



```
/*This document is a code snippet, showcasing how to build a multiple linear
regression model native on hana */
```

```
-----
/* MULTIPLE LINEAR REGRESSION: MODEL*/
-----
```

```
-----
/*DATA TABLE DEFFINITION*/
-----
```

```
/*
```

The data definition phase, has the sole purpose of laying the foundation for the procedure, when generating the procedure, we must feed the parameter tables.

The parameter tables have no data in them so far and their structure must be defined.

All the tables will be in another table ("RM_PAL_MLR_PDATA_TBL"), this will be specified in the procedure generation wrapper.

```
*/
```

```
SET SCHEMA PAL;
```

```
DROP TYPE RM_PAL_MLR_DATA_T;
```

```
CREATE TYPE RM_PAL_MLR_DATA_T AS TABLE (
```

```
    /*first column must always be a unique value ( primary key ) */
```

```
    "ID" INTEGER NOT NULL PRIMARY KEY GENERATED BY DEFAULT AS IDENTITY,
    /*automatically generated id for our input data */
```

```
    "SALES" DOUBLE, /*dependant variable(target column */
```

```
    /*regression function by default views the second column from the dataset
as the target column */
```



```
/*the rest of the columns are independent variables*/
```

```
"STORE" INTEGER,  
"STORE_TYPE" VARCHAR(10),  
"ASSORTMENT" varchar(10),  
"PROMO" INTEGER,  
"PROMO2" INTEGER,  
"PROMO2_SINCE_WEEK" INTEGER,  
"COMP_OP_Y" INTEGER,  
"SCHOOL_H" VARCHAR(10)  
);
```

```
-----  
/* RESULT TABLE DEFFINITION */  
-----
```

```
DROP TYPE RM_PAL_MLR_RESULT_T;  
CREATE TYPE RM_PAL_MLR_RESULT_T AS TABLE("Coefficient"  
varchar(50),"CoefficientValue" DOUBLE);
```

```
-----  
/* FITTED VALUES TABLE DEFFINITION */  
-----
```

```
DROP TYPE RM_PAL_MLR_FITTED_T;  
CREATE TYPE RM_PAL_MLR_FITTED_T AS TABLE("ID" INT,"Fitted" DOUBLE);
```



```
/*COEFFICIENT TABLE DEFFINITION */
```

```
-----
```

```
DROP TYPE RM_PAL_MLR_SIGNIFICANCE_T;
```

```
CREATE TYPE RM_PAL_MLR_SIGNIFICANCE_T AS TABLE("NAME" varchar(50),"VALUE"  
DOUBLE);
```

```
-----
```

```
/* PMML MODEL TABLE DEFFINITION */
```

```
-----
```

```
DROP TYPE RM_PAL_MLR_PMMLMODEL_T;
```

```
CREATE TYPE RM_PAL_MLR_PMMLMODEL_T AS TABLE("ID" INT,"Model" varchar(5000));
```

```
-----
```

```
/* CONTROL TABLE DEFFINITION */
```

```
-----
```

```
DROP TYPE RM_PAL_CONTROL_T;
```

```
CREATE TYPE RM_PAL_CONTROL_T AS TABLE("NAME" VARCHAR(100), "INTARGS" INT,  
"DOUBLEARGS" DOUBLE,"STRINGARGS" VARCHAR(100));
```



