ECL DOCUMENTATION

Go Up

Name	LogisticRegression
Version	3.0.0
Description	Logistic Regression implementation
License	http://www.apache.org/licenses/LICENSE-2.0
Copyright	Copyright (C) 2017 HPCC Systems®
Authors	HPCCSystems
DependsOn	ML_Core 3.2.1, PBblas
Platform	6.2.0

OVERVIEW

LogisticRegression

Bundle for binomial, multinomial, and ordinal logistic regression.

Table of Contents

Matrix multiply when either A or B is a diagonal and is passed as a vector
enum_workitems.ecl Create an enumeration of string contents to be used as work items
ExtractBeta.ecl Extract the beta values form the model dataset
ExtractBeta_CI.ecl Extract the beta values and confidence intervals from the model dataset
ExtractBeta_full.ecl Extract the coefficient information including confidence intervals, z and p values
ExtractBeta_pval.ecl Extract the beta values including z and p value from the model
ExtractReport.ecl Create a model report from a model
LogitPredict.ecl Predict the category values with the logit function and the the supplied beta coefficients
LogitScore.ecl Calculate the score using the logit function and the the supplied beta coefficients
LUCI_Model.ecl Create a LUCI model file description of the model(s) from the external version of the model
Model_Deviance.ecl Model Deviance Report
Named_Model.ecl Apply external labels for work items and field names to a model
Null_Deviance.ecl Return Deviance information for the null model, that is, a model with only an intercept
Types.ecl Type definitions for LogisticRegression bundle

BinomialConfusion

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

BINOMIALCONFUSION

/ EXPORT DATASET(Types.Binomial_Confusion_Summary)	**BinomialConfusion**
(DATASET(Core_-Types.Confusion_Detail) d)	

Calculate the binomial confusion matrix. Work items with multinomial responses are ignored by this function. The higher value lexically is considered to be the positive indication.

PARAMETER **d** ||| TABLE (Confusion_Detail) — confusion detail for the work item and classifier.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , UNSIGNED8 true_positive , UNSIGNED8 true_negative , UNSIGNED8 false_positive , UNSIGNED8 false_negative , UNSIGNED8 cond_pos , UNSIGNED8 pred_pos , UNSIGNED8 cond_neg , UNSIGNED8 pred_neg , REAL8 prevalence , REAL8 accuracy , REAL8 true_pos_rate , REAL8 false_neg_rate , REAL8 false_pos_rate , REAL8 true_neg_rate , REAL8 pos_pred_val , REAL8 false_disc_rate , REAL8 false_omit_rate , REAL8 neg_pred_val })** — confusion matrix for a binomial classifier in Binomial_Confusion_Summary format.

SEE Types.Binomial_Confusion_Summary

BinomialLogisticRegression

Go Up

IMPORTS

Constants | ML_Core.Interfaces | ML_Core.Types |

DESCRIPTIONS

BINOMIALLOGISTICREGRESSION

/ EXPORT	**BinomialLogisticRegression**
(UNSIGNED max_iter=200, REAL8 epsilon=Constants.default_epsilon, REAL8 ridge=Constants.default_ridge)	

Binomial logistic regression using iteratively re-weighted least squares.

PARAMETER **max_iter** ||| UNSIGNED8 — (Optional) The maximum number of iterations to try. Default = 200.

PARAMETER **epsilon** ||| REAL8 — (Optional) The minimum change in the Beta value estimate to continue

PARAMETER **ridge** ||| REAL8 — (Optional) A value to populate a diagonal matrix that is added to a matrix help assure that the matrix is invertible.

PARENT **ML_Core.Interfaces.IClassify** </home/lily/source/ML_Core/Interfaces/IClassify.ecl>

Children

1. GetModel : Calculate the model to fit the observation data to the observed classes

2. Classify : Classify the observations using a model as previously returned from GetModel
3. Report : Report the confusion matrix for the classifier and training data

---

GETMODEL

BinomialLogisticRegression /

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

Calculate the model to fit the observation data to the observed classes.

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 model in Layout_Model format.

SEE ML_Core.Types.NumericField

SEE ML_Core.Types.DiscreteField

SEE ML_Core.Types.Layout_Model

OVERRIDE

CLASSIFY

BinomialLogisticRegression /

```
DATASET(Types.Classify_Result)   Classify
(DATASET(Types.Layout_Model) model,
DATASET(Types.NumericField) new_observations)
```

Classify the observations using a model as previously returned from GetModel.

PARAMETER **model** ||| TABLE (Layout_Model) — The model in Layout_Model format.

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

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

SEE ML_Core.Types.Layout_Model

SEE ML_Core.Types.NumericField

SEE ML_Core.Types.Classify_Result

OVERRIDE

REPORT

BinomialLogisticRegression /

```
DATASET(Types.Confusion_Detail)   Report
(DATASET(Types.Layout_Model) model,
DATASET(Types.NumericField) observations,
DATASET(Types.DiscreteField) classifications)
```

Report the confusion matrix for the classifier and training data.

PARAMETER **model** ||| TABLE (Layout_Model) — the encoded model as returned from GetModel.

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

PARAMETER **classifications** ||| TABLE (DiscreteField) — the actual classifications associated with the observations (i.e. ground truth) 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.Types.DiscreteField

SEE ML_Core.Types.ConfusionDetail

OVERRIDE

Confusion

Go Up

IMPORTS

ML_Core | ML_Core.Types | Types |

DESCRIPTIONS

CONFUSION

`/ EXPORT 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.

