The Shark

Machine Learning Library

Institut für Neuroinformatik - Lehrstuhl für Theoretische Biologie -

HRI Europe
Honda Research Institute

Tobias Glasmachers

12/12/2008

Outline

- Introduction
- Supervised learning with Shark
- Roundtrip through all the rest
- Conclusions

What is Shark?

Introduction

What is Shark?

- machine learning library

Introduction

What is Shark?

- machine learning library
- C++

What is Shark?

- machine learning library
- C++
- modular

Introduction

What is Shark?

- machine learning library
- C++
- modular
- free: GPL

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
    - OS-independent: Linux, MacOS, Windows

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
    - OS-independent: Linux, MacOS, Windows
    - pure machine learning, just algorithms

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
  - OS-independent: Linux, MacOS, Windows
  - pure machine learning, just algorithms ⇒ no own GUI

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
  - OS-independent: Linux, MacOS, Windows
  - pure machine learning, just algorithms ⇒ no own GUI
  - completely self-contained:
    - flexible arrays
    - random number generator
    - linear algebra
    - ... other tools

What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
  - OS-independent: Linux, MacOS, Windows
  - pure machine learning, just algorithms ⇒ no own GUI
  - completely self-contained:
    - flexible arrays
    - random number generator
    - linear algebra
    - . . . other tools

  ⇒ no dependencies!

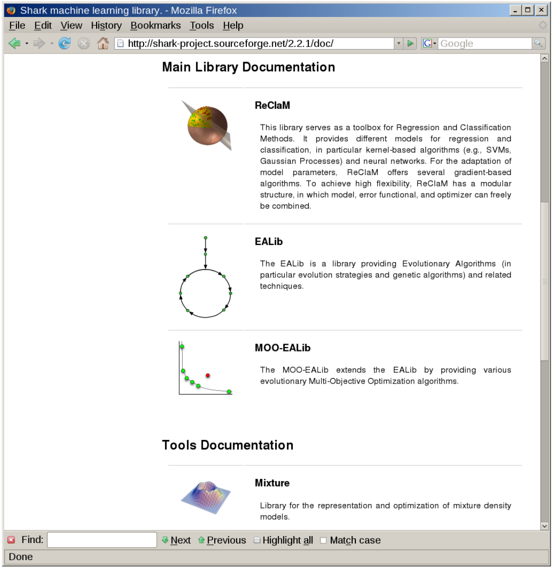
What is Shark?

- machine learning library
- C++
- modular
- free: GPL
- portable
  - OS-independent: Linux, MacOS, Windows
  - pure machine learning, just algorithms ⇒ no own GUI
  - completely self-contained:
    - flexible arrays
    - random number generator
    - linear algebra
    - . . . other tools

    ⇒ no dependencies!
- growing, actively maintained

What is Shark?

- well-documented



Shark machine learning library. - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://shark-project.sourceforge.net/2.2.1/doc/
Google

Main Library Documentation

ReClaM

This library serves as a toolbox for Regression and Classification Methods. It provides different models for regression and classification, in particular kernel-based algorithms (e.g., SVMs, Gaussian Processes) and neural networks. For the adaptation of model parameters, ReClaM offers several gradient-based algorithms. To achieve high flexibility, ReClaM has a modular structure, in which model, error functional, and optimizer can freely be combined.

EALib

The EALib is a library providing Evolutionary Algorithms (in particular evolution strategies and genetic algorithms) and related techniques.

MOO-EALib

The MOO-EALib extends the EALib by providing various evolutionary Multi-Objective Optimization algorithms.

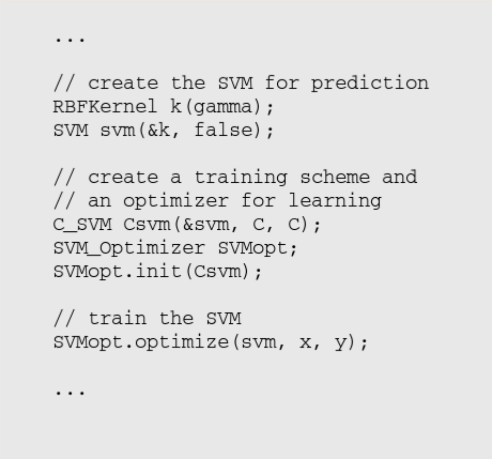
Tools Documentation

Mixture

Library for the representation and optimization of mixture density models.

What is Shark?

