root

Go Up

Name	SupportVectorMachines
Version	1.1
Description	Support Vector Machines Bundle
License	http://www.apache.org/licenses/LICENSE-2.0
Copyright	Copyright (C) 2017 HPCC Systems
Authors	HPCCSystems
DependsOn	ML_Core, PBblas
Platform	6.2.0

Table of Contents

Support Vector Machine Regression
train.ecl Train SVM classification and regression models
Types.ecl SupportVectorMachines type definitions
Datasets
Examples
libsvm
Performance
Validation

SupportVectorMachines

July 6, 2022

Support Vector Machine implementations.

Prerequisites: this bundle is a wrapper for the LibSVM library, and so requires LibSVM be installed on all nodes of the cluster. Note: some distributions may have a development version, and that the development version is required if there is more than one version.

Confusion

Go Up

IMPORTS

_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

CONFUSION Confusion

DATASET(Confusion_Detail)	**Confusion**
(DATASET(DiscreteField) dependents, DATASET(DiscreteField) predicts)	

Generate the confusion matrix, to compare actual versus predicted response variable values.

PARAMETER **dependents** ||| TABLE (DiscreteField) — the original response values.

PARAMETER **predicts** ||| TABLE (DiscreteField) — the predicted responses.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , INTEGER4 actual_class , INTEGER4 predict_class , UNSIGNED4 occurs , BOOLEAN correct , REAL8 pctActual , REAL8 pctPred })** — confusion matrix in Confusion_Detail format.

INTERNAL

SEE ML_Core.Types.Confusion_Detail

Converted

Go Up

IMPORTS

_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

CONVERTED Converted

	Converted

Module for various data conversions between standard ML Bundle formats and those used by libSVM.

INTERNAL

Children

1. ToInstance : Convert dataset in NumericField format (with separate NumericFields for dependent and independent variables) to an SVM_Instance format
2. FromModel : Convert from SVM Model type to standardized Layout_Model format
3. ToModel : Convert from standardized ML Layout_Model format to SVM Model format used by libSVM

TOINSTANCE ToInstance

Converted \

	ToInstance
	`(DATASET(ML_Types.NumericField) Ind, DATASET(ML_Types.NumericField) Dep=dummy)`

Convert dataset in NumericField format (with separate NumericFields for dependent and independent variables) to an SVM_Instance format.

PARAMETER **Ind** ||| TABLE (NumericField) — NumericField dataset of independent variables.

PARAMETER **Dep** ||| TABLE (NumericField) — NumericField dataset of dependent variable(s) (default: empty dataset).

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 rid , REAL8 y , REAL8 max_value , TABLE (SVM_Feature) x })** — Dataset converted to SVM_Instance format.

SEE SupportVectorMachines.Types.SVM_Instance

FROMMODEL FromModel

Converted \

	FromModel
	`(UNSIGNED id_base = 1000, DATASET(SVM.Types.Model) mdl)`

Convert from SVM Model type to standardized Layout_Model format. The Layout_Model format is harder to interpret, but more generalized.

PARAMETER **id_base** ||| UNSIGNED8 — Base number from which to start model IDs (default: 1000).

PARAMETER **mdl** ||| TABLE (Model) — Object of SVM Model type (see Types.Model).

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value })** — Convert SVM model in Layout_Model format (see ML_Core.Types for format definition).

SEE Types.Model

SEE ML_Core.Types.Layout_Model

TOMODEL ToModel

Converted \

	ToModel
`(DATASET(ML_Types.Layout_Model) mdl)`	

Convert from standardized ML Layout_Model format to SVM Model format used by libSVM. The SVM model format is less general, but easier to interpret.

PARAMETER **mdl** ||| TABLE (Layout_Model) — Trained SVM in Layout_Model format.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 gamma , REAL8 coef0 , UNSIGNED4 k , UNSIGNED4 l , BOOLEAN scale , TABLE (FeatureStats) scaleInfo , TABLE (SVM_SV) sv , TABLE (R8Entry) sv_coef , TABLE (R8Entry) rho , TABLE (R8Entry) probA , TABLE (R8Entry) probB , TABLE (I4Entry) labels , TABLE (I4Entry) nSV })** — Converted SVM model in SVM Model format.

SEE ML_Core.Types.Layout_Model

SEE Types.Model

CrossValidate

Go Up

IMPORTS

Types | libsvm.Types | libsvm | libsvm.Converted |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

CROSSVALIDATE CrossValidate

	CrossValidate
	`( DATASET(Parms) p = DATASET(Types.Training_Parameters_Default), DATASET(ML_Types.NumericField) observations, DATASET(ML_Types.NumericField) actuals, UNSIGNED2 folds = 10)`

Perform n-fold cross-validation to assess the performance of the given model parameters.

PARAMETER **p** ||| TABLE (Parms) — The parameters which define the model(s) to be trained.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values.

PARAMETER **actuals** ||| TABLE (NumericField) — The dependent variable(s).

PARAMETER **folds** ||| UNSIGNED2 — No Doc

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq })** — Dataset of cross-validated scores.

INTERNAL

GridSearch

Go Up

IMPORTS

Types | std.system.Thorlib | _versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

GRIDSEARCH GridSearch

	GridSearch
	`( Grid_Plan plan = Types.SVM_Grid_Plan_Default, T_Base base = Types.Training_Base_Default, DATASET(ML_Types.NumericField) observations, DATASET(ML_Types.NumericField) actuals)`

Perform grid search over parameters gamma and C. The grid resolution is increased automatically to utilize any otherwise idle nodes. For a single given set of model parameters, models can be tuned to a number of datasets by concatenating multiple datasets into single 'observations' and 'actuals' datasets, with separate datasets being identified by a work ID column, 'wi'.

PARAMETER **plan** ||| ROW (Grid_Plan) — A structure defining preferences for the grid resolution and number of CV folds used in evaluation of candidate models. In SVM_Grid_Plan format.

PARAMETER **base** ||| ROW (T_Base) — The fixed model parameters (those other than gamma and C) in Training_Base format.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **actuals** ||| TABLE (NumericField) — The observed dependent varible(s) used to build the model(s) in NumericField format.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 coef0 , REAL8 eps , REAL8 nu , REAL8 p , INTEGER4 nr_weight , BOOLEAN shrinking , BOOLEAN prob_est , BOOLEAN scale , TABLE (I4Entry) lbl , TABLE (R8Entry) weight , REAL8 gamma , REAL8 C })** — Dataset with sets of model parameters and corresponding cross-validated scores in GridSearch_Result format.

INTERNAL

SEE Types.SVM_Grid_Plan

SEE Types.Training_Base

SEE ML_Core.Types.NumericField

SEE Types.GridSearch_Result

ModelSummary

Go Up

IMPORTS

_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

MODELSUMMARY ModelSummary

DATASET({UNSIGNED4 r, STRING60 txt})	**ModelSummary**
(DATASET(Core_Types.Layout_Model) model)	

Generate human-readable summary of SVM models.

PARAMETER **model** ||| TABLE (Layout_Model) — Trained SVM models in Layout_Model format.

RETURN **TABLE ({ UNSIGNED4 r , STRING60 txt })** — A single-column dataset which contains human-readable information about the SVM models.

INTERNAL

predict

Go Up

IMPORTS

libsvm | _versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

PREDICT Predict

	Predict
(DATASET(Model) models, DATASET(ML_Types.NumericField) observations)	

Module for generating predictions on data from SVM models.

PARAMETER **models** ||| TABLE (Model) — Trained SVM models.

PARAMETER **observations** ||| TABLE (NumericField) — Independent variables to apply model to and generate predictions.

INTERNAL

Children

1. Prediction : Get predictions (classes or values) only
2. Pred_Values : Get predictions (classes or values) and decision values
3. Pred_Prob_Est : Get predictions (classes or values) and probabilities

PREDICTION Prediction

Predict \

	Prediction

Get predictions (classes or values) only.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED8 rid , REAL8 target_y , REAL8 predict_y })** — Dataset with predictions.

