gesvd_result

NAME

gesvd_result - a structure to model the output of frovedis singular value decomposition methods.

SYNOPSIS

import com.nec.frovedis.matrix.GesvdResult

Public Member Functions

`SingularValueDecomposition[RowMatrix,Matrix]` to_spark_result(SparkContext sc)
Unit save(String svec, String umat, String vmat)
Unit savebinary(String svec, String umat, String vmat)
Unit load_as_colmajor(String svec, String umat, String vmat)
Unit load_as_blockcyclic(String svec, String umat, String vmat)
Unit loadbinary_as_colmajor(String svec, String umat, String vmat)
Unit loadbinary_as_blockcyclic(String svec, String umat, String vmat)
Unit debug_print()
Unit release()
Int stat()

DESCRIPTION

GesvdResult is a client spark side pseudo result structure containing the proxies of the in-memory SVD results created at frovedis server side. It can be used to convert the frovedis side SVD result to spark equivalent data structures.

Public Member Function Documentation

`SingularValueDecomposition[RowMatrix,Matrix]` to_spark_result(SparkContext sc)

This function can be used to convert the frovedis side SVD results to spark equivalent result structure (`SingularValueDecomposition[RowMatrix,Matrix]`). Internally it uses the SparkContext object while performing this conversion.

save(String svec, String umat, String vmat)

This function can be used to save the result values in different text files at server side. If saving of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

savebinary(String svec, String umat, String vmat)

This function can be used to save the result values in different little-endian binary files at server side. If saving of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

load_as_colmajor(String svec, String umat, String vmat)

This function can be used to load the target result object with the values in given text files. If loading of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

If "umat" and/or "vmat" filenames are given, they will be loaded as frovedis distributed column major matrix.

load_as_blockcyclic(String svec, String umat, String vmat)

This function can be used to load the target result object with the values in given text files. If loading of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

If "umat" and/or "vmat" filenames are given, they will be loaded as frovedis distributed blockcyclic matrix.

loadbinary_as_colmajor(String svec, String umat, String vmat)

This function can be used to load the target result object with the values in given little-endian binary files. If loading of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

If "umat" and/or "vmat" filenames are given, they will be loaded as frovedis distributed column major matrix.

loadbinary_as_blockcyclic(String svec, String umat, String vmat)

This function can be used to load the target result object with the values in given little-endian binary files. If loading of U and V components are not required, "umat" and "vmat" can be null, but "svec" should have a valid filename.

If "umat" and/or "vmat" filenames are given, they will be loaded as frovedis distributed blockcyclic matrix.

Unit debug_print()

This function can be used to print the result components at server side user terminal. This is useful in debugging purpose.

Unit release()

This function can be used to release the in-memory result components at frovedis server.

Int stat()

This function returns the exit status of the scalapack native gesvd routine on calling of which the target result object was obtained.