ГОСУДАРСТВЕННЫЙ НАУЧНЫЙ ЦЕНТР РОССИЙСКОЙ ФЕДЕРАЦИИ

ИНСТИТУТ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ

ИФВЭ 2011-23
ОЭФ, ОМВТ

И.В. Ажиненко, С.И. Битюков, Н.В. Красников[1],
В.В. Смирнова

Программное обеспечение статистической обработки информационных потоков в задачах физики высоких энергий

Направлено в *ИИИФ*

[1]Институт ядерных исследований РАН

Протвино 2011

УДК 519.223.2 М-24

Аннотация

Ажиненко И.В., Битюков С.И., Красников Н.В., Смирнова В.В. Программное обеспечение статистической обработки информационных потоков в задачах физики высоких энергий: Препринт ИФВЭ 2011-23. – Протвино, 2011. – 25 с., 1 рис., библиогр.: 52.

В работе дан обзор основных направлений развития программного обеспечения статистической обработки информационных потоков в задачах физики высоких энергий.

Abstract

Azhinenko I.V., Bityukov S.I., Krasnikov N.V., Smirnova V.V. The Development of Statistical Software for Usage in High Energy Physics Tasks: IHEP Preprint 2011-23. – Protvino, 2011. – p. 25, fig. 1, refs.: 52.

The paper reviews main ways of the development of statistical software for usage in high energy physics tasks.

Введение

Базовым инструментарием для обработки информационных потоков в задачах физики высоких энергий является система программ ROOT [1]. ROOT – пакет объектно-ориентированных программ и библиотек, разработанный в ЦЕРН. Проект базируется на свободном программном обеспечении. Наряду со специальными средствами программирования и стандартных математических вычислений, ROOT обеспечивает пользователя средствами для построения и анализа гистограмм и графиков функций, средствами фитирования и подбора теоретических и экспериментальных зависимостей, инструментарием для проведения статистического (в том числе многофакторного) анализа данных [2].

Большинство программных наработок, по возможности, либо встраиваются в ROOT, либо являются надстройкой над ROOT, то есть базируются на ROOTовских библиотеках. Проводятся работы по совмещению и/или по созданию интерфейса между пакетом ROOT и астрофизическим языком и оболочкой для статистических расчетов и построения графиков R [3].

Обзор основных пакетов

Надстройкой над ROOT является пакет программ RooFit [4]. RooFit – это инструментарий для моделирования ожидаемых распределений событий в физическом анализе. Пакет был изначально ориентирован на эксперимент BaBar. Пакет удобен для

быстрого Монте Карло розыгрыша событий и статистической обработки полученных распределений. Со временем он стал универсальным и базовым для пакетов, расширяющих его возможности, например RooStats [5].

Байесовская парадигма реализована в пакете программ BAT – The Bayesian Analysis Toolkit [6]. Анализ базируется на теореме Байеса и использует Монте Карло моделирование Марковских цепей. Это позволяет строить апостериорное распределение вероятностей, производить оценивание параметров, строить доверительные интервалы, осуществлять перенос неопределенностей.

Много внимания Статистические группы экспериментов уделяют статистическим пакетам многомерного анализа. Было проведено несколько мини-совещаний по данной тематике, например, Miniworkshop on Statistical Tools (2008) DESY [7], CMS tutorials on Multivariate Analysis Methods (2007) CERN [8]. Здесь можно выделить пакет TMVA – Toolkit for MultiVariate Data Analysis [9]. Данный инструментарий также встроен в среду ROOT и ориентирован на использование многомерных классификационных алгоритмов для решения широкого спектра задач. Для обработки информационных потоков в задачах физики высоких энергий и астрофизики также разрабатывается пакет StatPatternRecognition (SPR) [10]. Оба пакета имеют как перекрывающиеся, так и дополняющие друг друга возможности. Интересной новой разработкой является система поддержки принятия решений при выборе переменных в многомерном анализе и уменьшения размерности задачи PARADIGM [11].

Проект RooStats

Для анализа экспериментальных данных коллабораций CMS и ATLAS разрабатывается проект RooStats [5], основанный на комплексе программ ROOT [1]. Главные цели проекта:

- предоставить пользователю компьютерные программы с наиболее распространенными статистическими методами, которые применяются при анализе данных в физике высоких энергий;
- стандартизовать используемые методы для легкого сравнения результатов полученных разными группами и разными экспериментами.

