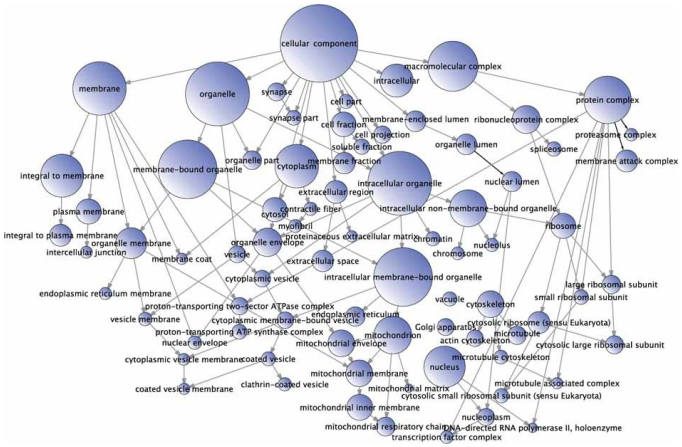
Ensembles for predicting structured outputs

Dragi Kocev

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

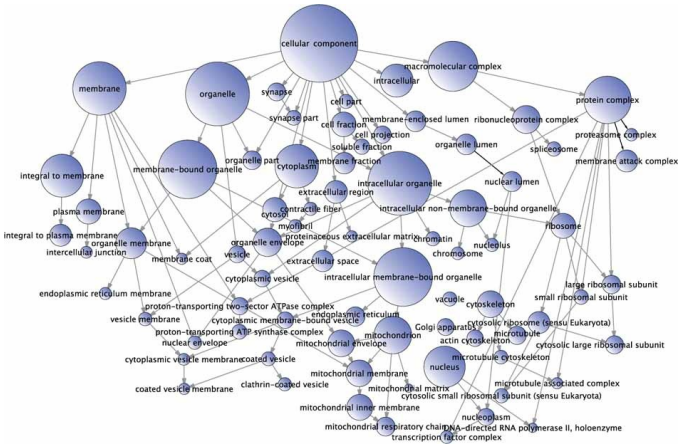
Motivation

- Increasing amounts of structured data
  - Vectors
  - Hierarchies – trees, DAGs,...
  - Sequences – time series



Motivation

- Increasing amounts of structured data
  - Vectors
  - Hierarchies – trees, DAGs,...
  - Sequences – time series



- Success of ensemble methods in simple classification and regression

Outline

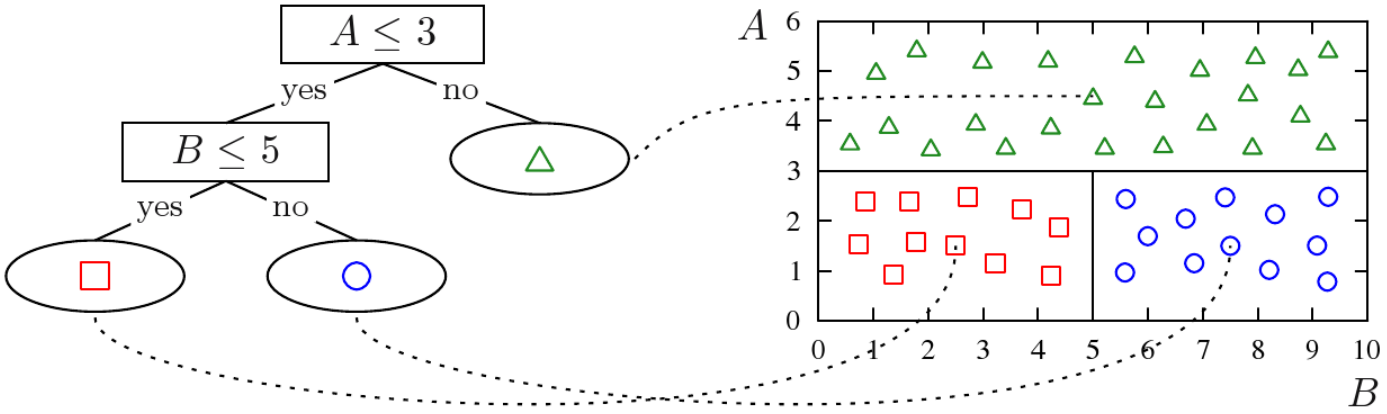
- Background
  - Structured outputs
  - Predictive Clustering Trees (PCTs)
  - PCTs for HMLC
- Ensembles of PCTs
- Experimental evaluation
- Application in functional genomics
- Conclusions

Structured outputs

- Target in supervised learning
  - Single discrete or continuous variable
- Target in structured prediction
  - Vector of discrete or continuous variables
  - Hierarchy – tree or DAG
  - Sequence – time series
- Solutions
  - De-composition to simpler problems
  - Exploitation of the structure

Predictive Clustering Trees

- Standard Top-Down Induction of DTs
- Hierarchy of clusters
- Distance measure: minimization of intra-cluster variance
- Instantiation of the variance for different tasks



CLUS

- System where the PCTs framework is implemented (KULeuven & JSI)

procedure PCT(I) **returns** tree

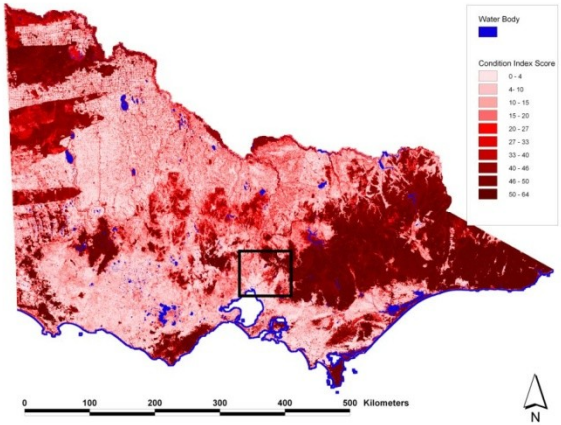
1: $(t^*, \mathcal{P}^*) = \text{BestTest}(I)$
2: **if** $t^* \neq none$ **then**
3: **for each** $I_k \in \mathcal{P}^*$ **do**
4: $tree_k = \text{PCT}(I_k)$
5: **return** node$(t^*, \bigcup_k \{tree_k\})$
6: **else**
7: **return** leaf(Prototype(I))

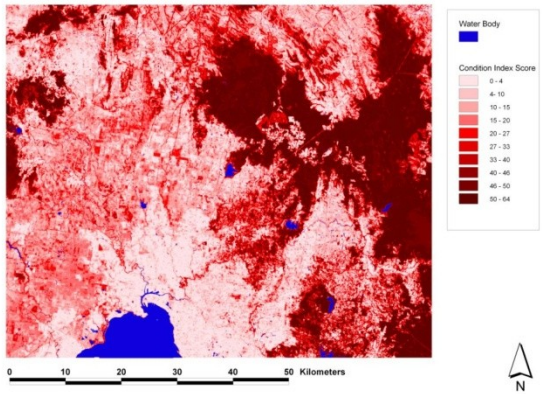
procedure BestTest(I)

1: $(t^*, h^*, \mathcal{P}^*) = (none, 0, \emptyset)$
2: **for each** possible test t **do**
3: $\mathcal{P} =$ partition induced by t on I
4: $h = \text{Var}(I) - \sum_{I_k \in \mathcal{P}} \frac{|I_k|}{|I|} \text{Var}(I_k)$
5: **if** $(h > h^*) \wedge \text{Acceptable}(t, \mathcal{P})$ **then**
6: $(t^*, h^*, \mathcal{P}^*) = (t, h, \mathcal{P})$
7: **return** $(t^*, \mathcal{P}^*)$

PCTs – Multiple numeric targets

- Condition of vegetation in Victoria, Australia
- Habitat Hectares Index
  - Large Trees, Tree Canopy Cover, Understorey, Litter, Logs, Weeds and Recruitment





- Euclidean distance $$Var(E) = \sum Var(E, y_t)$$

Kocev et al., Ecological Modelling 220(8):1159-1168, 2009

PCTs – Multiple discrete targets

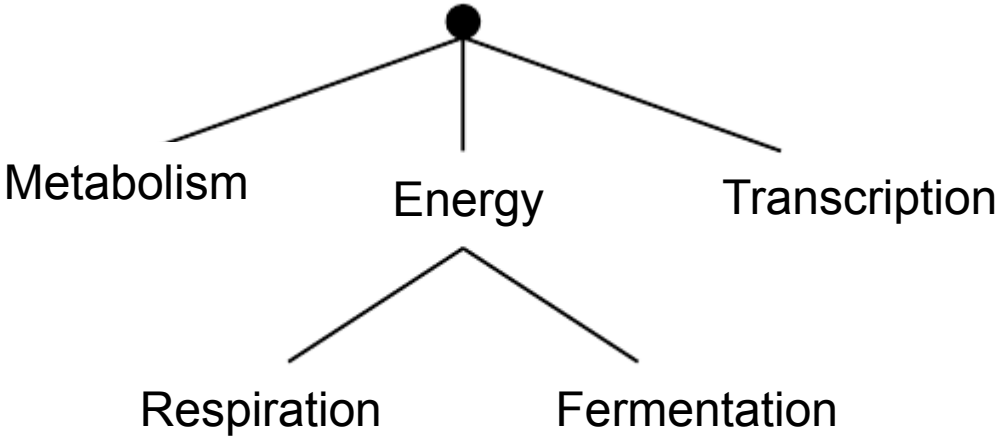
- Mediana – Slovenian media space
  - 8000 Questionnaires about reading and TV habits and life style of Slovenians
- What type of people read:
  - Delo, Dnevnik, Ekipa, Slovenske Novice, Večer

