Hierarchical Hardness Models for SAT

Lin Xu, Holger H. Hoos
and Kevin Leyton-Brown

University of British Columbia

{xulin730, hoos, kevinlb}@cs.ubc.ca

SATzilla-07

- Old SATzilla [Nudelman, et. al, 2003]
  - **2nd Random**
  - **2nd Handmade (SAT)**
  - **3rd Handmade**
- SATzilla-07 [Xu, et. al, 2007]
  - **1st Handmade**
  - **1st Handmade (UNSAT)**
  - **1st Random**
  - **2nd Handmade (SAT)**
  - **3rd Random (UNSAT)**

Outline

- Introduction
- Predicting the satisfiability of SAT instances
- Hierarchical Hardness Models
- Conclusions and future work

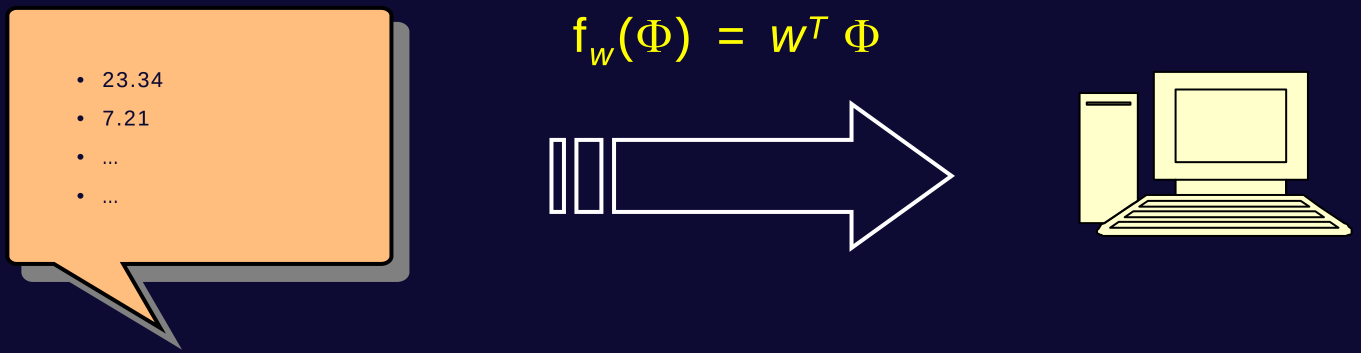
Introduction

Empirical Hardness Model (EHM)

- Predicting algorithm's runtime based on poly-time computable features
  - **Features:** anything can **characterize** the problem instance and can be represented by a real number
    - 9 category features [Nudelman, et al, 2004]
  - **Prediction:** any machine learning technique can return prediction of a **continuous value**
    - Linear basis function regression [Leyton-Brown et al, 2002; Nudelman, et al, 2004]

Linear Basis Function Regression

$f_w(\Phi) = w^T \Phi$



Features (Φ)

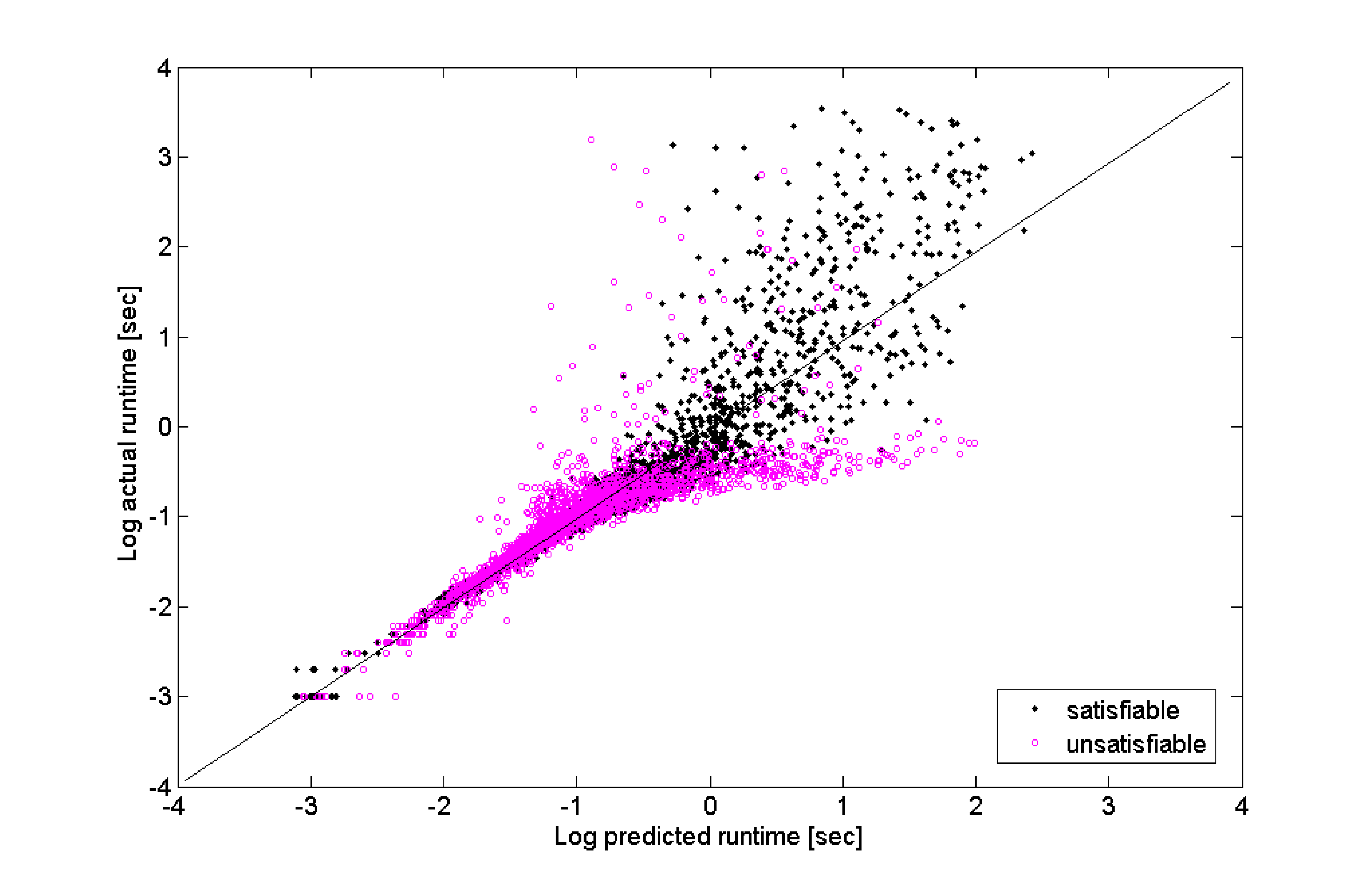
Runtime (y)

