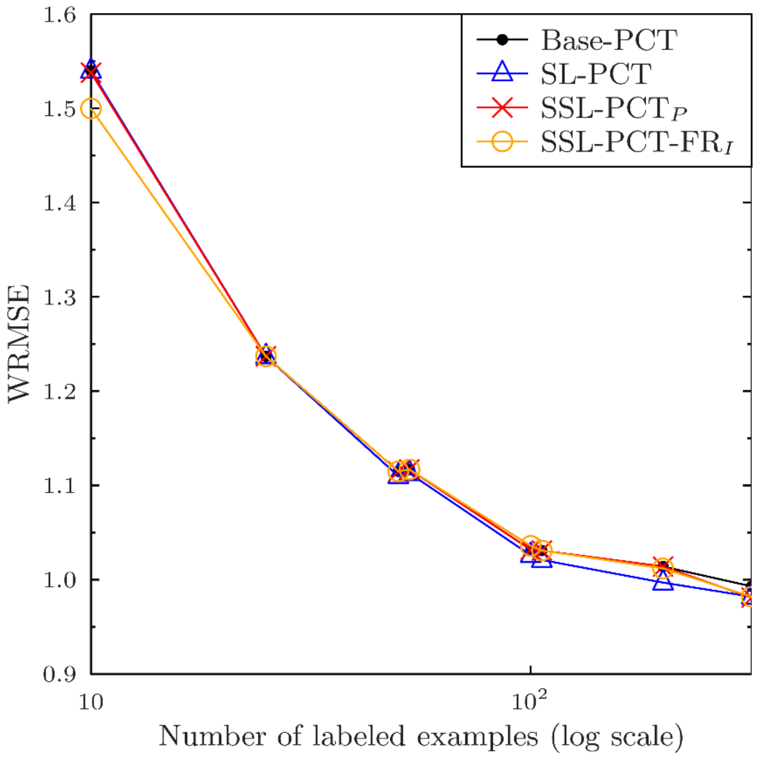
3) SSL improves a bit only for the smallest amount of labeled data (25 labeled ex.)



Forestry LIDAR IRS



Forestry LIDAR Spot



Water quality

Results: Influence of the $\boldsymbol{w}$ parameter

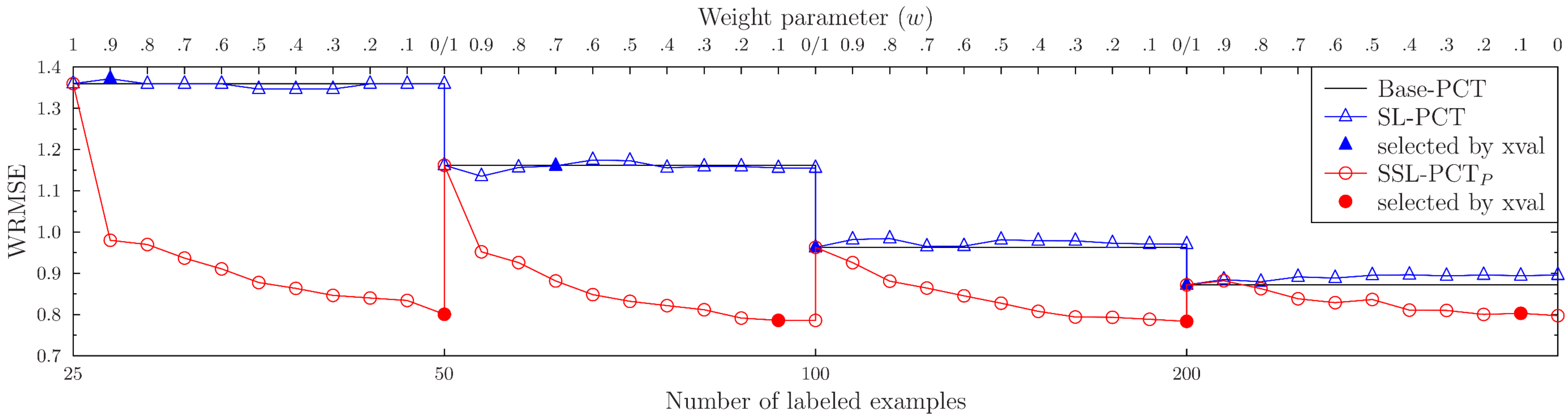
- controls the amount of „supervision“
  - $\boldsymbol{w} = 0 \Rightarrow$ unsupervised learning
  - $\boldsymbol{w} = 1 \Rightarrow$ supervised learning

$$Var(E) = \frac{1}{T+D} \cdot \left(w \cdot \sum_{i=1}^{T} Var(Y_i) + (1-w) \cdot \sum_{j=1}^{D} Var(X_j) \right)$$

- w provides a safety mechanism to SSL-PCTs
- w is optimized via 3-fold cross-validation on labeled part