$$Var(E) = \sum Entropy(E, y_t)$$

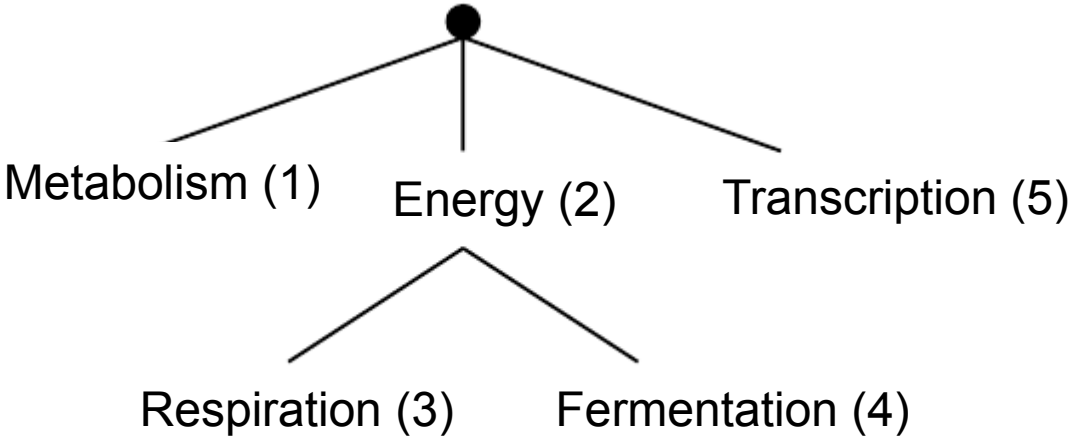
$$Var(E) = \sum Gini(E, y_t)$$

Skranjc et al., Informatica 25(3):357-363, 2001

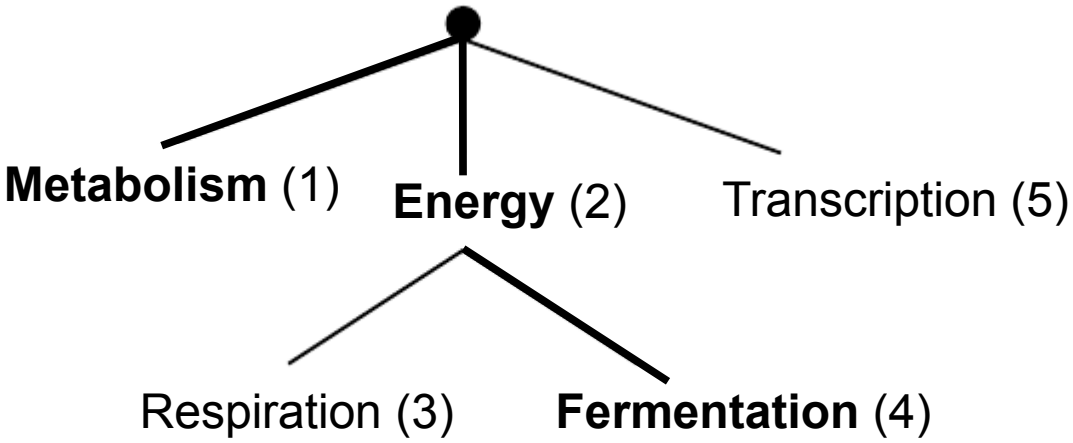
PCTs - HMLC


Metabolism
Energy
Transcription
Respiration
Fermentation

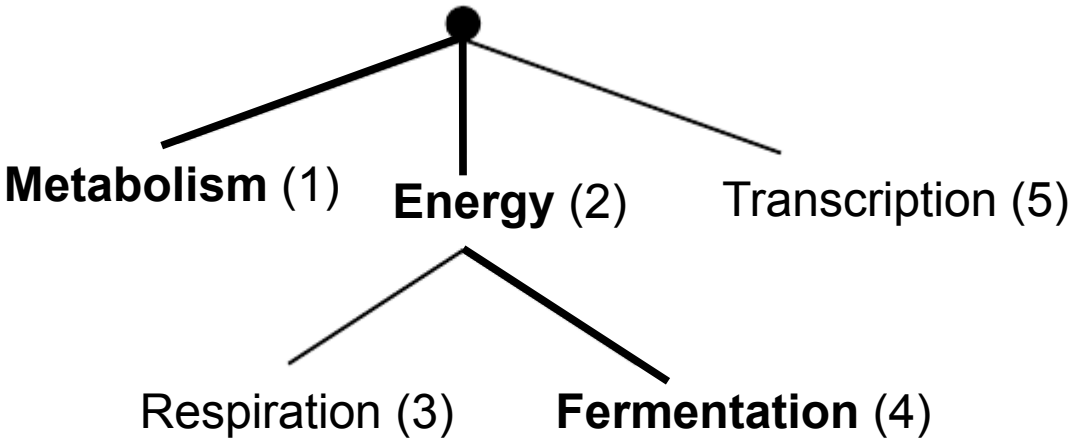
PCTs - HMLC


Metabolism (1)
Energy (2)
Transcription (5)
Respiration (3)
Fermentation (4)

PCTs - HMLC


Metabolism (1)
Energy (2)
Transcription (5)
Respiration (3)
Fermentation (4)