RooStats использует три наиболее распространенных подхода в статистике:

1) частотный подход;
2) метод максимального правдоподобия;
3) Байесовский подход.

Заметим, что программа *RooStats* постоянно развивается, совершенствуется и, на сегодняшний день, она содержит программы, позволяющие решать следующие задачи:

1. Точная оценка для оценки наилучшего в некотором смысле (например, оценки с минимальной дисперсией или наиболее вероятные значения) значения параметра.
2. Определение доверительного интервала: областей параметров функции распределения, не противоречащих наблюдаемым данным.
3. Проверка гипотез: оценка значения вероятности p для одной или нескольких гипотез (значимость).
4. Оценка качества фита - количественное определение насколько хорошо модель описывает данные

Программа *RooStats* написана на языке *C++* и содержит следующие классы, позволяющие решать эти задачи.

— *ProfileLikelihoodCalculator* вычисляет значимость сигнала и определяет наилучшее значение сигнала на основе метода максимального правдоподобия. Возможность учета систематических эффектов также включена в калькулятор.

— *ProfileLikelihood* возможно использовать для оценки интервалов доверия. Возможно вычисление верхнего и нижнего пределов, а также центрального доверительного интервала.

— $BayesianCalculator$ позволяет решать задачи на основе метода Байеса. Причем Байесовское интегрирование может производиться численно, аналитически и методом Монте Карло с помощью Марковских цепей (MCMCCalculator - Monte Carlo Markov Chain Calculator). Здесь, конечно, очень важно – выбор функции приора $\pi(\lambda)$, а также выбор интервала – центрального интервала, интервала минимальной длины или одностороннего интервала.

— *HybridCalculator* вычисляет частотную вероятность событий. Учет систематики проводится с помощью метода Кузинса-Хайлэнда. В частности вычисляет □□-значения (p-value). Вычисления производятся с помощью Монте Карло розыгрыша псевдоэкспериментов. Также возможно осуществить построение Неймана (NeymanConstruction) для определения интервалов доверия частотным способом. Предоставляется возможность использовать несколько правил конструирования интервалов (интервал минимальной длины, центральный интервал, метод Фельдмана-Кузинса). Также при определении значения верхних пределов на сигнал можно использовать чисто P_{sl} частотный подход и $CL_s = \frac{P_{sl}}{1-P_B}$ – модифицированнный частотный подход.

— *HypoTestInverter* преобразует результат по проверке гипотез (HybridCalculator) в доверительный интервал (или в предел доверия) для параметра.

— *HistFactory* обеспечивает использование статистического инструментария пакета программ RooStats без необходимости использовать язык моделирования данных для пакета RooFit.

— *BATCalculator*. Сам пакет Байесовских вычислений с помощью Монте Карло цепей Маркова (BAT) является внешним к RooStats, но как класс в пакете программ RooStats полезен.

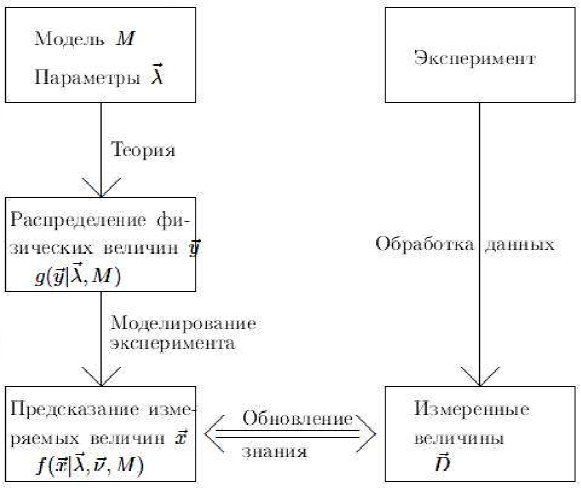
Проект BAT

BAT (the Bayesian Analysis Toolkit) [12] – инструментарий для статистического анализа также как и RooStats возник недавно. Статистический анализ данных в пакете программ BAT основывается на теореме Байеса и реализуется с помощью метода Монте Карло с использованием цепей Маркова [13]. Это позволяет строить апостериорные распределения параметров и, соответственно, проводить оценку параметров, устанавливать доверительные пределы и интервалы, а также осуществлять непосредственно перенос неопределенностей на уровне апостериорных распределений.