$$P(y \mid \Phi) = N(y \mid w^T \Phi, \beta)$$

Previous work [Nudelman, et al, 2004]

M_{sat}/M_{unsat}: focused on one type of instances

$M_{uncond.}$: Trained on mixture of instances



Solver: satelite; Dataset: Quasi-group completion problem

The Key Idea

- Knowing satisfiability of an instance allows better prediction

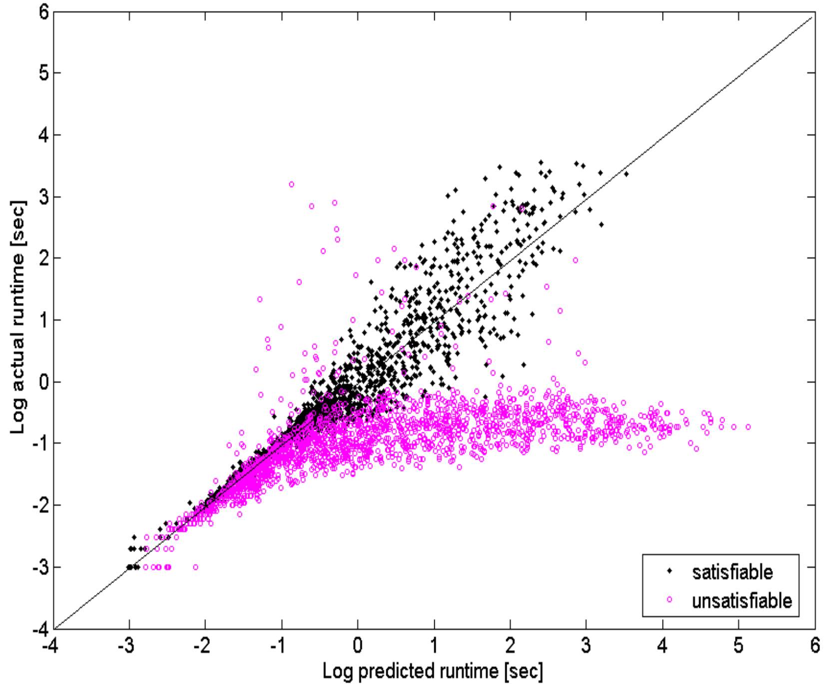
Problem: the **only way** to know this is to **solve** the instance!

- **Idea:** predict satisfiability

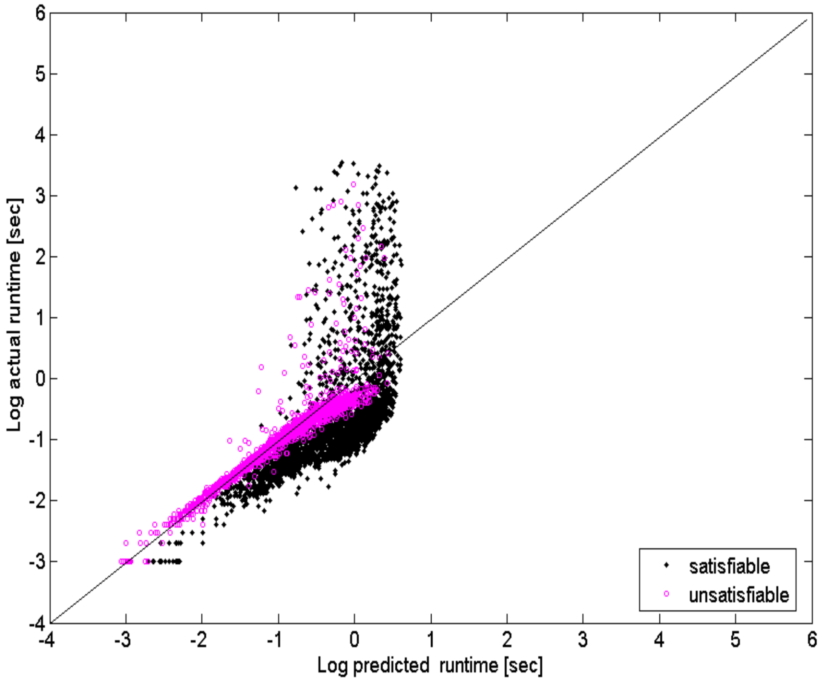
→ even though such a classifier **can't be 100% accurate**, maybe it can be accurate enough to help