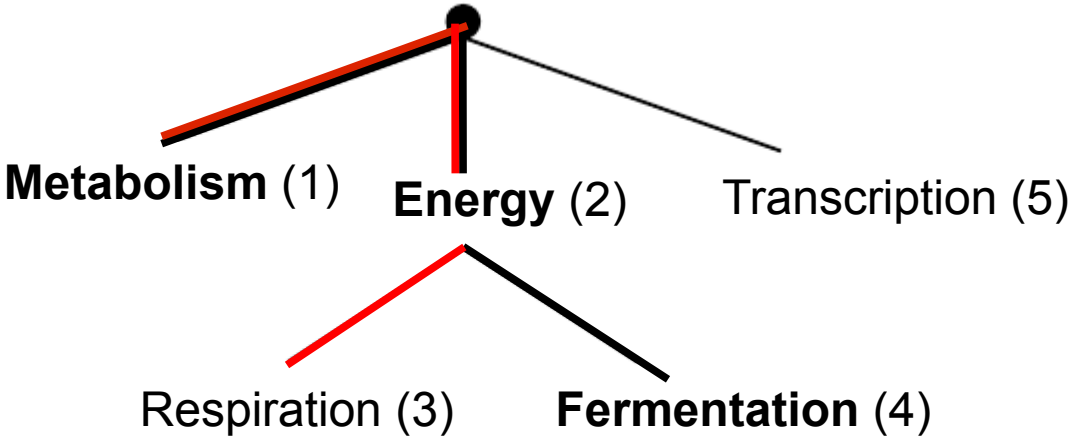
PCTs - HMLC



$$(1)(2)(3)(4)(5)$$

$$v_i = [1, 1, 0, 1, 0]$$

PCTs - HMLC

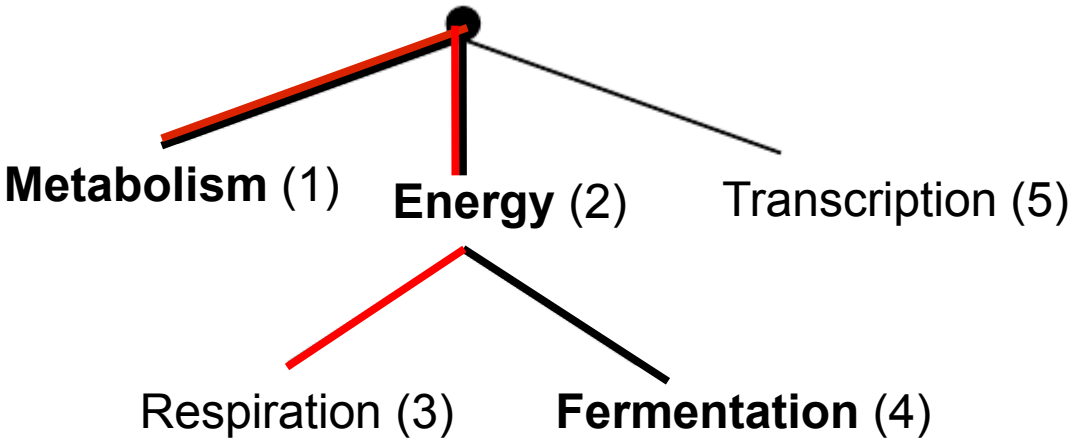


$$(1)(2)(3)(4)(5)$$

$$v_i = [1,1,0,1,0]$$

$$v_i = [1,1,1,0,0]$$

PCTs - HMLC



$$(1)(2)(3)(4)(5)$$

$$v_i = [1,1,0,1,0]$$

$$v_i = [1,1,1,0,0]$$

$$Var(E) = \frac{\sum d(v_i, \hat{v})^2}{|E|}$$

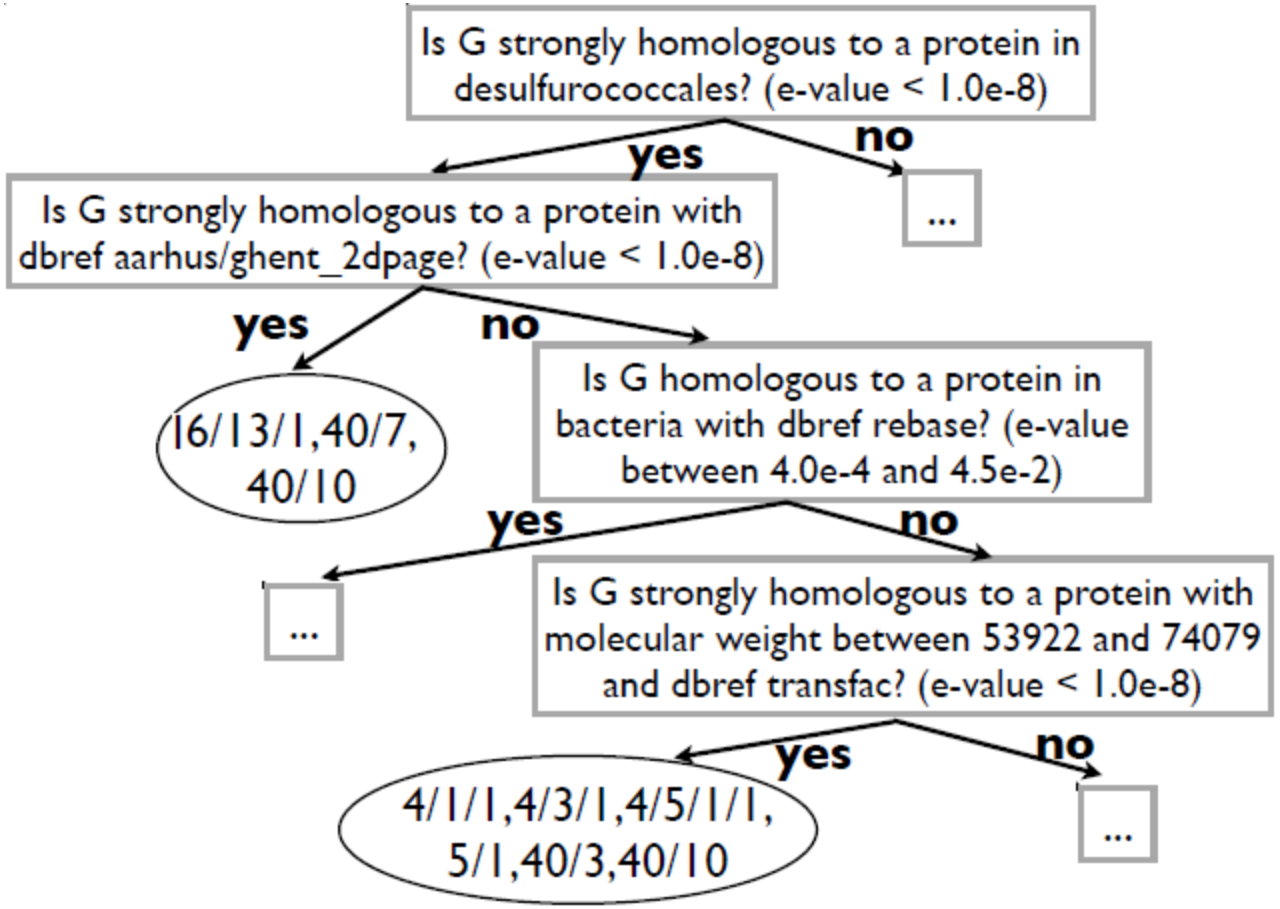
$$d(v_i, \hat{v}) = \sqrt{\sum \omega(c_i) \cdot (v_{1,i} - v_{2,i})^2}$$

$$\omega(c_i) = \omega_0 \cdot \omega(par(c_i))$$

Vens et al., Machine Learning 73(2):185-214, 2008

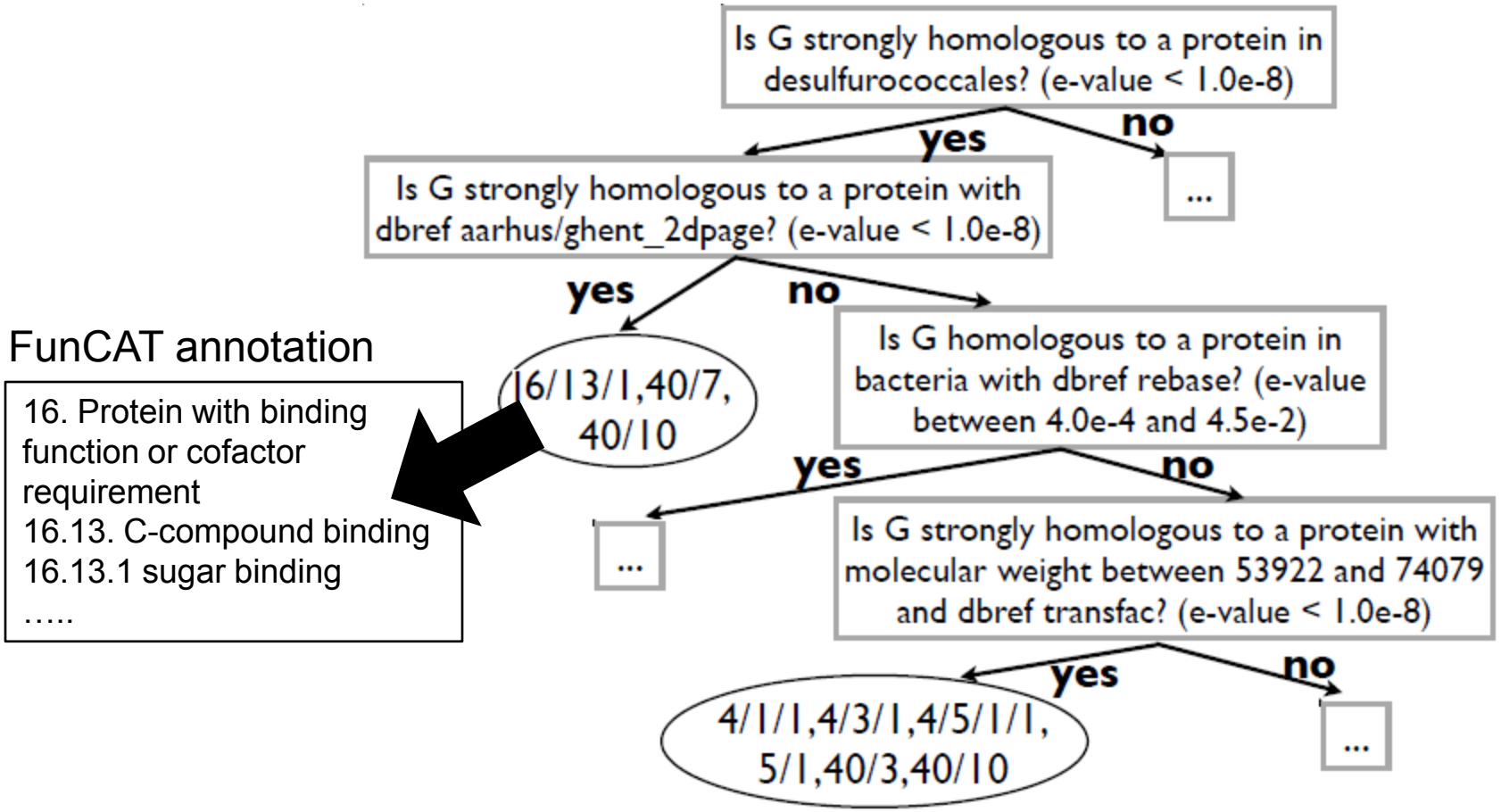
PCTs - HMLC

- A leaf stores the mean label: proportion of the examples belonging to each class
- Prediction is made by using a user-defined threshold



PCTs - HMLC

- A leaf stores the mean label: proportion of the examples belonging to each class
- Prediction is made by using a user-defined threshold



Outline

- Background
  - Structured outputs
  - Predictive Clustering Trees (PCTs)
  - PCTs for HMLC
- Ensembles of PCTs
- Experimental evaluation
- Application in functional genomics
- Conclusions

Ensemble Methods

- Ensembles are a set of predictive models
  - Unstable base classifiers
- Voting schemes to combine the predictions into a single prediction
- Ensemble learning approaches
  - Modification of the data
  - Modification of the algorithm

Ensemble Methods

- Ensembles are a set of predictive models
  - Unstable base classifiers
- Voting schemes to combine the predictions into a single prediction
- Ensemble learning approaches
  - Modification of the data ← **Bagging**
  - Modification of the algorithm