PRED_VALUES Pred_Values

Predict \

	Pred_Values

Get predictions (classes or values) and decision values. Values are empty if not supported.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED8 rid , REAL8 target_y , REAL8 predict_y , TABLE (R8Entry) decision_values })** — Dataset with predictions and decision values.

PRED_PROB_EST Pred_Prob_Est

Predict \

	Pred_Prob_Est

Get predictions (classes or values) and probabilities. Values are empty if not supported.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED8 rid , REAL8 target_y , REAL8 predict_y , TABLE (R8Entry) prob_estimates })** — Dataset with predictions and probabilities.

Scale

Go Up

IMPORTS

Types | libsvm | libsvm.Types | _versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

SCALE Scale

	Scale
(Problem prob, BOOLEAN Scale = true)	

Scaling and column statistics for datasets in problem format.

PARAMETER **prob** ||| ROW (Problem) — Dataset in problem format.

PARAMETER **Scale** ||| BOOLEAN — Optional boolean value indicating whether or not data should be centered and scaled (default: true).

INTERNAL

Children

1. stats : Get means and standard deviations for a set of predictors
2. problemScaled : Standardize (center and scale) dataset in problem data format

STATS stats

Scale \

	stats

Get means and standard deviations for a set of predictors.

RETURN **TABLE ({ INTEGER4 indx , REAL8 mean , REAL8 sd })** — Dataset with one row per predictor with columns for the mean and standard deviation of each. This can be passed later to ”problemScaled” to standardize data appropriately.

PROBLEMSCALED problemScaled

Scale \

	problemScaled

Standardize (center and scale) dataset in problem data format.

RETURN **ROW (Problem)** — A standardized replica of the input problem dataset.

SVC

Go Up

IMPORTS

Types | libsvm.Types | _versions.PBblas.V3_0_2.PBblas |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |
_versions.ML_Core.V3_2_2.ML_Core.Interfaces |

DESCRIPTIONS

SVC SVC

	SVC

```
( Types.SVM_Type svmType = LibSVM_Types.LibSVM_Type.C_SVC, Types.Kernel_Type
kernelType = LibSVM_Types.LibSVM_Kernel.RBF, REAL8 gamma = 0.05, REAL8 C = 1,
INTEGER4 degree = 3, REAL8 coef0 = 0.0, REAL8 eps = 0.001, REAL8 nu = 0.5, REAL8 p
= 0.1, BOOLEAN shrinking = true, BOOLEAN prob_est = true, BOOLEAN scale = true,
INTEGER4 nr_weight = 0, DATASET(Types.I4Entry) lbl = DATASET([], Types.I4Entry),
DATASET(Types.R8Entry) weight = DATASET([], Types.R8Entry))
```

Support vector machine classification.

Utilizes the open-source libSVM under the hood.

This module is appropriate for small to medium sized Machine Learning problems or multitudes of small-to-medium problems using the Myriad interface.

This is due to both scaling limitations endemic to SVM, as well as the fact that libSVM runs independently on each node, and cannot, therefore scale to very large single problems.

Other techniques should be employed for Machine Learning with more than 10,000 data points.

This module also provides a mechanism for doing a grid search for regularization parameters using the full resources of the HPCC cluster rather than searching sequentially (see GridSearch.ecl).

PARAMETER **svmType** ||| UNSIGNED2 — The SVC type, which may be one of 0 (C_SVC, default), 1 (NU_SVC), or 2 (ONE_CLASS).

PARAMETER **kernelType** ||| UNSIGNED2 — The kernel used in training and predicting, which may be one of 0 (LINEAR), 1 (POLY), 2 (RBF, default), 3 (SIGMOID), or 4 (PRECOMPUTED).

PARAMETER **gamma** ||| REAL8 — Parameter needed for all kernels except LINEAR (default: 0.05).

PARAMETER **C** ||| REAL8 — Cost of constraint violation (default: 1).

PARAMETER **degree** ||| INTEGER4 — Parameter needed for kernel of type POLY (default: 3).

PARAMETER **coef0** ||| REAL8 — Parameter needed for kernels of type POLY and SIGMOID (default: 0).

PARAMETER **eps** ||| REAL8 — Tolerance of termination criterion (default: 0.001).

PARAMETER **nu** ||| REAL8 — Parameter needed for NU_SVC and ONE_CLASS (default: 0.5).

PARAMETER **p** ||| REAL8 — Epsilon in the insensitive-loss function (default: 0.1).

PARAMETER **shrinking** ||| BOOLEAN — Flag indicating the use of shrinking-heuristics (default: true).

PARAMETER **prob_est** ||| BOOLEAN — Whether to train for probability estimates (default true).

PARAMETER **scale** ||| BOOLEAN — Whether to standardize the data (subtract mean, divide by sd) before fitting.

PARAMETER **nr_weight** ||| INTEGER4 — The number of elements in the 'lbl' parameter (default: 0).

PARAMETER **lbl** ||| TABLE (I4Entry) — Labels to indicate classes, used with the 'weight' parameter (default: []).

PARAMETER **weight** ||| TABLE (R8Entry) — Class weights, assigned to classes using the 'lbl' parameter (default: []).

PARENT **_versions.ML_Core.V3_2_2.ML_Core.Interfaces.IClassify** </home/amar/.HPCCSystems/bundles/_versions/ML_Core/V3_2_2/ML_Core/Interfaces/IClassify.ecl>

Children

1. GetModel : Calculate a model to fit the observation data to the observed classes
2. Classify : Classify the values for new observations using models trained by the GetModel function

3. Report : Report the confusion matrix for the classifier and training data
4. Tune : Perform a regularization tuning in order to align the granularity of the algorithm with the complexity of the data
5. GetTunedModel : Choose the best set of regularization parameters and use it to train the models
6. CrossValidate : Perform n-fold cross-validation of a given model for each work-item
7. ModelSummary : Generate human-readable model summary of trained SVM model(s)

GETMODEL GetModel

SVC \

DATASET(Layout_Model)	**GetModel**
(DATASET(NumericField) observations, DATASET(DiscreteField) classifications)	

Calculate a model to fit the observation data to the observed classes. For a single given set of model parameters, models can be fit to a number of datasets by concatenating multiple datasets into single 'observations' and 'classifications' datasets, with separate datasets being identified by a work-item number ('wi'), in the NumericField and DiscreteField datasets.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **classifications** ||| TABLE (DiscreteField) — The observed classification used to build the model in DiscreteField format.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value })** — The encoded models in Layout_Model format.

SEE ML_Core.Types.NumericField

SEE ML_Core.Types.DiscreteField

SEE ML_Core.Types.Layout_Model

OVERRIDE

CLASSIFY Classify

SVC \

DATASET(ML_Types.Classify_Result)	**Classify**
(DATASET(Layout_Model) model, DATASET(NumericField) new_observations)	

Classify the values for new observations using models trained by the GetModel function.

PARAMETER **model** ||| TABLE (Layout_Model) — The models, which should be produced by a corresponding GetModel function.

PARAMETER **new_observations** ||| TABLE (NumericField) — Observations to be classified in NumericField format.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , INTEGER4 value , REAL8 conf })** — Classifications with a probability value in Classify_Results format.

SEE ML_Core.Types.NumericField

SEE ML_Core.Types.Classify_Results

OVERRIDE

REPORT Report

SVC \

DATASET(ML_Types.Confusion_Detail)	**Report**
(DATASET(Layout_Model) model, DATASET(NumericField) observations, DATASET(DiscreteField) classifications)	

Report the confusion matrix for the classifier and training data.

PARAMETER **model** ||| TABLE (Layout_Model) — The models, which should be produced by a corresponding GetModel function.

PARAMETER **observations** ||| TABLE (NumericField) — The explanatory values in NumericField format.