Одна из главных целей анализа данных – сравнить модельные предсказания с экспериментальными данными и либо сделать заключение о корректности модели

по отношению к данным, либо построить области доверия с той или иной точностью для параметров заданной модели.

На следующем рисунке представлена основная идея, заложенная в пакете программ BAT.



Модель может меняться от модели, описывающую природу явлений, до простой параметризации данных, используемой в предсказании результатов или в представлении результатов исследований. Теория или модель должны быть обеспечены прямыми распределениями вероятности (вероятностей в дискретном случае), то есть относительными частотами возможных результатов эксперимента, которые можно получить при многократном повторении эксперимента в идентичных условиях[1]. Это возможно, так как модель это математическая конструкция, которая позволяет вычислять (или моделировать) возможные результаты ее использования. Однако предсказания модели обычно нельзя использовать напрямую для сравнения с экспериментальными данными. Нужно или модифицировать предсказание, чтобы учесть особенности

[1] Термин "метод Монте Карло" относится к моделированию процессов с использованием генератора случайных чисел. Термин Монте Карло (Монте Карло – это город, известный своим казино) произошел из того факта, что "число шансов" (в методах моделирования Монте Карло) было использовано с целью нахождения интегралов от сложных уравнений при разработке первых ядерных бомб (интегралы квантовой механики). С помощью формирования больших выборок случайных чисел, например, из нескольких распределений, интегралы этих (сложных) распределений могут быть аппроксимированы из (сгенерированных) данных. Уравнения со сложно решаемыми интегралами часто используются в Байесовском анализе.

в проведении эксперимента, или учитывать эти особенности при представлении результата эксперимента. Очевидно, что необходимо точное описание таких особенностей эксперимента, чтобы сделать надежные выводы из экспериментальных данных.

Функция $g(\vec{y}|\vec{\lambda},M)$ описывает относительную частоту возможного результата $\vec{y}$ в рамках модели M при условии наличия параметров $\vec{\lambda}$. При этом выполнены условия

$$g(\vec{y}|\vec{\lambda},M) \geq 0, \quad (1)$$

где для дискретной величины

$$\sum_i g(\vec{y_i}|\vec{\lambda},M) = 1,$$

а для непрерывной величины

$$\int g(\vec{y}|\vec{\lambda},M)\, d\vec{y}=1. \quad (2)$$

Остановимся на непрерывном случае. Моделирование эксперимента добавляет в модель дополнительные предположения и параметры. Назовем такие параметры вспомогательными (nuisance parameters), хотя часто они определяют область допустимых значений основных параметров $\vec{\lambda}$ модели, и обозначим их $\vec{\nu}$.

Рассмотрим, например, радиоактивный распад нестабильных ядер. В качестве модели возьмем экспоненциальное распределение времени распада, то есть $P(t|\tau) = \frac{1}{\tau}e^{-\frac{t}{\tau}}$, с параметром времени жизни τ. Пусть $\vec{t} = (t_1, t_2, \cdots, t_n)$ – вектор n наблюдений. Одна из возможностей для определения $g(\vec{t}|\tau)$ – это перемножение плотностей вероятности для каждого одиночного наблюдения (события) времени жизни t_i □

$$g(\vec{t}|\tau) = \prod_{i=1}^{n} \frac{1}{\tau} e^{-t_i/\tau}.$$

Другая возможность определения $g(\vec{t}|\tau)$ – подсчет событий (количество распадов) в различные промежутки времени и сравнение их числа с ожидаемым числом распадов, полученным из распределения Пуассона

$$g(\vec{t}|\tau) = \prod_{j=1}^{n} \frac{\eta_j^{k_j}}{k_j!} e^{-\eta_j},$$

где k_j – число событий наблюденных в интервал времени $\Delta\, t_j$, а η_j – ожидаемое число событий для этого интервала времени

$$\eta_j = \int_{\Delta t_j} \frac{N}{\tau} e^{-\frac{t}{\tau}} dt.$$

N – полное число наблюденных событий. Можно представить, что, при измерениях исследуемого образца, наш прибор не способен разделить два близко расположенных по времени распада, то есть имеет некоторое мертвое время. Также, следует предположить, что прибор имеет конечное временное разрешение. Часто полагается, что точность измерения времени распада распределена по нормальному закону. Параметры, описывающие временное разрешение и мертвое время прибора, обычно не являются параметрами, определяющими модель, и могут рассматриваться как вспомогательные. Тем не менее их можно учесть с помощью моделирования методом Монте Карло и получить для измеренной выборки $\vec{t}^{meas} = (t_1^{meas}, t_2^{meas}, \cdots t_n^{meas})$ плотность распределения величины $f(\vec{t}^{meas}|\tau, \sigma_t)$.

