

IMSL C Numerical Library

The IMSL C Numerical Library takes full advantage of the intrinsic characteristics and desirable features of the C language.

IMSL .NET Numerical Library

As the only numerical library of its kind to offer unprecedented analytic capabilities and charting, the IMSL .NET Numerical Library can be referenced from any .NET language including C#, F#, and Visual Basic.

JMSL Numerical Library

The JMSL Library provides robust data analysis and visualization technology for the Java platform and a fast, scalable framework for tailored analytical applications.

IMSL Fortran Numerical Library

The IMSL Fortran Library combines the powerful and flexible interface features of the Fortran language with the performance benefits of both distributed memory and shared memory multiprocessing architectures.

IMSL NUMERICAL LIBRARIES

The mathematical and statistical power you need when results matter

Analyzing data has never been more important — or harder. Getting actionable results from your large and very large datasets can often determine if your organization meets its goals. Create competitive differentiation and unlock innovation by using the most trusted, tested, and reliable algorithms available. Backed by a team of mathematicians and statisticians, IMSL® Numerical Libraries allow you to address complex problems quickly by using the right algorithm. With IMSL you get consistency from prototype to production.

Accelerate development with proven quality and reliability

IMSL Libraries save development time by providing mathematical and statistical algorithms that can be embedded into C, C++, .NET, Java™, and Fortran applications, including many databases. IMSL enhances application performance, reliability, portability, scalability, and maintainability as well as developer productivity. IMSL Libraries are supported across a wide range of languages as well as hardware and operating system environments including Windows, Linux, and many UNIX platforms.

Robust, flexible algorithms for results that matter

Developers can save weeks, months, or even years of effort by embedding the algorithms from IMSL Libraries versus building them in-house or using open source. Instead of writing hundreds of lines of code to create new algorithms, a developer can make one simple call to a routine that is fully tested, supported, and documented which allows faster time to market. Plus, IMSL Libraries offer superior error handling. Development time is quicker with input verification. Build bullet proof applications by capturing error messages that check algorithm progress, ill-conditioning and numerical instability, and offer suggestions on what to do next.

- Optimized and validated for compatibility, numerical accuracy, and performance on widely adopted platforms
- Numerical algorithms are developed, tested, documented, supported by Rogue Wave Software
- Consistent commercial-quality interfaces yields faster time to market
- Save up to 95 percent of the time required to research and develop algorithms
- IMSL Libraries products includes libraries written in the standard languages of C, C#, Java, and Fortran



USA **800-487-3217**
FRANCE **+33 (0) 1 46 93 94 20**
GERMANY **+49 (0) 6103 5934 0**
UK **+44 (0) 8450 549950**
JAPAN **+81 (0) 3 5211 7760**

www.roguewave.com

© 2014 Rogue Wave Software, Inc.
All Rights Reserved

Rogue Wave provides software development tools for mission-critical applications. Our trusted solutions address the growing complexity of building great software and accelerates the value gained from code across the enterprise. The Rogue Wave portfolio of complementary, cross-platform tools helps developers quickly build applications for strategic software initiatives. With Rogue Wave, customers improve software quality and ensure code integrity, while shortening development cycle times.

The IMSL Libraries and support services emphasize application reliability, performance, embeddability, portability, and supportability.

The algorithms you need to support the work you do

IMSL Libraries delivers a comprehensive set of functional data mining and forecasting, statistics, and mathematical areas.

Mathematical functionality	Statistical functionality	Data mining and forecasting functionality
Matrix Operations	Basic Statistics	Regression
Linear Algebra	Time Series and Forecasting	Cluster Analysis
Eigensystem Analysis	Nonparametric Tests	Neural Networks
Interpolation and Approximation	Correlation and Covariance	Auto_ARIMA
Quadrature	Data Mining	ARMA, GARCH
Differential Equations	Regression	Genetic Algorithm
Feynman-Kac Solver	Analysis of Variance	Naïve Bayes
Transforms	Goodness of Fit	Logistic Regression
Nonlinear Equations	Distribution Functions	Principal Components Analysis
Optimization	Random Number Generation	Factor Analysis
Special Functions	Neural Networks	Variances & Covariances
Utilities	Genetic Algorithm	Discriminant Analysis
	Naïve Bayes	Analysis of Variance
	Classification	Visualization