

Frequently Used IMLPlus Classes and Methods

DataObject Class

Create a Data Object or Write Data

Method	Source/Destination of Data
Create	SAS/IML matrix
CreateFromFile	SAS data set on PC
CreateFromR	R data frame
CreateFromServerDataSet	SAS data set in libref
WriteToServerDataSet	SAS data set in libref
WriteVarsToServerDataSet	SAS data set in libref

Data Query and Retrieval

Set Method	Description
AddVar/AddVars	Add new variable
GetNumObs	Get number of observations
GetNumVar	Get number of variables
GetVarData	Get data values
GetVarNames	Get names of variables

Variable Metadata Properties

Set Method	Description
SetNominal	Set variable to be nominal
SetVarFormat	Set SAS format
SetVarInformat	Set SAS informat
SetVarLabel	Set label
SetRoleVar	Set role: ROLE_LABEL,...

Observation Metadata Properties

Set Method	Description
SetMarkerColor	Set color of marker
SetMarkerShape	Set shape of marker
IncludeInAnalysis	Set analysis indicator variable
IncludeInPlots	Set plot indicator variable
SelectObs	Select observations
GetSelectedObsNumbers	Get indices of selected observations

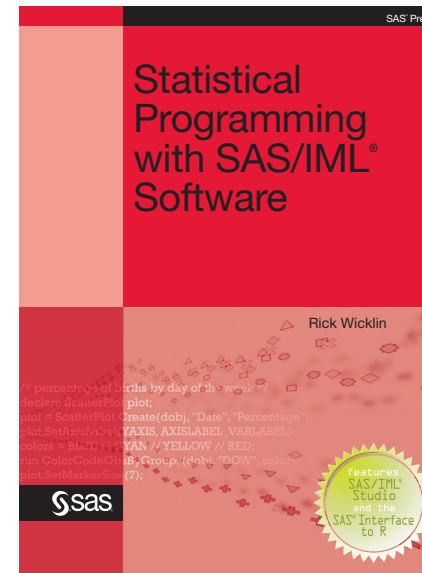
Plot Classes

Plot Attributes

Method	Description
SetAxisLabel	Set label for axis
SetAxisTickAnchor	Set anchor position for ticks
SetAxisTickUnit	Set unit between ticks
SetAxisTicks	Set tick locations
SetAxisViewRange	Set min/max of axis
SetGraphAreaMargins	Set margins in graph area
SetPlotAreaMargins	Set margins in plot area
SetObsLabelVar	Set label variable for obs
SetMarkerSize	Set size of marker
SetTitleText	Set the graph title
ShowObs	Toggle display of non-selected observations

Drawing Methods

Method	Description
DrawSetRegion	Sets drawing region: PLOTFOREGROUND, GRAPHFOREGROUND,...
DrawUseDataCoordinates	Figures are drawn in the data coordinate system
DrawUseNormalizedCoordinates	Specifies a coordinate system for drawing
DrawSetPenAttributes	Set color, style, and width of graphical pen
DrawSetPenColor	Color: RED, 0FFFF0x,...
DrawSetPenStyle	Style: SOLID, DASHED,...
DrawSetPenWidth	Width: 1, 2,...
DrawSetBrushColor	Set color of graphical brush for filling figure interiors
DrawSetTextAlignment	Align text: ALIGN_LEFT, ALIGN_CENTER,...
DrawLine	Draw line or polyline
DrawMarker	Draw marker
DrawPolygon	Draw polygon
DrawRectangle	Draw rectangle
DrawText	Draw text



This is a companion piece to the following book:

Wicklin, Rick. 2010. *Statistical Programming with SAS/IML Software*. Cary, NC: SAS Institute Inc.

Copyright © 2010, SAS Institute Inc., Cary, NC, USA

ISBN 978-1-60764-663-1

ISBN 978-1-60764-770-6 (electronic book)

All rights reserved. Produced in the United States of America.

support.sas.com/publishing/authors/wicklin.html

Author's blog: blogs.sas.com/iml

Questions? Comments? Tips?

Join the SAS/IML® and SAS/IML® Studio

discussion forum at support.sas.com/forums



SAS Institute Inc. World Headquarters +1 919 677 8000

To contact your local SAS office, please visit: www.sas.com/offices

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration. Other brand and product names are trademarks of their respective companies. Copyright © 2011 SAS Institute Inc. All rights reserved. 285710_01.11

Tip Sheet

from

Statistical Programming with SAS/IML® Software

IMLPlus Statements

The IMLPlus language extends the SAS/IML® language. IMLPlus provides support for the following:

- traditional SAS/IML® syntax for computations
- calling SAS® procedures from a SAS/IML® program
- passing parameters from SAS/IML® matrices to SAS procedures
- creating and modifying statistical graphs
- dynamic linking of graphs and data tables
- reading data from diverse sources into a data object
- calling functions in the R language
- features for debugging programs

You can develop, debug, and run IMLPlus programs in SAS/IML® Studio, which is free with a SAS/IML® license. SAS/IML® Studio runs on a Windows PC.

This tip sheet presents the basic syntax of frequently used IMLPlus statements. It also gives you a variety of examples to type in, study, and modify.

