



Java Machine Learning

by Jason Brownlee on July 19, 2014 in Uncategorized

Like 96 Tweet +1 37 Share 32

Are you a Java programmer and looking to get started or practice machine learning?

Writing programs that make use of machine learning is the best way to learn machine learning. You can write the algorithms yourself from scratch, but you can make a lot more progress if you leverage an existing open source library.

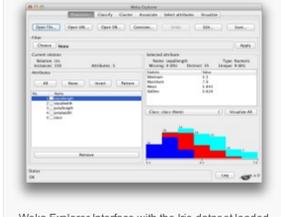
In this post you will discover the major platforms and open source machine learning libraries you can use in Java.

Environments

This section describes Java-based environments or workbenches that can be used for machine learning. They are called environments because they provided graphical user interfaces for performing machine learning tasks, but also provided Java APIs for developing your own applications.

Weka

[Waikato Environment for Knowledge Analysis \(Weka\)](#) is a machine learning platform developed by the University of Waikato, New Zealand. It is written in Java and provides a graphical user interface, command line interface and Java API. It is perhaps the most popular Java machine learning library and a great place to start or practice machine learning.



Weka Explorer Interface with the Iris dataset loaded

KNIME

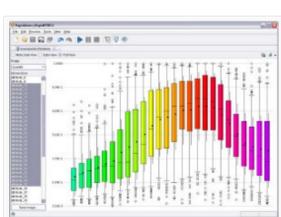
The [Konstanz Information Miner \(KIME\)](#) is an analytics and reporting platform developed by Konstanz University, Germany. It was developed with a focus on pharmaceutical research, but has expanded into general business intelligence. It provides a graphical user interface (based on Eclipse) and a Java API.



Screenshot of KNIME

RapidMiner

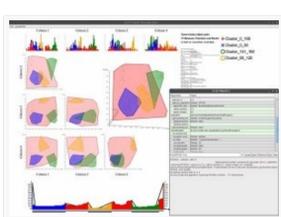
[RapidMiner](#) used to be called Yet Another Learning Environment (YALE) and was developed at Technical University of Dortmund, Germany. It provides a GUI and a Java API for developing your own applications. It provides data handling, visualization and modeling with machine learning algorithms.



RapidMiner Screenshot

ELKI

The [Environment for DeveLoping KDD-Applications Supported by Index-Structures \(ELKI\)](#) is a data mining workbench developed in Java by the Ludwig Maximilian University of Munich, Germany. It has a focus on working with data in relational database for tasks such as outlier detection and classification (distance function based methods). It provides a mini GUI, command line interface and Java API.



ELKI Screenshot

Libraries

Practically every project listed on this page is/has a library with a Java API, those projects listed in this section only provide a Java API. They are machine learning libraries in the narrow sense.

Java-ML

The [Java Machine Learning Library \(Java-ML\)](#) provides a collection of machine learning algorithms implemented in Java. It provides a standard interface for each algorithm, no UIs and references to the relevant scientific literature for further reading. It includes methods for data manipulation, clustering, feature selection and classification. Note that at the time of writing, the last release was in 2012.

JSAT

The [Java Statistical Analysis Tool \(JSAT\)](#) provides pure Java implementations of standard machine learning algorithms for modest sized problems. The author comments that he developed the library partly as a self-education exercise and partly to get things done. Nevertheless the list of algorithms is impressive. It includes classification, regression, ensemble, clustering and feature selection methods.

Big Data

This section lists Java projects intended for use with Big Data, such as on clusters of machines.

Mahout (Hadoop)

[Apache Mahout](#) provides implementations of machine learning algorithms for use on the Apache Hadoop platform (distributed map-reduce). The project provides a focus on clustering and classification algorithms and a popular application driving implementation is its use in collaborative filtering for recommender systems. Reference implementations of algorithms that run on a single node are also included.

MLlib (Spark)

[Apache Machine Learning Library](#) provides implementations of machine learning algorithms for use on the Apache Spark platform (HDFS, but not map-reduce). Although Java, the library and the platform support Java, Scala and Python bindings. The library is new and the list of algorithms is short, but growing quickly.

MOA

[Massive Online Analysis \(MOA\)](#) is an open source platform designed for data stream mining by University of Waikato, New Zealand. Like Weka (developed at the same place), it provides a GUI, command line interface and Java API. It provides a long list of algorithms with focus on classification and support for outlier detection and addressing concept drift. MOA uses the [Advanced Data mining And Machine learning System \(ADAMS\)](#) for managing workflows also developed at the same place.

SAMOA

[Scalable Advanced Massive Online Analysis \(SAMOA\)](#) is a distributed streaming machine learning framework developed by Yahoo!. It is designed to run on Apache Storm and Apache S4. The system can leverage the algorithms provided by the MOA project for tasks like classification.

