

NIPS 2008 Workshop Machine Learning Open Source Software



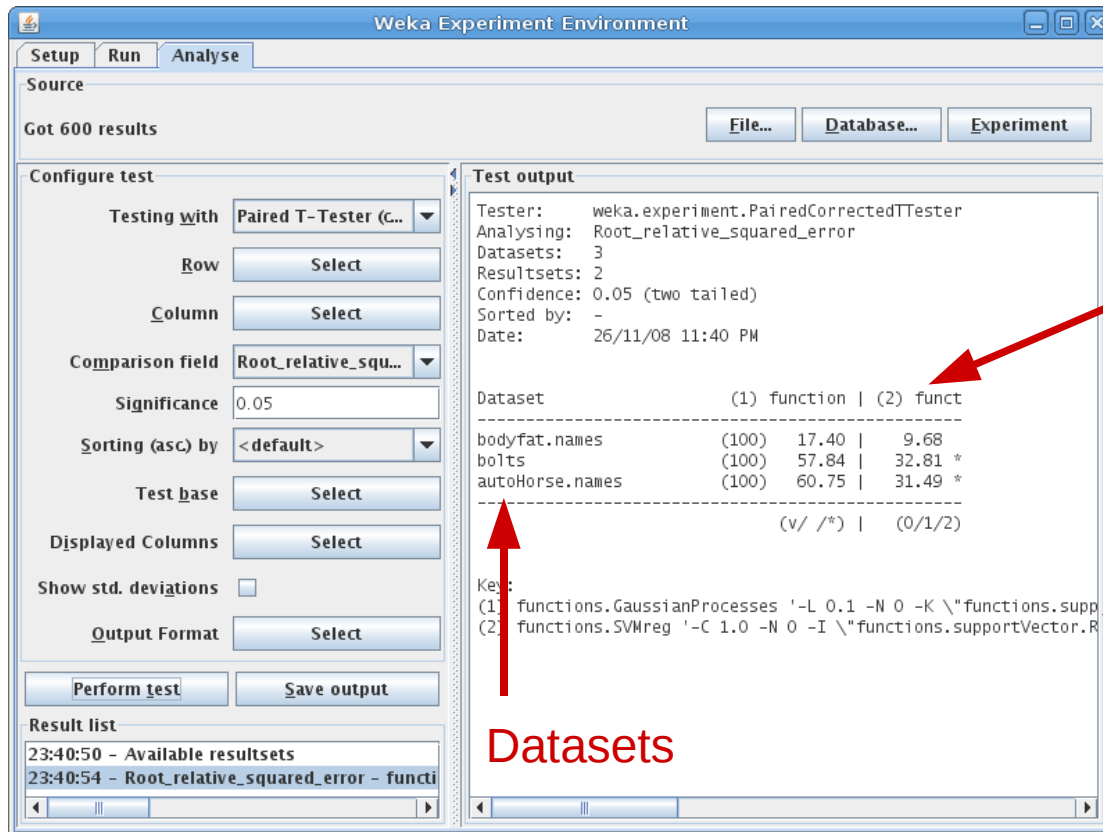
“BenchMarking Via WEKA”

Peter Reutemann

fracpete@waikato.ac.nz

What is WEKA?

- **W**aikato **E**nvironment for **K**nowledge **A**nalysis
- Machine learning work-bench written in Java



Schemes

Datasets

Key for Scheme setups

“WEKA Experimenter”

Motivation

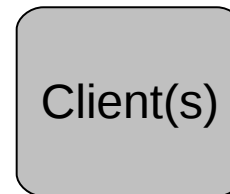
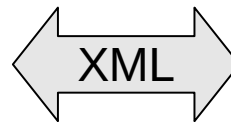
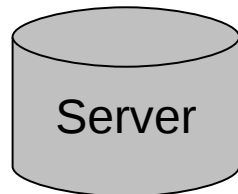
- WEKA's benefits
 - plug-in architecture for schemes
 - framework for statistical evaluation/comparison
 - experiments are relatively easy to reproduce
- WEKA's limitations
 - written in Java, limited support for Python (via Jython)
 - no benchmarks of standard schemes publicly available

How to integrate other languages/frameworks, but keep benefits of centralized statistical framework and easily reproduce experiments???

The BMW Framework

- Client-server architecture
- Communication via XML protocol
 - ⇒ programming language agnostic
- Clients:
 - obtain dataset from server
 - build scheme
 - send predictions back
- Server evaluates predictions and stores results

- Datasets
- Experiment setups
- Evaluation
- Results



- WEKA (Java)
- MLPY (Python)
- etc.

Reproducibility of Experiments

- Datasets need to be uploaded to server
 - ⇒ Publicly available
- Single experiment is uniquely identified by:
 - Type of evaluation (cross-validation, percentage split, etc.)
 - Scheme setup (name, parameters, version)
 - Dataset (name, version)
- Controlled, centralized evaluation:
 - Server generates training and test data for client
 - Server calculates statistics based on predictions from client

Experiments Screenshot

Datasets × Schemes matrix

Datasets

Key for Scheme setups

The screenshot shows the 'BenchMarking Via Weka @ jossan' window. The 'Experiment results' section displays a table of correlation coefficients for three datasets: bodyfat/1.0.0, bolts/1.0.0, and autoHorse/1.0.0. The 'Displayed measure' is set to 'correlation_coefficient'. Below the table, two scheme setups are listed, both using the 'weka.classifiers.functions.SVMreg' classifier. The log at the bottom shows the execution status for the datasets.

Dataset	(1)	(2)
bodyfat/1.0.0	0.979	0.989
bolts/1.0.0	0.69	0.896
autoHorse/1.0.0	0.787	0.952

```
(1) bmvw.schemes.Weka (3.5.8) -classifier "weka.classifiers.functions.GaussianProcesses -L 0.1 -N 0 -K \"weka
(2) bmvw.schemes.Weka (3.5.8) -classifier "weka.classifiers.functions.SVMreg -C 1.0 -N 0 -I \"weka.classifier
```

2008-12-09 06:37:23: Datasets (numeric): execute/start
2008-12-09 06:37:24: Datasets (numeric): execute/finished = true

Conclusion

- Step towards straight-forward experiment reproducibility
- Evaluation framework for different tools across programming languages (WEKA, MLPY, ...)
- Generation of ExpML output easy
- Submission of results to Experiment Database easy



Bench Marking Via Weka

<http://www.scms.waikato.ac.nz/~fracpete/projects/bmvw/>

...thank you!