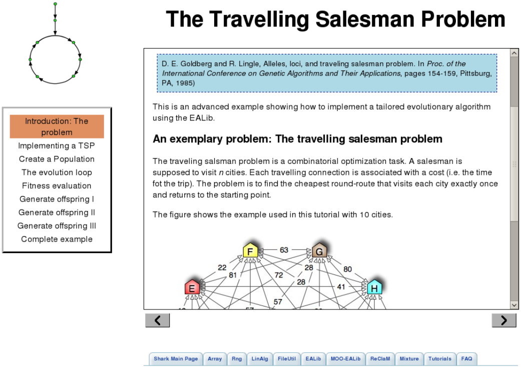
- well-documented
- more than 60 example programs



```
...

// create the SVM for prediction
RBFKernel k(gamma);
SVM svm(&k, false);

// create a training scheme and
// an optimizer for learning
C_SVM Csvm(&svm, C, C);
SVM_Optimizer SVMopt;
SVMopt.init(Csvm);

// train the SVM
SVMopt.optimize(svm, x, y);

...
```

What is Shark?

- well-documented
- more than 60 example programs
- more than 20 tutorials



The Travelling Salesman Problem

D. E. Goldberg and R. Lingle, Alleles, loci, and traveling salesman problem. In *Proc. of the International Conference on Genetic Algorithms and Their Applications*, pages 154-159, Pittsburg, PA, 1985)

This is an advanced example showing how to implement a tailored evolutionary algorithm using the EALib.

An exemplary problem: The travelling salesman problem

The traveling salsman problem is a combinatorial optimization task. A salesman is supposed to visit *n* cities. Each travelling connection is associated with a cost (i.e. the time fot the trip). The problem is to find the cheapest round-route that visits each city exactly once and returns to the starting point.

The figure shows the example used in this tutorial with 10 cities.

Outline

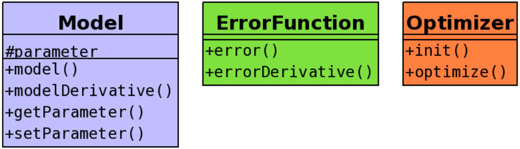
- Introduction
- Supervised learning with Shark
- Roundtrip through all the rest
- Conclusions

Supervised learning with Shark

Learning $\Leftrightarrow$ Optimization

Supervised learning with Shark

Learning ⇔ Optimization



Model

#parameter

+model()

+modelDerivative()

+getParameter()

+setParameter()

ErrorFunction

+error()

+errorDerivative()

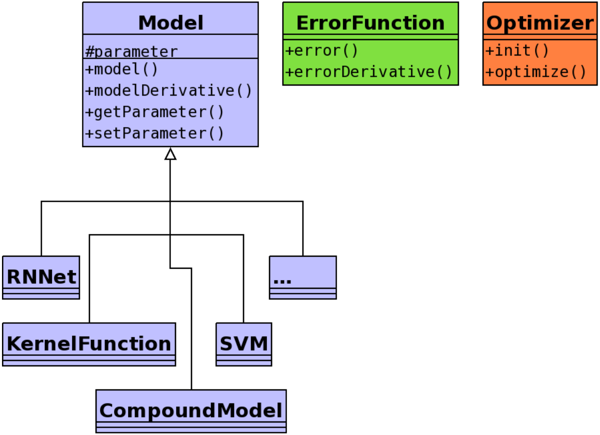
Optimizer

+init()

+optimize()

