CERN - Summer Student Report

Simone Azeglio[1, 2]

[1] Student in Physics of Complex Systems – University of Turin

[2] Visiting Student Researcher – University of Ottawa

Assigned Task

[TMVA experimental] Implement a variable plotter as TMVA Gui replacement #4211

Basic Ideas

- Make it modular ⟶ No monolithic tool
- Provide tight integration with RDataFrame & RTensor
- Plotting from model output files (.root)
- Create Tutorials

Variable Plotter - RDataFrame

RDataFrame offers a high level interface for analyses of data stored in TTree (columnar dataset), CSV's etc …

It is built in a *modular* way:

- **Build a dataframe** object by specifying your data-set

```
const std::string filename = "http://root.cern.ch/files/tmva_class_example.root";
ROOT::RDataFrame sig1("TreeS", filename);
```

- **Apply** a series of **transformations** to your data

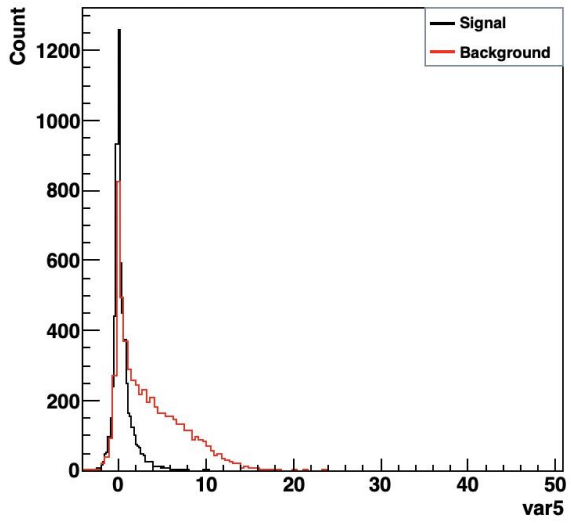
```
auto transform_ = [](ROOT::RDF::RNode df) { return df.Define("var5", "var1 * var2"); };
auto sig2 = transform_(sig1);
```

- **Apply actions to the transformed data** (e.g. fill a histograms)

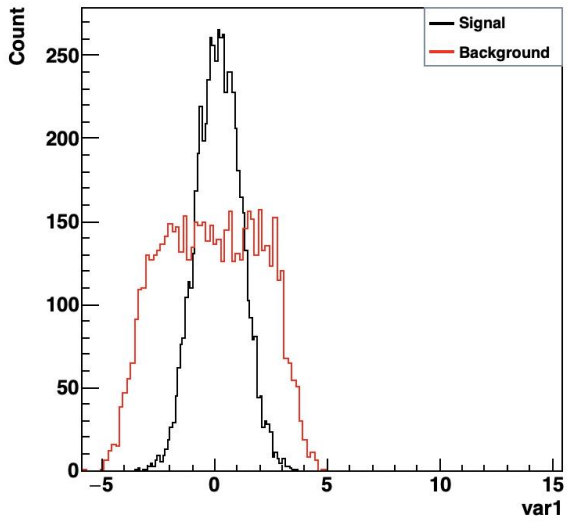
```
const std::vector<std::string> vars = sig2.GetColumnNames();
ROOT::RDF::RResultPtr<TH1D> h = sig2.Histo1D(vars[0]);
```