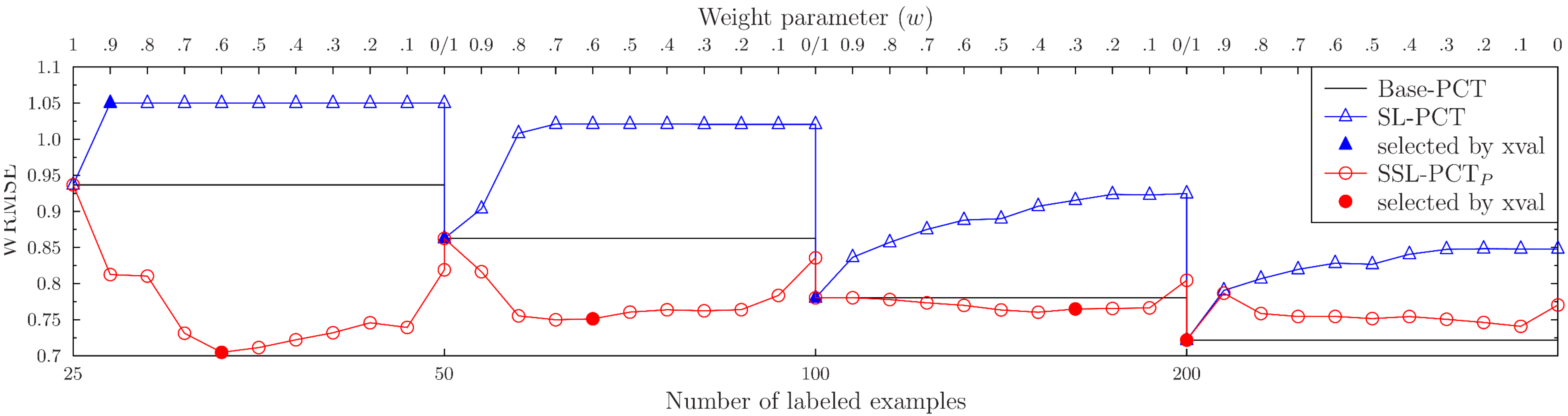
Results: Influence of the *w* parameter

RF1



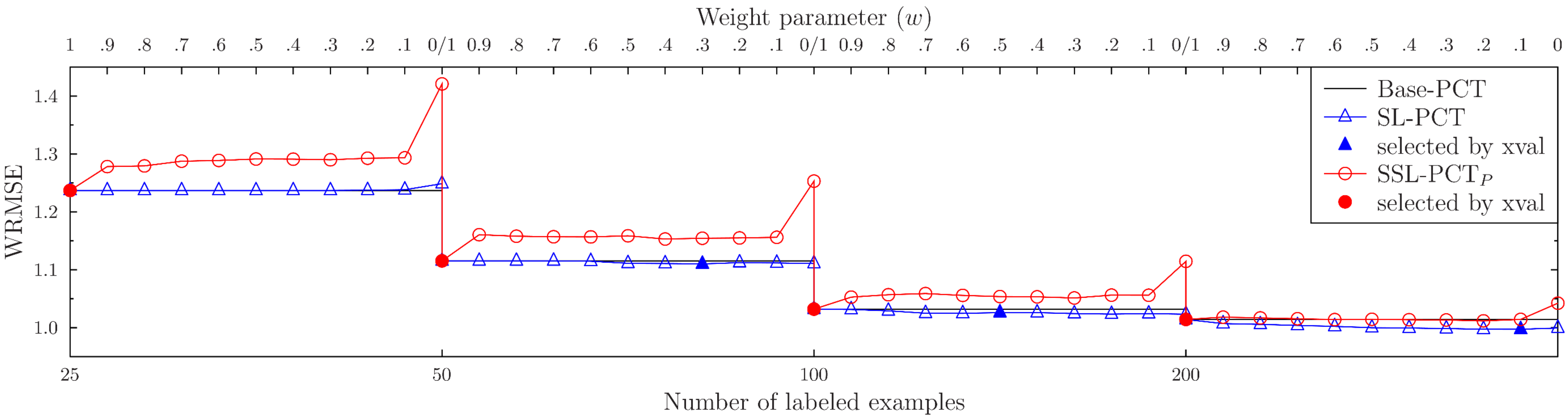
Results: Influence of the *w* parameter

SCM1D

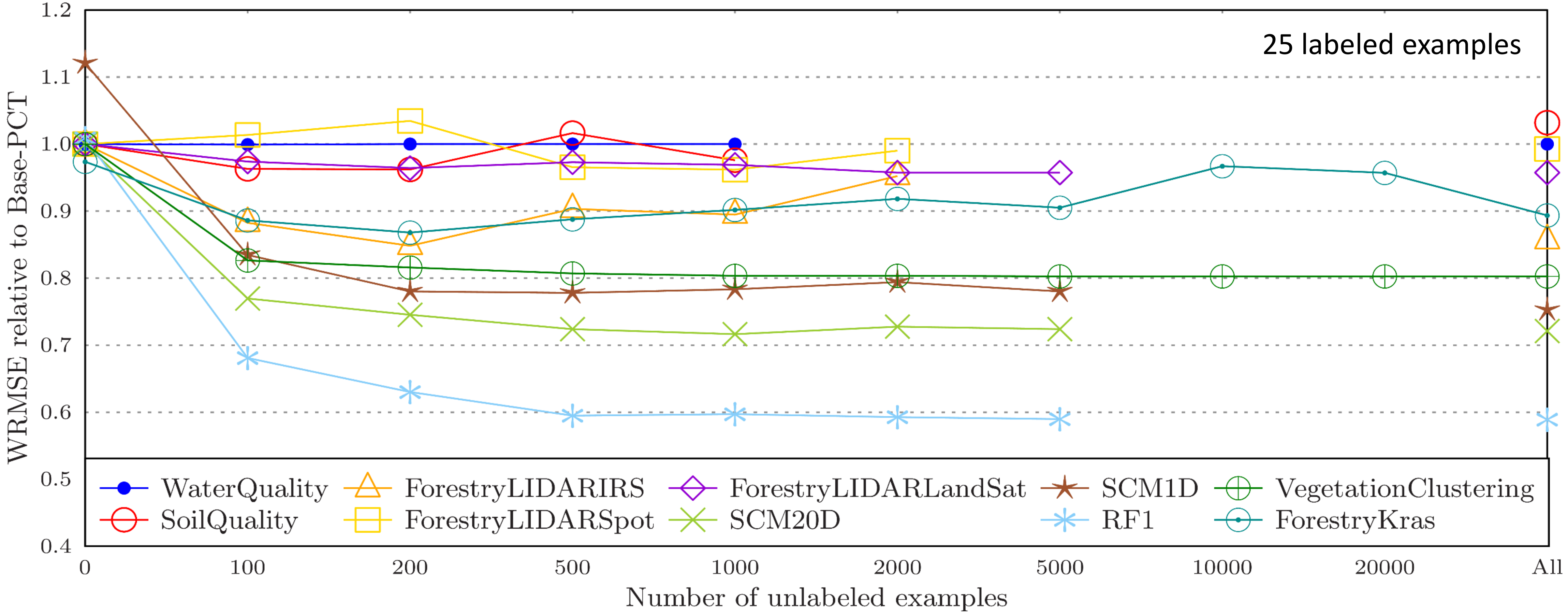


Results: Influence of the *w* parameter

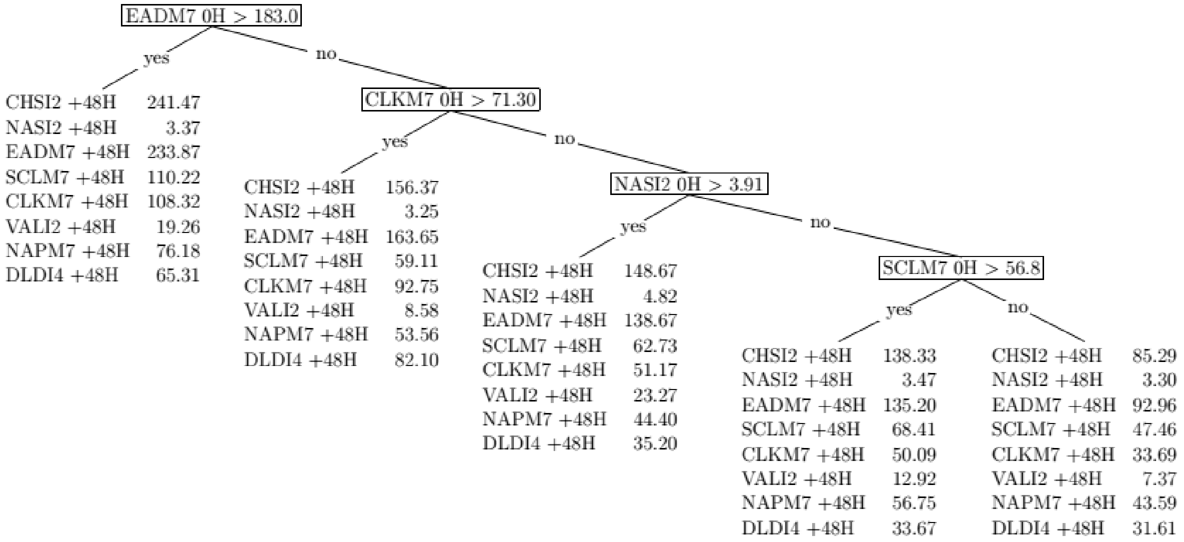
Water Quality



Results: Variable amount of unlabeled data

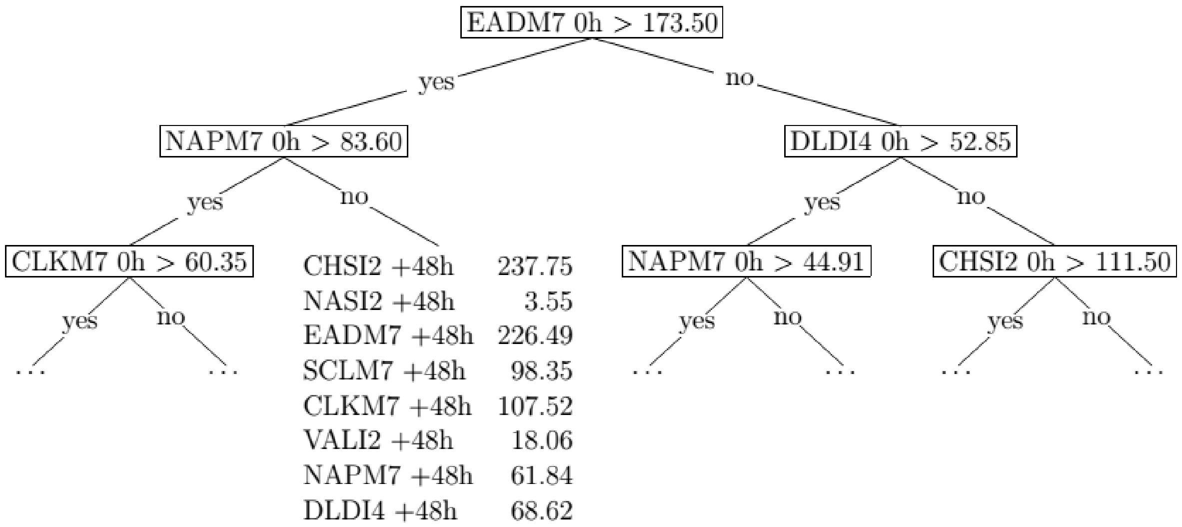