Запишем непрерывную вероятность модели M как P(M) со следующим свойством

$$0 \leq P(M) \leq 1. \quad (3)$$

Пусть вспомогательные параметры, описывающие условия эксперимента, не коррелируют с основными параметрами физической модели, то есть

$$P(\vec{\lambda}, \vec{\nu}|M) = P(\vec{\lambda}|M)P(\vec{\nu}). \quad (4)$$

Пусть выполнены следующие условия:

$$P(\vec{\lambda}|M) \geq 0, \quad (5)$$

$$\int P(\vec{\lambda}|M) d\vec{\lambda} = 1, \quad (6)$$

$$P(\vec{\nu}) \geq 0, \quad (7)$$

$$\int P(\vec{\nu}) d\vec{\nu} = 1. \quad (8)$$

В Байесовском подходе P(M) и $P(\vec{\lambda}|M)$ рассматриваются как плотности вероятности, хотя они не являются распределениями частот[2] и их следует описывать как распределение степени веры (degrees-of-belief) [16]. Это распределение степени веры обновляется при сравнении очередных данных эксперимента и предсказания модели, поправленного на результаты предыдущих измерений. Параметр и его модельное

[2] В частотном подходе они рассматриваются как доверительные распределения [15].

распределение степени веры являются предметом изучения и содержат наше знание о природе исследуемого явления. Целью измерений в этом смысле является модифицирование распределения степени веры. При этом утверждение $P(M)$=1 означает то, что модель M верна, а утверждение $P(M)$=0 означает то, что модель M неверна.

Этот построение позволяет сформулировать правило обучения в данном подходе как

$$P_{i+1}(\vec{\lambda},\vec{\nu},M|\vec{D}) \propto f(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M)P_i(\vec{\lambda},\vec{\nu},M), \tag{9}$$

где индекс при P обозначает текущее состояния знания.

Из условия нормализации

$$\sum_M \int P(\vec{\lambda},\vec{\nu},M)\, d\vec{\lambda}d\vec{\nu} = \sum_M P(M)\left[\int P\left(\vec{\lambda}\middle|M\right)d\vec{\lambda}\int P(\vec{\nu})d\vec{\nu}\right] = 1 \tag{10}$$

имеем

$$P_{i+1} = \left(\vec{\lambda},\vec{\nu},M\middle|\vec{D}\right) = \frac{f(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M)P_i(\vec{\lambda},\vec{\nu},M)}{\sum_M \int f(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M)P_i(\vec{\lambda},\vec{\nu},M)d\vec{\lambda}d\vec{\nu}}. \tag{11}$$

Обычно P_i называется приором и обозначается как P_0. Он содержит всю информацию о модели и значениях параметров, которой мы владеем до проведения эксперимента. Апостериорная вероятность P_{i+1} обычно обозначается просто P. Она описывает состояние наших знаний после анализа данных эксперимента.

Для заданной модели M f является функцией основных параметров модели, вспомогательных параметров измерения и возможного результата измерений. Если f зависит только от параметров $(\vec{\lambda},\vec{\nu})$ для конкретной выборки $\vec{x}=\vec{D}$, то она рассматривается как функция правдоподобия. В данной формулировке f это относительная частота появления выборки $\vec{x}=\vec{D}$ при моделировании. Если f нормализованы, то можно записать

$$P(\vec{D}|\vec{\lambda},\vec{\nu},M) = f(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M). \tag{12}$$

Знаменатель в Ур. (11) это вероятность получить данные $\vec{D}$, в условиях модели M и параметризации эксперимента. Он описывает все возможные результаты экспери-

мента и его можно записать в виде $P(\vec{D})$. В этих обозначениях формула (11) принимает классический вид (например, [14])

$$P(\vec{\lambda},\vec{\nu},M|\vec{D}) = \frac{P(\vec{D}|\vec{\lambda},\vec{\nu},M)P(\vec{\lambda},\vec{\nu},M)}{P(\vec{D})}. \quad (13)$$

В рамках данной модели оценка параметров осуществляется с помощью соотношения

$$P(\vec{\lambda},\vec{\nu}|\vec{D},M) = \frac{P(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M)P_0(\vec{\lambda},\vec{\nu}|M)}{\int P(\vec{x}=\vec{D}|\vec{\lambda},\vec{\nu},M)P_0(\vec{\lambda},\vec{\nu}|M)}. \quad (14)$$

Результатом оценки является нормализованная плотность вероятности параметров, включая все корреляции. Эта плотность позволяет строить доверительные интервалы для параметров, находить доверительные пределы, проверять согласованность экспериментальных данных и моделей и многое другое.