Variable Plotter - RDataFrame

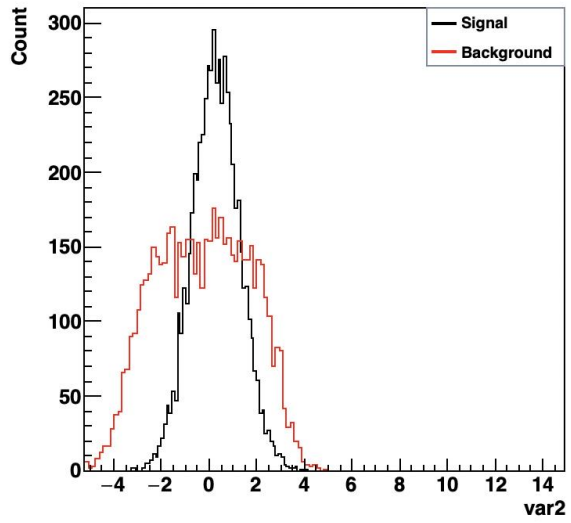
var5



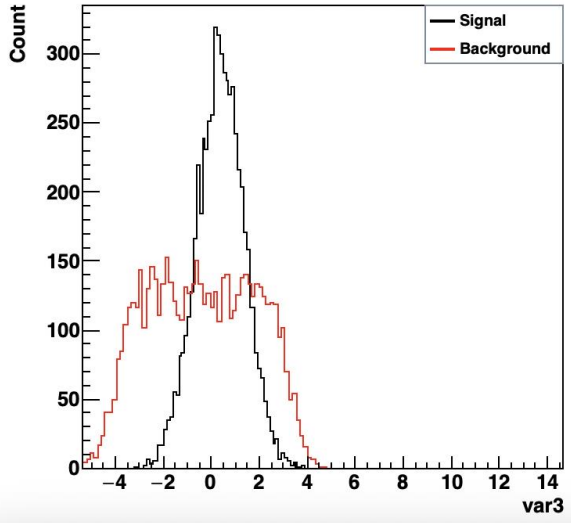
var1



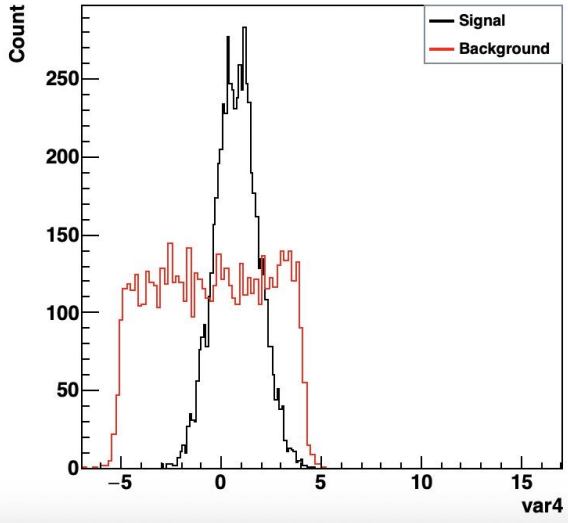
var2



var3



var4



Variable Plotter - RTensor

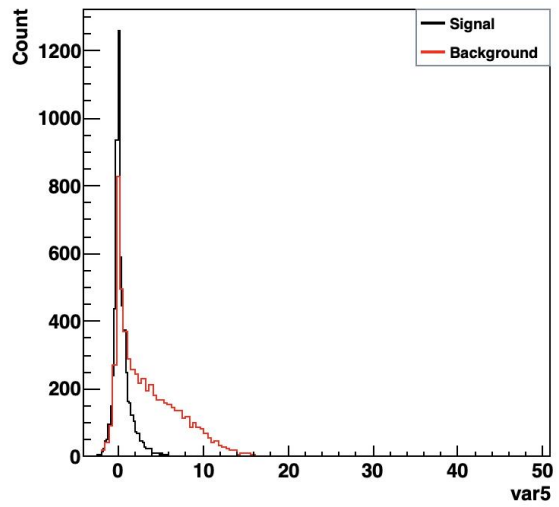
RTensor is a vector-like container with additional shape information.

- We'd like to re-use the plotting methods developed for **RDataFrames**
- A possible way is to convert **RTensors** in **RDataFrames**

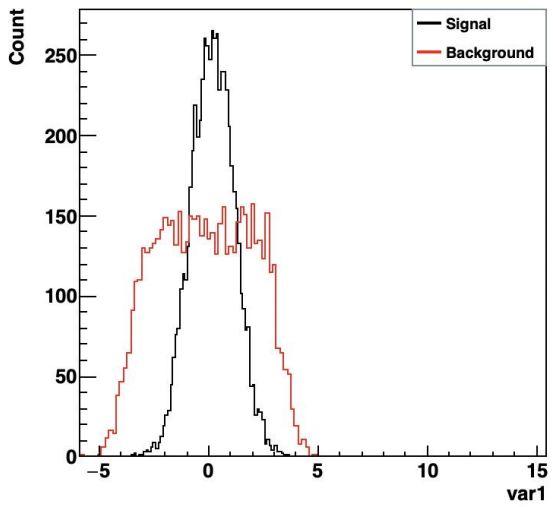
```
ROOT::RDF::RNode TMVA::RVariablePlotter::TensorToNode(const TMVA::Experimental::RTensor<Float_t>& tensor,
                                                      const std::vector<std::string>& variables){

   auto dfSig = ROOT::RDataFrame(tensor.GetShape()[0]).DefineSlotEntry(variables[0],
                                [=] (unsigned, ULong64_t entry) { return tensor(entry, 0); } );

   std::size_t nvar = variables.size();
   if(tensor.GetShape()[1]!= variables.size())
       nvar = std::min(tensor.GetShape()[1], variables.size());

   for (std::size_t j = 1; j < nvar; j++){
       dfSig = dfSig.DefineSlotEntry(variables[j],  [=] (unsigned, ULong64_t entry) { return tensor(entry,j);});
           }
```

