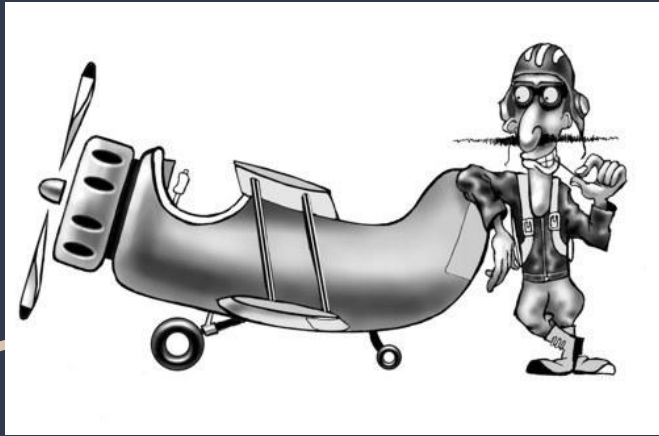
TMVA и классификация данных

(На примере написанной программы)

Задача классификации

Задача классификации в общем виде: Разделить данные по категориям. Например, некоторый алгоритм (примененный в данной задаче), определяет, какие пилоты погибнут при следующем вылете, а какие нет, исходя из их характеристик, таких, как рост, вес, цвет глаз, пол, итд. (Выделена характеристика, которая влияет на результат жизни пилотов).



Не погибнет	Погибнет
Красные глаза	**Синие** глаза
Рост 180 см	Рост 185 см
Мужчина	Мужчина
Двое детей	Двое детей

Используемые инструменты

Задача классификации пилотов по признаку их выживаемости решалась в среде ROOT при помощи TMVA. Итак:

ROOT

ROOT - программная среда, используемая для анализа данных, разработанная CERN(Conseil européen pour la recherche nucléaire). Включает в себя, такие объекты, как, например:

- Графики
- Математический анализ
- Математические функции
- Методы работы с изображениями

TMVA

TMVA - The toolkit for multivariate analysis. Root - интегрированная среда для классификации и анализа данных.

Сигнал/Бэкграунд

Понятия сигнальных и бэкграундных событий:

- Сигнал - события, которые нужно выделить среди остальных событий, те которые имеют ценность для поставленной задачи.
- Бэкргграунд - фоновые события, которые не имеют ценности для поставленной задачи.

Постановка задачи:

Проанализировать данные, содержащие характеристики пилотов, такие, как рост, цвет глаз, вес. На основе этих данных сделать вывод о том, какие пилоты погибнут, а какие останутся в живых. Между данными и вероятностью смерти пилотов есть явное соответствие.

Программа:

Написанная программа состоит из 3-х файлов:

- Neuro.hpp - файл-заголовок.
- Neuro.cpp - файл-описание.
- Neuro-pilots.cpp - Главная программа.

Программа состоит из структуры, в которую занесены переменные - характеристики “пилотов”, такие как рост, вес итд, функций, создающих рандомную генерацию этих характеристик и функции, записывающей сгенерированные характеристики в структуру данных “дерево”.

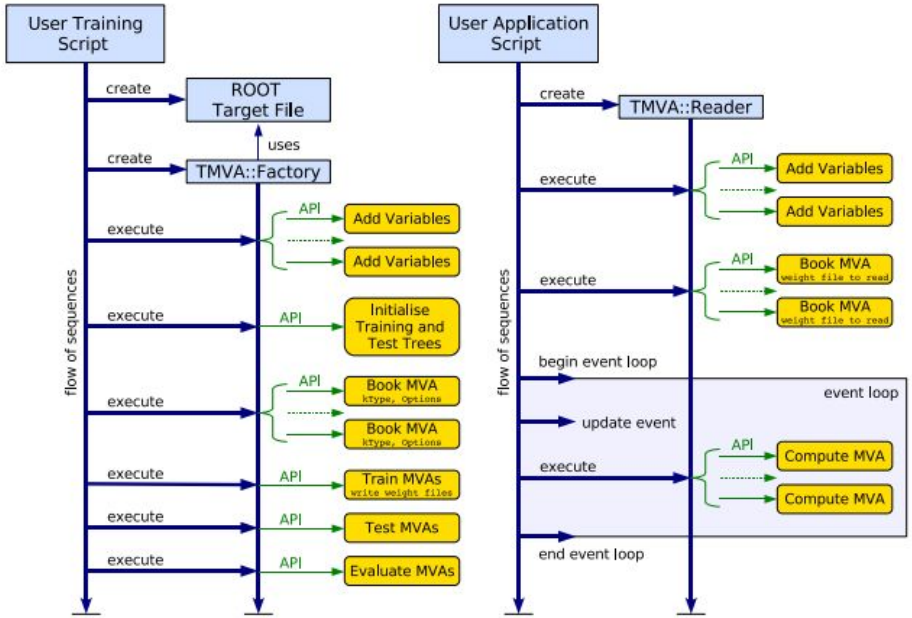
Работа TMVAClassification.C

Данный скрипт считывает из “дерева” переменные - данные, записанные в дерево, работает с этими переменными различными классификаторами, и предоставляет информацию, такую, как: Матрицы корреляций переменных, зависимости между выходными переменными, график эффективности работы каждого классификатора итд. Так же, формирует директорию, в которую записываются “веса” для каждой переменной - значения определенной функции.

Работа TMVAClassification Application.C

Скрипт получает набор данных в виде дерева из файла .root и применяет к каждому событию веса, посчитанные скриптом TMVAClassification.C, после чего записывает гистограмму (график с количеством событий по оси Y) для каждого примененного метода, в которой указывается распределение весов для каждого события для этих данных. Этот скрипт является проверочным, а не тренировочным.

Схема работы обоих скриптов:



Важно, что оба скрипта используются для решения одной и той же задачи, но представляют разные этапы решения.

Результат работы:

Сравнивая графики, полученные скриптом TMVAClassificationApplication.C и графики для тренировочного набора данных, показывающие разделения сигнальных и бэкграундных событий, можно сказать, какой диапазон значений классификатора соответствует сигнальным событиям в тестовом наборе данных.