Example PCTs



WRMSE = 0.96 #Nodes = 9

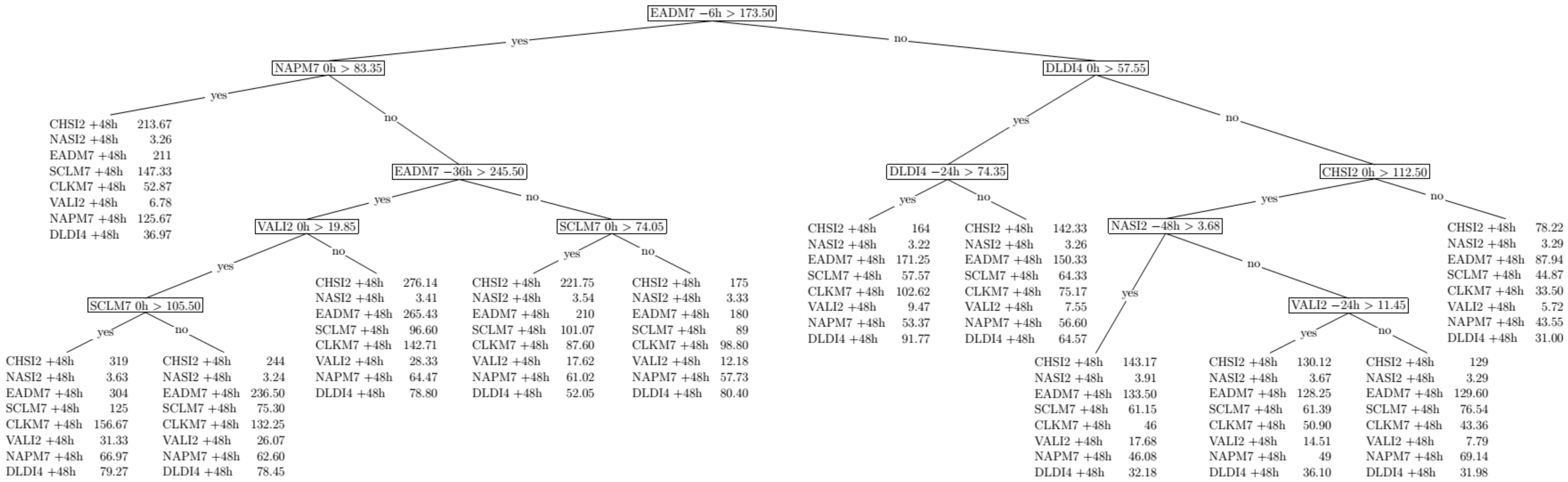
(a) Base-PCT with 50 labeled examples



WRMSE = 0.51 #Nodes = 1671

(b) Base-PCT with 9125 labeled examples

Example semi-supervised PCT



WRMSE = 0.71 #Nodes = 23

(c) SSL-PCT$_P$ with 50 labeled examples and 9075 unlabeled examples

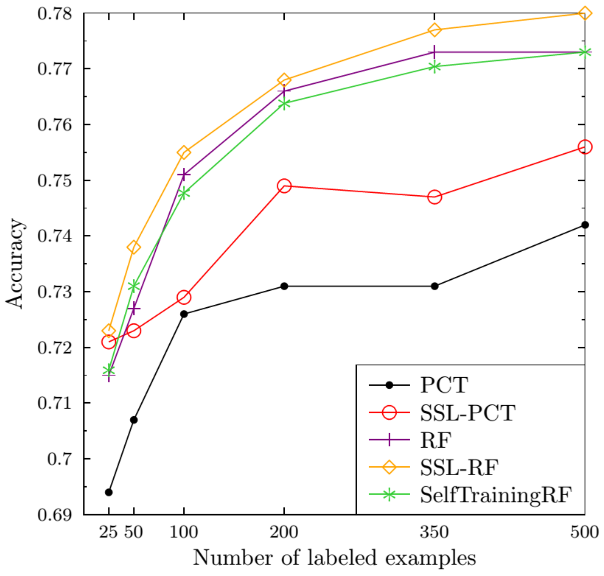
Summary

- Global and interpretable semi-supervised method for the task of multi-target regression
- Can improve the performance of supervised PCTs by a large degree
- The most effective in scenarios especially relevant for SSL
  - When few labeled examples are available
- Very seldom degenerates the performance of supervised PCTs
  - Mechanism to control the amount of influence of unlabeled examples
- The performance saturates after considering ~1000 unlabeled examples