Variable Plotter - RTensor

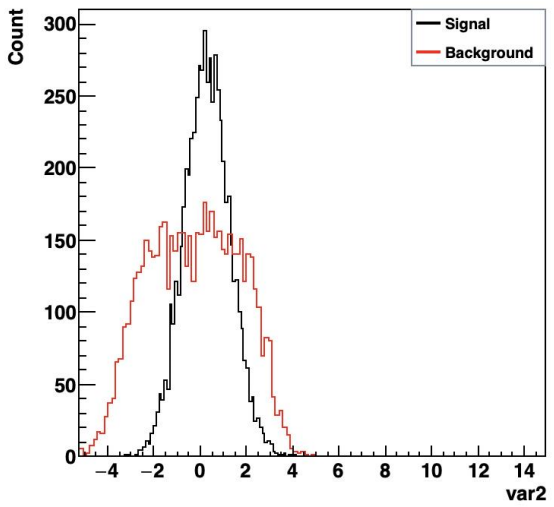
var5



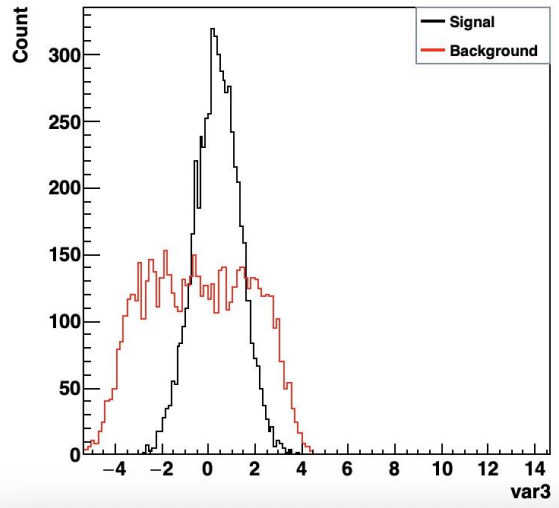
var1



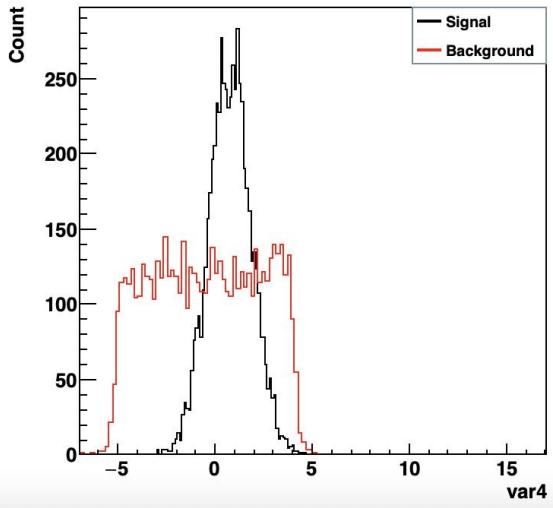
var2



var3



var4

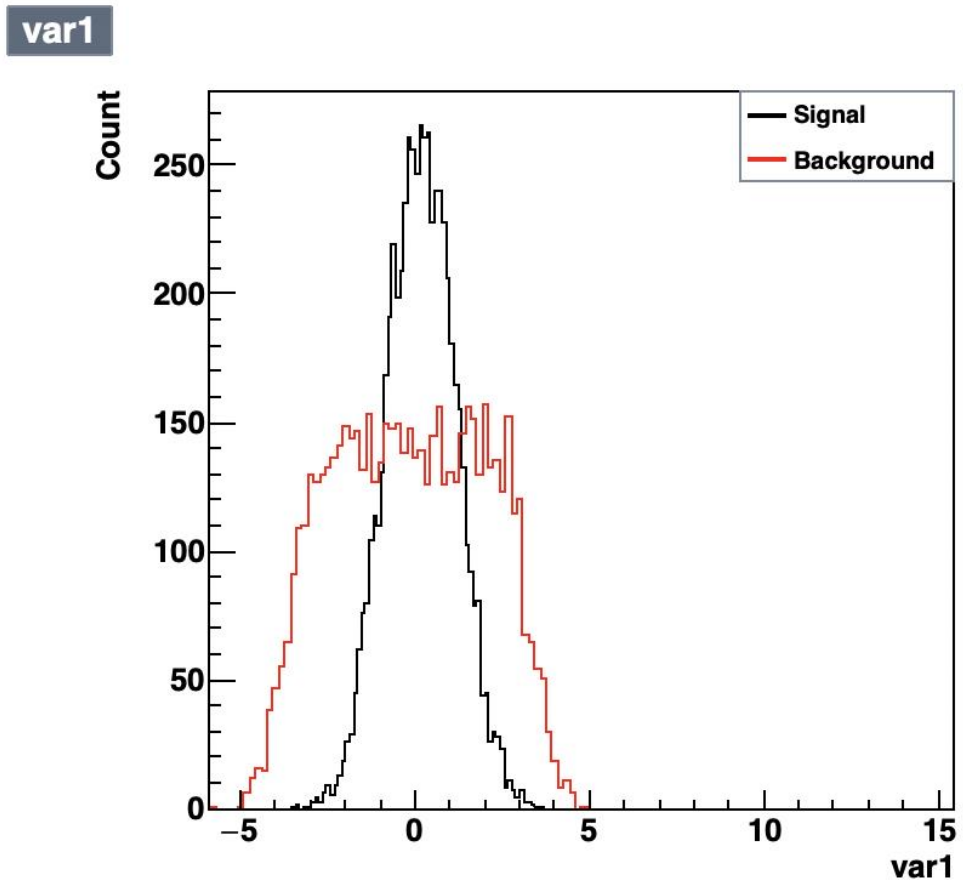


TMVA Plotting Style

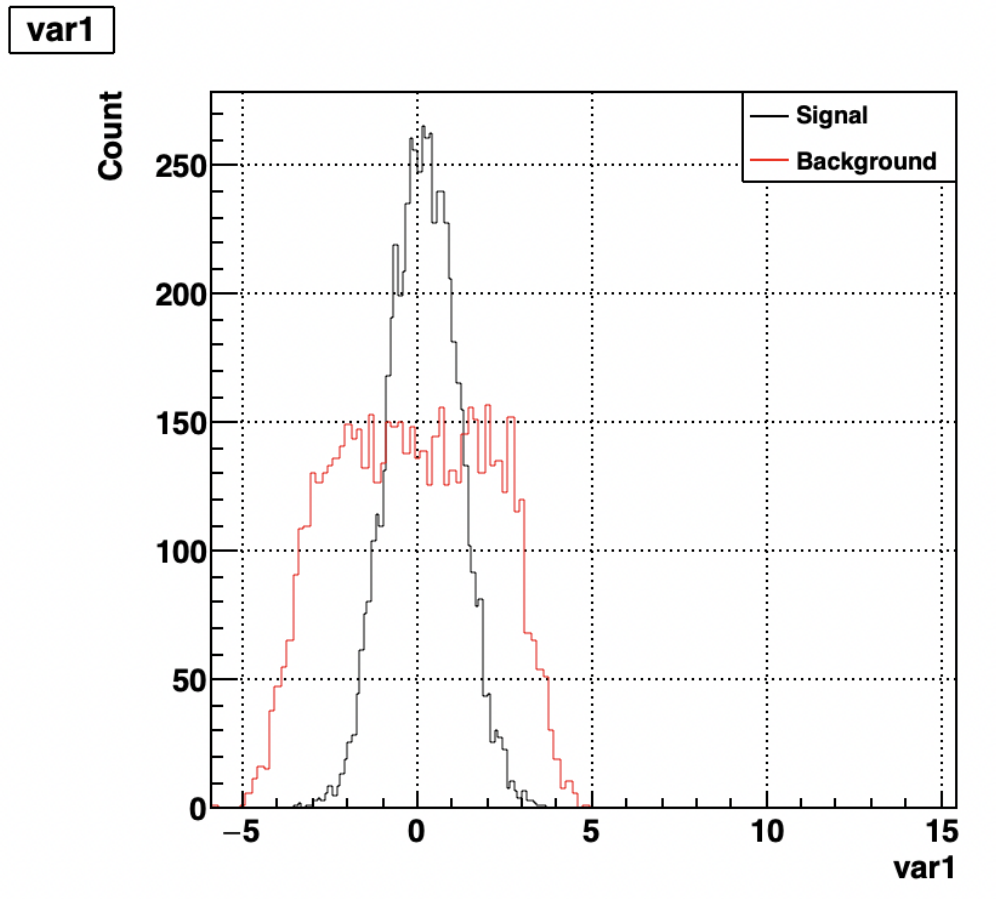
```
void TMVA::RVariablePlotter::InitializeStyle(bool useTMVAStyle){
   // set custom style
   if (!useTMVAStyle) {
      gROOT->SetStyle("Plain");
      gStyle->SetOptStat(0);
      gPad->SetMargin(0.2, 0.9, 0.1, 0.9);
      gPad->SetGrid(1,1);
      return;
   }

   TMVA::TMVAGlob::SetTMVAStyle();
   gPad->SetMargin(0.2, 0.9, 0.1, 0.9);
}
```

