Exercise 8: Trees and Fishers

N. Berger (nberger@physi.uni-heidelberg.de)

3.12.2012

Please send your solutions to nberger@physi.uni-heidelberg.de until 10. 12. 2011, 12:00. Put your answers in an email (subject line *SMIPP:Exercise08*).

1. **Root trees and simple cuts** On the course website, you find a root file, named tree_ex8.root. It contains a made-up analysis *N-tuple*, i.e. a series of numbers for each measured event. For each event there are two measurements x, y and a variable *type*, which is 0 for the signal Monte Carlo, 1 for the background Monte Carlo and 2 for the "data".

   Explore this file in interactive root. You easily access files by creating a TBrowser object on the command line. You can also draw the distributions for each variable by clicking it in the TBrowser. You can also draw distributions using the command line, giving the name of the tree (in this case searchTree) and using the Draw command, e.g.

   ```
   > searchTree->Draw("x");
   ```

   You can plot correlations by specifying two variables,

   ```
   > searchTree->Draw("x:y");
   ```

   You can apply cuts (and weights) in the second argument

   ```
   > searchTree->Draw("x:y","type == 2");
   ```

   and you can specify drawing options in the third argument

   ```
   > searchTree->Draw("x:y","type==2","BOX");
   ```

   Explore the given file.

   If you want to read the file in a script or program, do something like the following:

   ```
   double x,y;
   int type;

   TFile * f = new TFile("tree_ex8.root","READ");

   TTree * tree = (TTree *)(f->Get("searchTree"));

   tree->SetBranchAddress("x",&x);
   tree->SetBranchAddress("y",&y);
   ```

```
tree->SetBranchAddress("type",&type);

for(int event =0; event < tree->GetEntries(); event++){
     tree->GetEntry(event);

     < Do stuff with x,y,type...>
}
```

Try to separate signal and background MC via cuts in either x, y. For an efficiency of 50%, what purity do you obtain?

2. **2D Fisher discriminant**

   Construct the Fisher discriminant from x and y and the MC events. For an efficiency of 50%, what purity do you obtain?

   (Tips: You can use TMatrixD in root for the matrix inversion and other linear algbra stuff. You find a good description of the method at `http://research.cs.tamu.edu/prism/lectures/pr/pr_l10.pdf`)