PARAMETER **classifications** ||| TABLE (DiscreteField) — The classifications associated with the observations in DiscreteField format.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , INTEGER4 actual_class , INTEGER4 predict_class , UNSIGNED4 occurs , BOOLEAN correct , REAL8 pctActual , REAL8 pctPred })** — The confusion matrix showing correct and incorrect results in Confusion_Detail format.

SEE ML_Core.Types.NumericField

SEE ML_Core.Tyeps.DiscreteField

SEE ML_Core.Types.Confusion_Detail

OVERRIDE

TUNE Tune

SVC \

DATASET(Types.GridSearch_Result)	**Tune**

```
( INTEGER4 folds = 10, REAL8 start_log2C = -5,
REAL8 stop_log2C = 15, REAL8 maxIncr_log2C = 2,
REAL8 start_log2gamma = -15, REAL8 stop_log2gamma =
3, REAL8 maxIncr_log2gamma = 2,
DATASET(NumericField) observations,
DATASET(DiscreteField) classifications)
```

Perform a regularization tuning in order to align the granularity of the algorithm with the complexity of the data. This is to avoid under or over fitting of the data.

Finds a reasonable setting for the regularization parameters gamma and C by performing a grid search over them and testing each using cross-validation. The parameters that provide the lowest out-of-sample error (i.e. when tested on data not in the training set) are the ones chosen.

Returns a set of training parameter combinations and their results that can then be passed to GetTunedModel below to acquire a model that has been properly regularized.

The grid resolution is increased automatically to utilize any otherwise idle nodes.

For a single given set of model parameters, models can be tuned to a number of datasets by concatenating multiple datasets into single 'observations' and 'classifications' datasets, with separate datasets being identified by a work ID column, 'wi'.

PARAMETER **folds** ||| INTEGER4 — The number of cross-validation folds for evaluating each candidate model.

PARAMETER **start_log2C** ||| REAL8 — The lower bound for log2(C): C >= 2^(start_log2C).

PARAMETER **stop_log2C** ||| REAL8 — The upper bound for log2(C): C <= 2^(start_log2C).

PARAMETER **maxIncr_log2C** ||| REAL8 — Taximum allowable exponential increment for C.

PARAMETER **start_log2gamma** ||| REAL8 — The lower bound for log2(gamma): gamma >= 2^(start_log2gamma).

PARAMETER **stop_log2gamma** ||| REAL8 — The upper bound for log2(gamma): gamma <= 2^(start_log2gamma).

PARAMETER **maxIncr_log2gamma** ||| REAL8 — Taximum allowable exponential increment for gamma.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **classifications** ||| TABLE (DiscreteField) — The observed classification used to build the model in DiscreteField format.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 coef0 , REAL8 eps , REAL8 nu , REAL8 p , INTEGER4 nr_weight , BOOLEAN shrinking , BOOLEAN prob_est , BOOLEAN scale , TABLE (I4Entry) lbl , TABLE (R8Entry) weight , REAL8 gamma , REAL8 C })** — Dataset with sets of model parameters and corresponding cross-validated scores in GridSearch_Result format.

SEE GetTunedModel

SEE Types.GridSearch_Result

GETTUNEDMODEL GetTunedModel

SVC \

DATASET(Layout_Model)	**GetTunedModel**
(DATASET(Types.GridSearch_Result) tuneResult, DATASET(NumericField) observations, DATASET(DiscreteField) classifications)	

Choose the best set of regularization parameters and use it to train the models. Using the output of Tune(), find the best set of modeling parameters for each work id, and train the corresponding models. The the most regularized (i.e. coarsest) set of parameters that achieved near-maximum performance is used to create the models.

PARAMETER **tuneResult** ||| TABLE (GridSearch_Result) — The results of a grid search over C and gamma, produced by Tune().

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **classifications** ||| TABLE (DiscreteField) — The observed classification used to build the model in DiscreteField format.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value })** — The encoded models in Layout_Model format.

SEE Tune

SEE ML_Core.Types.NumericField

SEE ML_Core.Types.DiscreteField

SEE ML_Core.Types.Layout_Model

CROSSVALIDATE CrossValidate

SVC \

DATASET(Types.CrossValidate_Result)	**CrossValidate**
(INTEGER4 folds = 10, DATASET(NumericField) observations, DATASET(DiscreteField) classifications)	

Perform n-fold cross-validation of a given model for each work-item.

For a single given set of model parameters, models can be cross-validated against a number of datasets by concatenating multiple datasets into single 'observations' and 'classifications' datasets, with separate datasets being identified by a work ID column, 'wi'.

PARAMETER **folds** ||| INTEGER4 — The number of cross-validation folds.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **classifications** ||| TABLE (DiscreteField) — The observed classification used to build in DiscreteField format.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq })** — Dataset of cross-validated scores in CrossValidate_Result format.

SEE ML_Core.NumericField

SEE ML_Core.DiscreteField

SEE Types.CrossValidate_Result

MODELSUMMARY ModelSummary

SVC \

DATASET({UNSIGNED4 r, STRING60 Txt})	**ModelSummary**
(DATASET(Layout_Model) model)	

Generate human-readable model summary of trained SVM model(s).

Multiple models can be simultaneously summarized by concatenating a number of models into a single 'model' object, with separate models being identified by a work ID column, 'wi'.

PARAMETER **model** ||| TABLE (Layout_Model) — The models, which should be produced by a corresponding GetModel function.

RETURN **TABLE ({ UNSIGNED4 r , STRING60 txt })** — Single-column dataset with textual description of models.

SVR

Go Up

IMPORTS

Types | libsvm.Types | _versions.PBblas.V3_0_2.PBblas |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |
_versions.ML_Core.V3_2_2.ML_Core.Interfaces |

DESCRIPTIONS

SVR SVR

	SVR
	(DATASET(NumericField) X = DATASET([], NumericField), DATASET(NumericField) Y = DATASET([], NumericField), Types.SVM_Type svmType = LibSVM_Types.LibSVM_Type.C_SVC, Types.Kernel_Type kernelType = LibSVM_Types.LibSVM_Kernel.RBF, REAL8 gamma = 0.05, REAL8 C = 1, INTEGER4 degree = 3, REAL8 coef0 = 0.0, REAL8 eps = 0.001, REAL8 nu = 0.5, REAL8 p = 0.1, BOOLEAN shrinking = true, BOOLEAN prob_est = true, BOOLEAN scale = true, INTEGER4 nr_weight = 0, DATASET(Types.I4Entry) lbl = DATASET([], Types.I4Entry), DATASET(Types.R8Entry) weight = DATASET([], Types.R8Entry))

Support Vector Machine Regression.

Utilizes the open-source libSVM under the hood.

This module is appropriate for small to medium sized Machine Learning problems or multitudes of small-to-medium problems using the Myriad interface.

This is due to both scaling limitations endemic to SVM, as well as the fact that libSVM runs

independently on each node, and cannot, therefore scale to very large single problems.

Other techniques should be employed for Machine Learning with more than 10,000 data points.

This module also provides a mechanism for doing a grid search for regularization parameters using the full resources of the HPCC cluster rather than searching sequentially (see GridSearch.ecl).

PARAMETER **X** ||| TABLE (NumericField) — The observed explanatory values in NumericField format.

PARAMETER **Y** ||| TABLE (NumericField) — The observed values the model aims to fit in NumericField format.

PARAMETER **svmType** ||| UNSIGNED2 — The SVR type, which may be one of 3 (EPSILON_SVR, default), or 4 (NU_SVR).

PARAMETER **kernelType** ||| UNSIGNED2 — The kernel used in training and predicting, which may be one of 0 (LINEAR), 1 (POLY), 2 (RBF, default), 3 (SIGMOID), or 4 (PRECOMPUTED).

PARAMETER **gamma** ||| REAL8 — regularization parameter needed for all kernels except LINEAR (default: 0.05).