/* PARAMATER TABLE CREATION */

/*this table will be used to specify the input and output parameters of the procedure and the structure of the parameters*/

```
DROP TABLE RM_PAL_MLR_PDATA_TBL;
CREATE COLUMN TABLE RM_PAL_MLR_PDATA_TBL("POSITION" INT, "SCHEMA_NAME"
NVARCHAR(256), "TYPE_NAME" NVARCHAR(256), "PARAMETER_TYPE" VARCHAR(7));
INSERT INTO RM_PAL_MLR_PDATA_TBL values (1,'PAL','RM_PAL_MLR_DATA_T','IN');
INSERT INTO RM_PAL_MLR_PDATA_TBL values (2,'PAL','RM_PAL_CONTROL_T','IN');
INSERT INTO RM_PAL_MLR_PDATA_TBL values (3,'PAL','RM_PAL_MLR_RESULT_T','OUT');
INSERT INTO RM_PAL_MLR_PDATA_TBL values (4,'PAL','RM_PAL_MLR_FITTED_T','OUT');
INSERT INTO RM_PAL_MLR_PDATA_TBL values
(5,'PAL','RM_PAL_MLR_SIGNIFICANCE_T','OUT');
INSERT INTO RM_PAL_MLR_PDATA_TBL values
(6,'PAL','RM_PAL_MLR_PMMLMODEL_T','OUT');
```



```
-----  
/* PROCEDURE GENERATION */  
-----
```

```
/*before generating a procedure wrapper, make sure that you erase any previously  
created one with the same name*/
```

```
CALL SYS.AFLLANG_WRAPPER_PROCEDURE_DROP('PAL','RM_PAL_LR_PROC');
```

```
/*generate procedure wrapper,
```

```
“ALFPAL” is the area where the “LRREGRESSION” Function is,
```

```
“PAL” is the schema name where the procedure will be stored,
```

```
“RM_PAL_LR_PROC” is the name of generated procedure,
```

```
“RM_PAL_MLR_PDATA_TBL” is the parameter table
```

```
*/
```

```
CALL
```

```
SYS.AFLLANG_WRAPPER_PROCEDURE_CREATE('ALFPAL','LRREGRESSION','PAL','RM_PAL_LR  
_PROC',RM_PAL_MLR_PDATA_TBL);
```

```
/*Now that the parameters have been defined and the procedure created it's time  
to insert the training data and specify the regression function*/
```

```
-----  
/* CONTROL TABLE: INPUT TABLE */  
-----
```

```
/*this table will contain the specifications for the 'LRREGRESSION' function */
```

```
DROP TABLE #RM_PAL_CONTROL_TBL;
```

```
CREATE LOCAL TEMPORARY COLUMN TABLE #RM_PAL_CONTROL_TBL ("NAME"  
VARCHAR(100), "INTARGS" INT, "DOUBLEARGS" DOUBLE,"STRINGARGS" VARCHAR(100));
```

```
INSERT INTO #RM_PAL_CONTROL_TBL VALUES ('THREAD_NUMBER',15,null,null);
```

```
/*specifies number of threads ; valid only when ALG is 1,5 or 6 */
```



```
INSERT INTO #RM_PAL_CONTROL_TBL VALUES ('ALG', 6, null, null); /*Specifies
algorithms for solving the least square problem*/
INSERT INTO #RM_PAL_CONTROL_TBL VALUES ('ENET_LAMBDA', null, 0.03194, null);
/*penalized weight ; must be greater or equal than 0 */
INSERT INTO #RM_PAL_CONTROL_TBL VALUES ('ENET_ALPHA', null, 0.65, null);
/*The elastic net mixing parameter. The value range is between 0 and 1 inclusively.
*/
INSERT INTO #RM_PAL_CONTROL_TBL VALUES ('PMML_EXPORT',0,null,null);
/*specifies if the MLR will be or not exported in PMML*/
```

```
-----
/* TRAINING DATA TABLE CREATION:INPUT TABLE */
-----
```

```
/*we are going to populate our training data table from our hana Calculation View
*/
```

```
SET SCHEMA PAL;
```

```
DROP TABLE RM_PAL_MLR_DATA_TBL;
```

```
CREATE COLUMN TABLE RM_PAL_MLR_DATA_TBL LIKE RM_PAL_MLR_DATA_T; --
structure of table must be exactly the same like the on specified in the procedure
generation
```

```
INSERT INTO
RM_PAL_MLR_DATA_TBL("SALES","STORE","STORE_TYPE","ASSORTMENT","PROMO","PRO
MO2","PROMO2_SINCE_WEEK","COMP_OP_Y",
    "SCHOOL_H")
SELECT
    "GPE_RMSLS" ,
    CAST("GPE_RMSTR" AS INTEGER),
    "GPE_RMSTP",
    "GPE_RMART",
    CAST("GPE_RMPRM" AS INTEGER),
```



```
CAST("GPE_RMPR2" AS INTEGER),
CAST("GPE_RMP2M" AS INTEGER),
CAST("GPE_RMCOY" AS INTEGER),
"GPE_RMIHD"
FROM "_SYS_BIC"."ZGPE.ROSSMAN/Z_PA_ROSSMAN_TRAIN_SET_CV"
WHERE CAST("GPE_RMSTR" AS INTEGER)<=10 AND CAST("GPE_RMOPN" AS
INTEGER)=1;
```