SEE ML_Core.Types.Confusion_Detail

Constants

Go Up

DESCRIPTIONS

CONSTANTS

Constants

Constants used by Logistic Regression. Most of these are the nominal values used by the Model data set. A few are used to control behavior.

Children

1. limit_card : No Documentation Found
2. default_epsilon : No Documentation Found
3. default_ridge : No Documentation Found
4. local_cap : No Documentation Found
5. id_base : No Documentation Found
6. id_iters : No Documentation Found
7. id_delta : No Documentation Found
8. id_correct : No Documentation Found
9. id_incorrect : No Documentation Found
10. id_stat_set : No Documentation Found
11. id_betas : No Documentation Found
12. id_betas_coef : No Documentation Found
13. id_betas_SE : No Documentation Found
14. base_builder : No Documentation Found

15. base_max_iter : No Documentation Found
16. base_epsilon : No Documentation Found
17. base_ind_vars : No Documentation Found
18. base_dep_vars : No Documentation Found
19. base_obs : No Documentation Found
20. builder_irls_local : No Documentation Found
21. builder_irls_global : No Documentation Found
22. builder_softmax : No Documentation Found

LIMIT_CARD

Constants /

UNSIGNED2	**limit_card**

No Documentation Found

RETURN **UNSIGNED2** —

DEFAULT_EPSILON

Constants /

REAL8	**default_epsilon**

No Documentation Found

RETURN **REAL8** —

DEFAULT_RIDGE

Constants /

REAL8	**default_ridge**

No Documentation Found

RETURN **REAL8** —

LOCAL_CAP

Constants /

UNSIGNED4	**local_cap**

No Documentation Found

RETURN **UNSIGNED4** —

ID_BASE

Constants /

	id_base

No Documentation Found

RETURN **INTEGER8** —

ID_ITERS

Constants /

```
id_iters
```

No Documentation Found

RETURN **INTEGER8** —

ID_DELTA

Constants /

```
id_delta
```

No Documentation Found

RETURN **INTEGER8** —

ID_CORRECT

Constants /

```
id_correct
```

No Documentation Found

RETURN **INTEGER8** —

ID_INCORRECT

Constants /

id_incorrect

No Documentation Found

RETURN **INTEGER8** —

ID_STAT_SET

Constants /

id_stat_set

No Documentation Found

RETURN **SET (INTEGER8)** —

ID_BETAS

Constants /

id_betas

No Documentation Found

RETURN **INTEGER8** —

ID_BETAS_COEF

Constants /

id_betas_coef

No Documentation Found

RETURN **INTEGER8** —

ID_BETAS_SE

Constants /

id_betas_SE

No Documentation Found

RETURN **INTEGER8** —

BASE_BUILDER

Constants /

base_builder

No Documentation Found

RETURN **INTEGER8** —

BASE_MAX_ITER

Constants /

base_max_iter

No Documentation Found

RETURN **INTEGER8** —

BASE_EPSILON

Constants /

base_epsilon

No Documentation Found

RETURN **INTEGER8** —

BASE_IND_VARS

Constants /

base_ind_vars

No Documentation Found

RETURN **INTEGER8** —

BASE_DEP_VARS

Constants /

base_dep_vars

No Documentation Found

RETURN **INTEGER8** —

BASE_OBS

Constants /

base_obs

No Documentation Found

RETURN **INTEGER8** —

BUILDER_IRLS_LOCAL

Constants /

builder_irls_local

No Documentation Found

RETURN **INTEGER8** —

BUILDER_IRLS_GLOBAL

Constants /

builder_irls_global

No Documentation Found

RETURN **INTEGER8** —

BUILDER_SOFTMAX

Constants /

builder_softmax

No Documentation Found

RETURN **INTEGER8** —

DataStats

Go Up

IMPORTS

LogisticRegression.Types | LogisticRegression.Constants | ML_Core.Types |

DESCRIPTIONS

DATASTATS

```
/ EXPORT DATASET(Types.Data_Info) DataStats
(DATASET(Core_Types.NumericField) indep,
DATASET(Core_Types.DiscreteField) dep, BOOLEAN
field_details=FALSE)
```

Produce summary information about the datasets.

When field_details = FALSE, indicates the range for the x and y (independent and dependent) columns.

When field_details = TRUE, the cardinality, minimum, and maximum values are returned. A zero cardinality is returned when the field cardinality exceeds the Constants.limit_card value.

Note that a column of all zero values cannot be distinguished from a missing column.

PARAMETER **indep** ||| TABLE (NumericField) — data set of independent variables.

PARAMETER **dep** ||| TABLE (DiscreteField) — data set of dependent variables.

PARAMETER **field_details** ||| BOOLEAN — Boolean directive to provide field level info.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 dependent_fields , UNSIGNED4 dependent_records , UNSIGNED4 independent_fields , UNSIGNED4 independent_records , UNSIGNED4 dependent_count , UNSIGNED4 independent_count , TABLE (Field_Desc) dependent_stats , TABLE (Field_Desc) independent_stats })** — a data set of information on each work item in Data_Info format.

SEE Types.Data_Info

SEE Constants.limit_card

Deviance_Analysis

Go Up

IMPORTS

Types | ML_Core.Math |

DESCRIPTIONS

DEVIANCE_ANALYSIS

```
/ EXPORT DATASET(Types.AOD_Record) Deviance_Analysis
(DATASET(Types.Deviance_Record) proposed,
DATASET(Types.Deviance_Record) base)
```

Analysis of Deviance Report.