The Cost of Using the Wrong Model

Only use SAT model



Only use UNSAT model



Solver: satelite; Dataset: Quasi-group completion problem

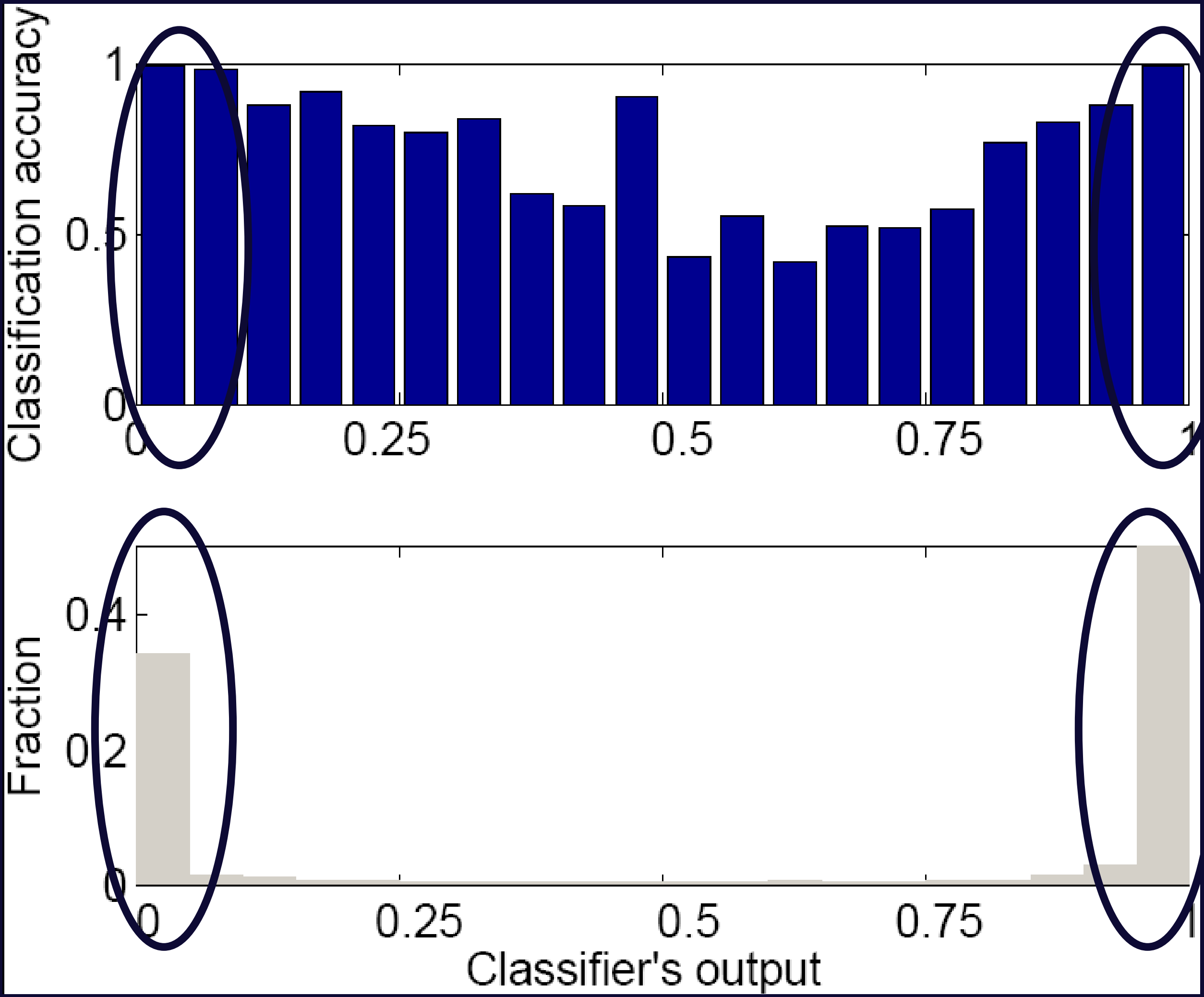
Predicting Satisfiability of SAT Instances