*/*we will take into consideration the sales history of 10 stores only on dates when it is open*/*

/ RESULTS TABLE CREATION:OUTPUT TABLE */*

SET SCHEMA PAL;
DROP TABLE RM_PAL_MLR_RESULTS_TBL;
CREATE COLUMN TABLE RM_PAL_MLR_RESULTS_TBL like RM_PAL_MLR_RESULT_T;

/ FITTED VALUES TABLE CREATION:OUTPUT TABLE */*

*/*this table will contain the fitted values of the target column */*
DROP TABLE RM_PAL_MLR_FITTED_TBL;
CREATE COLUMN TABLE RM_PAL_MLR_FITTED_TBL like RM_PAL_MLR_FITTED_T;

/ COEFFICIENTS TABLE CREATION:OUTPUT TABLE */*



/*this table will contain the model accuracy measures*/

DROP TABLE PAL_MLR_SIGNIFICANCE_TBL;

CREATE COLUMN TABLE PAL_MLR_SIGNIFICANCE_TBL like
RM_PAL_MLR_SIGNIFICANCE_T;

/* PMML MODEL TABLE CREATION:OUTPUT TABLE */

DROP TABLE RM_PAL_MLR_PMMLMODEL_TBL;

CREATE COLUMN TABLE RM_PAL_MLR_PMMLMODEL_TBL like
RM_PAL_MLR_PMMLMODEL_T;

/* CALL PREVIOUSLY GENERATED PROCEDURE AND SPECIFY THE INPUT TABLES AS
PARAMETERS */

CALL PAL.RM_PAL_LR_PROC(RM_PAL_MLR_DATA_TBL,"#RM_PAL_CONTROL_TBL",
RM_PAL_MLR_RESULTS_TBL, RM_PAL_MLR_FITTED_TBL, PAL_MLR_SIGNIFICANCE_TBL,
RM_PAL_MLR_PMMLMODEL_TBL) WITH OVERVIEW;



```
-----  
/* VIEW RESULTS */  
-----
```

```
SELECT * FROM RM_PAL_MLR_RESULTS_TBL;
```

```
SELECT * FROM RM_PAL_MLR_FITTED_TBL;
```

```
SELECT * FROM PAL_MLR_SIGNIFICANCE_TBL;
```

```
SET SCHEMA PAL;
```

```
DROP TABLE RM_MLR_ALL;
```

```
CREATE COLUMN TABLE RM_MLR_ALL("ID" INTEGER NOT NULL PRIMARY KEY  
GENERATED BY DEFAULT AS IDENTITY,"STORE" INTEGER, "DATE" INTEGER);
```

```
INSERT INTO RM_MLR_ALL("STORE","DATE")
```

```
    SELECT CAST("GPE_RMSTR" AS INTEGER),
```

```
    CAST("GPE_RMDAT" AS INTEGER)
```

```
    FROM "_SYS_BIC"."ZGPE.ROSSMAN/Z_PA_ROSSMAN_TRAIN_SET_CV"
```

```
        WHERE CAST("GPE_RMSTR" AS INTEGER)<=10 AND CAST("GPE_RMOPN"  
AS INTEGER)=1;
```

```
DROP VIEW RM_MLR_STATS_V;
```

```
CREATE VIEW RM_MLR_STATS_V AS
```

```
SELECT A."STORE",C."DATE",A."SALES", B."Fitted"
```

```
FROM RM_PAL_MLR_DATA_TBL A
```

```
JOIN RM_PAL_MLR_FITTED_TBL B ON A."ID"=B."ID"
```

```
JOIN RM_MLR_ALL C ON A."ID"=C."ID";
```