Compare deviance information between two models, a base and and proposed model.

Analysis of Deviance is analogous to the Analysis of Variance (ANOVA) used in least-squares modeling, but adapted to the general linear model (GLM). In this case it is adapted specifically to the logistic model.

The inputs are the deviance records for each model as obtained from a call to Model_Deviance.

PARAMETER **proposed** ||| TABLE (Deviance_Record) — deviance records of the proposed model.

PARAMETER **base** ||| TABLE (Deviance_Record) — deviance records of the base model for comparison.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , UNSIGNED8 residual_df , UNSIGNED8 df , REAL8 residual_dev , REAL8 deviance , REAL8 p_value })** — the comparison of the deviance between the models in AOD_Record format.

SEE Model_Deviance

SEE Types.Deviance_Record

SEE Types.AOD_Record

Deviance_Detail

Go Up

IMPORTS

ML_Core | ML_Core.Types | Types |

DESCRIPTIONS

DEVIANCE_DETAIL

/ EXPORT DATASET(Types.Observation_Deviance)	**Deviance_Detail**
(DATASET(Core_Types.DiscreteField) dependents, DATASET(Types.Raw_Prediction) predicts)	

Deviance detail report.

Provides deviance information for each observation.

Analysis of Deviance is analogous to the Analysis of Variance (ANOVA) used in least-squares modeling, but adapted to the general linear model (GLM). In this case it is adapted specifically to the logistic model.

PARAMETER **dependents** ||| TABLE (DiscreteField) — original dependent records for the model

PARAMETER **predicts** ||| TABLE (Raw_Prediction) — the predicted values of the response variable

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 classifier , INTEGER4 actual , INTEGER4 predicted , REAL8 mod_ll , REAL8 mod_dev_component , REAL8 mod_dev_residual , REAL8 nil_ll , REAL8 nil_dev_component , REAL8 nil_dev_residual })** — the deviance information by observation and the log likelihood of the predicted result in Observation_Deviance format.

SEE Types.Observation_Deviance

dimm

Go Up

IMPORTS

std.blas | std.BLAS.Types |

DESCRIPTIONS

DIMM

Types.matrix_t	**dimm**
(BOOLEAN transposeA, BOOLEAN transposeB, BOOLEAN diagonalA, BOOLEAN diagonalB, Types.dimension_t m, Types.dimension_t n, Types.dimension_t k, Types.value_t alpha, Types.matrix_t A, Types.matrix_t B, Types.value_t beta=0.0, Types.matrix_t C=[])	

Matrix multiply when either A or B is a diagonal and is passed as a vector.

Computes: alpha*op(A) op(B) + beta*C where op() is transpose.

PARAMETER **transposeA** ||| BOOLEAN — true when transpose of A is used.

PARAMETER **transposeB** ||| BOOLEAN — true when transpose of B is used.

PARAMETER **diagonalA** ||| BOOLEAN — true when A is the diagonal matrix.

PARAMETER **diagonalB** ||| BOOLEAN — true when B is the diagonal matrix.

PARAMETER **m** ||| UNSIGNED4 — number of rows in product.

PARAMETER **n** ||| UNSIGNED4 — number of columns in product.

PARAMETER **k** ||| UNSIGNED4 — number of columns/rows for the multiplier/multiplicand.

PARAMETER **alpha** ||| REAL8 — scalar used on A.

PARAMETER **A** ||| SET (REAL8) — matrix A.

PARAMETER **B** ||| SET (REAL8) — matrix B.

PARAMETER **beta** ||| REAL8 — scalar for matrix C.

PARAMETER **C** ||| SET (REAL8) — matrix C or empty.

RETURN **SET (REAL8)** — result matrix in matrix_t format.

SEE Std.BLAS.Types.matrix_t

---

enum_workitems

Go Up

DESCRIPTIONS

ENUM_WORKITEMS

enum_workitems

```
(dsIn, dsOut, src_field, wi_name)
```

Create an enumeration of string contents to be used as work items.

This macro produces 2 external symbols, dsOut and dsOut_Map.

The dsOut extends the input dataset with a numeric work-item number.

The dsOut_Map dataset captures the relationship between the strings that name the work items and the nominal assigned in Workitem_Mapping format.

PARAMETER **dsIn** ||| INTEGER8 — the input recordset.

PARAMETER **dsOut** ||| INTEGER8 — the symbol to use for the appended data.

PARAMETER **src_field** ||| INTEGER8 — a field name to use to discriminate work-items.

PARAMETER **wi_name** ||| INTEGER8 — the field name for the work item value assigned.

RETURN — Nothing. The macro creates the symbols 'dsOut' and 'dsOut_Map' inline.

SEE Types.Workitem_Mapping

ExtractBeta

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

EXTRACTBETA

/ EXPORT	ExtractBeta
(DATASET(Core_Types.Layout_Model) mod_ds)	

Extract the beta values form the model dataset.

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 ind_col , UNSIGNED4 dep_nom , REAL8 w , REAL8 SE })** — the beta values as Model_Coef records, with zero as the constant term.

SEE Types.Model_Coef

ExtractBeta_CI

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

EXTRACTBETA_CI

/ EXPORT DATASET(Types.Confidence_Model_Coef)	**ExtractBeta_CI**
(DATASET(Core_Types.Layout_Model) mod_ds, REAL8 level)	

Extract the beta values and confidence intervals from the model dataset.