Performance of Classification

- Classifier: SMLR [Krishnapuram, et al. 2005]
- Features: same as regression
- Datasets
  - rand3sat-var
  - rand3sat-fix
  - QCP
  - SW-GCP

Dataset	Classification Accuracy		
	sat.	unsat.	overall
rand3sat-var	0.979	0.989	0.984
rand3sat-fix	0.848	0.881	0.865
QCP	0.980	0.932	0.960
SW-GCP	0.752	0.711	0.734

Performance of Classification (QCP)



Hierarchical Hardness Models

Improve EHM by using classification

Key idea: use prediction of satisfiability to improve runtime prediction

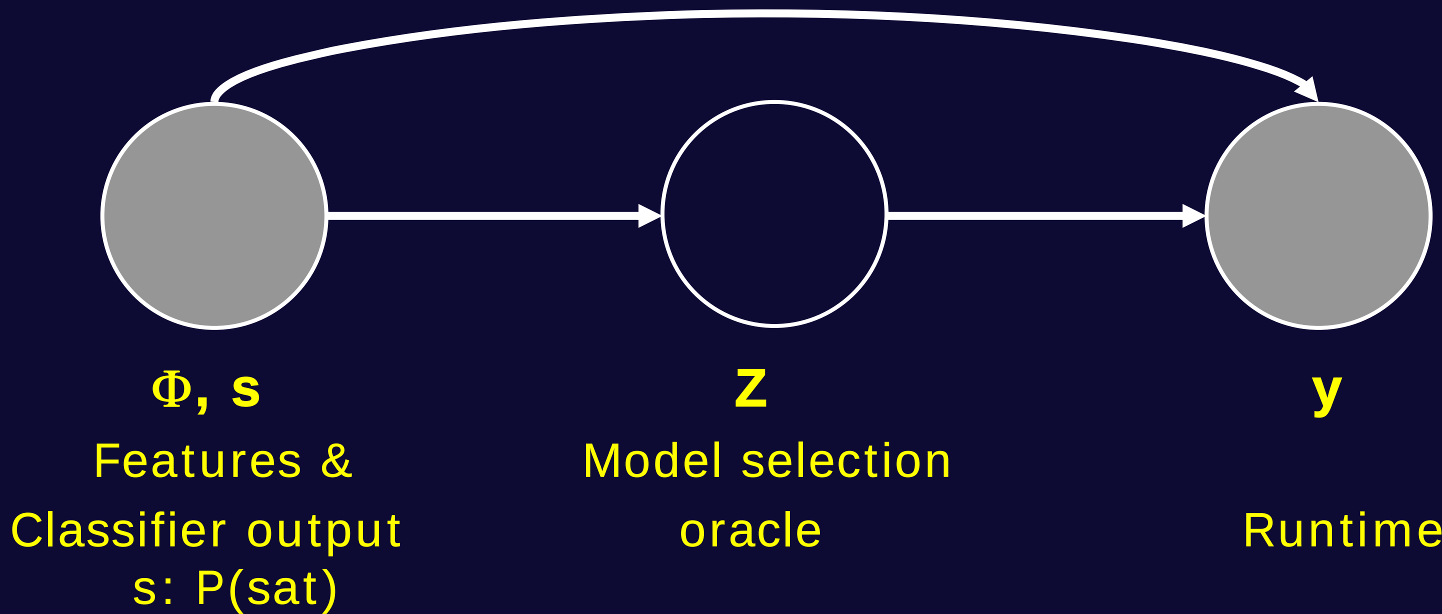
- Does this mean that I just use the (eg) unsat model if the instance is predicted to be unsat?

  NO! need to consider the **error distribution**

- Note: best performance would be achieved by **model selection oracle**!

  Question: How to **approximate model selection oracle** based on features

Hierarchical Hardness Models



$$P(y \mid \Phi, s) = \sum_{z \in \{sat, unsat\}} P(z \mid \Phi, s) N(y \mid w_z^T \Phi, \beta_z)$$