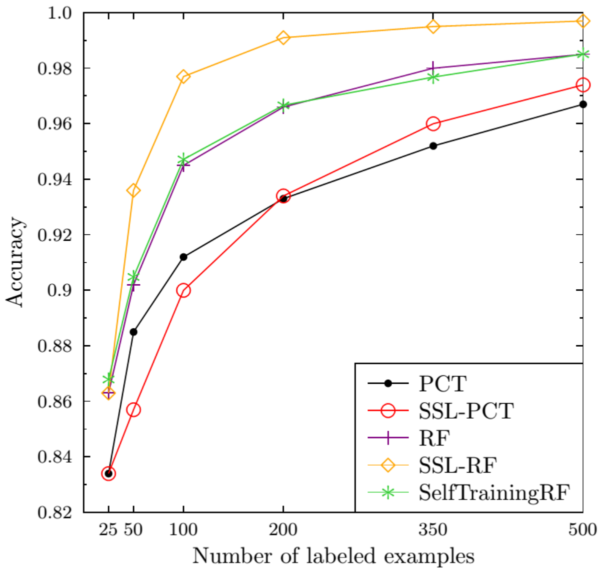
SSL PCTs for classification tasks

- Datasets
  - 12 binary classification datasets
  - 10 multi-class classification datasets
  - 14 multi-label classification datasets
- Evaluation in an ensemble setting
- We explore the influence of the amount of labeled data
  - 25, 50, 100, 200, 350 and 500 labeled examples
- Parametar w ranges from 0 (unsupervised) to 1 (supervised)
- Transductive evaluation scenario: unlabeled examples = test examples
- 10 runs with different random initialization

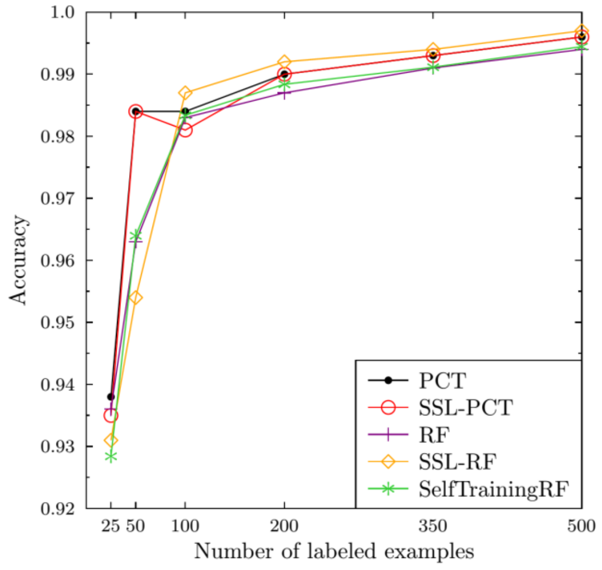
Binary classification results



a) Abalone



d) Banknote

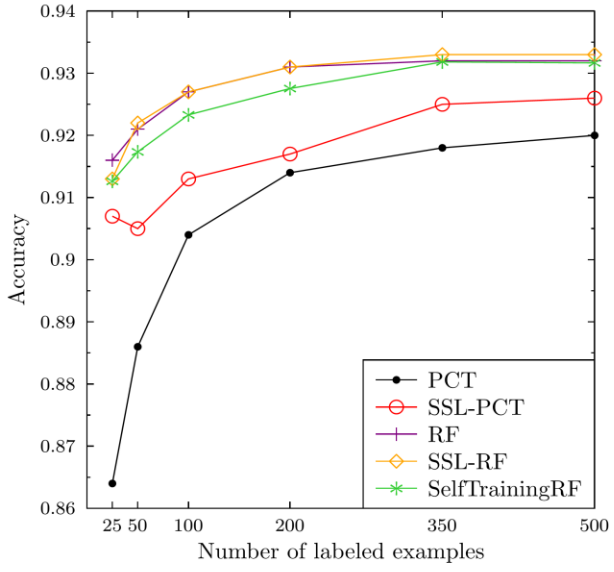


i) Mushroom

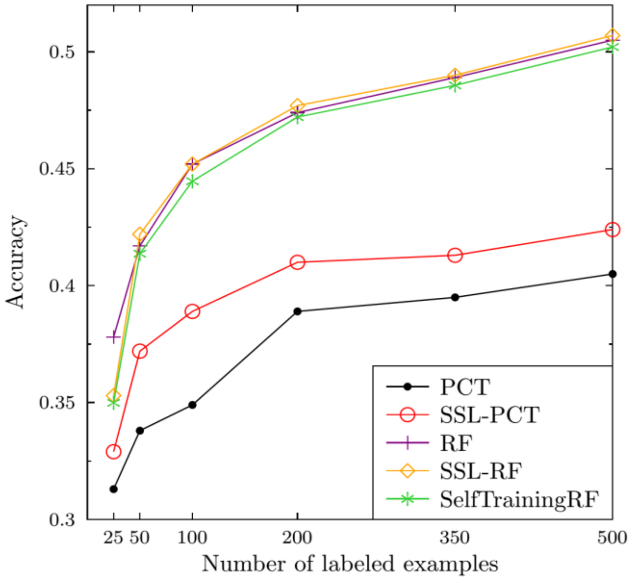
Wilcoxon test to assess the statistical significance of different performances

Methods	25	50	100	200	350	500
PCT vs. SSL-PCT	**0.009 (+)**	0.388 (+)	0.066 (+)	**0.005 (+)**	**0.019 (+)**	**0.019 (+)**
RF vs. SSL-RF	0.529 (+)	0.192 (+)	**0.002 (+)**	0.099 (+)	0.093 (+)	**0.012 (+)**
SelfTrainingRF vs. SSL-RF	**0.015 (+)**	0.072 (+)	**0.005 (+)**	**0.005 (+)**	**0.015 (+)**	**0.016 (+)**

