# Variables correlation study

In data analysis, machine learning and statistics, feature selection, also known as “variable selection” or “variable subset selection”, is the process of selecting a subset of relevant features for use in model construction. The central assumption when using a feature selection technique is that the data contains many redundant or irrelevant features. Redundant variables are those which provide no more information than the currently selected variables, and irrelevant variables provide no useful information in any context.

The Correlation Feature Selection analysis evaluates subsets of variables (features) on the basis of the following hypothesis: **“Good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other”**. Hence, in the context of data analysis, one possible way to determine which variables should be used for selecting a sample of events is to study the correlation between a large set of variables and choose the ones for which the correlation is weaker still providing a good background rejection.

The simplest (not necessary the best!) method to determine if two variables are correlated is to look at the linear correlation coefficient using the Pearson product-moment method.

**Product-moment correlation coefficient.** The correlation r between two variables is:

$$r = \Sigma (xy) / \text{sqrt}[ ( \Sigma x^2 ) * ( \Sigma y^2 ) ]$$

where $\Sigma$ is the summation symbol, $x = x_i - \bar{x}$, $x_i$ is the x value for observation i, $\bar{x}$ is the mean x value, $y = y_i - \bar{y}$, $y_i$ is the y value for observation i, and $\bar{y}$ is the mean y value.

The strongest correlations (r = 1.0 and r = -1.0 ) occur when data points fall exactly on a straight line. The correlation becomes weaker as the data points become more scattered. If the data points fall in a random pattern, the correlation is equal to zero.

# Goal of these exercises

**The aim of these exercises is to study the correlation of two variables in a given data sample and look at their capability in selecting the signal while rejecting the background.**

Basic informations: you will work on a set of events coming out from a PAMELA experiment simulation. The given file

```
/home/mocchiut/pamela/data/pamelasimu.root
```

contains the TTree pamcalotree, storing data with the PamCalo class, header file:

```
/home/mocchiut/pamela/PamCalo/inc/PamCalo.h
```

so library:

```
/home/mocchiut/pamela/PamCalo/lib/Linux/libPamCalo.so .
```

The ROOT file contains about 5.700.000 events: protons, electrons and positrons mixed together in an energy range from 10 to 300 GeV.

Positrons to electrons ratio in this file is about 0.1 . Electrons to protons ratio in this file is about 0.007 .

**In these exercises, consider only the energy range [45-55] GeV. We want to study the correlation between the variables qt1≡qtr/qtot and qt2≡qcyl/qtot. We will determine two selection criteria for qt1 and qt2 and we will look how the qncore≡qcore/ncore variable distribution behaves when applying the selection on signal and background events. Signal-like events are represented by electrons (`pID==1`), background-like events are represented by protons (`pID==0`).**

# Exercise 1

Write an an executable compiled program which reads the input file

`/home/mocchiut/pamela/data/pamelasimu.root`

and gives as output a new ROOT file containing a TTree with four variables (a TBranch for each one):

- pID
- qt1 (≡qtr/qtot)
- qt2 (≡qcyl/qtot)
- qncore (≡qcore/ncore)

Save into the new file events which satisfy the following condition:

1. the event lays in the energy range 45 – 55 GeV (hint: pay attention to the sign of "energy", use "fabs"!).

Hints:

- to compile, remember to add also the compilation flags:

```
-I/home/mocchiut/pamela/PamCalo/inc
-L/home/mocchiut/pamela/PamCalo/lib/Linux/
-lPamCalo
```

- to run, remember to export LD_LIBRARY_PATH:

```
export LD_LIBRARY_PATH=/home/mocchiut/pamela/PamCalo/lib/Linux/:$LD_LIBRARY_PATH
```

- the output file should have a size of about 1.2M, if you have quota problem you can write the output on the linux temporary directory "/tmp/".

