

# Interfaces

---

[Go Up](#)

## Table of Contents

<a href="#">IClassify.ecl</a>
Interface definition for Classification
<a href="#">IRegression.ecl</a>
Interface Definition for Regression Modules Regression learns a function that maps a set of input data to one or more output variables

# ML\_Core/ Interfaces/ IClassify

---

[Go Up](#)

## IMPORTS

ML\_Core | ML\_Core.Types |

## DESCRIPTIONS

### **MODULE** IClassify

	IClassify
--	-----------

Interface definition for Classification. Actual implementation modules will probably take parameters.

#### Children

1. [GetModel](#) : Calculate the model to fit the observation data to the observed classes
2. [Classify](#) : Classify the observations using a model
3. [Report](#) : Report the confusion matrix for the classifier and training data

---

### **FUNCTION** GetModel

[IClassify](#) \

<code>DATASET(Types.Layout_Model)</code>	<b>GetModel</b>
<code>(DATASET(Types.NumericField) observations,  DATASET(Types.DiscreteField) classifications)</code>	

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

**PARAMETER** observations the observed explanatory values

**PARAMETER** classifications the observed classification used to build the model

**RETURN** the encoded model

## FUNCTION Classify

`IClassify \`

<code>DATASET(Types.Classify_Result)</code>	<b>Classify</b>
<code>(DATASET(Types.Layout_Model) model,  DATASET(Types.NumericField) new_observations)</code>	

Classify the observations using a model.

**PARAMETER** model The model, which must be produced by a corresponding getModel function.

**PARAMETER** new\_observations observations to be classified

**RETURN** Classification with a confidence value

## FUNCTION Report

`IClassify \`

<code>DATASET(Types.Confusion_Detail)</code>	<b>Report</b>
<code>(DATASET(Types.Layout_Model) model,  DATASET(Types.NumericField) observations,  DATASET(Types.DiscreteField) classifications)</code>	

Report the confusion matrix for the classifier and training data.

**PARAMETER** model the encoded model

**PARAMETER** observations the explanatory values.

**PARAMETER** classifications the classifications associated with the observations

**RETURN** the confusion matrix showing correct and incorrect results

---

# ML\_Core/ Interfaces/ IRegression

---

[Go Up](#)

## IMPORTS

ML\_Core | ML\_Core.Types |

## DESCRIPTIONS

### **MODULE** IRegression

IRegression
(DATASET(NumericField) X=empty_data, DATASET(NumericField) Y=empty_data)

Interface Definition for Regression Modules Regression learns a function that maps a set of input data to one or more output variables. The resulting learned function is known as the model. That model can then be used repetitively to predict (i.e. estimate) the output value(s) based on new input data.

**PARAMETER** X The independent data in DATASET(NumericField) format. Each statistical unit (e.g. record) is identified by 'id', and each feature is identified by field number (i.e. 'number').

**PARAMETER** Y The dependent variable(s) in DATASET(NumericField) format. Each statistical unit (e.g. record) is identified by 'id', and each feature is identified by field number (i.e. 'number').

### Children

1. [GetModel](#) : Calculate and return the 'learned' model The model may be persisted and later used to make predictions using 'Predict' below
2. [Predict](#) : Predict the output variable(s) based on a previously learned model

---

## ATTRIBUTE GetModel

IRegression \

DATASET(Layout_Model)	GetModel
-----------------------	----------

Calculate and return the 'learned' model The model may be persisted and later used to make predictions using 'Predict' below.

**RETURN** DATASET(LayoutModel) describing the learned model parameters

---

## FUNCTION Predict

IRegression \

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

Predict the output variable(s) based on a previously learned model

**PARAMETER** newX DATASET(NumericField) containing the X values to b predicted.

**RETURN** DATASET(NumericField) containing one entry per observation (i.e. id) in newX. This represents the predicted values for Y.

---