TMVA vs Custom Plotting Style

TMVA



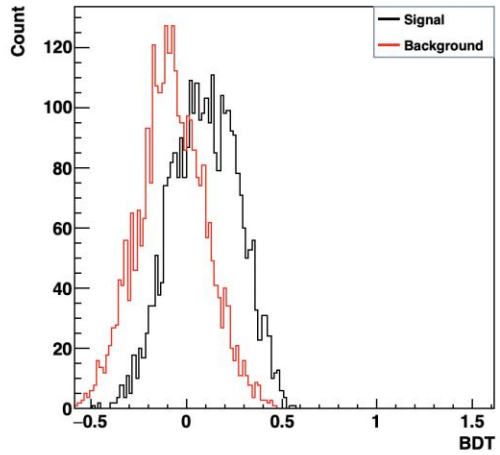
Custom



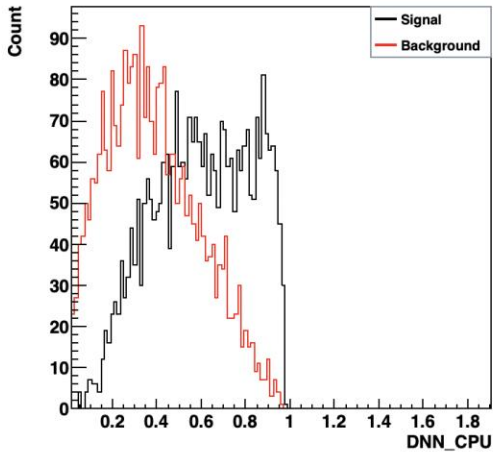
Higgs Classification Outputs

TMVA_Higgs_Classification.C generates a **.root** file which contains predictions etc..

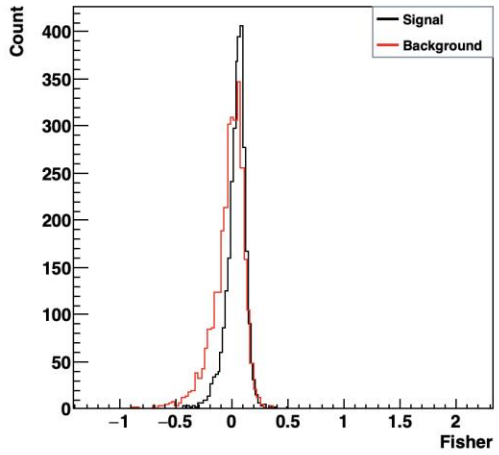
BDT



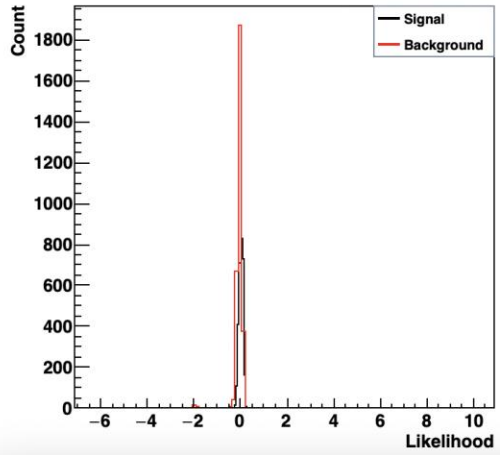
DNN_CPU



Fisher



Likelihood



Examples & Conclusions

The following examples have been developed

- *tmva005_RVariablePlotter.C*
- *tmva006_RVariablePlotter_RTensor.C*
- *tmva007_RVariablePlotter_Higgs_Output.C*
- *tmva008_RVariablePlotter_RTensor_Output.C* (based on *tmva003_RReader.C*)

https://github.com/sazio/root/tree/tmva-variable-plotter