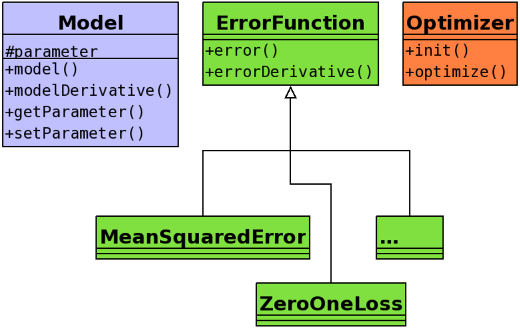
Supervised learning with Shark

Learning ⇔ Optimization



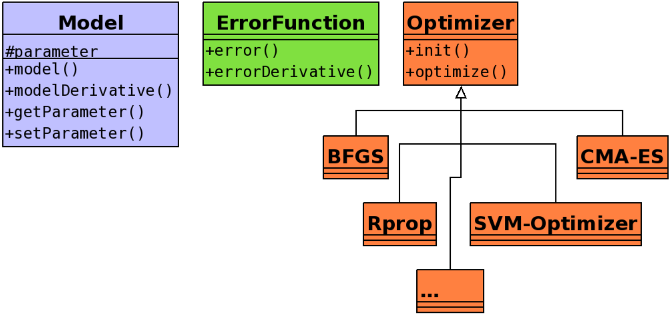
Supervised learning with Shark

Learning ⇔ Optimization



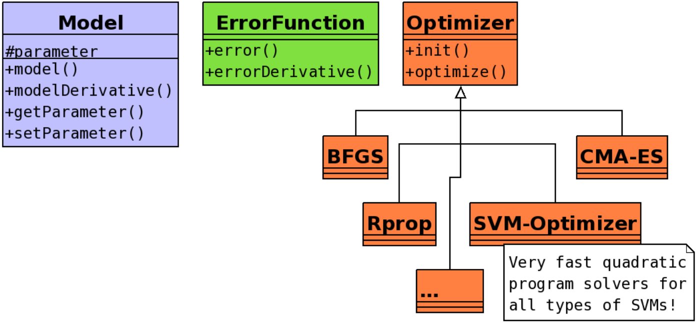
Supervised learning with Shark

Learning ⇔ Optimization



Supervised learning with Shark

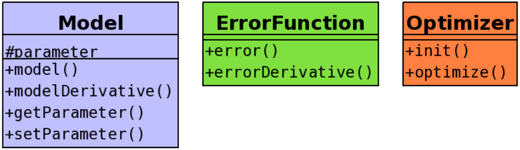
Learning ⇔ Optimization



Very fast quadratic program solvers for all types of SVMs!

Supervised learning with Shark

Learning ⇔ Optimization



Model

#parameter

+model()

+modelDerivative()

+getParameter()

+setParameter()

ErrorFunction

+error()

+errorDerivative()

Optimizer

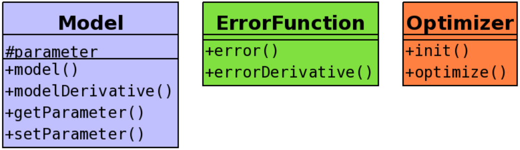
+init()

+optimize()

- Communication only through top level interfaces

Supervised learning with Shark

Learning ⇔ Optimization



Model

#parameter

+model()

+modelDerivative()

+getParameter()

+setParameter()

ErrorFunction

+error()

+errorDerivative()

Optimizer

+init()

+optimize()

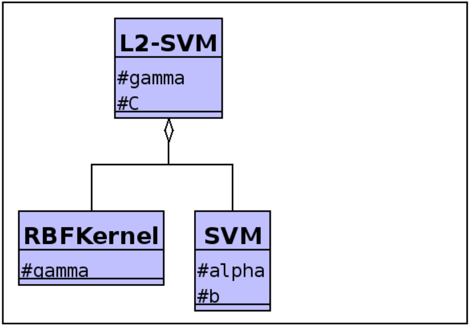
- Communication only through top level interfaces
- Apply any combination of Model, ErrorFunction, and Optimizer

Supervised learning with Shark

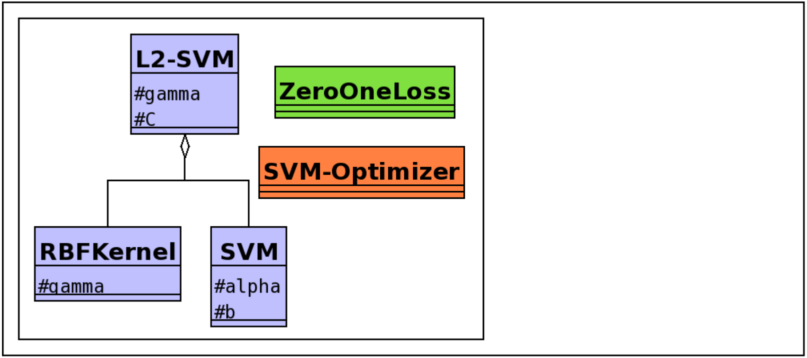
Application: Training a Support Vector Machine

