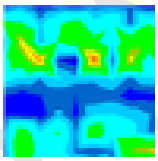
ADAMS

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

Module: adams-heatmap



Peter Reutemann

June 23, 2015

DRAFT

©2011-2015



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato



Except where otherwise noted, this work is licensed under
http://creativecommons.org/licenses/by-sa/3.0/

Contents

1 Introduction **7**

2 Flow **9**

3 Tools **11**

Bibliography **13**

List of Figures

3.1 Heatmap viewer displaying a heatmap with the associated histogram. 11

Chapter 1

Introduction

According to WikiPedia [2], a "heat map is a graphical representation of data where the individual values contained in a matrix are represented as colors."

Chapter 2

Flow

The following conversions are available:

- *BufferedImageToHeatmap* – turns a BufferedImage into a heat map, using the RGB values (but not alpha).
- *HeatmapToArray* – generates a double array from a heat map.
- *HeatmapToBufferedImage* – generates an image from heat map.
- *HeatmapToSpreadSheet* – converts the heat map into a spreadsheet object.
- *SpreadSheetToHeatmap* – creates a heat map from an all-numeric spreadsheet.

The following transformers are available:

- *NewHeatmap* – creates an empty heatmap.

The following transformers are available:

- *HeatmapArrayStatistics* – interprets a heatmap as array (row-wise concatenated) and generates statistics from it.
- *HeatmapFeatureGenerator* – generates features from the heatmap passing through in various formats using the specific feature generator.
- *HeatmapFileReader* – reads a heat map from disk with a specified reader.
- *HeatmapFileWriter* – writes a heat map back to disk with a custom writer.
- *HeatmapFilter* – transform a heat map using a filter.
- *HeatmapGetValue* – retrieves one or more values from a heatmap.
- *HeatmapInfo* – outputs information on a heatmap.
- *HeatmapInstanceGenerator* – turns a heat map into a WEKA instance.
- *HeatmapLocateObjects* – generates sub-heatmaps from objects that were located in the heatmap.
- *HeatmapSetValue* – set one or more values in a heatmap.

The following sinks are available:

- *HeatmapDisplay* – displays a heatmap.
- *HeatmapHistogram* – displays a histogram generated from a heatmap.

Chapter 3

Tools

The *Heatmap viewer* allows you to load, process and analyze heatmaps from a variety of formats. See Figure 3.1 for a screenshot.

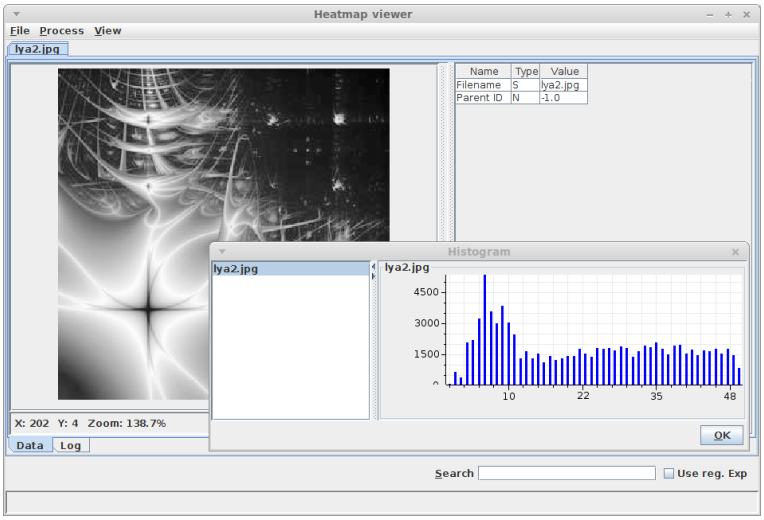


Figure 3.1: Heatmap viewer displaying a heatmap with the associated histogram.

The viewer also allows you to add custom plugins to the menu. The superclass for all plugins is:

`adams.gui.visualization.heatmap.plugins.AbstractHeatmapViewerPlugin`

- *canExecute(HeatmapPanel)* – determines whether the plugin can be applied to the current heatmap panel.
- *doExecute()* – executes the plugin and returns a string depending on success (*null*) or failure (*error message*).
- *createLogEntry()* – can be used to output a string that should appear the in viewer's log tab; return *null* if that does not apply.

Bibliography

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

[2] *Heat map* – WikiPedia article
`http://en.wikipedia.org/wiki/Heat_map`