IMLPlus Language Tip Sheet

Call SAS Procedures

```
submit;
proc univariate data=Sashelp.Class;
  var Height Weight;
  ods select Moments;
run;
endsubmit;
```

Pass Parameters to Procedures

```
DSName = "Sashelp.Class";
vars = {"Height" "Weight"};
submit DSName vars;
proc univariate data=&DSName;
  var &vars;
  ods select Moments;
run;
endsubmit;
```

Handle Errors in Procedures

```
submit / ok=mOK;
proc univariate data=DataNotFound;
run;
endsubmit;
if ^mOK then do; /* handle error */
  msg = symget("syserrortext");
  /* ... */
end;
```

Call R Functions

```
run ExportDataSetToR("Sashelp.Class","cls");
submit / R;
lm.obj <- lm(Weight~Height, data=cls)
coef <- coef(lm.obj)
endsubmit;
run ImportMatrixFromR(c, "coef");

/* import R data frame */
declare DataObject dobjR =
  DataObject.CreateFromR("f", "faithful");
run ImportDataSetFromR("Work.f","faithful");
```

DataObject Class Methods

```
declare DataObject dobj;
dobj = DataObject.CreateFromServerDataSet(
  "Sashelp.Class");

dobj.GetVarData("Weight", w);
dobj.AddVar("LogWeight", log(w));

dobj.GetVarData("sex", sex);
idx = loc(upcase(sex)="M");
dobj.SetMarkerShape(idx, MARKER_TRIANGLE);
dobj.SetMarkerColor(idx, RED);

/* simulate data */
m = rannor(j(100, 3)); /* 100 x 3 matrix */
declare DataObject d2;
d2 = DataObject.Create("rnd", 'x1': 'x3', m);
```

Plot Class Methods

```
declare ScatterPlot p;
p = ScatterPlot.Create(dobj, "Height",
  "Weight");
p.SetAxisLabel(XAXIS, "Height (lbs)");
p.SetAxisTickUnit(YAXIS, 5);

declare BarChart b;
b = BarChart.Create(dobj, "sex");
b.ShowPercentage();
b.ShowBarLabels();
dobj.SelectObsWhere("sex",WHERE_EQ,"male");
b.SetWindowPosition(50, 50, 50, 50);

declare BoxPlot box;
box = BoxPlot.Create(dobj, "sex", "Weight");
box.SetWindowPosition(0, 50, 50, 50);

/* histogram of simulated data */
declare Histogram h;
h = Histogram.Create(d2, "x1");
h.ShowDensity();
h.ReBin(0, 0.5);
h.DrawUseDataCoordinates();
h.SetWindowPosition(0, 0, 50, 50);
x = do(-3, 3, 0.05);
y = pdf("normal", x);
h.DrawLine(x, y);
h.SetTitleText("Standard Normal", true);
```

IMLPlus Modules

```
Colors = RED//BLUE;
Markers = MARKER_SQUARE//MARKER_TRIANGLE;
Blend = BlendColors(IntToRGB(Colors), 7);
run ColorCodeObs(dobj, "Age", Colors, 7);
run ColorCodeObsByGroups(dobj, "Sex", Colors);
run DrawLegend(p, {"Female" "Male"}, 12,
  Colors, -1, Markers, -1, "IRB");

LabelProps = j(5, 1, .);
run DrawInset(h, {"Mean" "Sigma"}, {0 1},
  LabelProps, "Arial", -1, "IRT");

rgb = IntToRGB(Colors);
hex = RGBToInt(rgb);
```

Query for User Input

```
title = "IMLPlus Language Tip Sheet";
msg = "A message box";
run DoMessageBoxOK(title, msg);

N=5; msg = "Enter Number 1-10";
ok = DoDialogModifyDouble(N, title, msg);
if ok & N>0 & N<11 then
  print "You entered" N;

msg = "Select options for PROC MEANS";
options = {N MEAN MEDIAN Q1 Q3 STD IQR};
ok = DoDialogGetListItems(selections,
  title, msg, options, "Options");
if ok then do;
  opt = options[selections];
  submit opt;
  proc means data=Sashelp.Class &opt;
  run;
  endsubmit;
end;
```

Case Study: Linear Regression

```
declare DataObject dobj;
dobj = DataObject.CreateFromServerDataSet(
  "Sashelp.Class");
declare ScatterPlot plot =
  ScatterPlot.Create(dobj, "Height", "Weight");

/* if data not in libref, create data set */
vars = {"Height" "Weight"};
dobj.WriteVarsToServerDataSet(vars,
  "Work", "In", true);

submit;
proc glm data=In;
  model Weight = Height;
  output out=Out P=Pred R=Resid;
run;
endsubmit;

use Out;
  read all var {"Height" "Pred"} into curve;
close Out;
call sort(curve, 1);

plot.DrawUseDataCoordinates();
plot.DrawSetPenAttributes(BLUE, SOLID, 1);
plot.DrawLine(curve[,1], curve[,2]);
run DrawLegend(plot, "Predicted", 12,
  BLUE, SOLID, _NULL_, -1, "IRB");

VarDS = {"Pred" "Resid"}; /* names in Out */
VarDobj={"Pred" "Resid"}; /* names in dobj */
Lbl = {"Predicted Values" "Residual Values"};
ok = CopyServerDataToDataObject("Work",
  "Out", dobj, VarDS, VarDobj, Lbl, 1);

declare Histogram hist =
  Histogram.Create(dobj, "Resid");
Hist.SetWindowPosition(50, 50, 50, 50);
dobj.SelectObs({9 12 16 17});
```

From
Wicklin, Rick. 2010. *Statistical Programming
with SAS/IML® Software*. Cary, NC: SAS
Institute Inc.