PARAMETER **C** ||| REAL8 — Cost of constraint violation regularization parameter(default: 1).

PARAMETER **degree** ||| INTEGER4 — Parameter needed for kernel of type POLY (default: 3).

PARAMETER **coef0** ||| REAL8 — Parameter needed for kernels of type POLY and SIGMOID (default: 0).

PARAMETER **eps** ||| REAL8 — Tolerance of termination criterion (default: 0.001).

PARAMETER **nu** ||| REAL8 — Parameter needed for NU_SVC and ONE_CLASS (default: 0.5).

PARAMETER **p** ||| REAL8 — Epsilon in the insensitive-loss function (default: 0.1).

PARAMETER **shrinking** ||| BOOLEAN — Flag indicating the use of shrinking-heuristics (default: true).

PARAMETER **prob_est** ||| BOOLEAN — Whether to train for probability estimates (default true).

PARAMETER **scale** ||| BOOLEAN — Whether to standardize the data (subtract mean, divide by sd) before fitting.

PARAMETER **nr_weight** ||| INTEGER4 — No Doc

PARAMETER **lbl** ||| TABLE (I4Entry) — No Doc

PARAMETER **weight** ||| TABLE (R8Entry) — No Doc

SEE ML_Core.Types.NumericField

PARENT **_versions.ML_Core.V3_2_2.ML_Core.Interfaces.IRegression**
</home/amar/.HPCCSystems/bundles/_versions/ML_Core/V3_2_2/ML_Core/Interfaces/IRegression.e

Children

1. GetModel : Train and return a model that fits the observation data to the observed values
2. Predict : Predict values for the new observations using models trained by the GetModel function
3. Tune : Perform a regularization tuning in order to align the granularity of the algorithm with the complexity of the data
4. GetTunedModel : Choose the best set of regularization parameters and use it to train the models
5. CrossValidate : Perform n-fold cross-validation of a given model for each work ID
6. ModelSummary : Generate human-readable model summary of trained SVM model(s)

GETMODEL GetModel

SVR \

DATASET(Layout_Model)	**GetModel**

Train and return a model that fits the observation data to the observed values. For a single given set of model parameters, models can be fit to a number of datasets by concatenating multiple datasets into single 'X' and 'Y' datasets, with separate datasets being identified by a work-item column, 'wi'.

RETURN — The encoded models in Layout_Model format.

SEE ML_Core.Types.Layout_Model

OVERRIDE

PREDICT Predict

SVR \

DATASET(NumericField)	**Predict**
(DATASET(NumericField) newX, DATASET(Layout_Model) model)	

Predict values for the new observations using models trained by the GetModel function.

PARAMETER **model** ||| TABLE (Layout_Model) — The models, which should be produced by a corresponding GetModel function.

PARAMETER **newX** ||| TABLE (NumericField) — Observations to be classified in NumericField format.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value })** — Predictions in NumericField format.

SEE ML_Core.Types.NumericField

OVERRIDE

TUNE Tune

SVR \

DATASET(Types.GridSearch_Result)	**Tune**
(INTEGER4 folds = 10, REAL8 start_log2C = -5, REAL8 stop_log2C = 15, REAL8 maxIncr_log2C = 2, REAL8 start_log2gamma = -15, REAL8 stop_log2gamma = 3, REAL8 maxIncr_log2gamma = 2)	

Perform a regularization tuning in order to align the granularity of the algorithm with the complexity of the data. This is to avoid under or over fitting of the data.

Finds a reasonable setting for the regularization parameters gamma and C by performing a grid search over them and testing each using cross-validation. The parameters that provide the lowest out-of-sample error (i.e. when tested on data not in the training set) are the ones chosen.

Returns a set of training parameter combinations and their results that can then be passed to GetTunedModel below to acquire a model that has been properly regularized.

The grid resolution is increased automatically to utilize any otherwise idle nodes.

For a single given set of model parameters, models can be tuned to a number of datasets by concatenating multiple datasets into single 'observations' and 'classifications' datasets, with separate datasets being identified by a work ID column, 'wi'.

PARAMETER **folds** ||| INTEGER4 — The number of cross-validation folds for evaluating each candidate model.

PARAMETER **start_log2C** ||| REAL8 — The lower bound for log2(C): C >= 2^(start_log2C).

PARAMETER **stop_log2C** ||| REAL8 — The upper bound for log2(C): C <= 2^(start_log2C).

PARAMETER **maxIncr_log2C** ||| REAL8 — Taximum allowable exponential increment for C.

PARAMETER **start_log2gamma** ||| REAL8 — The lower bound for log2(gamma): gamma >= 2^(start_log2gamma).

PARAMETER **stop_log2gamma** ||| REAL8 — The upper bound for log2(gamma): gamma <= 2^(start_log2gamma).

PARAMETER **maxIncr_log2gamma** ||| REAL8 — Taximum allowable exponential increment for gamma.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 coef0 , REAL8 eps , REAL8 nu , REAL8 p , INTEGER4 nr_weight , BOOLEAN shrinking , BOOLEAN prob_est , BOOLEAN scale , TABLE (I4Entry) lbl , TABLE (R8Entry) weight , REAL8 gamma , REAL8 C })** — Dataset with sets of model parameters and corresponding cross-validated scores in GridSearch_Result format.

SEE GetTunedModel

SEE Types.GridSearch_Result

GETTUNEDMODEL GetTunedModel

SVR \

DATASET(Layout_Model)	**GetTunedModel**
(DATASET(Types.GridSearch_Result) tuneResult)	

Choose the best set of regularization parameters and use it to train the models. Using the output of Tune(), find the best set of modeling parameters for each work id, and train the corresponding models. The the most regularized (i.e. coarsest) set of parameters that achieved near-maximum performance is used to create the models.

PARAMETER **tuneResult** ||| TABLE (GridSearch_Result) — The results of a grid search over C and gamma, produced by Tune().

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value })** — The encoded models.

CROSSVALIDATE CrossValidate

SVR \

DATASET(Types.CrossValidate_Result)	**CrossValidate**
(INTEGER4 folds = 10)	

Perform n-fold cross-validation of a given model for each work ID. For a single given set of model parameters, models can be cross-validated against a number of datasets by concatenating multiple datasets into single 'X' and 'Y' datasets, with separate datasets being identified by a work ID column, 'wi'.

PARAMETER **folds** ||| INTEGER4 — The number of cross-validation folds.

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq })** — Dataset of cross-validated scores i CrossValidate_Result format.

SEE Types.CrossValidate_Result

MODELSUMMARY ModelSummary

SVR \

DATASET({UNSIGNED4 r, STRING60 Txt})	**ModelSummary**
(DATASET(Layout_Model) model)	

Generate human-readable model summary of trained SVM model(s).

Multiple models can be simultaneously summarized by concatenating a number of models into a single 'model' object, with separate models being identified by a work ID column, 'wi'.

PARAMETER **model** ||| TABLE (Layout_Model) — The models, which should be produced by a corresponding GetModel function.

RETURN **TABLE ({ UNSIGNED4 r , STRING60 txt })** — Single-column dataset with textual description of models.

---

train

Go Up

IMPORTS

Types | libsvm.Types | libsvm | libsvm.Converted |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

TRAIN Train

DATASET(Model)	**Train**
(DATASET(Parms) p = DATASET(Types.Training_Parameters_Default), DATASET(ML_Types.NumericField) observations, DATASET(ML_Types.NumericField) actuals)	

Train SVM classification and regression models.

PARAMETER **p** ||| TABLE (Parms) — The parameters which define the model(s) to be trained.

PARAMETER **observations** ||| TABLE (NumericField) — The observed explanatory values.

PARAMETER **actuals** ||| TABLE (NumericField) — The dependent variable(s).