Natural Language Processing

This section is dedicated to Java libraries and projects for addressing problems from the subfield of machine learning called Natural Language Processing (NLP).

NLP is not my area, so I'll just point to the key libraries.

- **OpenNLP:** [Apache OpenNLP](#) is a toolkit for processing natural language text. It provides methods for NLP tasks such as tokenization, segmentation, and entity extraction.
- **LingPipe:** [LingPipe](#) is a toolkit for computational linguistics and includes methods for topic classification, entity extraction, clustering, and sentiment analysis.
- **GATE:** [The General Architecture for Text Engineering \(GATE\)](#) is an open source library for text processing. It provides an array of sub-projects targeted at different use cases.
- **MALLET:** [Machine Learning for Language Toolkit \(MALLET\)](#) is a Java toolkit for statistical natural language processing, document classification, clustering, topic modeling and information extraction.

Computer Vision

This section lists those libraries for the subfield of machine learning called Computer Vision (CV).

Again, CV is not my area, so I'll just point to the key libraries.

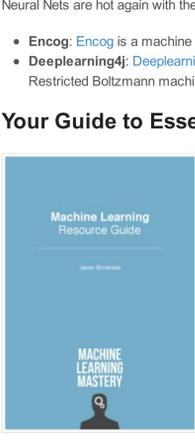
- **BoofCV:** [BoofCV](#) is an open source library for computer vision and robotics applications. It supports features such as image processing, features, geometric vision, calibration, recognition and image data IO.

Deep Learning

Neural Nets are hot again with the development of deep learning methods and faster hardware. This section lists key Java libraries for working with neural networks and deep learning.

- **Encog:** [Encog](#) is a machine learning library that provides algorithms such as SVM, classical neural networks, genetic programming, bayesian networks, HMM and genetic algorithms.
- **DeepLearning4j:** [DeepLearning4j](#) is claimed to be a commercial-grade deep learning library written in Java. It is described as being compatible with Hadoop and provides algorithms including Restricted Boltzmann machines, deep-belief networks and Stacked Denoising Autoencoders.

Your Guide to Essential Machine Learning Resources



I've hand-picked the essential machine learning resources and created an exclusive Resource Guide.

Included are the best books, videos, courses and MUCH more.

I can send it to you right now, for FREE!

[Download For Free](#)

You will also get emails with my best Tips and Tricks for getting started in machine learning.

Summary

In this round-up post we have touched on the big name options when selecting a library or platform for machine learning when working in Java.

These are the players and the popular projects, but by no means is this list complete. For example, take a look at [this page on MLOSS.org](#) that lists (at the time of writing) 71 Java-based open source machine learning projects. That's a lot and I'm sure there are more on GitHub and SourceForge.

The key is to think hard about your own project and it's requirements. Figure out what you need from a library or platform and then pick and learn a project that best fits your needs.



About Jason Brownlee

The editor-in-chief at MachineLearningMastery.com. Jason is a husband, father, researcher, author, professional programmer and a machine learning practitioner. [Learn more about him.](#)

[View all posts by Jason Brownlee](#) →

< How to Tune Algorithm Parameters with Scikit-Learn

Applied Machine Learning Lessons from A Case Study of Passenger Survival Prediction >

No comments yet.

Leave a Reply

[SUBMIT COMMENT](#)

Welcome!

Hi, my name is Jason and welcome to Machine Learning Mastery where we make programmers awesome at machine learning.

If you have any questions or you just want to chat about machine learning, [send me an email](#).

Free Guide to the Best Resources

Discover the best videos, books, courses and MUCH MORE in my Machine Learning Resource Guide.

Get your copy right now!

[Download For Free](#)

POPULAR



A Tour of Machine Learning Algorithms

NOVEMBER 25, 2013



How to Run Your First Classifier in Weka

FEBRUARY 17, 2014



Tutorial to Implement k-Nearest Neighbors in Python From Scratch

SEPTEMBER 12, 2014



Machine Learning for Programmers: Leap from developer to machine learning practitioner

AUGUST 17, 2015



How to Implement Naive Bayes From Scratch in Python

DECEMBER 8, 2014



Discover Feature Engineering, How to Engineer Features and How to Get Good at It

SEPTEMBER 26, 2014



4 Self-Study Machine Learning Projects

JANUARY 3, 2014

Best Machine Learning Resources for Getting Started

NOVEMBER 27, 2013

Find Your Machine Learning Tribe: Get Started And Avoid Getting The Wrong Advice

AUGUST 31, 2015



Looking for something? Search!

Search...

© 2016 Machine Learning Mastery. All Rights Reserved.

[Privacy](#) | [Contact](#) | [About](#)

Machine Learning Crash Course ✕

You can learn and apply Machine Learning even without a background in maths or a fancy degree. Find out how in this ridiculously practical (and totally free) email course.