} **Random Forest**

Ensemble Methods: Algorithm

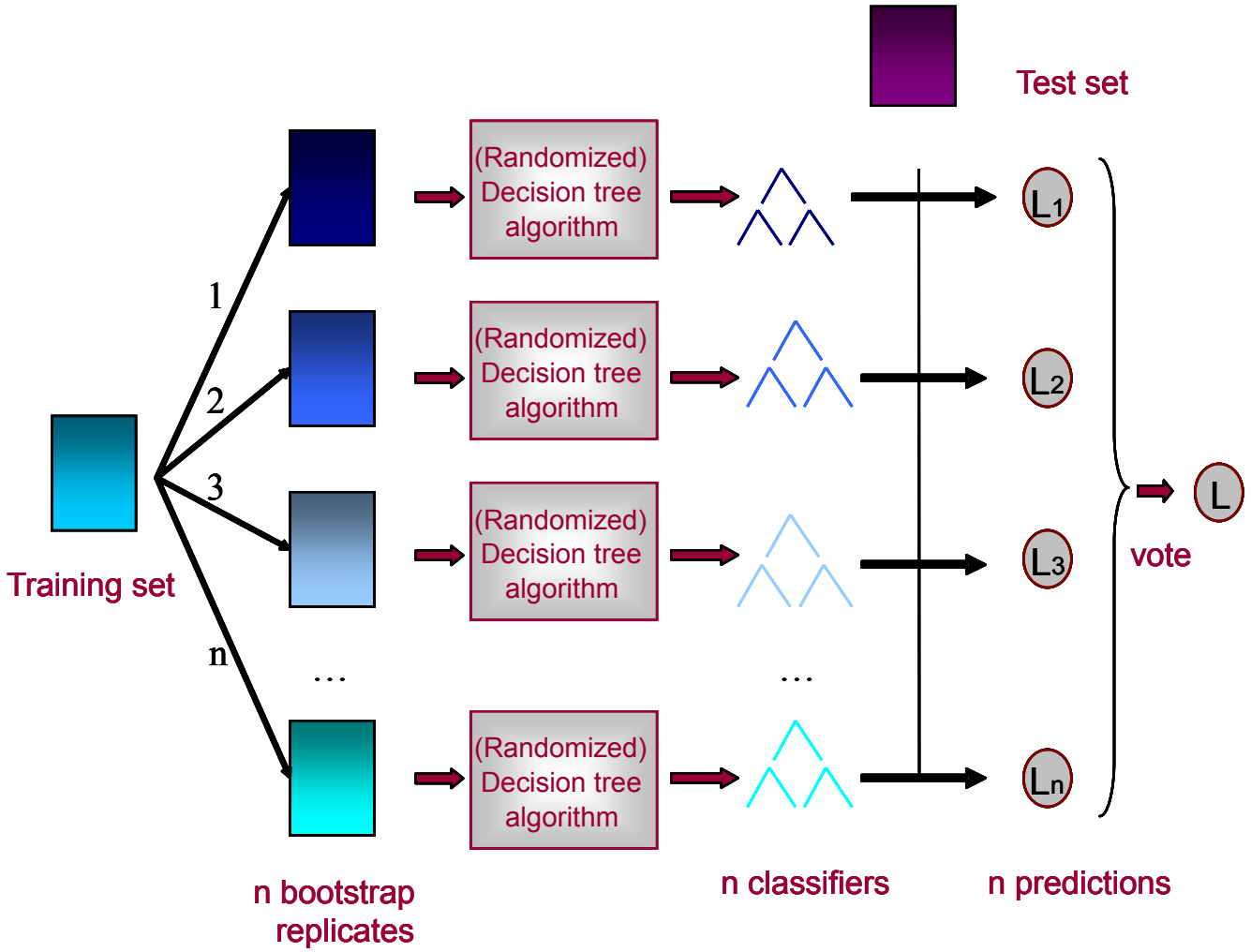


Image from van Assche, PhD Thesis, 2008

Ensemble Methods: Algorithm

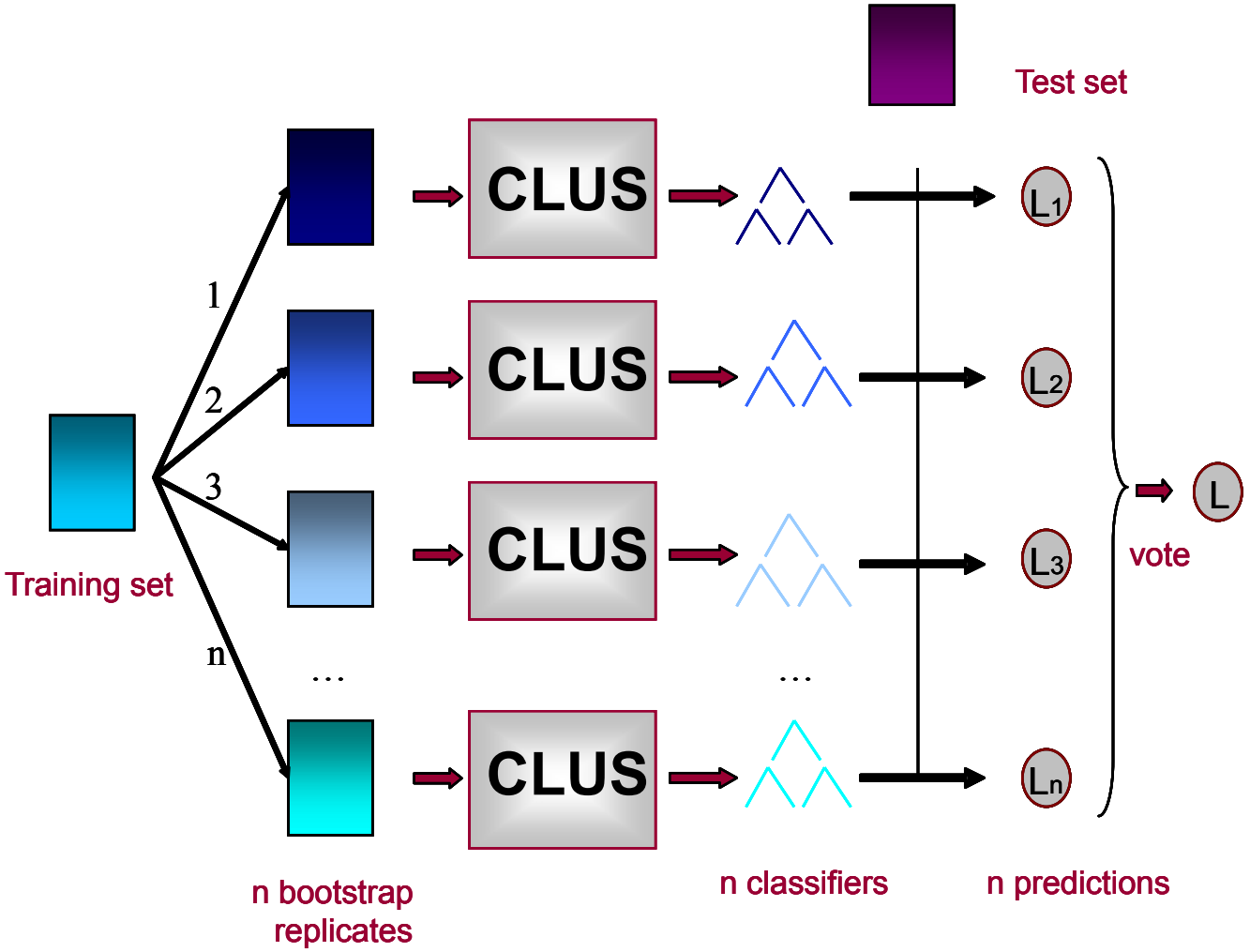


Image from van Assche, PhD Thesis, 2008

Ensembles for structured outputs

- PCTs as base classifiers
- Voting schemes for the structured outputs
  - MT Classification: majority and probability distribution vote
  - MT Regression and HMLC: average
  - For an arbitrary structure: prototype calculation function
- Memory efficient implementation

Outline

- Background
  - Structured outputs
  - Predictive Clustering Trees (PCTs)
  - PCTs for HMLC
- Ensembles of PCTs
- Experimental evaluation
- Application in functional genomics
- Conclusions

Experimental hypotheses

- How many base classifiers are enough?
- Do ensembles of PCTs lift the predictive performance of a single PCT?
- Are ensembles of MT/HMLC PCTs better than ensembles for each target/class separately?
- Which approach is more efficient in terms of time and size of the models?

Datasets

	Datasets	Examples	Descriptive attributes	Size of Output
MT Regression	14	154..60607	4..160	2..14
MT Classification	11	154..10368	4..294	2..14
HMLC	10	988..10000	80..74435	36..571

- Datasets with multiple targets
  - Mainly environmental data acquired in EU and Slovenian projects
- HMLC
  - Image classification
  - Text classification
  - Functional genomics

Experimental design

- Types of models
  - PCTs and ST trees (with F-test pruning)
  - Ensembles of PCTs and ensembles of ST trees
- PCTs for HMLC weight for the distance - 0.75
- Number of base classifiers (unpruned)
  - Classification: 10, 25, 50, 75, 100, 250, 500, 1000
  - Regression: 10, 25, 50, 75, 100, 150, 250
  - HMLC: 10, 25, 50, 75, 100

Experimental design (ctd.)

- Random forest – feature subset size
  - Multiple Targets: log
  - HMLC: 10%

- 10-fold cross-validation
- MT Classification - Accuracy
- MT Regression - correlation coefficient, RMSE, RRMSE
- HMLC - Precision-Recall (PR) curves, Area under PRCs

- Friedman and Nemenyi statistical tests

Precision-Recall Curves

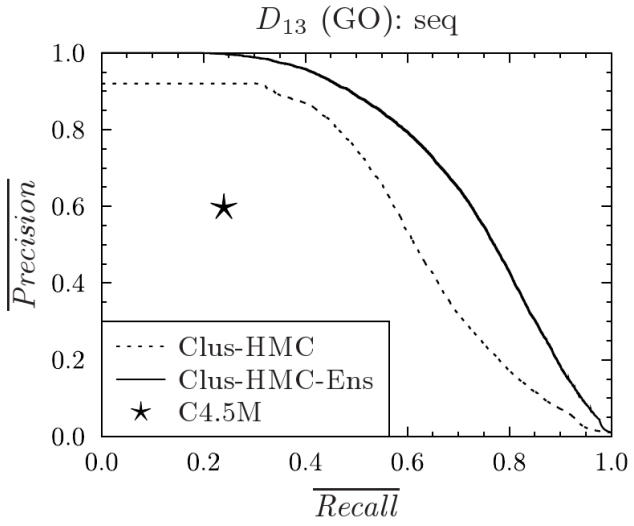
$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

- PR curve plots Precision as a function of the Recall
- Combination of the curves per class
  - Micro-averaging: Area under the average PRC
  - Macro-averaging: Average area under the PRCs

Precision-Recall Curves

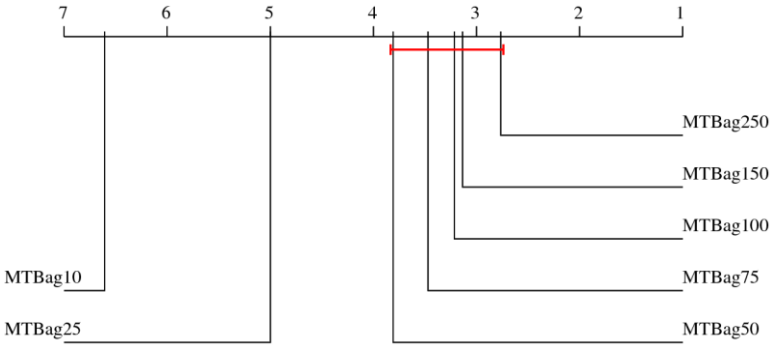
$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

- PR curve plots Precision as a function of the Recall
- Combination of the curves per class
  - Micro-averaging: Area under the average PRC
  - Macro-averaging: Average area under the PRCs

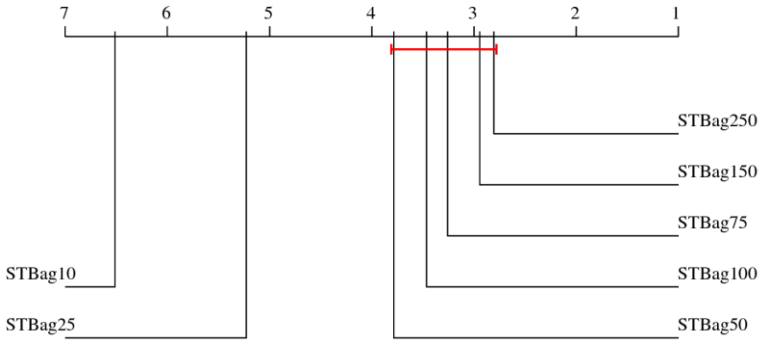


Results – Regression (RRMSE)