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 gamma , REAL8 coef0 , UNSIGNED4 k , UNSIGNED4 l , BOOLEAN scale , TABLE (FeatureStats) scaleInfo , TABLE (SVM_SV) sv , TABLE (R8Entry) sv_coef , TABLE (R8Entry) rho , TABLE (R8Entry) probA , TABLE (R8Entry) probB , TABLE (I4Entry) labels , TABLE (I4Entry) nSV })** — Trained SVM model(s) in SVM Model format.

INTERNAL

Types

Go Up

IMPORTS

libsvm.Types |

DESCRIPTIONS

TYPES Types

	Types

SupportVectorMachines type definitions.

Children

1. Model_ID
2. R8Entry
3. I4Entry
4. SVM_Type
5. Kernel_Type
6. SVM_Feature : Feature value data structure for communication with libSVM
7. SVM_Instance : Support Vector Machine Instance structure for communication with libSVM
8. SVM_SV : Support Vector data structure for communication with libSVM
9. SVM_Grid_Args : Arguments for grid search
10. SVM_Grid_Plan : This record provides the format for the argument to GridSearch

11. SVM_Grid_Plan_Default : Default Grid Plan
12. Training_Base : Training Parameter Base Record
13. Training_Base_Default : Default base training parameters
14. Training_Parameters
15. Training_Parameters_Default : Default Training Parameters
16. FeatureStats : Inforation about each feature
17. Model : Record to libSVM form of the model
18. CrossValidate_Result : Record to hold the results of call to CrossValidate
19. GridSearch_Result : Record for the results of call to GridSearch Contains both CrossValidate_Result and Training_Parameters
20. Feature_Scale : Record to hold scale information for each feature
21. Class_Scale
22. Scale_Parms
23. SVM_Scale
24. SVM_Prediction
25. SVM_Pred_Values
26. SVM_Pred_Prob_Est

MODEL_ID Model_ID

Types \

	Model_ID

RETURN **INTEGER4** —

INTERNAL

R8ENTRY R8Entry

Types \

	R8Entry

INTERNAL

I4ENTRY I4Entry

Types \

	I4Entry

INTERNAL

SVM_TYPE SVM_Type

Types \

	SVM_Type

INTERNAL

KERNEL_TYPE Kernel_Type

Types \

	Kernel_Type

INTERNAL

SVM_FEATURE SVM_Feature

Types \

	SVM_Feature

Feature value data structure for communication with libSVM

FIELD **nominal** ||| UNSIGNED4 — The feature identifier.

FIELD **v** ||| REAL8 — The feature value.

INTERNAL

SVM_INSTANCE SVM_Instance

Types \

	SVM_Instance

Support Vector Machine Instance structure for communication with libSVM

FIELD **wi** ||| UNSIGNED2 — The work-item number.

FIELD **rid** ||| UNSIGNED8 — The source identifier.

FIELD **y** ||| REAL8 — The Y (dependent) value.

FIELD **max_value** ||| REAL8 — Maximum value for feature Y.

FIELD **x** ||| TABLE (SVM_Feature) — Independent data for this observation in SVM_Feature format.

SEE SVM_Feature

INTERNAL

SVM_SV SVM_SV

Types \

	SVM_SV

Support Vector data structure for communication with libSVM

FIELD **v_ord** ||| UNSIGNED8 — Identifier for the vector

FIELD **features** ||| TABLE (SVM_Feature) — Dataset of SVM_Feature records

INTERNAL

SVM_GRID_ARGS SVM_Grid_Args

Types \

	SVM_Grid_Args

Arguments for grid search. This is a sub-format for SVM_Grid_Plan below.

FIELD **start** ||| REAL8 — The value at which to start the search.

FIELD **stop** ||| REAL8 — The value at which to stop the search.

FIELD **max_incr** ||| REAL8 — The maximum increment to use in the search.

SEE SVM_Grid_Plan

INTERNAL

SVM_GRID_PLAN SVM_Grid_Plan

Types \

	SVM_Grid_Plan

This record provides the format for the argument to GridSearch.

FIELD **log2_C** ||| ROW (SVM_Grid_Args) — Start, stop and increment values for the log base 2 of C, in SVM_Grid_Args format.

FIELD **log2_gamma** ||| ROW (SVM_Grid_Args) — Start, stop and increment values for the log base 2 of gamma, in SVM_Grid_Args format.

FIELD **folds** ||| UNSIGNED4 — No Doc

SEE SVM_Grid_Args

SEE GridSearch

INTERNAL

SVM_GRID_PLAN_DEFAULT SVM_Grid_Plan_Default

Types \

	SVM_Grid_Plan_Default

Default Grid Plan

RETURN **ROW (SVM_Grid_Plan)** —

SEE SVM_Grid_Plan

INTERNAL

TRAINING_BASE Training_Base

Types \

	Training_Base

Training Parameter Base Record

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **eps** ||| REAL8 — No Doc

FIELD **nu** ||| REAL8 — No Doc

FIELD **p** ||| REAL8 — No Doc

FIELD **nr_weight** ||| INTEGER4 — No Doc

FIELD **shrinking** ||| BOOLEAN — No Doc

FIELD **prob_est** ||| BOOLEAN — No Doc

FIELD **scale** ||| BOOLEAN — No Doc

FIELD **lbl** ||| TABLE (I4Entry) — No Doc

FIELD **weight** ||| TABLE (R8Entry) — No Doc

INTERNAL

TRAINING_BASE_DEFAULT Training_Base_Default

Types \

	Training_Base_Default

Default base training parameters

RETURN **ROW (Training_Base)** —

INTERNAL

TRAINING_PARAMETERS Training_Parameters

Types \

	Training_Parameters

FIELD **id** ||| INTEGER4 — No Doc

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **eps** ||| REAL8 — No Doc

FIELD **nu** ||| REAL8 — No Doc

FIELD **p** ||| REAL8 — No Doc

FIELD **nr_weight** ||| INTEGER4 — No Doc

FIELD **shrinking** ||| BOOLEAN — No Doc

FIELD **prob_est** ||| BOOLEAN — No Doc

FIELD **scale** ||| BOOLEAN — No Doc

FIELD **lbl** ||| TABLE (I4Entry) — No Doc

FIELD **weight** ||| TABLE (R8Entry) — No Doc

FIELD **gamma** ||| REAL8 — No Doc

FIELD **c** ||| REAL8 — No Doc

INTERNAL

TRAINING_PARAMETERS_DEFAULT Training_Parameters_Default

Types \

	Training_Parameters_Default

Default Training Parameters

RETURN **ROW (Training_Parameters)** —

INTERNAL

FEATURESTATS FeatureStats

Types \

	FeatureStats

Inforation about each feature

FIELD **indx** ||| INTEGER4 — No Doc

FIELD **mean** ||| REAL8 — No Doc

FIELD **sd** ||| REAL8 — No Doc

INTERNAL

MODEL Model

Types \

	Model

Record to libSVM form of the model

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| INTEGER4 — No Doc

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **gamma** ||| REAL8 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **k** ||| UNSIGNED4 — No Doc

FIELD **l** ||| UNSIGNED4 — No Doc

FIELD **scale** ||| BOOLEAN — No Doc

FIELD **scaleinfo** ||| TABLE (FeatureStats) — No Doc

FIELD **sv** ||| TABLE (SVM_SV) — No Doc

FIELD **sv_coef** ||| TABLE (R8Entry) — No Doc

FIELD **rho** ||| TABLE (R8Entry) — No Doc

FIELD **proba** ||| TABLE (R8Entry) — No Doc

FIELD **probb** ||| TABLE (R8Entry) — No Doc

FIELD **labels** ||| TABLE (I4Entry) — No Doc

FIELD **nsv** ||| TABLE (I4Entry) — No Doc

INTERNAL

CROSSVALIDATE_RESULT CrossValidate_Result

Types \

	CrossValidate_Result

Record to hold the results of call to CrossValidate

FIELD **wi** ||| UNSIGNED2 — The work-item number.