Supervised learning with Shark

Application: Training a Support Vector Machine

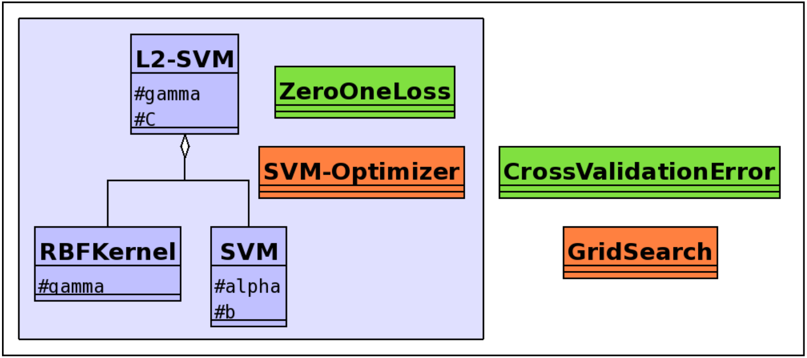


Application: Training a Support Vector Machine

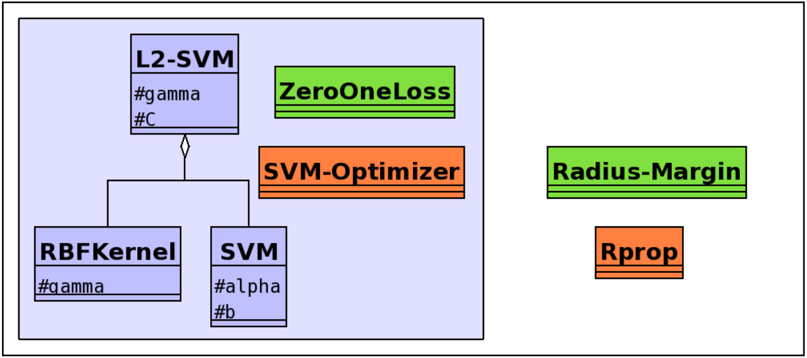


Supervised learning with Shark

Application: Training a Support Vector Machine

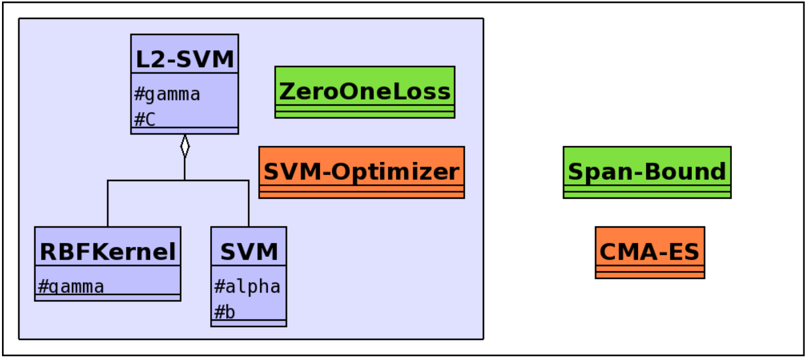


Application: Training a Support Vector Machine



Supervised learning with Shark

Application: Training a Support Vector Machine



Supervised learning with Shark

Advantage

We apply a single coherent optimization framework to different levels of inference!

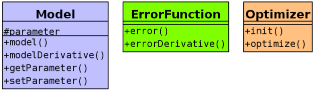
Outline

- Introduction
- Supervised learning with Shark
- Roundtrip through all the rest
- Conclusions

Roundtrip through all the rest

There is a lot more around ...

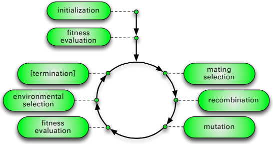
- Many more Models, ErrorFunctions, Optimizers



Roundtrip through all the rest

There is a lot more around ...

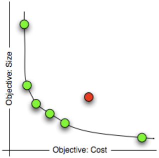
- Many more Models, ErrorFunctions, Optimizers
- Evolutionary Algorithms (EAs)

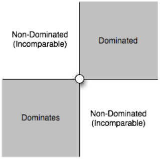


Roundtrip through all the rest

There is a lot more around ...

- Many more Models, ErrorFunctions, Optimizers
- Evolutionary Algorithms (EAs)
- Multi-Objective Optimization (MOO)

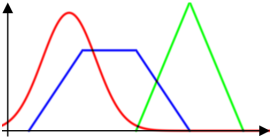




Roundtrip through all the rest

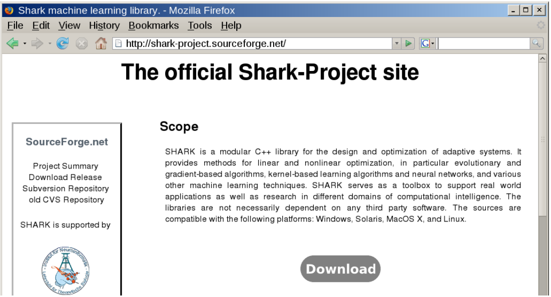
There is a lot more around ...