MT Bagging

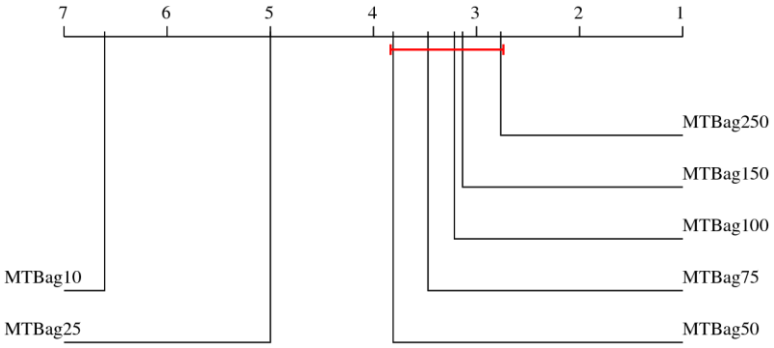


ST Bagging

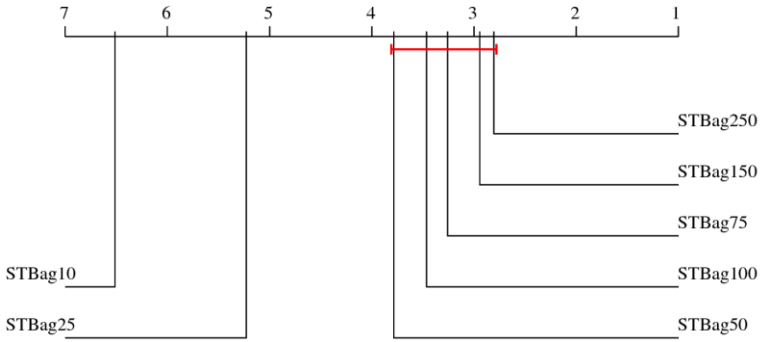


Results – Regression (RRMSE)

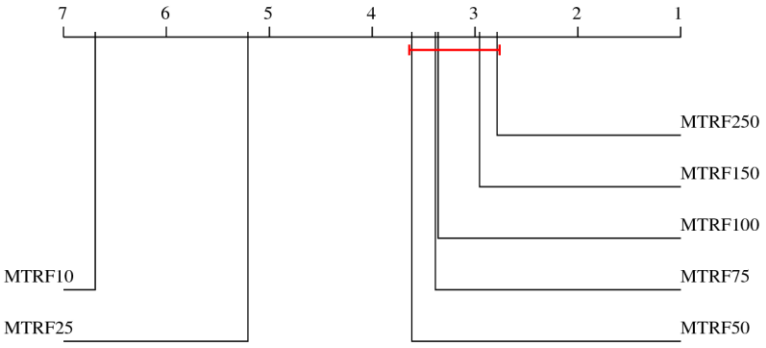
MT Bagging



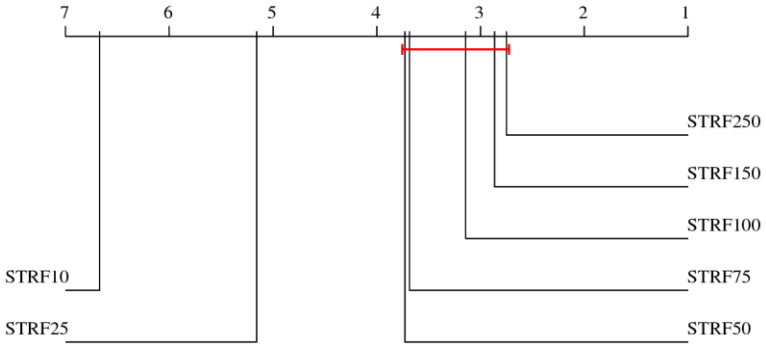
ST Bagging



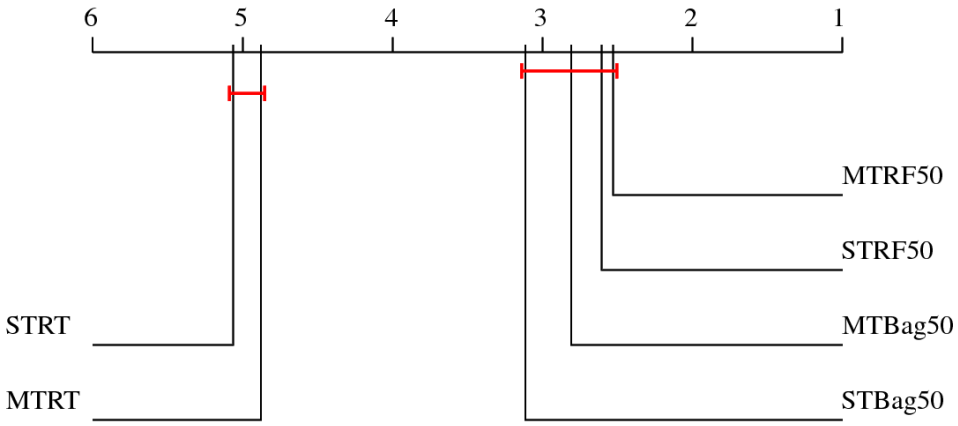
MT Random forest



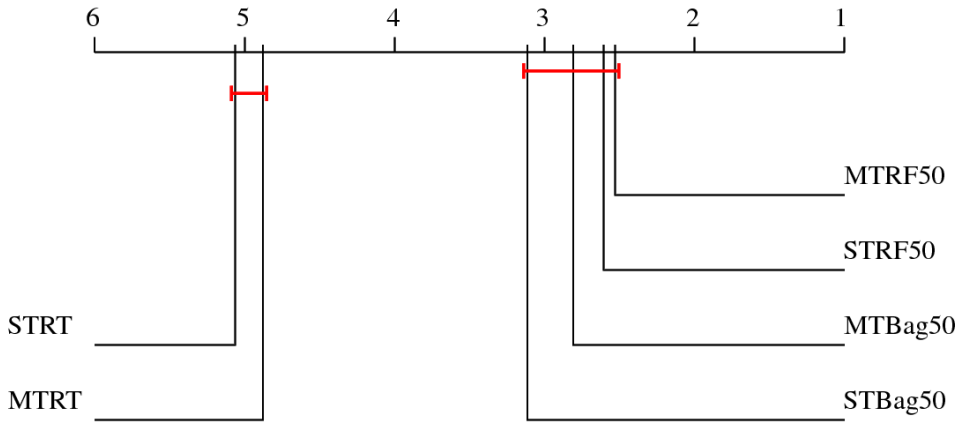
ST Random forest



Results – Regression (RRMSE)



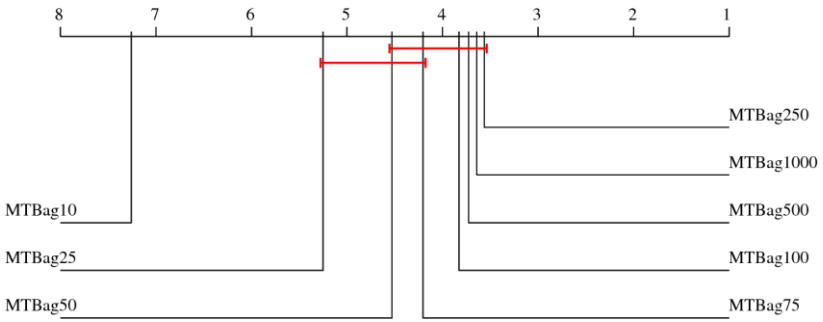
Results – Regression (RRMSE)



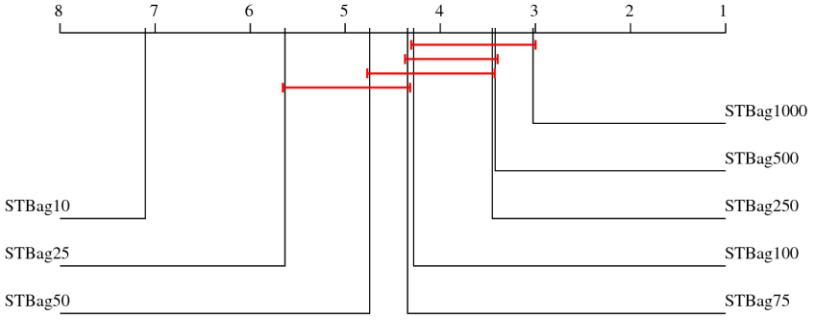
- MT Ensembles perform better than
  - single MT tree (stat. sign.)
  - ST Ensembles
  - faster to learn than ST ensembles (stat. sign.)
  - smaller models than ST ensembles (stat. sign.)