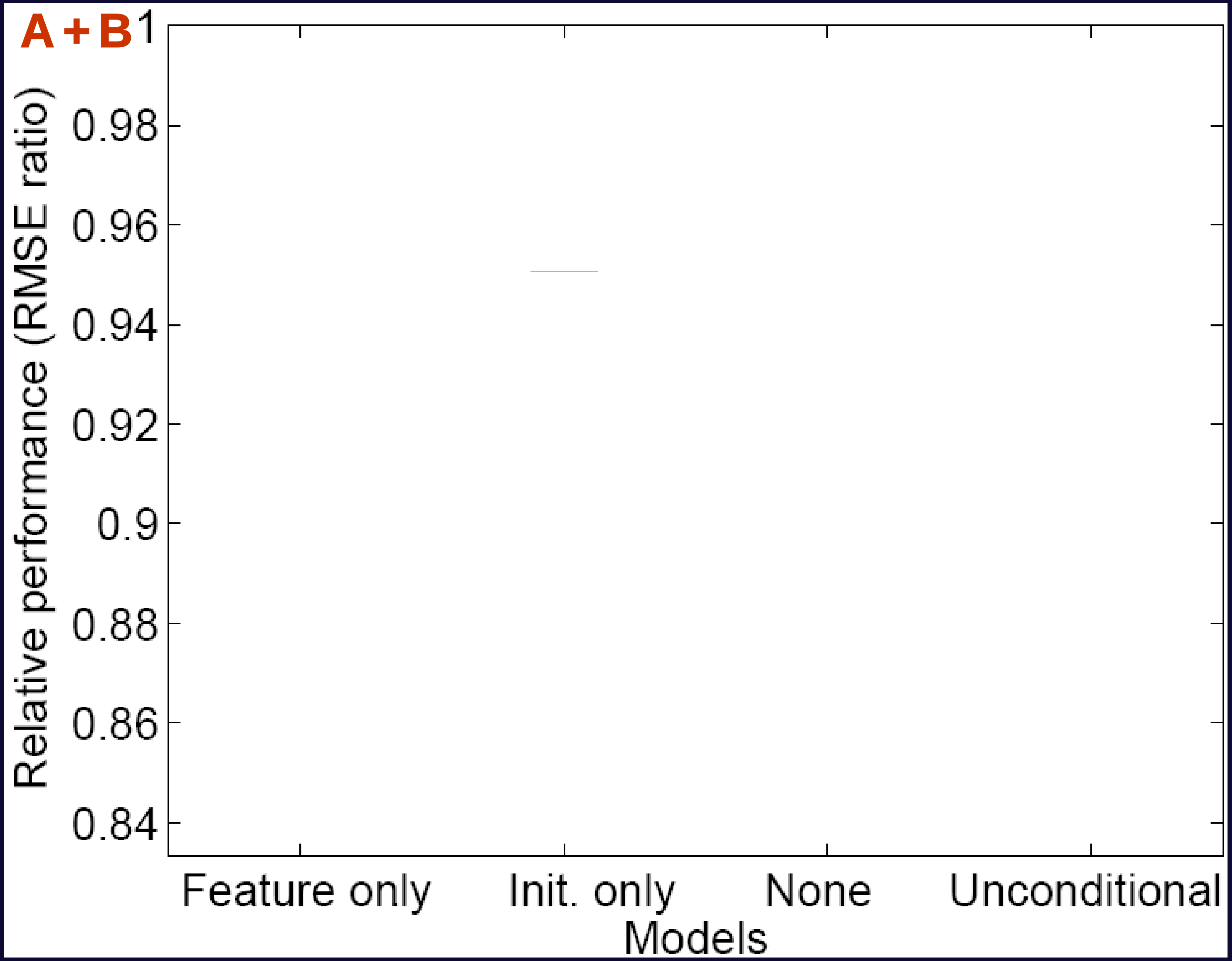
Mixture of experts problem with fixed experts, use EM to find the parameters for z [Murphy, 2001]

Importance of Classifier's Output

Two ways:

A: Using classifier's output as a feature

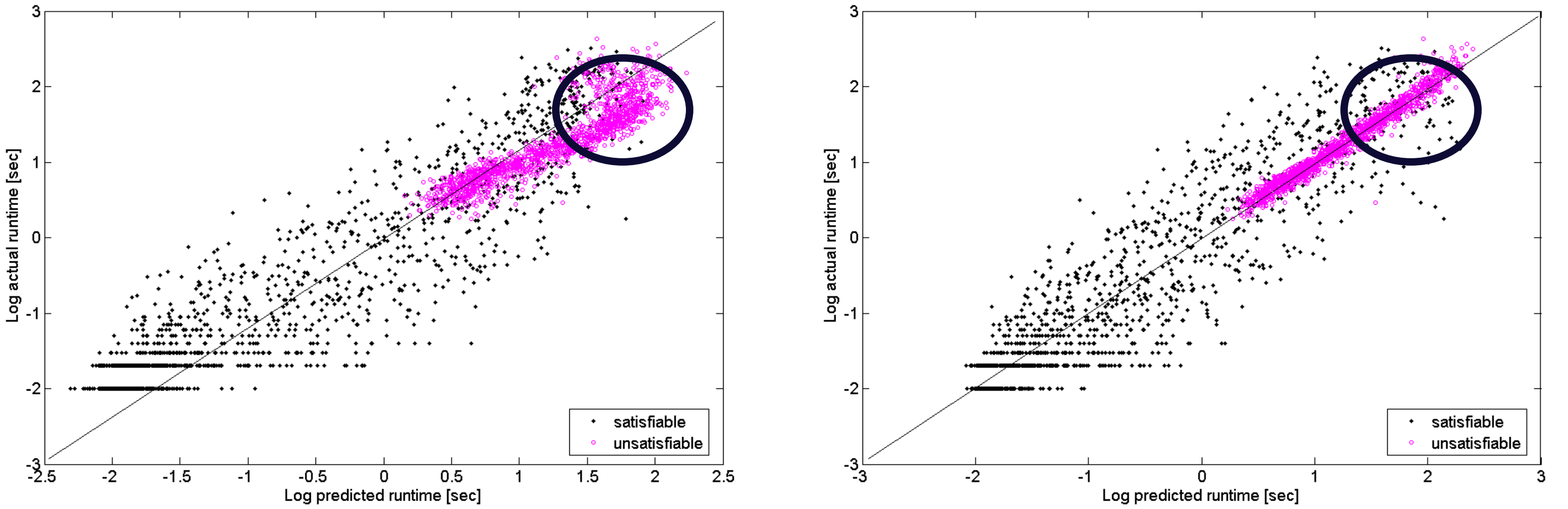
B: Using classifier's output for EM initialization