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

PARAMETER **level** ||| REAL8 — the significance value for the intervals.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 ind_col , UNSIGNED4 dep_nom , REAL8 w , REAL8 SE , REAL8 upper , REAL8 lower })** — the beta values with confidence intervals in Confidence_Model_Coef format, with zero as the constant term.

SEE Types.Confidence_Model_Coef

ExtractBeta_full

Go Up

IMPORTS

Types | ML_Core.Math | ML_Core.Types |

DESCRIPTIONS

EXTRACTBETA_FULL

/ EXPORT DATASET(Types.Full_Model_Coef)	**ExtractBeta_full**
(DATASET(Core_Types.Layout_Model) mod_ds, REAL8 level=0.05)	

Extract the coefficient information including confidence intervals, z and p values.

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

PARAMETER **level** ||| REAL8 — the significance value for the intervals.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 ind_col , UNSIGNED4 dep_nom , REAL8 w , REAL8 SE , REAL8 z , REAL8 p_value , REAL8 upper , REAL8 lower })** — the coefficient information for the model in Full_Model_Coef format, with zero as the constant term.

SEE Types.Full_Model_Coef

ExtractBeta_pval

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

EXTRACTBETA_PVAL

`/ EXPORT DATASET(Types.pval_Model_Coef)`	**ExtractBeta_pval**
`(DATASET(Core_Types.Layout_Model) mod_ds)`	

Extract the beta values including z and p value from the model.

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 ind_col , UNSIGNED4 dep_nom , REAL8 w , REAL8 SE , REAL8 z , REAL8 p_value })** — the beta values with p-values in pval_Model_Coef format, with zero as the constant term.

SEE Types.pval_Model_Coef

ExtractReport

Go Up

IMPORTS

Types | Constants | ML_Core.Types |

DESCRIPTIONS

EXTRACTREPORT

/ EXPORT DATASET(Types.Model_Report)	**ExtractReport**
(DATASET(Core_Types.Layout_Model) mod_ds)	

Create a model report from a model.

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 max_iterations , REAL8 epsilon , UNSIGNED4 dep_vars , UNSIGNED4 ind_vars , UNSIGNED8 obs , UNSIGNED2 builder , TABLE (Classifier_Stats) stats })** — the model report in Model_Report format.

SEE Types.Model_Report

LogitPredict

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

LOGITPREDICT

/ EXPORT DATASET(Classify_Result)	**LogitPredict**
(DATASET(Model_Coef) coef, DATASET(NumericField) independents)	

Predict the category values with the logit function and the the supplied beta coefficients.

PARAMETER **coef** ||| TABLE (Model_Coef) — the model beta coefficients as returned from ExtractBeta.

PARAMETER **independents** ||| TABLE (NumericField) — the observations.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , INTEGER4 value , REAL8 conf })** — the predicted category values and a confidence score in Classify_Result format.

SEE ExtractBeta

SEE ML_Core.Types.Classify_Result

LogitScore

Go Up

IMPORTS

Types | ML_Core.Types |

DESCRIPTIONS

LOGITSCORE

```
/ EXPORT DATASET(Raw_Prediction) LogitScore
(DATASET(Model_Coef) coef, DATASET(NumericField)
independents)
```

Calculate the score using the logit function and the the supplied beta coefficients.

PARAMETER **coef** ||| TABLE (Model_Coef) — the model beta coefficients as returned from ExtractBetas.

PARAMETER **independents** ||| TABLE (NumericField) — the observations.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED8 id , UNSIGNED4 number , REAL8 raw })** — the raw prediction value in Raw_Prediction format.

SEE ExtractBetas

SEE Types.Raw_Prediction

LUCI_Model

Go Up

IMPORTS

Types | std.Str | std.system.ThorLib |

DESCRIPTIONS

LUCI_MODEL

/ EXPORT DATASET(Types.LUCI_Rec)	**LUCI_Model**
(DATASET(Types.LUCI_Model_Rqst) rqst, DATASET(Types.External_Model) mod, STRING wi_field=’work_item’)	

Create a LUCI model file description of the model(s) from the external version of the model.

LUCI is a proprietary format used within LexisNexis.

The multi-score card per model case assumes that the score card selection is based solely upon the work item field. If this is not the case, the L1SE records will need to be patched.

The model id and name may have a ”$” character that is updated to match the work item when there are multiple models applied. If the strings do not have a ”$” character, the work item string is appended.

The score card name may have a ”$” character which is updated to match the work item. If the name is blank, the score card is named for the work item.

LUCI data fields may not contain comma characters. This function requires that the work item identification strings do not contain characters that need special handling for CSV data.

PARAMETER **rqst** ||| TABLE (LUCI_Model_Rqst) — the information to map work items to models in LUCI_Model_Rqst format.

PARAMETER **mod** ||| TABLE (External_Model) — the model with the external field names applied in External_Model format as returned from Named_Model.

PARAMETER **wi_field** ||| STRING — the field name holding the work item identification string.

RETURN **TABLE ({ STRING line })** — The lines of the LUCI file in LUCI_Rec format.