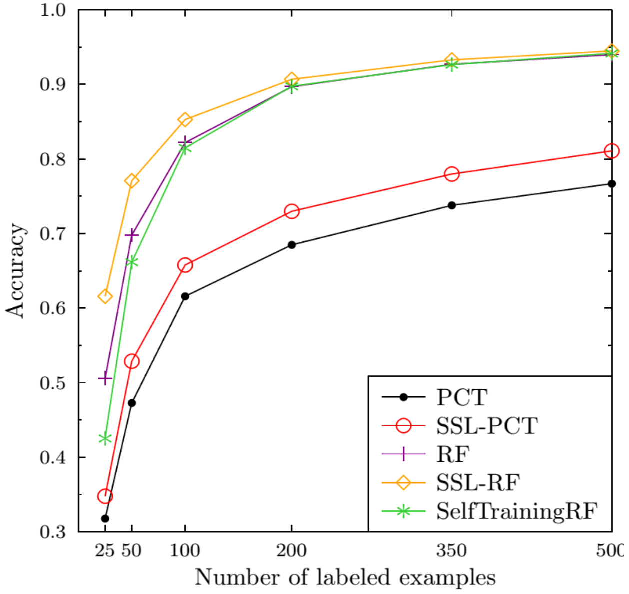
Multi-class classification results



a) Baseball



g) GesturePhase

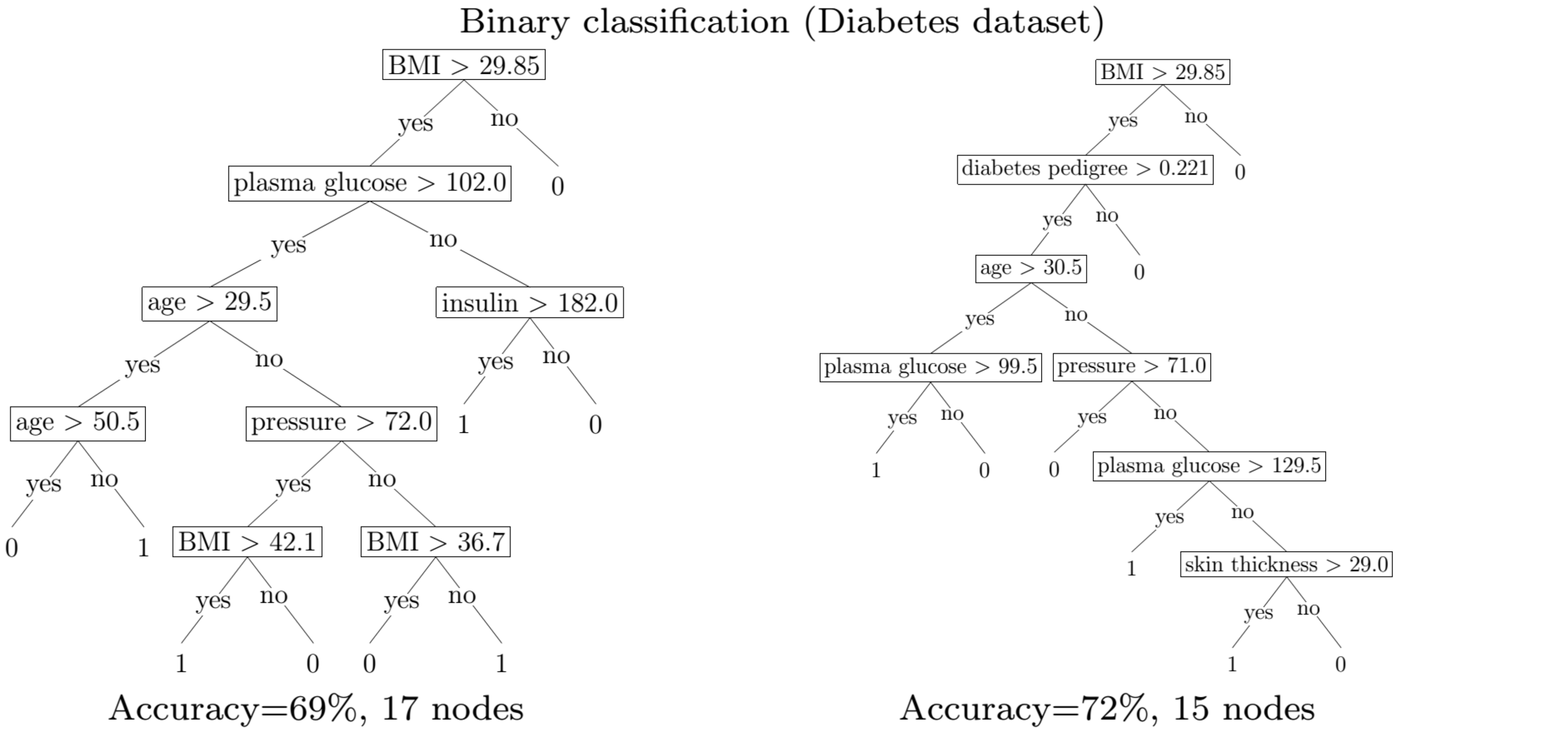


i) Optdigits

Wilcoxon test to assess the statistical significance of different performances

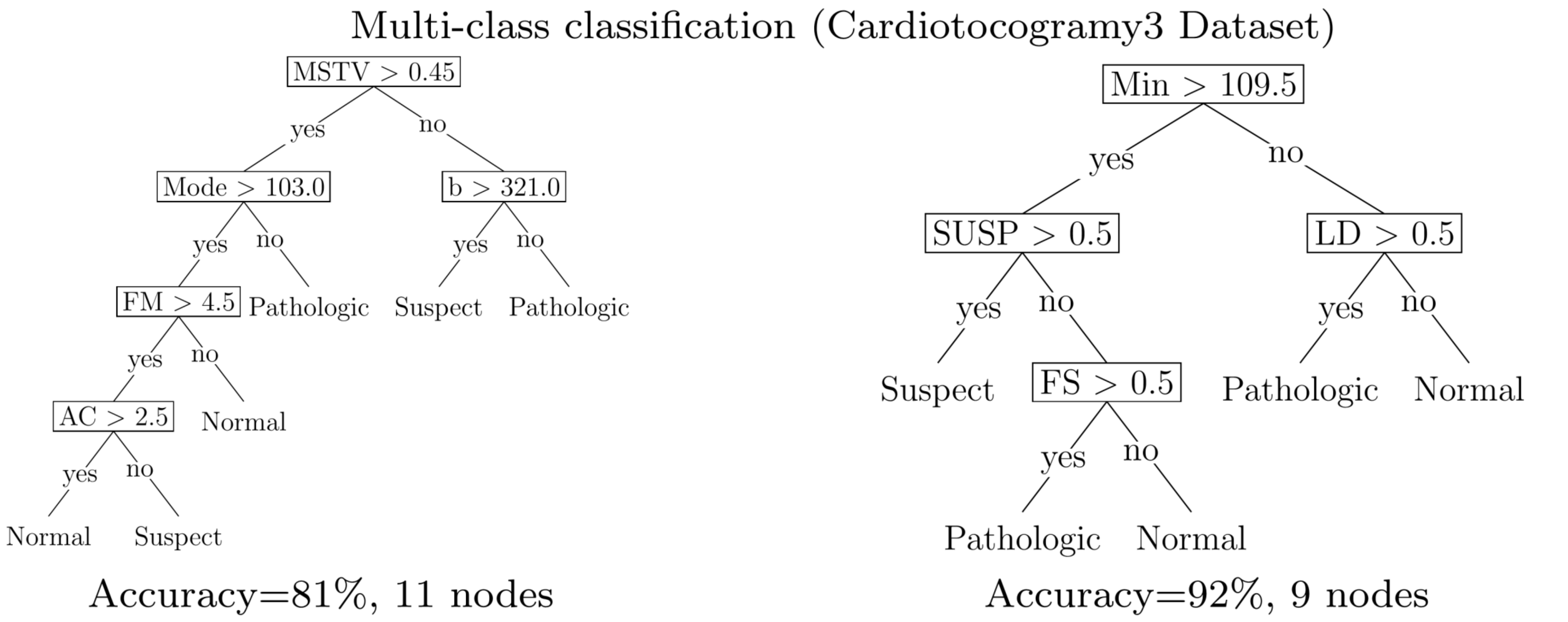
Methods	25	50	100	200	350	500
PCT vs. SSL-PCT	0.444 (+)	0.123 (+)	**0.044 (+)**	**0.019 (+)**	0.399 (+)	0.235 (+)
RF vs. SSL-RF	0.918 (+)	**0.022 (+)**	**0.019 (+)**	**0.006 (+)**	**0.005 (+)**	**0.03 (+)**