Апостериорная плотность вероятности Ур. (9) находится методом Монте-Карло, использующим цепи Маркова (например, [17]).

Пакет реализован на основе языка *C++*, имеет интерфейс с такими пакетами, как ROOT, RooStats, Minuit. Он позволяет использовать определенные пользователем функции и алгоритмы.

Многофакторные (многовариантные) методы в физике высоких энергий

Многофакторный метод это любой статистический инструментарий, в котором заложен статистический анализ многомерной переменной. Основными задачами, решаемыми в таком анализе, являются задачи классификации объектов, аппроксимации функций, оценки плотности распределений, сжатия данных, отбора переменных, оптимизации, сравнения моделей и проверки гипотез. Рассмотрим принципы и подходы к построению данного инструментария [18] и возможности, предоставляемые существующим программным обеспечением.

Информационные потоки экспериментальных данных с крупных установок многомерны. Например, типичное событие рождения одиночного топ-кварка в протон-

антипротоных соударениях в конечном состоянии включает электрон или мюон, от двух до четырех струй адронов и потерянную поперечную энергию. В работе [19] при обнаружении рождения одиночного топ-кварка было использовано около 60 переменных и применялось четыре метода многофакторного анализа.

Немного теории

Для многих приложений фундаментальную многофакторную задачу можно построить следующим образом: пусть существует набор обучающих данных $T = \{(y_1, x_1), (y_2, x_{2)}), \cdots\}$, где $y = (y_1, y_2 \cdots)$, y_i - действительное (или рациональное) число в интервале [-1,1] или [0,1], а x - входной вектор, компоненты которого $x_i \in R^d$, и существует функция f такая, что $y = f(x)$. Если f непрерывна, то задачу можно назвать регрессией. Если f дискретна, то говорят о задачах классификации или разделения. Величины y_i называют целями (значениями целевой функцией) или выходным вектором, а величины x_i входным вектором (или вектором особенности).

Многофакторные методы можно делить по способу обучения: методы с машинным или с Байесовским обучением.

Машинное обучение

Можно выделить два типа обучения. Обучение по прецедентам, или индуктивное обучение, основано на выявлении закономерностей в эмпирических данных. Дедуктивное обучение предполагает формализацию знаний экспертов и создание базы знаний. Дедуктивное обучение принято относить к области экспертных систем, поэтому термины машинное обучение и обучение по прецедентам будем считать синонимами.

Рассмотрим подход к построению приближения функции $y = f(x)$ как проблему задачи оптимизации, то есть приближение будет результатом минимизации соответствующего функционала. Основные элементы анализа это:

- класс параметрических функций $F = \{f(x, w)\}$, в которых параметры w находятся при минимизации функционала;

- ограничение $C(w)$на класс функций $F = \{f(x,w)\}$;
- функция потерь $L(y,f)$, которая оценивает потери при плохом выборе функции $f(x,w)$ из класса функций $F = \{f(x,w)\}$.

На практике выбранная функция $f(x,w)$ должна быть устойчива к выбору обучающих данных, то есть среднее значение функции потерь по обучающим данным должно быть минимальным (иметь лучшее значение). Это задается через функцию эмпирического риска $R(w)$

$$R(w) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, f_i), \tag{15}$$

где $f_i = f(x_i,w)$, а N – объем выборки обучающих данных. Практической задачей в данном случае является минимизация функции эмпирического риска $R(w)$ при наличии ограничения $C(w)$, то есть функционала

$$E(w) = R(w) + \lambda C(w), \tag{16}$$

где λ – параметр настройки, который определяет серьезность ограничения. При минимизации $E(w)$ будет выбрана функция $f(x,w^*)$, которая в пределе при $N \to \infty$ сходится в выбранной метрике к функции минимизирующей риск через функцию потерь $L(y,f)$. Хорошо известный пример минимизации эмпирической функции риска – это χ^2 фитирование (подгонка) с ограничениями.