Big Picture of HHM Performance

Solver	Oracular	Uncond.	Hier.	Oracular	Uncond.	Hier.
	RMSE (rand3-var)			RMSE (rand3-fix)		
satz	0.329	0.358	**0.344**	0.343	0.420	0.413
march_dl	0.283	0.396	**0.306**	0.444	0.542	0.533
kcnfs	0.294	0.373	**0.312**	0.397	0.491	0.486
Oksolver	0.356	0.443	**0.378**	0.479	0.596	0.587
Solver	RMSE (QCP)			RMSE (SW-GCP)		
Zchaff	0.303	0.675	**0.577**	0.657	0.993	0.983
Minisat	0.305	0.574	**0.500**	0.682	1.022	1.024
Satzoo	0.240	0.397	**0.334**	0.384	0.581	0.581
Satelite	0.247	0.426	**0.372**	0.618	0.970	0.978
Sato	0.375	0.711	**0.635**	0.723	1.352	1.345
oksolver	0.427	0.548	**0.506**	0.601	1.337	1.331

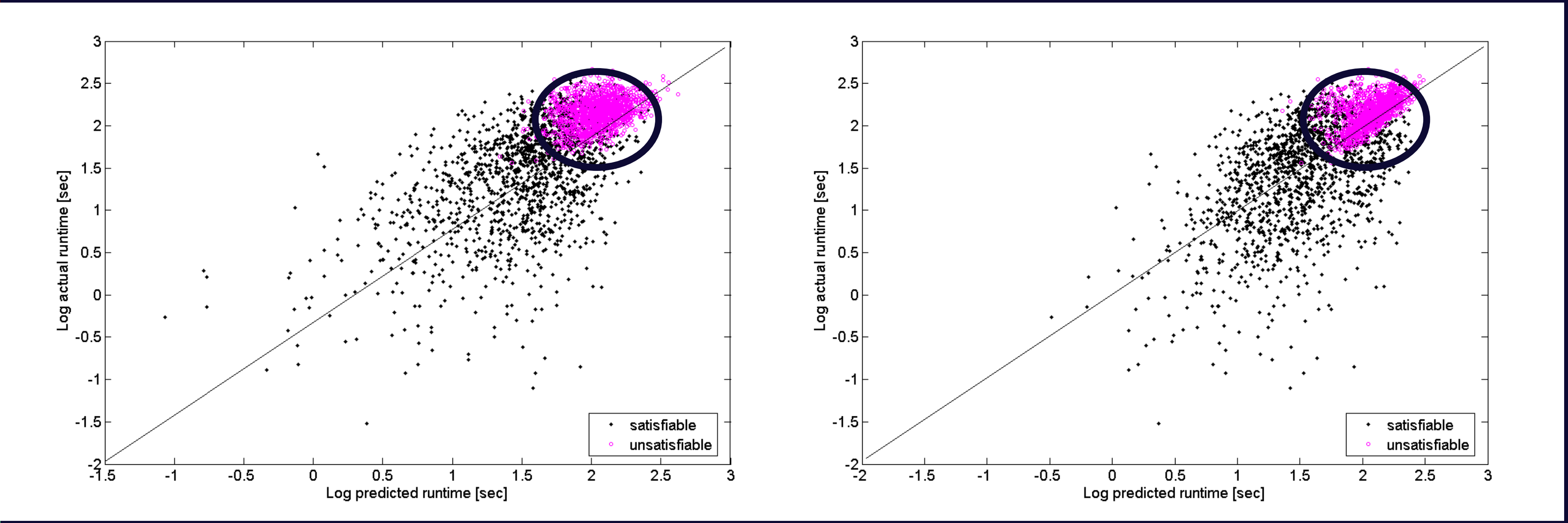
Example for rand3-var (Solver: satz)



Left: unconditional model

Right: hierarchical model

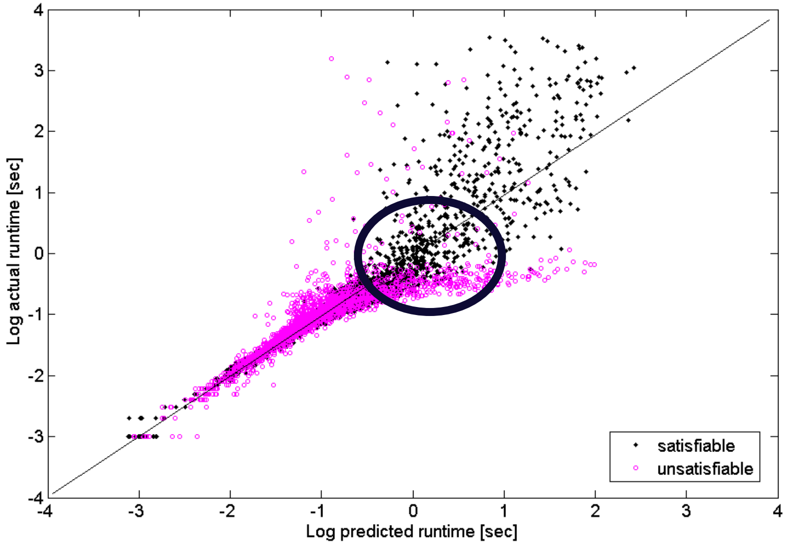
Example for rand3-fix (solver: satz)