Results – Classification

MT Bagging

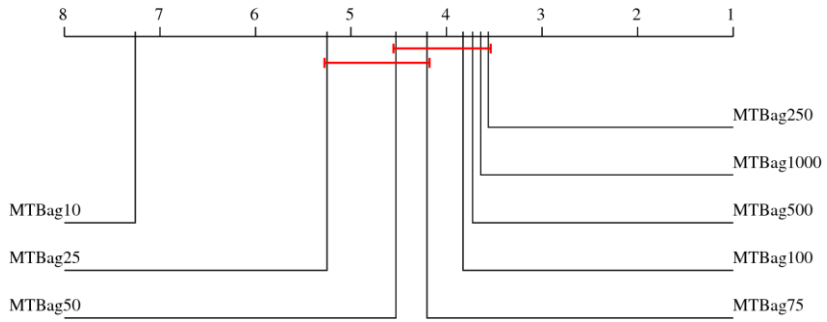


ST Bagging

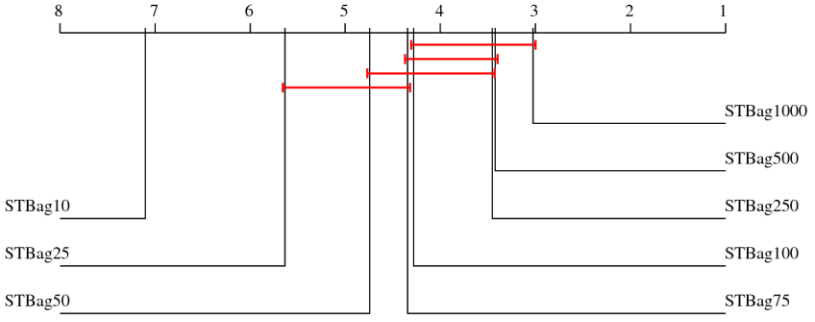


Results – Classification

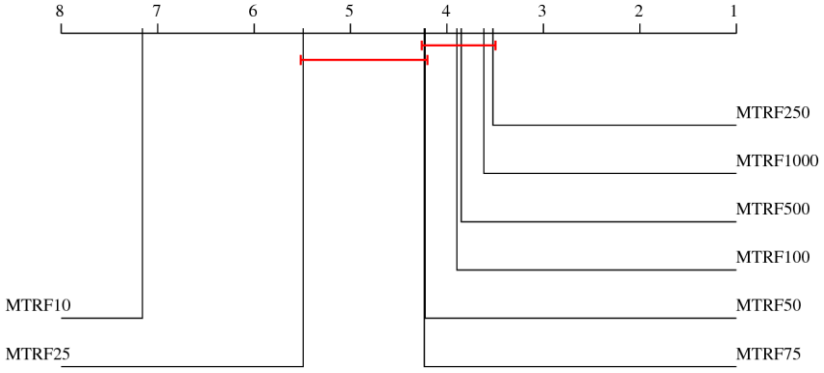
MT Bagging



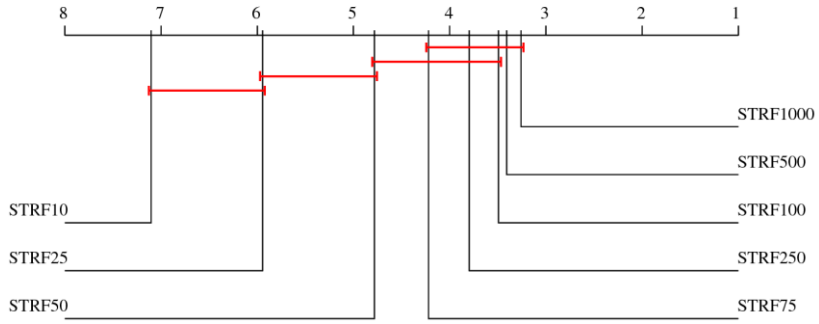
ST Bagging



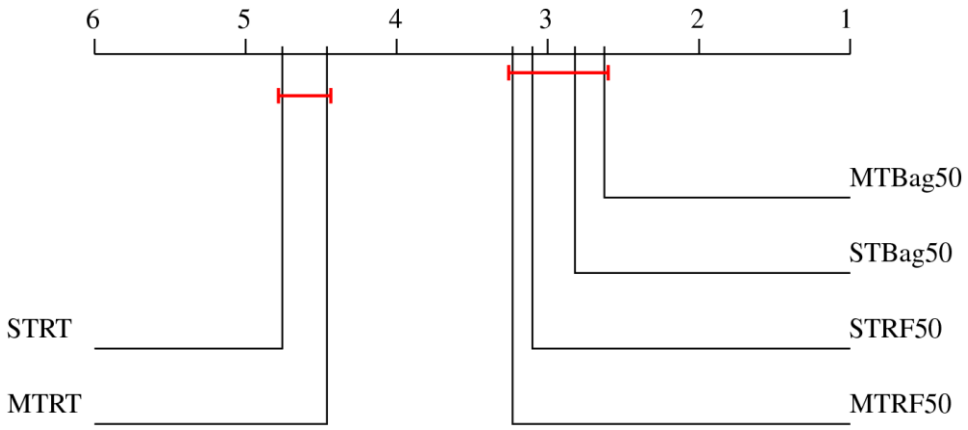
MT Random forest



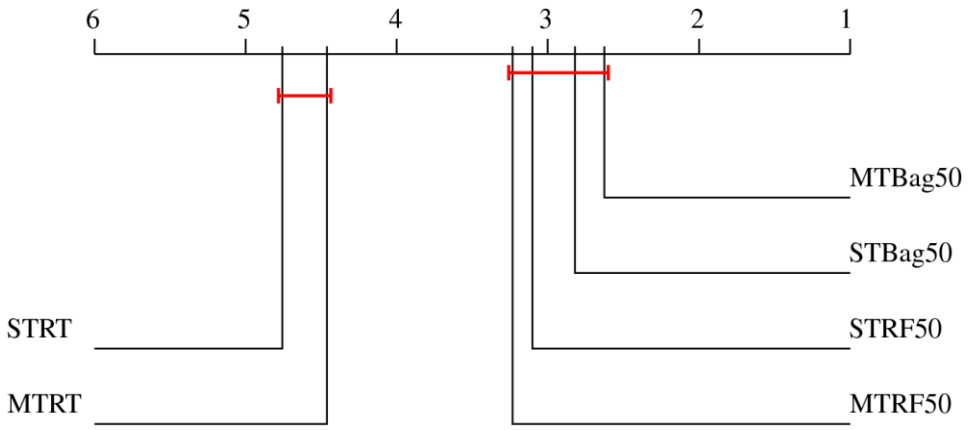
ST Random forest



Results – Classification



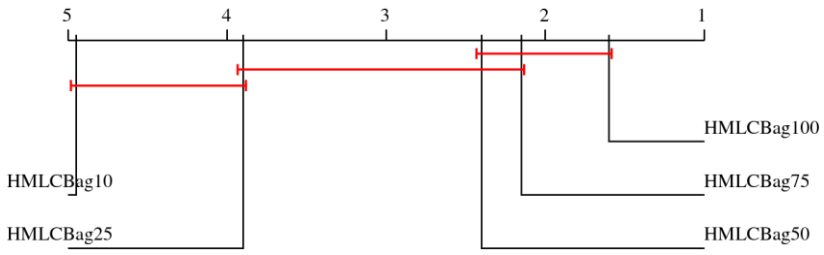
Results – Classification



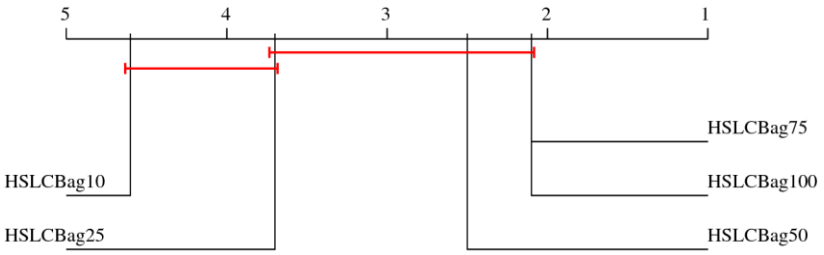
- MT Ensembles perform better than
  - single MT tree (stat. sign.)
  - faster to learn than ST ensembles (stat. sign.)
  - smaller models than ST ensembles (stat. sign.)
- MT Bagging is better than ST Bagging, while MT random forest is worse than ST random forest