FIELD **id** ||| INTEGER4 — The id of the cross-validation set (i.e. fold).

FIELD **correct** ||| REAL8 — The number of correct values.

FIELD **mse** ||| REAL8 — The mean squared error of the regression

FIELD **r__sq** ||| REAL8 — The R-squared value indicating the strength of the regression.

GRIDSEARCH_RESULT GridSearch_Result

Types \

	GridSearch_Result

Record for the results of call to GridSearch Contains both CrossValidate_Result and Training_Parameters.

FIELD **wi** ||| UNSIGNED2 — The work-item number.

FIELD **id** ||| INTEGER4 — The id of the cross-validation set (i.e. fold).

FIELD **correct** ||| REAL8 — The number of correct values.

FIELD **mse** ||| REAL8 — The mean squared error of the regression

FIELD **r__sq** ||| REAL8 — The R-squared value indicating the strength of the regression.

FIELD **gamma** ||| REAL8 — The gamma regularization parameter value.

FIELD **C** ||| REAL8 — The C regularization parameter value.

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **eps** ||| REAL8 — No Doc

FIELD **nu** ||| REAL8 — No Doc

FIELD **p** ||| REAL8 — No Doc

FIELD **nr__weight** ||| INTEGER4 — No Doc

FIELD **shrinking** ||| BOOLEAN — No Doc

FIELD **prob_est** ||| BOOLEAN — No Doc

FIELD **scale** ||| BOOLEAN — No Doc

FIELD **lbl** ||| TABLE (I4Entry) — No Doc

FIELD **weight** ||| TABLE (R8Entry) — No Doc

FEATURE_SCALE Feature_Scale

Types \

	Feature_Scale

Record to hold scale information for each feature

FIELD **nominal** ||| UNSIGNED4 — No Doc

FIELD **min_value** ||| REAL8 — No Doc

FIELD **max_value** ||| REAL8 — No Doc

INTERNAL

CLASS_SCALE Class_Scale

Types \

	Class_Scale

FIELD **y_min** ||| REAL8 — No Doc

FIELD **y_max** ||| REAL8 — No Doc

INTERNAL

SCALE_PARMS Scale_Parms

Types \

	Scale_Parms

FIELD **x_lower** ||| REAL8 — No Doc

FIELD **x_upper** ||| REAL8 — No Doc

FIELD **y_lower** ||| REAL8 — No Doc

FIELD **y_upper** ||| REAL8 — No Doc

INTERNAL

SVM_SCALE SVM_Scale

Types \

	SVM_Scale

FIELD **x_lower** ||| REAL8 — No Doc

FIELD **x_upper** ||| REAL8 — No Doc

FIELD **y_lower** ||| REAL8 — No Doc

FIELD **y_upper** ||| REAL8 — No Doc

FIELD **y_min** ||| REAL8 — No Doc

FIELD **y_max** ||| REAL8 — No Doc

FIELD **features** ||| TABLE (Feature_Scale) — No Doc

INTERNAL

SVM_PREDICTION SVM_Prediction

Types \

	SVM_Prediction

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| INTEGER4 — No Doc

FIELD **rid** ||| UNSIGNED8 — No Doc

FIELD **target_y** ||| REAL8 — No Doc

FIELD **predict_y** ||| REAL8 — No Doc

INTERNAL

SVM_PRED_VALUES SVM_Pred_Values

Types \

	SVM_Pred_Values

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| INTEGER4 — No Doc

FIELD **rid** ||| UNSIGNED8 — No Doc

FIELD **target_y** ||| REAL8 — No Doc

FIELD **predict_y** ||| REAL8 — No Doc

FIELD **decision_values** ||| TABLE (R8Entry) — No Doc

INTERNAL

SVM_PRED_PROB_EST SVM_Pred_Prob_Est

Types \

	SVM_Pred_Prob_Est

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| INTEGER4 — No Doc

FIELD **rid** ||| UNSIGNED8 — No Doc

FIELD **target_y** ||| REAL8 — No Doc

FIELD **predict_y** ||| REAL8 — No Doc

FIELD **prob_estimates** ||| TABLE (R8Entry) — No Doc

INTERNAL

Datasets

Go Up

Table of Contents

Datasets/

HeartScale

Go Up

IMPORTS

_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

HEARTSCALE heartScale

	heartScale

Defines the heart_scale example dataset.

Children

1. content : Specification of heart_scale dataset (270 rows, 15 columns)

CONTENT content

heartScale \

	content

Specification of heart_scale dataset (270 rows, 15 columns).

RETURN **TABLE (heart_scaleRec)** — The heart_scale dataset.

Examples

Go Up

Table of Contents

libsvm

Go Up

Table of Contents

Constants

Go Up

DESCRIPTIONS

CONSTANTS Constants

	Constants

No Documentation Found

Children

1. LibSVM_BadParm : No Documentation Found

LIBSVM_BADPARM LibSVM_BadParm

Constants \

	LibSVM_BadParm

No Documentation Found

RETURN **INTEGER8** —

---

libsvm/

Converted

Go Up

IMPORTS

libsvm.Types | Types | std |

DESCRIPTIONS

CONVERTED Converted

	Converted

No Documentation Found

Children

1. LIBSVMDATA2Instance : No Documentation Found
2. Instance2Problem : No Documentation Found

LIBSVMDATA2INSTANCE LIBSVMDATA2Instance

Converted \

DATASET(SVM_Instance)	LIBSVMDATA2Instance
(STRING fname)	

No Documentation Found

PARAMETER **fname** ||| STRING — No Doc

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 rid , REAL8 y , REAL8 max_value , TABLE (SVM_Feature) x }) —**

INSTANCE2PROBLEM Instance2Problem

Converted \

DATASET(ECL_LibSVM_ProblemList)	Instance2Problem
(DATASET(SVM_Instance) ds)	

No Documentation Found

PARAMETER **ds** ||| TABLE (SVM_Instance) — No Doc

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 elements , INTEGER4 entries , UNSIGNED4 features , REAL8 max_value , TABLE (R8Entry) y , TABLE (LibSVM_Node) x }) —**

---

libsvm/

LibSVMVersion

Go Up

DESCRIPTIONS

LIBSVMVERSION LibSVMVersion

	LibSVMVersion

No Documentation Found

---

libsvm/

SVMCrossValidate

Go Up

IMPORTS

libsvm.Types | libsvm.Constants | libsvm.Converted |

DESCRIPTIONS

SVMCROSSVALIDATE SVMCrossValidate

Result	SVMCrossValidate
(SVM_Parms prm, Problem prb, UNSIGNED2 nr_fold, UNSIGNED4 err_code=ErrCode)	

No Documentation Found

PARAMETER **prm** ||| ROW (SVM_Parms) — No Doc

PARAMETER **prb** ||| ROW (Problem) — No Doc

PARAMETER **nr_fold** ||| UNSIGNED2 — No Doc

PARAMETER **err_code** ||| UNSIGNED4 — No Doc

RETURN **ROW ({ REAL8 mse , REAL8 r_sq , REAL8 correct })** —

---

libsvm/

SVMPredict

Go Up

IMPORTS

libsvm.Types | libsvm.Constants | libsvm.Converted |

DESCRIPTIONS

SVMPREDICT SVMPredict

DATASET(R8Entry)	**SVMPredict**
(Model ecl_model, DATASET(Node) ecl_nodes, Rqst output_request)	

No Documentation Found

PARAMETER **ecl_model** ||| ROW (Model) — No Doc

PARAMETER **ecl_nodes** ||| TABLE (Node) — No Doc

PARAMETER **output_request** ||| UNSIGNED2 — No Doc

RETURN **TABLE (R8Entry)** —

---

libsvm/

SVMTrain

Go Up

IMPORTS

libsvm.Types | libsvm.Constants | libsvm.Converted |

DESCRIPTIONS

SVMTRAIN SVMTrain

ECL_LibSVM_Model	**SVMTrain**
(SVM_Parms prm, Problem prb)	