- Many more Models, ErrorFunctions, Optimizers
- Evolutionary Algorithms (EAs)
- Multi-Objective Optimization (MOO)
- Fuzzy-Logic



Shark on the web

- http://shark-project.sourceforge.net


Shark machine learning library. - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://shark-project.sourceforge.net/
SourceForge.net
Project Summary
Download Release
Subversion Repository
old CVS Repository
SHARK is supported by
Download

The official Shark-Project site

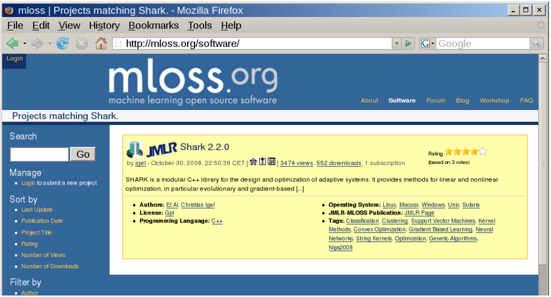
Scope

SHARK is a modular C++ library for the design and optimization of adaptive systems. It provides methods for linear and nonlinear optimization, in particular evolutionary and gradient-based algorithms, kernel-based learning algorithms and neural networks, and various other machine learning techniques. SHARK serves as a toolbox to support real world applications as well as research in different domains of computational intelligence. The libraries are not necessarily dependent on any third party software. The sources are compatible with the following platforms: Windows, Solaris, MacOS X, and Linux.

Roundtrip through all the rest

Shark on the web

- http://shark-project.sourceforge.net
- http://mloss.org



Shark 2.2.0

by igel - October 30, 2008, 22:50:38 CET [ ] 3474 views, 552 downloads, 1 subscription

Rating ★★★★☆ (based on 3 votes)

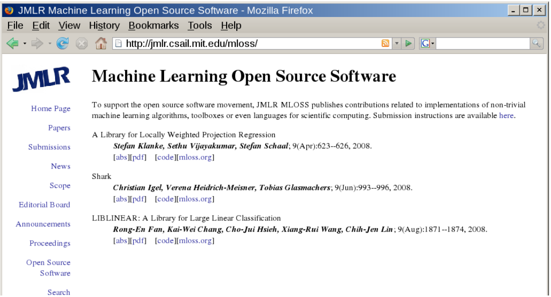
SHARK is a modular C++ library for the design and optimization of adaptive systems. It provides methods for linear and nonlinear optimization, in particular evolutionary and gradient-based [...]

- **Authors:** Et Al, Christian Igel
- **License:** Gpl
- **Programming Language:** C++
- **Operating System:** Linux, Macosx, Windows, Unix, Solaris
- **JMLR-MLOSS Publication:** JMLR Page
- **Tags:** Classification, Clustering, Support Vector Machines, Kernel Methods, Convex Optimization, Gradient Based Learning, Neural Networks, String Kernels, Optimization, Genetic Algorithms, Nips2008

Roundtrip through all the rest

Shark on the web

- http://shark-project.sourceforge.net
- http://mloss.org
- C. Igel, V. Heidrich-Meisner, T. Glasmachers. **Shark**. Journal of Machine Learning Research, Vol. 9, pp. 993–996, 2008.


JMLR Machine Learning Open Source Software - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://jmlr.csail.mit.edu/mloss/
JMLR
Home Page
Papers
Submissions
News
Scope
Editorial Board
Announcements
Proceedings
Open Source Software
Search

Machine Learning Open Source Software

To support the open source software movement, JMLR MLOSS publishes contributions related to implementations of non-trivial machine learning algorithms, toolboxes or even languages for scientific computing. Submission instructions are available here.

A Library for Locally Weighted Projection Regression
Stefan Klanke, Sethu Vijayakumar, Stefan Schaal; 9(Apr):623--626, 2008.
[abs][pdf] [code][mloss.org]

Shark
Christian Igel, Verena Heidrich-Meisner, Tobias Glasmachers; 9(Jun):993--996, 2008.
[abs][pdf] [code][mloss.org]

LIBLINEAR: A Library for Large Linear Classification
Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, Chih-Jen Lin; 9(Aug):1871--1874, 2008.
[abs][pdf] [code][mloss.org]

Outline

- Introduction
- Supervised learning with Shark
- Roundtrip through all the rest
- Conclusions

Conclusions

What makes Shak different?

Conclusions

What makes Shak different?

Three points to take home:

Conclusions

What makes Shak different?

Three points to take home:

- platform independent and self-contained

Conclusions

What makes Shak different?

Three points to take home:

- platform independent and self-contained
- flexible modular design

Conclusions

What makes Shak different?

Three points to take home:

- platform independent and self-contained
- flexible modular design
- covers multiple areas of machine learning

Thank you for your attention!