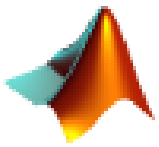
ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-matlab



Peter Reutemann

November 21, 2026

©2021-2022 THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

Except where otherwise noted, this work is licensed under
http://creativecommons.org/licenses/by-sa/4.0/

Contents

1 Introduction **5**

2 Flow **7**

Bibliography **9**

Chapter 1

Introduction

The *adams-matlab* module provides some basic input/output for binary Matlab .mat files (format 5). The heavy lifting under the hood is done by the Matlab File Library (MFL)[4].

Chapter 2

Flow

The following sources are available:

- *NewMat5File* – creates an empty Mat5File object
- *NewMat5Matrix* – creates a new Matrix object
- *NewMat5Struct* – creates a new Struct object

The following transformers are available:

- *Mat5ArrayInfo* – output information on a Matlab Array object.
- *Mat5ArraySubset* – can generate arbitrary subsets of arrays.
- *Mat5FileAddArray* – adds an array (matrix or cell) to a mat file
- *Mat5FileAddMap* – adds the Matlab arrays (of type matrix or cell) from a map to a mat file, using the map keys as entry names
- *Mat5FileInfo* – outputs information on a .mat file
- *Mat5FileReader* – loads a .mat file
- *Mat5GetArrayElement* – retrieves a specific element from a n-dimensional matrix.
- *Mat5GetStructField* – retrieves an array by name from a struct object.
- *Mat5SetArrayElement* – sets a specific element in a n-dimensional matrix.
- *Mat5SetStructField* – sets an array by name in a struct object.
- *Mat5StructInfo* – output information on a Matlab Struct object.
- *MultiMat5MatrixOperation* – performs an operation on an array of matrices passing through.

The following sinks are available:

- *Mat5FileWriters* – writes .mat files

The following conversions are available:

- *DoubleMatrixToMat5Array* – converts a double matrix into a 2-dimensional Matlab array
- *MapToMat5File* – converts a Java map into a Matlab file object
- *MapToMat5Struct* – converts a Java map into a Matlab struct object
- *Mat5ArrayToDoubleMatrix* – converts a (2-dimensional) Matlab array into a double matrix

- *Mat5ArrayToSpreadSheet* – converts a (2-dimensional) Matlab array into a spreadsheet
- *Mat5ArrayToString* – converts a (2-dimensional) Matlab array into a human-readable string
- *Mat5CharToString* – converts a Matlab Char object to a string
- *Mat5FileToMap* – converts a Matlab file object into a map
- *Mat5StructToMap* – converts a Matlab struct object into a map
- *StringToMat5Array* – parses a matrix string (e.g., $[1, 2; 3, 4]$) and generates a matrix array
- *SpreadSheetHeaderToMat5Array* – turns the spreadsheet header (ie the column names) into an array
- *SpreadSheetToMat5Array* – turns a spreadsheet into an array (matrix or cell)

The following boolean conditions are available:

- *IsMat5Array* – checks whether the token contains a Matlab array object.
- *IsMat5Matrix* – checks whether the token contains a Matlab matrix object.
- *IsMat5Struct* – checks whether the token contains a Matlab struct object.

Bibliography

[1] *ADAMS* – Advanced Data mining and Machine learning System
`https://adams.cms.waikato.ac.nz/`

[2] *Matlab* – a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks
`https://www.mathworks.com/products/matlab.html`

[3] *octave* – software featuring a high-level programming language, primarily intended for numerical computations
`https://gnu.org/software/octave/`

[4] *MAT File Library* – a Java library for reading and writing MAT Files that are compatible with MATLAB's MAT-File Format.
`https://github.com/HebiRobotics/MFL`