Байесовское обучение

В байесовском обучении исходя из полученных данных вычисляется вероятность каждой гипотезы и на основании этого делаются предсказания, то есть предсказания составляются с использованием всех гипотез, взвешенных по их вероятностям, а не с применением только одной "наилучшей" гипотезы. Таким образом, обучение сводится к вероятностному выводу. Это требует наличия

- класса параметрических функций $F = \{f(x,w)\}$, в которых параметры w находятся минимизацией функционала;

- плотности приора $\pi(f)$ по пространству функций, хотя на практике используют более простой путь определения приора по пространству параметров;
- функции правдоподобия $p(y|x,w)$ пропорциональной вероятности того, что величина функции $f(x)$ будет равна *y* при заданном входе *x*.

Преимущество Байесовского подхода в том, что все интерференционные проблемы решаются одним способом [20]. Проблема состоит в том, чтобы определить вероятные значения параметров *w*, и, следовательно, сделать вероятностный выбор функции *f(x,y)*, учитывая обучающие данные *T*. Это делается вычислением апостериорной вероятности *p(w|T)dw*, то есть вероятности того, что выбор *w* совместим с обучающими данными *T*. Стандартный способ произвести это вычисление состоит в использовании теоремы Байеса

$$\begin{aligned} p(w|T) &= \frac{p(T|w)\pi(w)}{p(T)} \\ &= \frac{p(y,x|w)\pi(w)}{p(x,y)} \\ &= \frac{p(y|x,w)p(x|w)\pi(w)}{p(y|x)p(x)} \\ &\sim p(y|x,w)\pi(w), \end{aligned} \tag{17}$$

здесь мы делаем разумное предположение, что *p(x|w)=p(x)* потому, что плотность вероятности входного вектора *x* не зависит от параметров функции, которая будет получена. Функция *p(w|T)dw* определяет значение функции *f(x,w)* в соседнем с точкой *w* отрезке [*w,w+dw*]. Большие значения *p(w|T)* подразумевают, что функция *f(x,w)* лучше согласована с обучающими данными.

В рамках Байесовского подхода [20] данная апостериорная плотность *p(w|T)* позволяет вычислить прогнозирующее распределение

$$p(y|x,T) = \int p(y|x,w)p(w|T)dw, \tag{18}$$

то есть плотность вероятности того, что функция *f* принимает значение *y* при данном значении входного вектора *x*. Часто используют точечную оценку значения *y=f(x)*. Она может быть получена при минимизации функции риска

$$y = arg_y min \int L(z,y)p(z|x,T)dz, \tag{19}$$

где $L(z,y)$, – некоторая функция потерь. Здесь минимизируются квадратичные потери, задаваемые как

$$L(a,b) = (a-b)^2. \tag{20}$$

При таком выборе вида функции потерь оптимальная оценка значения функции $y=f(x)$ определяется как среднее прогнозирующего распределения

$$y = \int zp(z|x,T)dz \tag{21}$$

Сравнение машинного и Байесовского обучения

Хотя рассмотренные подходы различны, они не так различаются, как кажется. Если мы рассмотрим следующие тождества

$$R(w) \sim \ln p(y|x,w)$$
$$= \sum_{i=1}^{N} \ln p(y_i|x_i,w), \tag{22}$$
$$\lambda C(w) \sim \ln \pi(w), \tag{23}$$

то становится ясно, что машинный подход обучения может быть рассмотрен как обеспечение Максимума АПостериорной (МАП) оценки функции $y=f(x)$, то есть при нахождении наилучшей оценки $f(x,w^*)$ функции $y=f(x)$ задача минимизации функции эмпирического риска с ограничениями (16) эквивалентна максимизации апостериорной плотности(17). В то время как в Байесовском подходе процедура присваивает определенную вероятность всем возможным выборам функции $f(x,w)$.

Регрессия и классификация

Регрессия (например, подгонка кривой) и классификация отличаются выбором целей y и функциями потерь L.