Results – HMLC

HMLC Bagging

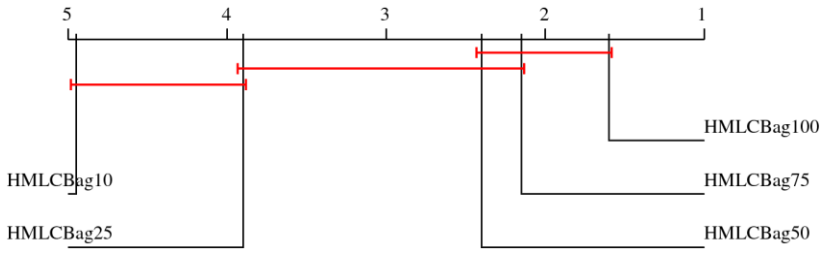


HSLC Bagging

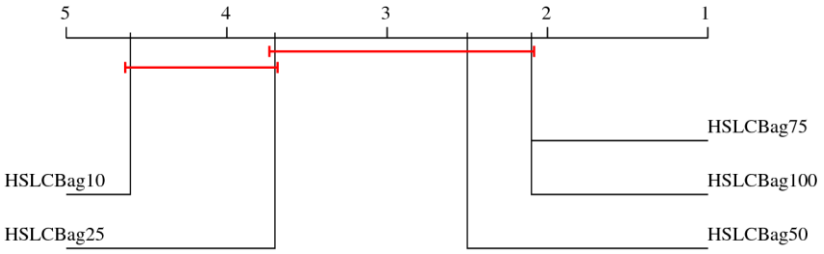


Results – HMLC

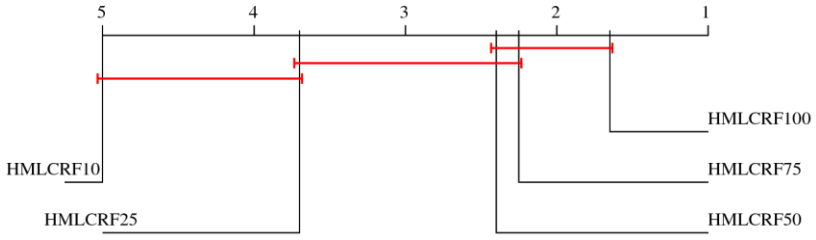
HMLC Bagging



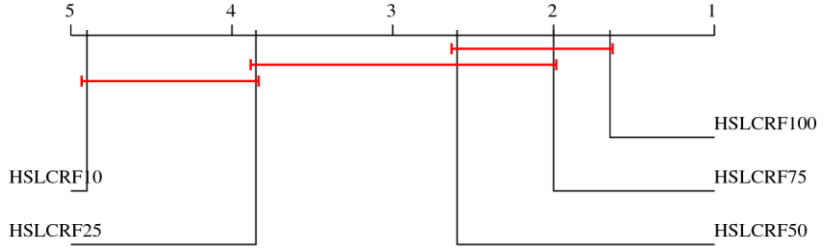
HSLC Bagging



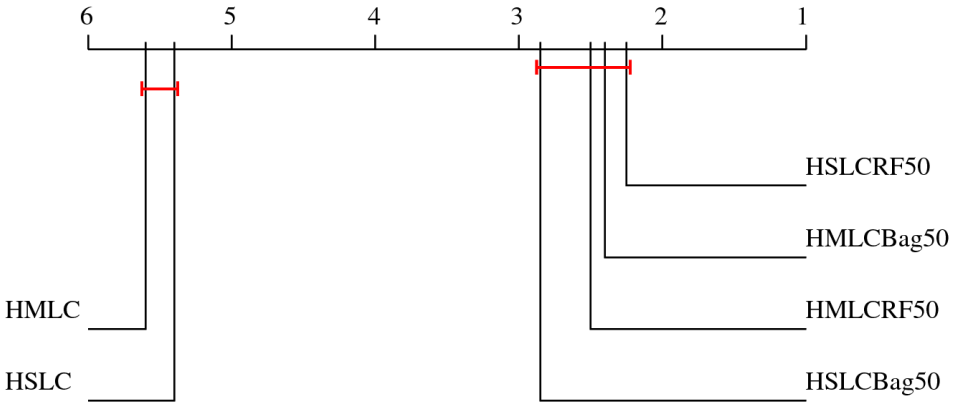
HMLC Random forest



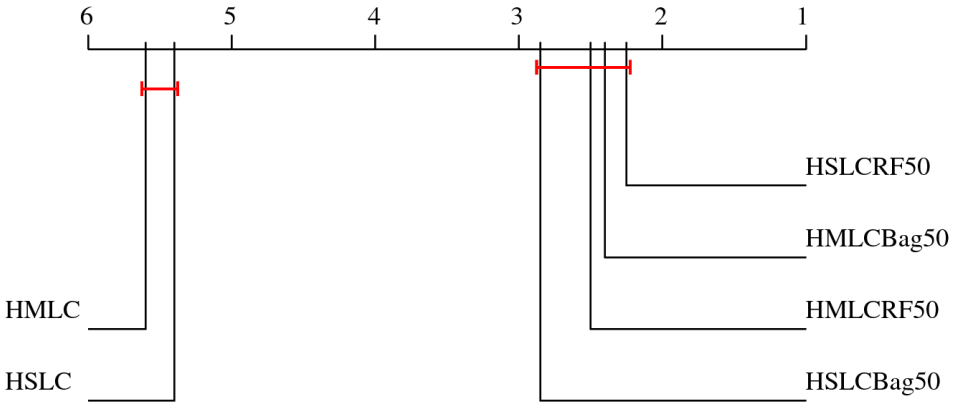
HSLC Random forest



Results – HMLC



Results – HMLC



- HMLC Ensembles perform better than
  - single HMLC tree (stat. sign.)
  - faster to learn than HSLC ensembles (stat. sign.)
  - smaller models than HSLC ensembles (stat. sign.)
- HMLC Bagging is better than HSLC Bagging, while HMLC random forest is worse than HSLC random forest

Results - Summary

Ensembles of PCTs:

- Perform significantly better than single PCT
- Perform better than ensembles for the sub-components
- Smaller and faster to learn than the ensembles for the sub-components
  - For multiple targets the ratio is ~ 3
  - For HMLC the ratio is ~ 4.5

Outline

- Background
  - Structured outputs
  - Predictive Clustering Trees (PCTs)
  - PCTs for HMLC
- Ensembles of PCTs
- Experimental evaluation
- Application in functional genomics
- Conclusions

Ensembles of PCTs - functional genomics

- Automatic prediction of the multiple functions of the ORFs in a genome
- Bagging of PCTs for HMLC compared with state-of-the-art approaches
- Datasets for three organisms
  - S. Cerevisae, A. Thaliana and M. musculus
- Two annotation schemes
  - FunCAT and GO (Gene Ontology)

Shietgat et al., BMC Bioinformatics 11:2, 2010

Background – network based approaches

- Usage of known functions of genes nearby in a functional association network

- **GeneFAS** (Chen and Xu, 2004)
- **GeneMANIA** (Mostafavi et al., 2008) – combines multiple networks from genomic and proteomic data

- **KIM** (Kim et al., 2008) – combines predictions of a network with predictions from Naïve Bayes
- **Funckenstein** (Tian et al., 2008) – logistic regression to combine predictions from network and random forest

Background – kernel based approaches

- **KLR** (Lee et al., 2006) – combination of Markov random fields and SVMs with diffusion kernels and using them in kernel logistic regression
- **CSVM** (Obozinski et al., 2008) – SVM for each function separately, and then reconciliated to enforce the hierarchy constraint
- **BSVM** (Barutcuoglu et al., 2006) – SVM for each function separately, then predictions combined using a Bayesian network
- **BSVM+** (Guan et al., 2008) – extension of BSVM, uses Naïve Bayes to combine results over the data sources

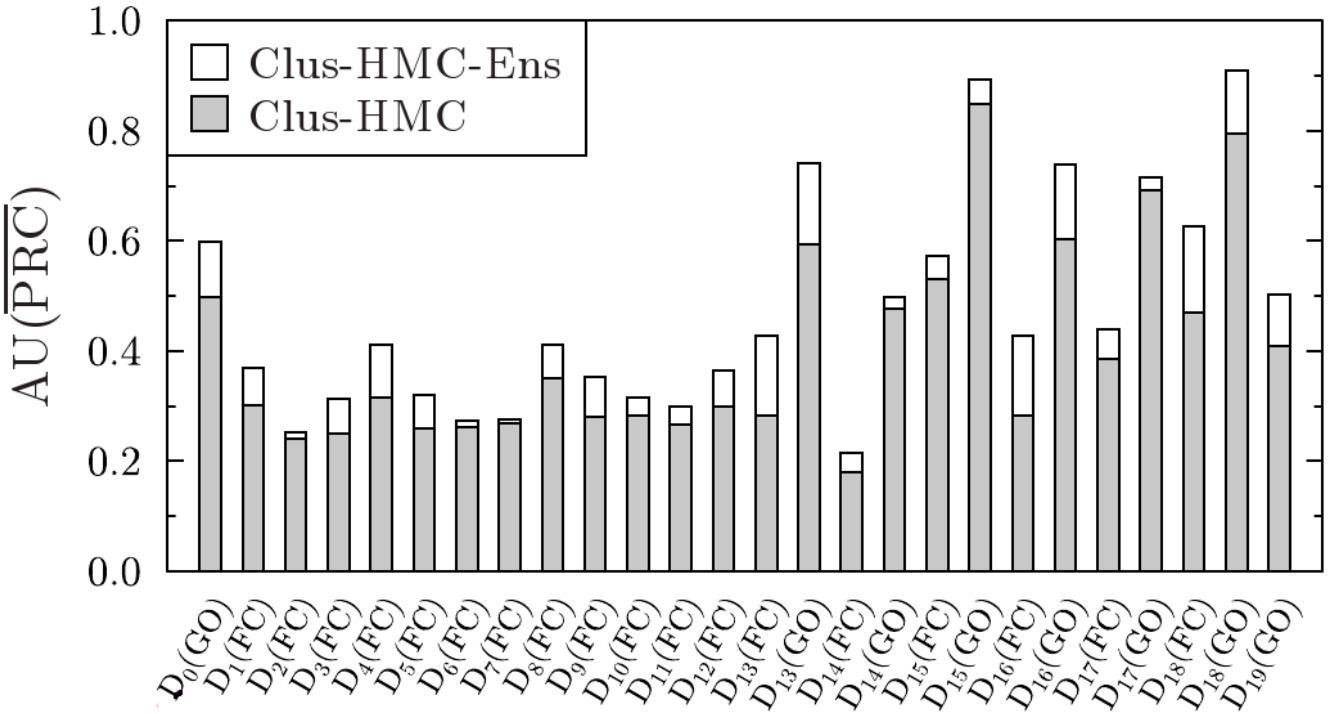
Datasets

- Saccharomyces Cerevisiae (D_0-D_{12})
  - sequence statistics, phenotype, secondary structure, homology and expression
- Arabidopsis Thaliana (D_{13}-D_{18})
  - sequence statistics, expression, predicted SCOP class, predicted secondary structure, InterPro and homology
  - Each dataset annotated with FunCAT and GO
- Mus Musculus (D_{19})
  - MouseFunc challenge, various data sources