SelfTrainingRF vs. SSL-RF	**0.012 (+)**	**0.008 (+)**	**0.003 (+)**	**0.003 (+)**	**0.011 (+)**	**0.05 (+)**

Binary classification: SSL PCTs example



(a) PCT, 100 labeled examples (b) SSL-PCT, 100 labeled and 668 unlabeled examples

MC classification: SSL PCTs example



(c) PCT, 50 labeled examples

(d) SSL-PCT, 50 labeled and 2076 unlabeled examples

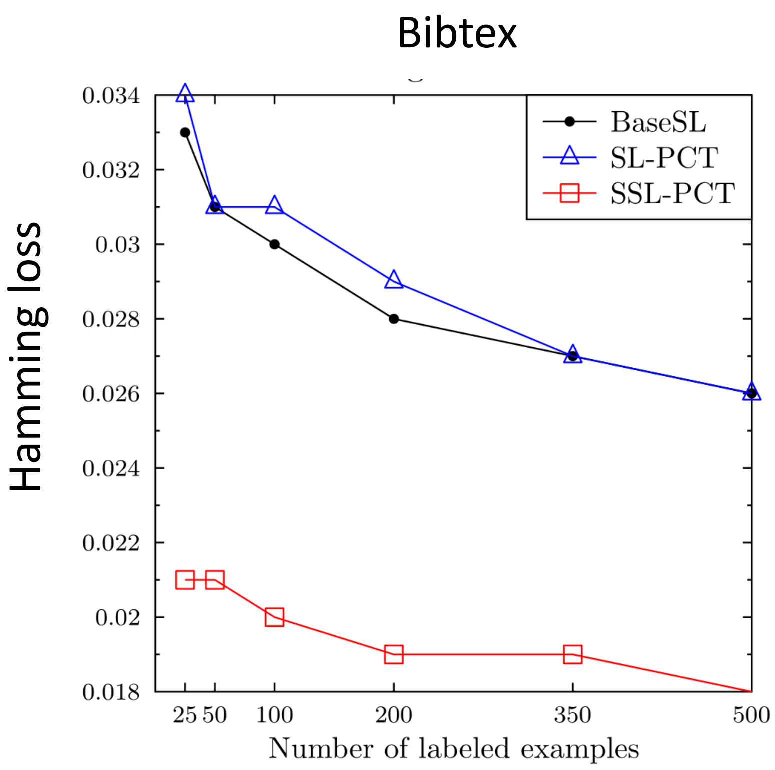
Binary/multi-class classification summary