SEE Types.External_Model

SEE Named_Model

SEE Types.LUCI_Model_Rqst

SEE Types.LUCI_Rec

Model_Deviance

Go Up

IMPORTS

Types |

DESCRIPTIONS

MODEL_DEVIANCE

/ EXPORT DATASET(Types.Deviance_Record)	**Model_Deviance**
(DATASET(Types.Observation_Deviance) od, DATASET(Types.Model_Coef) mod)	

Model Deviance Report.

Create a report of deviance information for a model.

Analysis of Deviance is analogous to the Analysis of Variance (ANOVA) used in least-squares modeling, but adapted to the general linear model (GLM). In this case it is adapted specifically to the logistic model.

PARAMETER **od** ||| TABLE (Observation_Deviance) — observation-deviance records, as obtained from a call to Deviance_Detail.

PARAMETER **mod** ||| TABLE (Model_Coef) — model co-efficients records, as obtained from a call to ExtractBeta.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , UNSIGNED8 df , REAL8 deviance , REAL8 AIC })** — model deviance in Deviance_Record format.

SEE Deviance_Detail

SEE ExtractBeta

SEE Types.Deviance_Record

Named_Model

Go Up

IMPORTS

Types |

DESCRIPTIONS

NAMED_MODEL

```
/ EXPORT DATASET(Types.External_Model)   Named_Model
(DATASET(Types.Layout_Model) mod_ds,
DATASET(Types.FieldName_Mapping) expl_map,
DATASET(Types.FieldName_Mapping) resp_map,
DATASET(Types.WorkItem_mapping) wi_map=empty,
REAL8 level=0.05)
```

Apply external labels for work items and field names to a model.

Returns an expanded model that includes:

- coefficients
- z and p-values
- independent variable field names
- dependent variable field names
- work-item names

PARAMETER **mod_ds** ||| TABLE (Layout_Model) — the model as returned from GetModel.

PARAMETER **expl_map** ||| TABLE (FieldName_Mapping) — the relation of the explanatory or independent variables to the field names for those variables in FieldName_Mapping format.

PARAMETER **resp_map** ||| TABLE (FieldName_Mapping) — the relation of the response variable column numbers to the field names in FieldName_Mapping format.

PARAMETER **wi_map** ||| TABLE (WorkItem_Mapping) — (optional) mapping of workitem strings to workitem nominals in FieldName_Mapping format.

PARAMETER **level** ||| REAL8 — (optional) value for confidence intervals. Default = 0.05.

RETURN **TABLE ({ STRING work_item , STRING response_field , UNSIGNED2 wi , UNSIGNED4 dep_nom , TABLE (External_Coef) coef })** — an expanded model in External_Model format.

SEE Types.FieldName_Mapping

SEE Types.External_Model

Null_Deviance

Go Up

IMPORTS

Types |

DESCRIPTIONS

NULL_DEVIANCE

/ EXPORT DATASET(Types.Deviance_Record)	**Null_Deviance**
(DATASET(Types.Observation_Deviance) od)	

Return Deviance information for the null model, that is, a model with only an intercept.

Analysis of Deviance is analogous to the Analysis of Variance (ANOVA) used in least-squares modeling, but adapted to the general linear model (GLM). In this case it is adapted specifically to the logistic model.

PARAMETER **od** ||| TABLE (Observation_Deviance) — Observation Deviance record set as returned from Deviance_Detail.

RETURN **TABLE ({ UNSIGNED2 wi , UNSIGNED4 classifier , UNSIGNED8 df , REAL8 deviance , REAL8 AIC })** — a data set of the null model deviances for each work item and classifier in Deviance_Record format.

SEE Types.Observation_Deviance

SEE Types.Deviance_Record

SEE Deviance_Detail

Types

Go Up

IMPORTS

ML_Core.Types |

DESCRIPTIONS

TYPES

Types

Type definitions for LogisticRegression bundle

Children

1. AnyField : No Documentation Found
2. NumericField : The NumericField layout defines a matrix of Real valued data-points
3. DiscreteField : The Discrete Field layout defines a matrix of Integer valued data-points
4. Layout_Model : No Documentation Found
5. t_work_item : No Documentation Found
6. t_RecordID : No Documentation Found
7. t_FieldNumber : No Documentation Found
8. t_FieldReal : No Documentation Found
9. t_Discrete : No Documentation Found

10. t_Universe : No Documentation Found
11. Field_Desc : Describe information about each field in a training set
12. Data_Info : Describes information about a training dataset composed of independent and dependent columns
13. NumericField_U : Record structure to add a ’Universe Number’ to a NumericField allowing multiple independent NumericField matrixes within a work-item
14. DiscreteField_U : Record structure to add a ’Universe Number’ to a DiscreteField allowing multiple independent DiscreteField matrixes within a work-item
15. Layout_Column_Map : Layout for a column map record that is used to remap column numbers
16. Classifier_Stats : Statistics about the effectiveness of each classifier in a model
17. Model_Report : Statistical information about a model
18. Binomial_Confusion_Summary : Accuracy stats for binomial classifications
19. Model_Coef : Model Coefficients
20. Confidence_Model_Coef : Model Coefficients with confidence intervals
21. pval_Model_Coef : Model coefficients with z and p-value
22. Full_Model_Coef : Model coefficients with confidence intervals and p-value
23. External_Coef : Model coefficients, confidence intervals, and p-value, plus independent field names, for each coefficient
24. External_Model : Expanded version of a model with statistics and field names
25. Raw_Prediction : Record for raw prediction without confidence information
26. Observation_Deviance : Record to contain deviance information about each observation
27. Deviance_Record : Record to hold deviance summary information about a model
28. AOD_Record : Record to hold Analysis of Deviance (AOD) information for a model
29. FieldName_Mapping : Layout used to hold the mapping between a field’s number and its name
30. WorkItem_Mapping : Layout used to hold the mapping between a work-item number and a textual name for that work-item
31. LUCI_Rec : Layout to store the lines of a generated LUCI model file
32. LUCI_Model_Rqst : Format for information to guide the generation of a LUCI file