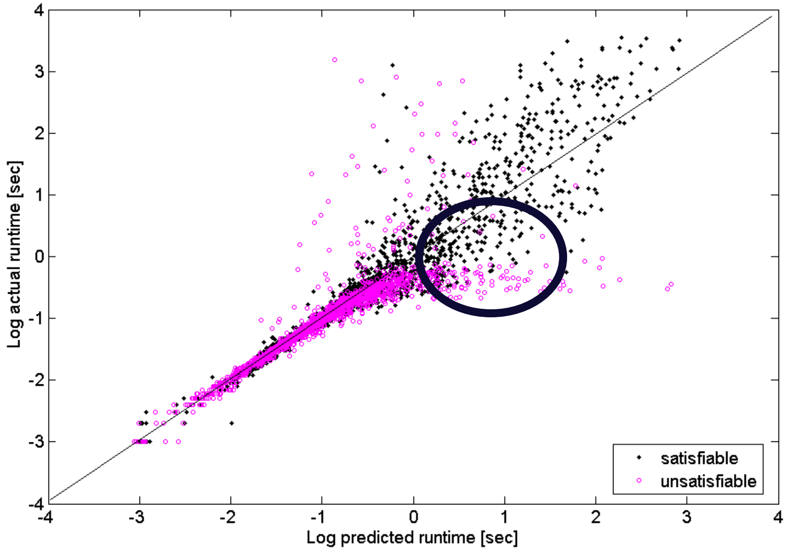


Left: unconditional model

Right: hierarchical model

Example for QCP (solver: satelite)