No Documentation Found

PARAMETER **prm** ||| ROW (SVM_Parms) — No Doc

PARAMETER **prb** ||| ROW (Problem) — No Doc

RETURN **ROW ({ UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 gamma , REAL8 coef0 , UNSIGNED4 k , UNSIGNED4 l , UNSIGNED4 elements , UNSIGNED4 pairs_A , UNSIGNED4 pairs_B , UNSIGNED4 nr_label , UNSIGNED4 nr_nSV , TABLE (LibSVM_Node) sv , TABLE (R8Entry) sv_coef , TABLE (R8Entry) rho , TABLE (R8Entry) probA , TABLE (R8Entry) probB , TABLE (I4Entry) labels , TABLE (I4Entry) nSV }) —**

Types

Go Up

DESCRIPTIONS

TYPES Types

	Types

No Documentation Found

Children

1. LibSVM_Output : No Documentation Found
2. LibSVM_Kernel : No Documentation Found
3. LibSVM_Type : No Documentation Found
4. LibSVM_Node : No Documentation Found
5. R8Entry : No Documentation Found
6. I4Entry : No Documentation Found
7. ECL_LibSVM_ProblemList : No Documentation Found
8. ECL_LibSVM_Problem : No Documentation Found
9. ECL_LibSVM_Parameter : No Documentation Found
10. ECL_LibSVM_Train_Param : No Documentation Found
11. ECL_LibSVM_Model : No Documentation Found
12. CrossValidate_Result : No Documentation Found
13. LibSVM_Version_set : No Documentation Found

LIBSVM_OUTPUT LibSVM_Output

Types \

	LibSVM_Output

No Documentation Found

RETURN **UNSIGNED2** —

LIBSVM_KERNEL LibSVM_Kernel

Types \

	LibSVM_Kernel

No Documentation Found

RETURN **UNSIGNED2** —

LIBSVM_TYPE LibSVM_Type

Types \

	LibSVM_Type

No Documentation Found

RETURN **UNSIGNED2** —

LIBSVM_NODE LibSVM_Node

Types \

	LibSVM_Node

No Documentation Found

FIELD **indx** ||| INTEGER4 — No Doc

FIELD **value** ||| REAL8 — No Doc

R8ENTRY R8Entry

Types \

	R8Entry

No Documentation Found

FIELD **v** ||| REAL8 — No Doc

I4ENTRY I4Entry

Types \

	I4Entry

No Documentation Found

FIELD **v** ||| INTEGER4 — No Doc

ECL_LIBSVM_PROBLEMLIST ECL_LibSVM_ProblemList

Types \

	ECL_LibSVM_ProblemList

No Documentation Found

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **elements** ||| UNSIGNED4 — No Doc

FIELD **entries** ||| INTEGER4 — No Doc

FIELD **features** ||| UNSIGNED4 — No Doc

FIELD **max_value** ||| REAL8 — No Doc

FIELD **y** ||| TABLE (R8Entry) — No Doc

FIELD **x** ||| TABLE (LibSVM_Node) — No Doc

ECL_LIBSVM_PROBLEM ECL_LibSVM_Problem

Types \

	ECL_LibSVM_Problem

No Documentation Found

FIELD **elements** ||| UNSIGNED4 — No Doc

FIELD **entries** ||| INTEGER4 — No Doc

FIELD **features** ||| UNSIGNED4 — No Doc

FIELD **max_value** ||| REAL8 — No Doc

FIELD **y** ||| TABLE (R8Entry) — No Doc

FIELD **x** ||| TABLE (LibSVM_Node) — No Doc

ECL_LIBSVM_PARAMETER ECL_LibSVM_Parameter

Types \

	ECL_LibSVM_Parameter

No Documentation Found

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **gamma** ||| REAL8 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

ECL_LIBSVM_TRAIN_PARAM ECL_LibSVM_Train_Param

Types \

	ECL_LibSVM_Train_Param

No Documentation Found

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **gamma** ||| REAL8 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **cache_size** ||| REAL8 — No Doc

FIELD **eps** ||| REAL8 — No Doc

FIELD **c** ||| REAL8 — No Doc

FIELD **nu** ||| REAL8 — No Doc

FIELD **p** ||| REAL8 — No Doc

FIELD **nr__weight** ||| INTEGER4 — No Doc

FIELD **shrinking** ||| UNSIGNED2 — No Doc

FIELD **prob__est** ||| UNSIGNED2 — No Doc

FIELD **lbl** ||| TABLE (I4Entry) — No Doc

FIELD **weight** ||| TABLE (R8Entry) — No Doc

ECL_LIBSVM_MODEL ECL_LibSVM_Model

Types \

	ECL__LibSVM__Model

No Documentation Found

FIELD **svmtype** ||| UNSIGNED2 — No Doc

FIELD **kerneltype** ||| UNSIGNED2 — No Doc

FIELD **degree** ||| INTEGER4 — No Doc

FIELD **gamma** ||| REAL8 — No Doc

FIELD **coef0** ||| REAL8 — No Doc

FIELD **k** ||| UNSIGNED4 — No Doc

FIELD **l** ||| UNSIGNED4 — No Doc

FIELD **elements** ||| UNSIGNED4 — No Doc

FIELD **pairs__a** ||| UNSIGNED4 — No Doc

FIELD **pairs__b** ||| UNSIGNED4 — No Doc

FIELD **nr__label** ||| UNSIGNED4 — No Doc

FIELD **nr__nsv** ||| UNSIGNED4 — No Doc

FIELD **sv** ||| TABLE (LibSVM_Node) — No Doc

FIELD **sv__coef** ||| TABLE (R8Entry) — No Doc

FIELD **rho** ||| TABLE (R8Entry) — No Doc

FIELD **proba** ||| TABLE (R8Entry) — No Doc

FIELD **probb** ||| TABLE (R8Entry) — No Doc

FIELD **labels** ||| TABLE (I4Entry) — No Doc

FIELD **nsv** ||| TABLE (I4Entry) — No Doc

CROSSVALIDATE_RESULT CrossValidate_Result

Types \

	CrossValidate_Result

No Documentation Found

FIELD **mse** ||| REAL8 — No Doc

FIELD **r_sq** ||| REAL8 — No Doc

FIELD **correct** ||| REAL8 — No Doc

LIBSVM_VERSION_SET LibSVM_Version_set

Types \

SET OF UNSIGNED2	**LibSVM_Version_set**

No Documentation Found

RETURN **SET (UNSIGNED2)** —

Performance

Go Up

Table of Contents

GetModelPerformanceMyriad

Go Up

IMPORTS

_versions.PBblas.V3_0_2.PBblas | _versions.PBblas.V3_0_2.PBblas.internal |
_versions.PBblas.V3_0_2.PBblas.internal.Types |
_versions.PBblas.V3_0_2.PBblas.Types |
_versions.PBblas.V3_0_2.PBblas.internal.MatDims |
_versions.PBblas.V3_0_2.PBblas.test |
_versions.PBblas.V3_0_2.PBblas.test.MakeTestMatrix |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

GETMODELPERFORMANCEMYRIAD GetModelPerformanceMyriad

XPORT	GetModelPerformanceMyriad
(UNSIGNED4 test_size, BOOLEAN regress)	

No Documentation Found

PARAMETER **test_size** ||| UNSIGNED4 — No Doc

PARAMETER **regress** ||| BOOLEAN — No Doc

RETURN TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value }) —

PredictPerformanceMyriad

Go Up

IMPORTS

_versions.PBblas.V3_0_2.PBblas | _versions.PBblas.V3_0_2.PBblas.internal |
_versions.PBblas.V3_0_2.PBblas.internal.Types |
_versions.PBblas.V3_0_2.PBblas.Types |
_versions.PBblas.V3_0_2.PBblas.internal.MatDims |
_versions.PBblas.V3_0_2.PBblas.test |
_versions.PBblas.V3_0_2.PBblas.test.MakeTestMatrix |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