ANYFIELD

Types /

AnyField

No Documentation Found

NUMERICFIELD

Types /

NumericField

The NumericField layout defines a matrix of Real valued data-points. It acts as the primary Dataset layout for interacting with most ML Functions. Each record represents a single cell in a matrix. It is most often used to represent a set of data-samples or observations, with the 'id' field representing the data-sample or observation, and the 'number' field representing the various fields within the observation.

FIELD **wi** ||| — The work-item id, supporting the Myriad style interface. This allows multiple independent matrixes to be contained within a single dataset, supporting independent ML activities to be processed in parallel.

FIELD **id** ||| — This field represents the row-number of this cell of the matrix. It is also considered the record-id for observations / data-samples.

FIELD **number** ||| — This field represents the matrix column number for this cell. It is also considered the field number of the observation

FIELD **value** ||| — The value of this cell in the matrix.

DISCRETEFIELD

Types /

DiscreteField

The Discrete Field layout defines a matrix of Integer valued data-points. It is similar to the NumericField layout above, except for only containing discrete (integer) values. It is typically used to convey the class-labels for classification algorithms.

FIELD **wi** ||| — The work-item id, supporting the Myriad style interface. This allows multiple independent matrixes to be contained within a single dataset, supporting independent ML activities to be processed in parallel.

FIELD **id** ||| — This field represents the row-number of this cell of the matrix. It is also considered the record-id for observations / data-samples.

FIELD **number** ||| — This field represents the matrix column number for this cell. It is also considered the field number of the observation

FIELD **value** ||| — The value of this cell in the matrix.

LAYOUT_MODEL

Types /

Layout_Model

No Documentation Found

T_WORK_ITEM

Types /

t_work_item

No Documentation Found

RETURN **UNSIGNED2** —

T_RECORDID

Types /

t_RecordID

No Documentation Found

RETURN **UNSIGNED8** —

T_FIELDNUMBER

Types /

t_FieldNumber

No Documentation Found

RETURN **UNSIGNED4** —

T_FIELDREAL

Types /

t_FieldReal

No Documentation Found

RETURN **REAL8** —

T_DISCRETE

Types /

t_Discrete

No Documentation Found

RETURN **INTEGER4** —

T_UNIVERSE

Types /

t_Universe

No Documentation Found

RETURN **UNSIGNED1** —

FIELD_DESC

Types /

Field_Desc

Describe information about each field in a training set.

FIELD **number** ||| UNSIGNED4 — the column (feature) number.

FIELD **cardinality** ||| UNSIGNED4 — the number of unique values in the field.

FIELD **min_value** ||| REAL8 — the minimum value for the field.

FIELD **max_value** ||| REAL8 — the maximum value for the field.

DATA_INFO

Types /

Data_Info

Describes information about a training dataset composed of independent and dependent columns.

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

FIELD **dependent_fields** ||| UNSIGNED4 — the number of fields in the dependent data.

FIELD **dependent_records** ||| UNSIGNED4 — the number of records in the dependent data.

FIELD **independent_fields** ||| UNSIGNED4 — the number of fields in the independent data.

FIELD **independent_records** ||| UNSIGNED4 — the number of records in the independent data.

FIELD **dependent_stats** ||| TABLE (Field_Desc) — dataset of Field_Desc records describing each of the fields of the dependent data.

FIELD **independent_stats** ||| TABLE (Field_Desc) — dataset of Field_Desc records describing each of the fields of the independent data.

FIELD **dependent_count** ||| UNSIGNED4 — No Doc

FIELD **independent_count** ||| UNSIGNED4 — No Doc

SEE Field_Desc

NUMERICFIELD_U

Types /

NumericField_U

Record structure to add a 'Universe Number' to a NumericField allowing multiple independent NumericField matrixes within a work-item.

FIELD **u** ||| UNSIGNED1 — the ’universe’ number identifying a distinct matrix within a NumericField dataset and work-item.

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| UNSIGNED8 — No Doc

FIELD **number** ||| UNSIGNED4 — No Doc

FIELD **value** ||| REAL8 — No Doc

DISCRETEFIELD_U

Types /

DiscreteField_U

Record structure to add a ’Universe Number’ to a DiscreteField allowing multiple independent DiscreteField matrixes within a work-item.

FIELD **u** ||| UNSIGNED1 — the ’universe’ number identifying a distinct matrix within a DiscreteField dataset and work-item.

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| UNSIGNED8 — No Doc

FIELD **number** ||| UNSIGNED4 — No Doc

FIELD **value** ||| INTEGER4 — No Doc

LAYOUT_COLUMN_MAP

Types /

Layout_Column_Map

Layout for a column map record that is used to remap column numbers.

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

FIELD **orig_number** ||| UNSIGNED4 — the original field number.

FIELD **remap_number** ||| UNSIGNED4 — the mapped-to field number.

CLASSIFIER_STATS

Types /

Classifier_Stats

Statistics about the effectiveness of each classifier in a model.