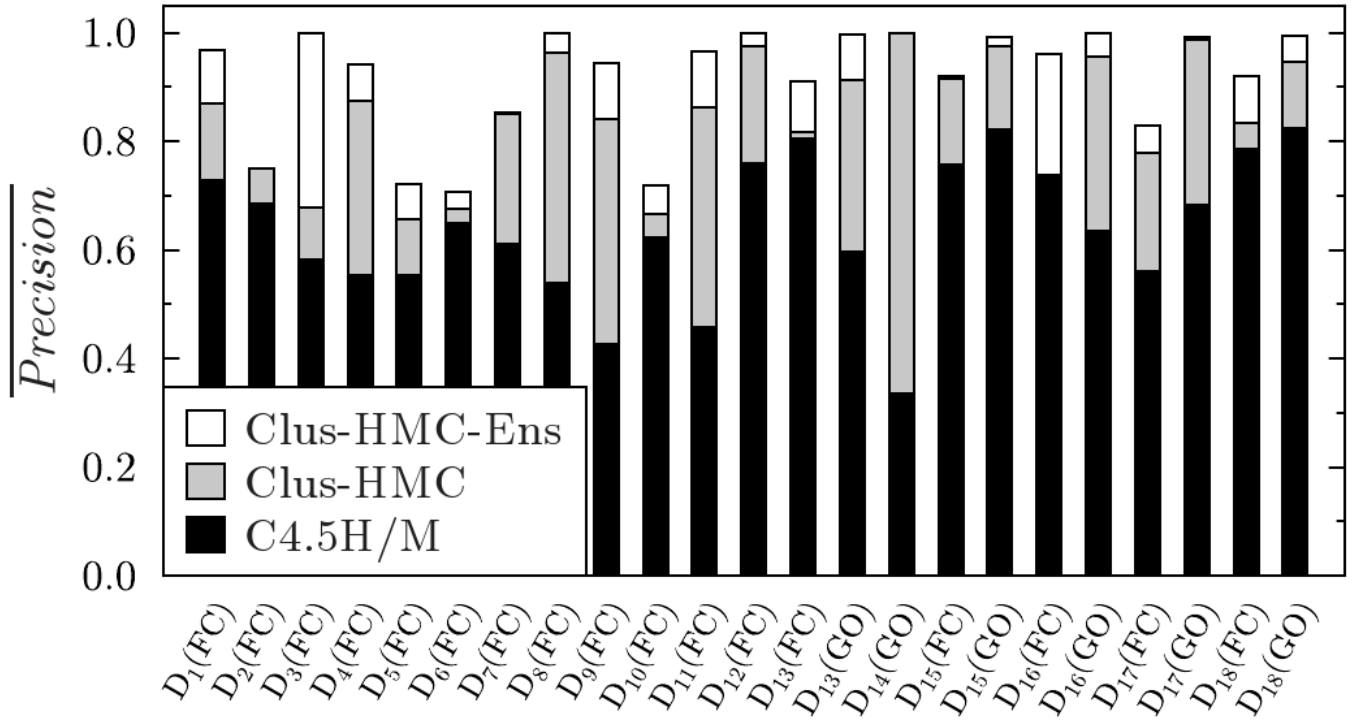
Experimental hypotheses

- Is bagging of PCTs better than single PCTs for HMLC and C4.5/H? (D_1-D_{18})
- Compare Bagging of PCTs with BSVM on D_0
- Compare Bagging of PCTs with the approaches from MouseFunc Challenge (D_{19})

Results – Bagging of PCTs vs. PCTs



Results – Bagging vs. PCTs vs. C4.5H



Results – Bagging of PCTs vs. BSVM

- Comparison by AUROC

- Average over the AUROC for all functions
  - Bagging of PCTs: 0.871
  - BSVM: 0.854

- Bagging of PCTs scores better on 73 of the 105 GO functions than BSVM ($p = 4.37 \cdot 10^{-5}$)

Results – MouseFunc

- Comparison by $AU\left(\overline{PRC}\right), \overline{AUPRC}, \overline{AUROC}$
- Different teams use different features
- Division of the dataset
  - Three branches of GO (BP, MF and CC)
  - Four ranges of specificity: number of genes by which each term is annotated with (3-10, 11-30, 31-100 and 101-300)

Results – MouseFunc (2)

$AU\left(\overline{PRC}\right)$

Subset	CLUS-HMC-ENS	BSVM$^+$	KLR	CSVM	GENEFAS	GENEMANIA	KIM	Funckenstein
BP_3-10	0.045	0.040⊖	0.028⊖	0.029⊖	0.028⊖	0.071⊕	0.029⊖	0.085⊕
BP_11-30	0.055	0.042⊖	0.053⊖	0.017⊖	0.012⊖	0.038⊖	0.031⊖	0.083⊕
BP_31-100	0.109	0.100⊖	0.135⊕	0.077⊖	0.033⊖	0.035⊖	0.044⊖	0.190⊕
BP_101-300	0.173	0.161⊖	0.174⊕	0.146⊖	0.078⊖	0.055⊖	0.051⊖	0.225⊕
CC_3-10	0.182	0.076⊖	0.060⊖	0.046⊖	0.050⊖	0.131⊖	0.128⊖	0.202⊕
CC_11-30	0.207	0.085⊖	0.128⊖	0.094⊖	0.038⊖	0.068⊖	0.112⊖	0.167⊖
CC_31-100	0.233	0.163⊖	0.161⊖	0.074⊖	0.107⊖	0.046⊖	0.127⊖	0.226⊖
CC_101-300	0.220	0.166⊖	0.225⊕	0.157⊖	0.110⊖	0.101⊖	0.094⊖	0.248⊕
MF_3-10	0.266	0.243⊖	0.191⊖	0.205⊖	0.174⊖	0.359⊕	0.189⊖	0.368⊕
MF_11-30	0.356	0.258⊖	0.285⊖	0.275⊖	0.136⊖	0.270⊖	0.215⊖	0.384⊕
MF_31-100	0.360	0.245⊖	0.294⊖	0.231⊖	0.120⊖	0.284⊖	0.191⊖	0.482⊕
MF_101-300	0.368	0.283⊖	0.331⊖	0.386⊕	0.184⊖	0.202⊖	0.140⊖	0.485⊕

- Bagging of PCTs is
  - significantly better ($p < 0.01$) than BSVM+, CSVM, GeneFAS and KIM
  - better than KLR and GeneMANIA
  - Significantly worse ($p < 0.01$) than Funckenstein

Results – MouseFunc (3)

$\overline{AUPRC}$

Subset	CLUS-HMC-ENS	BSVM$^+$	KLR	CSVM	GENEFAS	GENEMANIA	KIM	Funckenstein
BP_3-10	0.120	0.156⊕	0.075⊖	0.075⊖	0.108⊖	0.170⊕	0.108⊖	0.198⊕
BP_11-30	0.110	0.141⊕	0.087⊖	0.085⊖	0.074⊖	0.151⊕	0.107⊖	0.162⊕
BP_31-100	0.139	0.172⊕	0.158⊕	0.140⊕	0.094⊖	0.177⊕	0.116⊖	0.244⊕
BP_101-300	0.171	0.172⊕	0.169⊖	0.173⊕	0.104⊖	0.160⊖	0.056⊖	0.214⊕
CC_3-10	0.319	0.249⊖	0.119⊖	0.083⊖	0.233⊖	0.324⊕	0.271⊖	0.316⊖
CC_11-30	0.260	0.194⊖	0.212⊖	0.151⊖	0.131⊖	0.235⊖	0.178⊖	0.267⊕
CC_31-100	0.217	0.232⊕	0.197⊖	0.161⊖	0.191⊖	0.261⊕	0.144⊖	0.287⊕
CC_101-300	0.244	0.217⊖	0.259⊕	0.221⊖	0.177⊖	0.258⊕	0.118⊖	0.279⊕
MF_3-10	0.320	0.441⊕	0.258⊖	0.228⊖	0.427⊕	0.465⊕	0.304⊖	0.472⊕
MF_11-30	0.356	0.373⊕	0.347⊖	0.393⊕	0.350⊖	0.401⊕	0.302⊖	0.455⊕
MF_31-100	0.269	0.289⊕	0.230⊖	0.278⊕	0.242⊖	0.291⊕	0.255⊖	0.416⊕
MF_101-300	0.322	0.317⊖	0.321⊖	0.374⊕	0.295⊖	0.391⊕	0.172⊖	0.441⊕

- Bagging of PCTs is
  - significantly better ($p < 0.01$) than KIM
  - not different from BSVM+, KLR, CSVM, GeneFAS
  - Significantly worse ($p < 0.01$) than Funckenstein and GeneMANIA

Results – MouseFunc (4)

$\overline{AUROC}$

Subset	Clus-HMC-Ens	BSVM$^+$	KLR	CSVM	GeneFAS	GeneMANIA	KIM	Funckenstein
BP_3-10	0.695	0.808⊕	0.581⊖	0.588⊖	0.715⊕	0.873⊕	0.813⊕	0.790⊕
BP_11-30	0.748	0.808⊕	0.741⊖	0.659⊖	0.767⊕	0.849⊕	0.822⊕	0.796⊕
BP_31-100	0.831	0.874⊕	0.846⊕	0.778⊖	0.780⊖	0.872⊕	0.851⊕	0.880⊕
BP_101-300	0.823	0.853⊕	0.845⊕	0.813⊖	0.733⊖	0.840⊕	0.795⊖	0.838⊕
CC_3-10	0.748	0.845⊕	0.571⊖	0.618⊖	0.782⊕	0.899⊕	0.865⊕	0.837⊕
CC_11-30	0.791	0.873⊕	0.790⊖	0.785⊖	0.834⊕	0.907⊕	0.846⊕	0.850⊕
CC_31-100	0.863	0.896⊕	0.850⊖	0.851⊖	0.783⊖	0.887⊕	0.863	0.849⊖
CC_101-300	0.845	0.873⊕	0.851⊕	0.821⊖	0.750⊖	0.842⊖	0.808⊖	0.867⊕
MF_3-10	0.818	0.887⊕	0.630⊖	0.681⊖	0.850⊕	0.951⊕	0.880⊕	0.879⊕
MF_11-30	0.842	0.903⊕	0.861⊕	0.836⊖	0.865⊕	0.936⊕	0.884⊕	0.909⊕
MF_31-100	0.838	0.888⊕	0.892⊕	0.881⊕	0.843⊕	0.887⊕	0.884⊕	0.903⊕
MF_101-300	0.874	0.904⊕	0.894⊕	0.884⊕	0.843⊖	0.909⊕	0.844⊖	0.918⊕

- Bagging of PCTs is
  - not different from KLR, CSVM, GeneFAS and KIM
  - Significantly worse ($p < 0.01$) than Funckenstein, BSVM+ and GeneMANIA

Summary – Functional genomics

- Bagging of PCTs outperforms single PCT and C4.5H/M (Clare, 2003)
- Bagging of PCTs outperforms a statistical learner based on SVMs (BSVM) for S. Cerevisiae
- Bagging of PCTs is competitive to statistical and network based methods for the M. Musculus data

Outline

- Background
  - Structured outputs
  - Predictive Clustering Trees (PCTs)
  - PCTs for HMLC
- Ensembles of PCTs
- Experimental evaluation
- Application in functional genomics
- Conclusions

Conclusions

- Lift the predictive performance of single PCT
- Better than ensembles for each of the sub-component of the output
- Competitive with state-of-the-art approaches in functional genomics
- Applicability to wide range of problems
  - Different type and sizes of outputs
  - Small and large datasets

Questions?