PREDICTPERFORMANCEMYRIAD PredictPerformanceMyriad

XPORT	**PredictPerformanceMyriad**
`(UNSIGNED4 test_size, BOOLEAN regress)`	

No Documentation Found

PARAMETER **test_size** ||| UNSIGNED4 — No Doc

PARAMETER **regress** ||| BOOLEAN — No Doc

RETURN TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 value }) —

ToModelPerformanceMyriad

Go Up

IMPORTS

_versions.PBblas.V3_0_2.PBblas | _versions.PBblas.V3_0_2.PBblas.internal |
_versions.PBblas.V3_0_2.PBblas.internal.Types |
_versions.PBblas.V3_0_2.PBblas.Types |
_versions.PBblas.V3_0_2.PBblas.internal.MatDims |
_versions.PBblas.V3_0_2.PBblas.test |
_versions.PBblas.V3_0_2.PBblas.test.MakeTestMatrix |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

TOMODELPERFORMANCEMYRIAD ToModelPerformanceMyriad

XPORT	**ToModelPerformanceMyriad**
`(UNSIGNED4 test_size, BOOLEAN regress)`	

No Documentation Found

PARAMETER **test_size** ||| UNSIGNED4 — No Doc

PARAMETER **regress** ||| BOOLEAN — No Doc

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 gamma , REAL8 coef0 , UNSIGNED4 k , UNSIGNED4 l , BOOLEAN scale , TABLE (FeatureStats) scaleInfo , TABLE (SVM_SV) sv , TABLE (R8Entry) sv_coef , TABLE (R8Entry) rho , TABLE (R8Entry) probA , TABLE (R8Entry) probB , TABLE (I4Entry) labels , TABLE (I4Entry) nSV }) —**

Performance/

TunePerformanceMyriad

Go Up

IMPORTS

_versions.PBblas.V3_0_2.PBblas | _versions.PBblas.V3_0_2.PBblas.internal |
_versions.PBblas.V3_0_2.PBblas.internal.Types |
_versions.PBblas.V3_0_2.PBblas.Types |
_versions.PBblas.V3_0_2.PBblas.internal.MatDims |
_versions.PBblas.V3_0_2.PBblas.test |
_versions.PBblas.V3_0_2.PBblas.test.MakeTestMatrix |
_versions.ML_Core.V3_2_2.ML_Core |
_versions.ML_Core.V3_2_2.ML_Core.Types |

DESCRIPTIONS

TUNEPERFORMANCEMYRIAD TunePerformanceMyriad

XPORT	**TunePerformanceMyriad**
`(UNSIGNED4 test_size, REAL8 maxIncr_log2C, REAL8 maxIncr_log2Gamma, BOOLEAN regress)`	

No Documentation Found

PARAMETER **test_size** ||| UNSIGNED4 — No Doc

PARAMETER **maxincr_log2c** ||| REAL8 — No Doc

PARAMETER **maxincr_log2gamma** ||| REAL8 — No Doc

PARAMETER **regress** ||| BOOLEAN — No Doc

RETURN **TABLE ({ UNSIGNED2 wi , INTEGER4 id , REAL8 correct , REAL8 mse , REAL8 r_sq , UNSIGNED2 svmType , UNSIGNED2 kernelType , INTEGER4 degree , REAL8 coef0 , REAL8 eps , REAL8 nu , REAL8 p , INTEGER4 nr_weight , BOOLEAN shrinking , BOOLEAN prob_est , BOOLEAN scale , TABLE (I4Entry) lbl , TABLE (R8Entry) weight , REAL8 gamma , REAL8 C }) —**

Validation

Go Up

Table of Contents

Validation/

R_Validation

Go Up

DESCRIPTIONS

R_VALIDATION R_Validation

	R_Validation

No Documentation Found

Children

1. NumberOfSVs_SVC_NoScaling : No Documentation Found
2. SVCoefs_SVC_NoScaling : No Documentation Found
3. Predictions_SVC_NoScaling : No Documentation Found
4. Probabilities_SVC_NoScaling : No Documentation Found
5. NumberOfSVs_SVC_Scaling : No Documentation Found
6. SVCoefs_SVC_Scaling : No Documentation Found
7. Predictions_SVC_Scaling : No Documentation Found
8. Probabilities_SVC_Scaling : No Documentation Found
9. NumberOfSVs_SVR_NoScaling : No Documentation Found
10. SVCoefs_SVR_NoScaling : No Documentation Found
11. Predictions_SVR_NoScaling : No Documentation Found
12. NumberOfSVs_SVR_Scaling : No Documentation Found
13. SVCoefs_SVR_Scaling : No Documentation Found
14. Predictions_SVR_Scaling : No Documentation Found

15. GridSearchResults_SVC : No Documentation Found
16. GridSearchResults_SVR : No Documentation Found

NUMBEROFSVS_SVC_NOSCALING NumberOfSVs_SVC_NoScaling

R_Validation \

	NumberOfSVs_SVC_NoScaling

No Documentation Found

RETURN INTEGER8 —

SVCOEFS_SVC_NOSCALING SVCoefs_SVC_NoScaling

R_Validation \

	SVCoefs_SVC_NoScaling

No Documentation Found

RETURN TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —

PREDICTIONS_SVC_NOSCALING Predictions_SVC_NoScaling

R_Validation \

	Predictions_SVC_NoScaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —**

PROBABILITIES_SVC_NOSCALING Probabilities_SVC_NoScaling

R_Validation \

	Probabilities_SVC_NoScaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —**

NUMBEROFSVS_SVC_SCALING NumberOfSVs_SVC_Scaling

R_Validation \

	NumberOfSVs_SVC_Scaling

No Documentation Found

RETURN **INTEGER8 —**

SVCOEFS_SVC_SCALING SVCoefs_SVC_Scaling

R_Validation \

	SVCoefs_SVC_Scaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —**

PREDICTIONS_SVC_SCALING Predictions_SVC_Scaling

R_Validation \

	Predictions_SVC_Scaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —**

PROBABILITIES_SVC_SCALING Probabilities_SVC_Scaling

R_Validation \

	Probabilities_SVC_Scaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R }) —**

NUMBEROFSVS_SVR_NOSCALING NumberOfSVs_SVR_NoScaling

R_Validation \

	NumberOfSVs_SVR_NoScaling

No Documentation Found

RETURN **INTEGER8** —

SVCOEFS_SVR_NOSCALING SVCoefs_SVR_NoScaling

R_Validation \

	SVCoefs_SVR_NoScaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R })** —

PREDICTIONS_SVR_NOSCALING Predictions_SVR_NoScaling

R_Validation \

	Predictions_SVR_NoScaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R })** —

NUMBEROFSVS_SVR_SCALING NumberOfSVs_SVR_Scaling

R_Validation \

	NumberOfSVs_SVR_Scaling

No Documentation Found

RETURN **INTEGER8** —

SVCOEFS_SVR_SCALING SVCoefs_SVR_Scaling

R_Validation \

	SVCoefs_SVR_Scaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R })** —

PREDICTIONS_SVR_SCALING Predictions_SVR_Scaling

R_Validation \

	Predictions_SVR_Scaling

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , STRING20 Test , REAL8 R })** —

GRIDSEARCHRESULTS_SVC GridSearchResults_SVC

R_Validation \

	GridSearchResults_SVC

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , REAL8 Cost , REAL8 Gamma , REAL8 R }) —**

GRIDSEARCHRESULTS_SVR GridSearchResults_SVR

R_Validation \

	GridSearchResults_SVR

No Documentation Found

RETURN **TABLE ({ INTEGER8 ID , REAL8 Cost , REAL8 Gamma , REAL8 R }) —**