FIELD **column** ||| UNSIGNED4 — the classifier field number.

FIELD **max_delta** ||| REAL8 — the max_delta value for the classifier.

FIELD **iterations** ||| UNSIGNED4 — the number of iterations used to train the classifier.

FIELD **correct** ||| UNSIGNED4 — the number of classes predicted correctly in the training data.

FIELD **incorrect** ||| UNSIGNED4 — the number of classes predicted incorrectly in the training data.

MODEL_REPORT

Types /

Model_Report

Statistical information about a model.

One record is generated per work-item.

FIELD **wi** ||| UNSIGNED2 — the work-item

FIELD **max_iterations** ||| UNSIGNED4 — the maximum iterations use to train the model.

FIELD **epsilon** ||| REAL8 — the ’epsilon’ value used within the model.

FIELD **dep_vars** ||| UNSIGNED4 — the number of dependent variables (i.e. classifiers).

FIELD **ind_vars** ||| UNSIGNED4 — the number of independent variables (i.e. features).

FIELD **obs** ||| UNSIGNED8 — the number of observations (i.e. records) in the training data.

FIELD **builder** ||| UNSIGNED2 — the identifier for the builder used to train the model.

FIELD **stats** ||| TABLE (Classifier_Stats) — child dataset of Classifier_Stats, one for each classifier in the work-item.

SEE Classifier_Stats

BINOMIAL_CONFUSION_SUMMARY

Types /

Binomial_Confusion_Summary

Accuracy stats for binomial classifications.

One record per work-item and classifier.

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

FIELD **classifier** ||| UNSIGNED4 — the classifier field number (i.e. dependent field number).

FIELD **true_positive** ||| UNSIGNED8 — the count of true positive results (i.e. predicted = TRUE, actual = TRUE).

FIELD **true_negative** ||| UNSIGNED8 — the count of true negative results (i.e. predicted = FALSE, actual = FALSE).

FIELD **false_positive** ||| UNSIGNED8 — the count of false_positive results (i.e. predicted = TRUE, actual = FALSE).

FIELD **false_negative** ||| UNSIGNED8 — the count of false_negative results (i.e. predicted = FALSE, actual = TRUE).

FIELD **cond_pos** ||| UNSIGNED8 — the count of results where actual = TRUE.

FIELD **pred_pos** ||| UNSIGNED8 — the count of results where predicted = TRUE.

FIELD **cond_neg** ||| UNSIGNED8 — the count of results where actual = FALSE.

FIELD **pred_neg** ||| UNSIGNED8 — the count of results where predicted = FALSE.

FIELD **prevalence** ||| REAL8 — cond_pos / total.

FIELD **accuracy** ||| REAL8 — (true_positive + true_negative) / total.

FIELD **true_pos_rate** ||| REAL8 — true_positive / cond_pos.

FIELD **false_pos_rate** ||| REAL8 — false_positive / cond_neg.

FIELD **true_neg_rate** ||| REAL8 — true_negative / cond_neg.

FIELD **pos_pred_val** ||| REAL8 — true_positive / pred_pos.

FIELD **false_disc_rate** ||| REAL8 — false_positive / pred_pos.

FIELD **false_omit_rate** ||| REAL8 — false_negative / pred_neg.

FIELD **neg_pred_val** ||| REAL8 — true_negative / pred_neg.

FIELD **false_neg_rate** ||| REAL8 — No Doc

MODEL_COEF

Types /

Model_Coef

Model Coefficients.

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

FIELD **ind_col** ||| UNSIGNED4 — the independent column number (i.e feature number).

FIELD **dep_nom** ||| UNSIGNED4 — the dependent column number (i.e. classifier number).

FIELD **w** ||| REAL8 — the learned weight (i.e. coefficient).

FIELD **SE** ||| REAL8 — the Standard Error of the coefficient.

CONFIDENCE_MODEL_COEF

Types /

Confidence_Model_Coef

Model Coefficients with confidence intervals.

FIELD **upper** ||| REAL8 — the upper range of the confidence interval

FIELD **lower** ||| REAL8 — the lower range of the confidence interval

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **ind_col** ||| UNSIGNED4 — No Doc

FIELD **dep_nom** ||| UNSIGNED4 — No Doc

FIELD **w** ||| REAL8 — No Doc

FIELD **se** ||| REAL8 — No Doc

PVAL_MODEL_COEF

Types /

pval_Model_Coef

Model coefficients with z and p-value.

FIELD **z** ||| REAL8 — the z value.

FIELD **p_value** ||| REAL8 — the p_value of the coefficient.

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **ind_col** ||| UNSIGNED4 — No Doc

FIELD **dep_nom** ||| UNSIGNED4 — No Doc

FIELD **w** ||| REAL8 — No Doc

FIELD **se** ||| REAL8 — No Doc

FULL_MODEL_COEF

Types /

Full_Model_Coef

Model coefficients with confidence intervals and p-value

FIELD **z** ||| REAL8 — the z value.

FIELD **p_value** ||| REAL8 — the p_value of the coefficient.

FIELD **upper** ||| REAL8 — the upper range of the confidence interval

FIELD **lower** ||| REAL8 — the lower range of the confidence interval

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **ind_col** ||| UNSIGNED4 — No Doc

FIELD **dep_nom** ||| UNSIGNED4 — No Doc

FIELD **w** ||| REAL8 — No Doc

FIELD **se** ||| REAL8 — No Doc

EXTERNAL_COEF

Types /

External_Coef

Model coefficients, confidence intervals, and p-value, plus independent field names, for each coefficient.