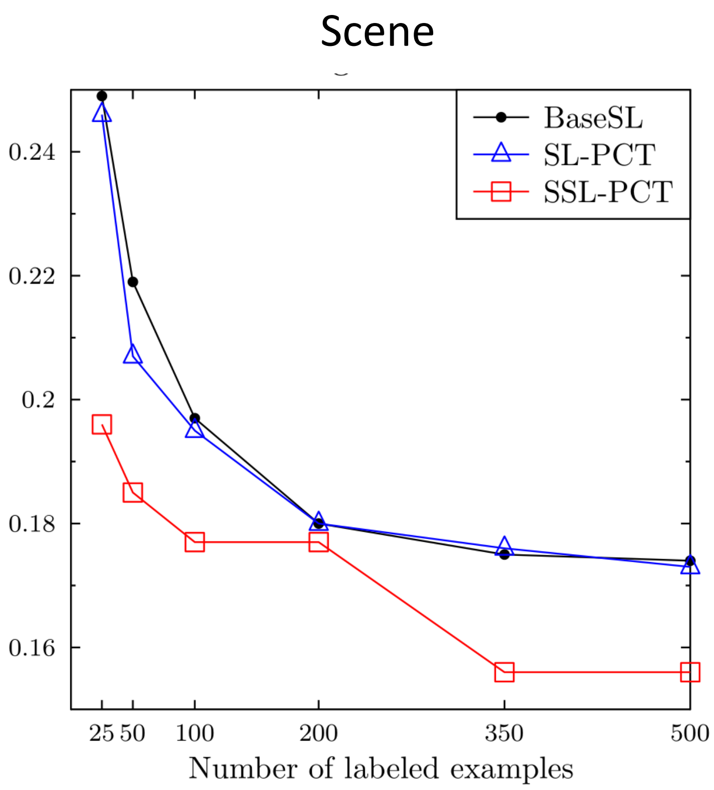
- Improvement usually doesn‘t saturate with increase in #labeled examples
  - In MTR SSL improved up to 200 labeled examples
- SSL generally does not help for „easy“ datasets (accuracy > 95%)
- The success of SSL-RF over RF is not directly connected with the success of its base model, i.e., SSL-PCTs
- Smaller interpretable models with better predictive performance

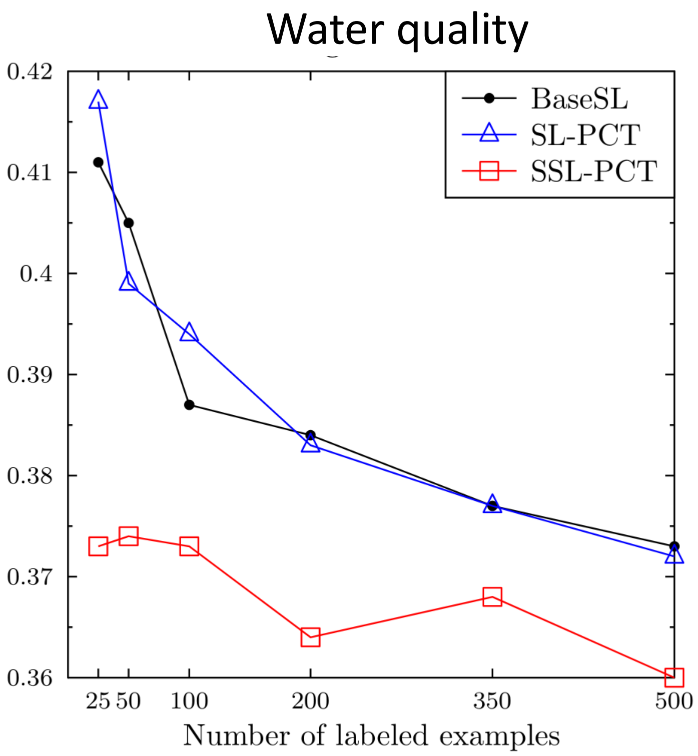
SSL PCTs for multi-label classification

- Not fully evaluated yet
- A variety of evaluation measures
  - Example based measures:Hamming loss, Accuracy, Precision, Recall, F1, Subset Accuracy
  - Label based measures: Macro{Precision, Recall, AUC}, Micro{Precision, Recall, AUC}
  - Ranking based measures: One error, Coverage, Ranking Loss, Average Precision

Sample results: multi-label classification







Conclusions

- Global semi-supervised method for multiple tasks
  - Multi-target regression
  - Binary, Multi-class and Multi-label classification
- Can improve the performance of supervised PCTs by a large degree
  - Especially when few labeled examples are available
- Very seldom degenerates the performance of supervised PCTs
  - Mechanism to control the amount of influence of unlabeled examples
- Easily interpretable models

Further work

- Consider additional tasks
  - Hierarchical multi-label classification, time series prediction
- Unsupervised learning/clustering for datasets with mixed variables
- Learning from partially labeled data
  - Two small case studies already performed
- Feature ranking in various settings
  - Unsupervised learning
  - Semi-supervised learning
  - Partially labeled