# Exercise 2

Write a ROOT-CINT script which reads the output file of exercise 2 (should be similar to this one: `/home/mocchiut/scripts/EM_output_250213.root` use this file if you are not able to complete or run exercise 1) and gives as output on the screen and on the disk (pdf format) a TCanvas divided into four pads (two columns, two rows – hint: `TCanvas::Divide`) which contains from top left clockwise:

1. a scatter plot (`TH2D`) of `qt1` vs `qt2` for the signal-like events (`pID==1`);
2. the event distribution histogram (`TH1D`) for `qt1` for the signal-like events (`pID==1`) fitted with a Gaussian function G1(N1, $\mu 1$,$\sigma 1$);
3. a scatter plot (`TH2D`) of `qt2` vs `qt1` for the signal-like events (`pID==1`);
4. the event distribution histogram (`TH1D`) for `qt2` for the signal-like events (`pID==1`) fitted with a Gaussian function G2(N2, $\mu 2$,$\sigma 2$);

Set the range of the TH1D histograms in order to be the same of the corresponding variable in the TH2D declaration. This way the two histograms in the TCanvas will represent the projection of the scatter plots on the x axis (when looking at pad columns) and the projection of the scatter plots on the y axis (when looking at pad rows).

Answer the following questions:

1. What is the value of the correlation coefficient between qt1 and qt2 for signal-like events? (Hints: print on the STDOUT the correlation factor of the two variables, look at method: `TH2:: GetCorrelationFactor` ).
2. Are the two variables correlated?

# Exercise 3

Update the script of exercise 2 in order to draw a new TCanvas divided into six pads (two columns, three rows – hint: `TCanvas::Divide`) which contains the `qncore` distributions (`TH1D` same binning for all histograms, range [0,65] should be fine) for signal-like events (`pID==1`) on the right column and background-like events (`pID==0`) on the left column, where:

- first row contains all the events; right panel signal, left panel background;
- second row contains the events which pass the one-sigma selection on qt1, where sigma and mean are obtained from exercise 2 (hints:

  `if (qt1` > $\mu 1-\sigma 1$ && `qt1 <` $\mu 1+\sigma 1$) ... , if unable to complete exercise 2 use $\mu 1=0.920$ and $\sigma 1=0.0125$); right panel signal, left panel background;
- third row contains the events which pass **both** the one-sigma selection on qt1 (see above) and the one-sigma selection on qt2, where again sigma and mean are obtained from exercise 2 (hints: `if (qt1` > $\mu 1-\sigma 1$ && `qt1 <` $\mu 1+\sigma 1$ && `qt2` > $\mu 2-\sigma 2$ && `qt2 <` $\mu 2+\sigma 2$ ) ... , if unable to complete exercise 2 use $\mu 2=0.820$ and $\sigma 2=0.0151$); the right panel signal, left panel background.

Get the number of entries (hint: `TH1::GetEntries`) for each histogram and answer the following questions:

1. What is the ratio between the number of entries of the second row over the first row for signal-like events? Is it the expected number?
2. What is the ratio between the number of entries of the third row over the second row for signal-like events? What would you expect in a similar case for uncorrelated variables?
3. What happens if you do the same of points 1 and 2 for background-like events? Can you guess the meaning of the results?

# Preparing the output

- create a directory named with the following format:
  YourInitials_C++2012
  (for example in my case it would be: EM_C++2012)
  put inside this directory ALL the files you want me to correct and look at.
- ALL files names format (but Makefile, if any) must be like:
  YourInitials_something.extension
  (for example in my case I would create files: EM_main.cpp,
  EM_myscript.C, EM_OutputHistogram1.pdf, etc. etc. )
- create a README text file (named like EM_README.txt), inside the file write:
  - **your name and surname**
  - a list of the files you are submitting
  - **in details** how to compile and run the programs
  - any other comment and answer to question(s) rised in the exercise description
- create a compressed tarfile containing the directory:

```
bash> ls
EM_C++2012
bash> tar zcf EM_C++2012.tar.gz EM_C++2012/
```

- copy the tarzipped file on the USB key I will circulate

# Timing and rules

- You have four hours time to do your work.
- You can search the web, look at manuals, look at any note you wrote during the course, etc.
- We will discuss what you have written at the oral examination on 2013/02/28, until that (if needed) you can change and improve your programs. In that case prepare an electronic version we can look at during the oral examination, we will compare it to the one handed in today and we will discuss any change and/or correction.