FIELD **isIntercept** ||| BOOLEAN — Boolean field is TRUE if this is the intercept coefficient, otherwise FALSE.

FIELD **field_name** ||| STRING — the name of the independent field for this coefficient.

FIELD **w** ||| REAL8 — the coefficient value (weight)

FIELD **SE** ||| REAL8 — the Standard Error of the coefficient

FIELD **z** ||| REAL8 — the z value.

FIELD **p_value** ||| REAL8 — the p-value.

FIELD **upper** ||| REAL8 — the upper bound of the confidence interval.

FIELD **lower** ||| REAL8 — the lower bound of the confidence interval.

FIELD **ind_col** ||| UNSIGNED4 — the field number of the independent field for this coefficient.

EXTERNAL_MODEL

Types /

External_Model

Expanded version of a model with statistics and field names.

Field names include independent data field names, dependent data field names and work-item names.

FIELD **work_item** ||| STRING — the work-item's name.

FIELD **response_field** ||| STRING — the name of the classifier field (i.e. dependent field name).

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

FIELD **dep_nom** ||| UNSIGNED4 — the field number of the classifier (i.e. dependent field number).

FIELD **coef** ||| TABLE (External_Coef) — child dataset of External_Coef format. One record per model coefficient.

SEE External_Coef

RAW_PREDICTION

Types /

Raw_Prediction

Record for raw prediction without confidence information.

FIELD **raw** ||| REAL8 — the raw prediction value.

FIELD **wi** ||| UNSIGNED2 — No Doc

FIELD **id** ||| UNSIGNED8 — No Doc

FIELD **number** ||| UNSIGNED4 — No Doc

OBSERVATION_DEVIANCE

Types /

Observation_Deviance

Record to contain deviance information about each observation.

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

FIELD **id** ||| UNSIGNED8 — the record id (i.e. observation number).

FIELD **classifier** ||| UNSIGNED4 — the dependent field number.

FIELD **actual** ||| INTEGER4 — the actual (i.e. ground truth value).

FIELD **predicted** ||| INTEGER4 — the value predicted by the model.

FIELD **mod_ll** ||| REAL8 — log likelihood of the model

FIELD **mod_dev_component** ||| REAL8 — the deviance explained by the model

FIELD **mod_dev_residual** ||| REAL8 — the deviance not explained by the model (i.e. the residual)

FIELD **nil** ||| — ll log likelihood of the nil model (i.e. model with only a constant term).

FIELD **nil_dev_component** ||| REAL8 — the deviance explained by the null model

FIELD **nil_dev_residual** ||| REAL8 — the deviance not explained by the null model (i.e. the residual)

FIELD **nil_ll** ||| REAL8 — No Doc

DEVIANCE_RECORD

Types /

Deviance_Record

Record to hold deviance summary information about a model.

FIELD **wi** ||| UNSIGNED2 — the work-item number

FIELD **classifier** ||| UNSIGNED4 — the classifier number (i.e. field number of the dependent variable).

FIELD **df** ||| UNSIGNED8 — degrees-of-freedom of the chi squared distribution.

FIELD **deviance** ||| REAL8 — the total deviance for this classifier.

FIELD **AIC** ||| REAL8 — the Akaike Information Criteria value.

AOD_RECORD

Types /

AOD_Record

Record to hold Analysis of Deviance (AOD) information for a model.

FIELD **wi** ||| UNSIGNED2 — the work-item number

FIELD **classifier** ||| UNSIGNED4 — the classifier number (i.e. field number of the dependent variable).

FIELD **df** ||| UNSIGNED8 — degrees of freedom of the chi squared distribution.

FIELD **residual_dev** ||| REAL8 — the deviance not explained by the model.

FIELD **deviance** ||| REAL8 — the total deviance.

FIELD **p** ||| — value the probability that the null hypothesis is correct.

FIELD **residual_df** ||| UNSIGNED8 — No Doc

FIELD **p_value** ||| REAL8 — No Doc

FIELDNAME_MAPPING

Types /

FieldName_Mapping

Layout used to hold the mapping between a field's number and its name.

FIELD **orig_name** ||| STRING — typically the field number as a text string (e.g. '2').

FIELD **assigned_name** ||| STRING — the textual name of the field (e.g. 'age').

WORKITEM_MAPPING

Types /

WorkItem_Mapping

Layout used to hold the mapping between a work-item number and a textual name for that work-item.

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

FIELD **orig_wi** ||| STRING — the work-item name.

LUCI_REC

Types /

LUCI_Rec

Layout to store the lines of a generated LUCI model file.

FIELD **line** ||| STRING — the text for a single line for the LUCI file.

LUCI_MODEL_RQST

Types /

LUCI_Model_Rqst

Format for information to guide the generation of a LUCI file.

FIELD **model_id** ||| STRING — a short textual name for the model as used in the LUCI L1MD format.

FIELD **model_name** ||| STRING — an expanded name for the model as used in the LUCI L1MD format.

FIELD **response_field** ||| STRING — name of the dependent field (aka classifier name).

FIELD **wi_list** ||| SET (STRING) — can be set to ['ALL'], or can be a list of work-item names.

FIELD **score_card_name** ||| STRING — the score card name pattern (see LUCI_Model.ecl for details).