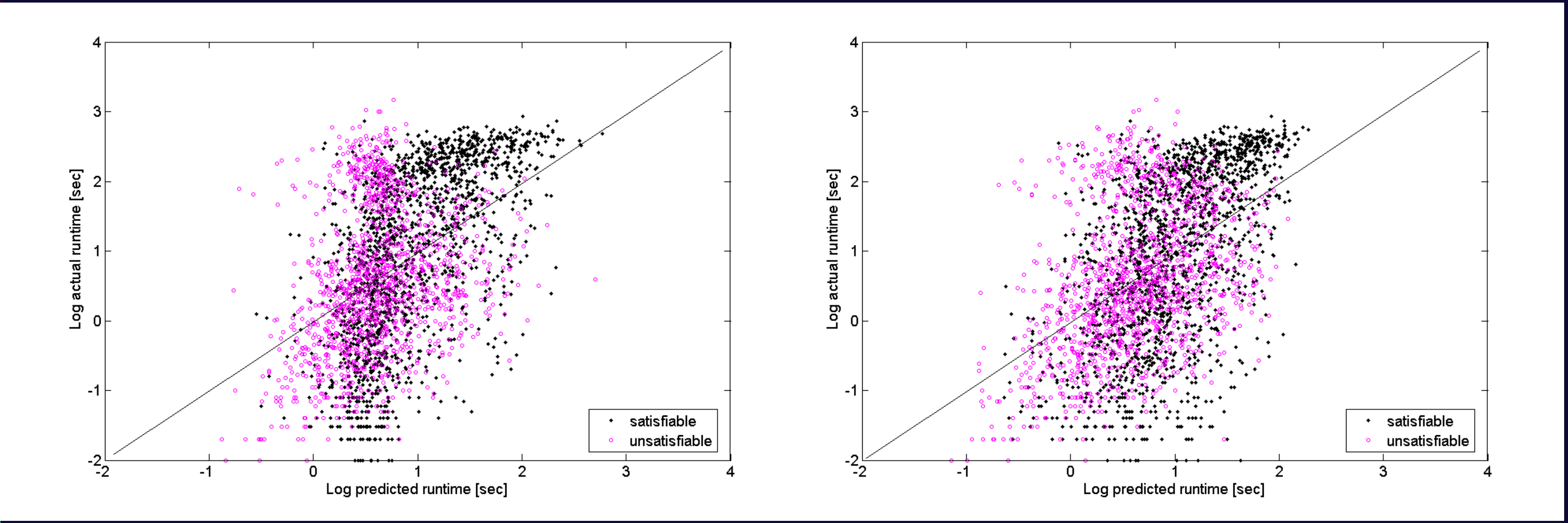




Left: unconditional model

Right: hierarchical model

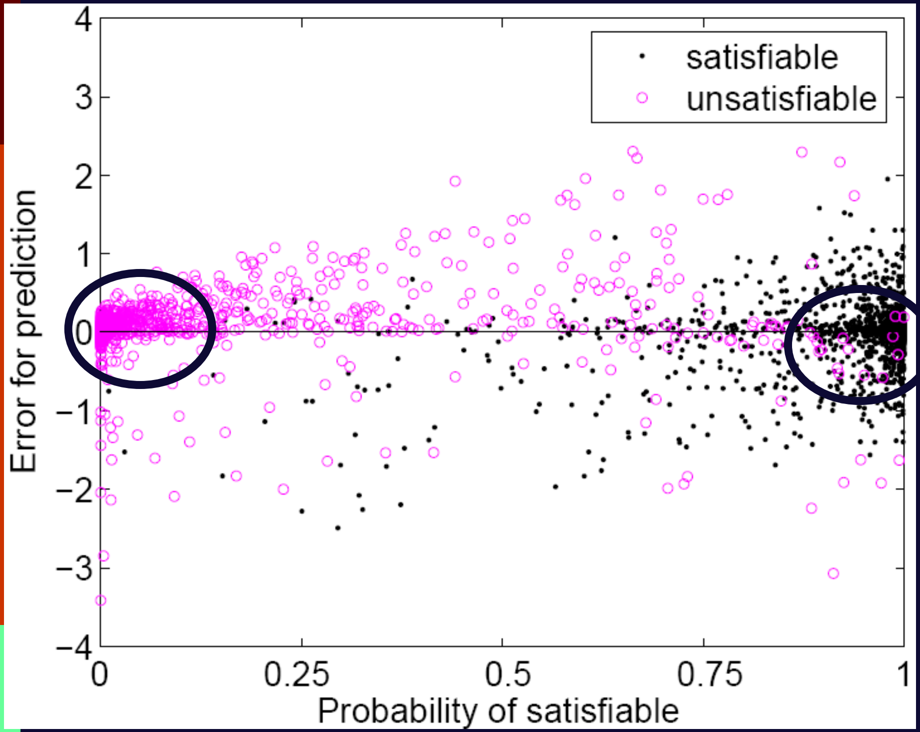
Example for SW-GCP (solver: zchaff)

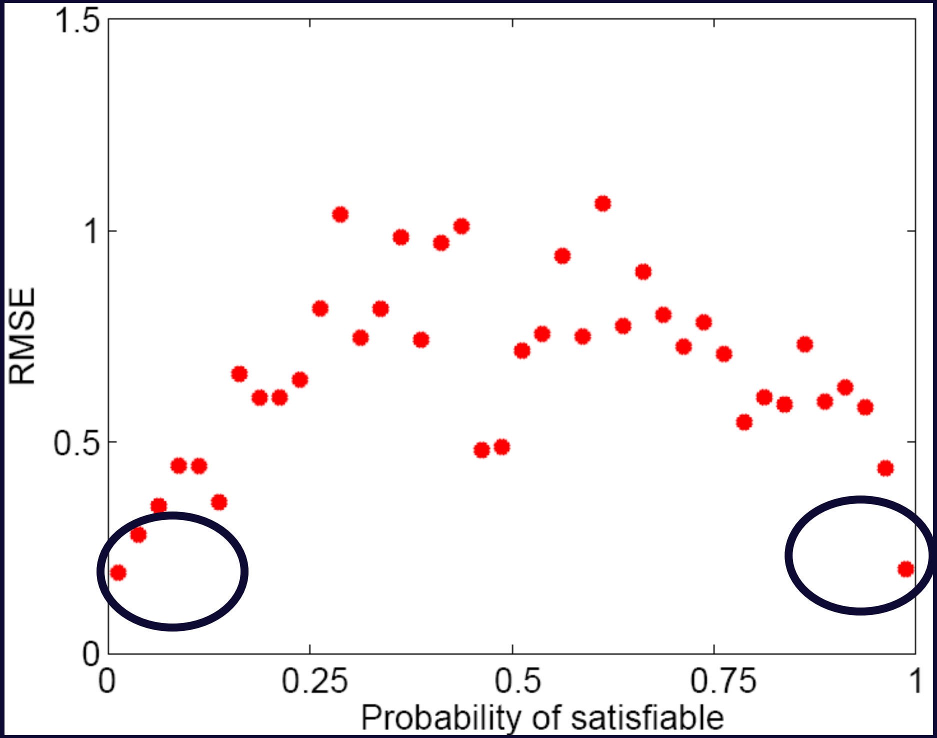


Left: unconditional model

Right: hierarchical model

Correlation Between Prediction Error and Classifier's Confidence





Left: Predicted P(sat.) vs runtime prediction error

Right: Predicted P(sat.) vs RMSE

Solver: satelite; Dataset: QCP

Conclusions and Future Work

Conclusions

- Models conditioned on SAT/UNSAT have much better runtime prediction accuracies.
- **A classifier can be used to distinguish SAT/UNSAT with high accuracy.**
- **Conditional models can be combined into hierarchical model with better performance.**
- Classifier's confidence correlates with prediction error.

Future Work

- Better features for SW-GCP
- Test on more real world problems
- Extend underlying experts beyond satisfiability