В случае регрессии цели это непрерывные функции входных данных, а функцией потерь, обычно, считаются квадратичные потери (20). Этот выбор дает функцию эмпирического риска следующего вида

$$\lambda R(w) = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i, w))^2, \quad (24)$$

или, что эквивалентно, функцию правдоподобия

$$p(y|x, w) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{NR(w)}{2\sigma^2}}. \quad (25)$$

Для классификации лучший выбор для функции правдоподобия (и, следовательно, функции эмпирического риска)

$$p(y|x, w) = \prod_{i=1}^{N} f(x_i, w)^{\omega_i y_i} [1 - f(x_i, w)]^{w_i(1-y_i)}, \quad (26)$$

где $0 < f(x_i, w) < 1$, а значения целевой функции y_i принимают дискретные значения 0 и 1. При классификации $f(x_i, w)$ интерпретируется как вероятность того, что входной вектор x принадлежит классу значений целевой функции y=1. В уравнении (26) вводятся дополнительные веса ω_i для каждой компоненты (или особенности) x_i входного вектора x. В физике частиц в методе Монте Карло модели обычно взвешиваются пособытийно, чтобы откорректировать недостатки моделирования. Соответствующей функцией эмпирического риска является

$$R(w) = \frac{1}{N}\sum_{i=1}^{N} \omega_i [y_i ln f(x_i, w) + (1 - y_i) \ln(1 - f(x_i, w))]. \quad (27)$$

Показано (например, [21]), что оптимальный выбор значения функции *f(x,w)* тот, который аппроксимирует вероятность $p(1|x)$ так, что входной вектор x принадлежит классу, помеченному 1

$$p(1|x) = \frac{p(x|1)p(1)}{p(x|1)p(1)+p(x|0)p(0)}, \quad (28)$$

где $p(1)$ и $p(0)$ – априорные вероятности для классов, помеченных соответственно 1 и 0, а $p(x|1)$ и $p(x|0)$ – соответствующие плотности вероятности для входных векторов. В задачах разделения сигнала и фона класс «сигнал» обычно помечается 1, а класс «фон» – соответственно 0 (иногда даже -1). В данном случае отношение $\frac{p(1)}{p(0)}$ будет априорным отношением сигнал/фон.

Для заданной вероятности $p(1|x)$ можно определить Байесовский классификатор:

Байесовский классификатор: Если $p(1|x) > q$, то x считается принадлежащим классу, помеченному 1, где величина q является порогом, выбранным для анализа[3]. Байесовский классификатор оптимален в том смысле, что при его использовании достигается наименьший процент ложных отбраковок. Например, если цель состоит в том, чтобы различить электроны и файкэлектроны (при ложной идентификации частицы как электрона) и сделать это с наименьшим количеством ошибок, тогда нужно использовать Байесовский классификатор. На практике обычно неизвестны априорные вероятности $p(1)$ и $p(0)$. Часто это именно те величины, что необходимо измерить. Поэтому кажется, что нельзя использовать Байесовский классификатор. Но это не так. Для классификации достаточно аппроксимировать дискриминант

$$D(x) = \frac{p(x|1)}{p(x|1)+p(x|0)} \tag{30}$$

потому, что $D(x)$ и $p(1|x)$ связаны взаимооднозначным соответствием, где $A = \frac{p(1)}{p(0)}$. Классификация с использованием $D(x)$ с заданным порогом эквивалентна классификации с использованием $p(1|x)$ с неизвестным порогом. Функция $p(1|x)$ оптимальна в другом смысле [22]. Если веса в смеси сигнальных и фоновых событий определяются весовой функцией $W(x) = p(1|x)$, то доля сигнала может быть оценена с нулевым смещением и минимальной дисперсией при условии, что зависимость веса от x может быть точно смоделировано и отношение A равно истинному значению. Так как мы не знаем истинного значения A, то, вероятно, можно построить итеративную процедуру сходящуюся к A. Оптимальность Байесовского классификатора зависит от решаемой задачи. Например, если нужно классифицировать объекты с наименьшим количеством ошибок, то он оптимален, если же нужно измерить массу Хиггса с минимальной неопределенностью, то Байесовский классификатор не обязательно оптимален.

[3] Формально, этот выбор определяется минимизацией выбранной функции потерь.

Многофакторные методы на практике

Для важной задачи разделения сигнал/фон существует большое количество многофакторных методов. Приведем неполный список этих методов.

— **Линейные и Нелинейные Дискриминанты** (Linear and Nonlinear Discriminants). Дискриминантный анализ это раздел вычислительной математики, представляющий основное средство решения задач распознавания образов, инструмент статистики, который используется для принятия решения о том, какие переменные разделяют возникающие наборы данных.

— **Метод опорных векторов** (Support Vector Machines). Это набор схожих алгоритмов вида «обучение с учителем», использующихся для задач регрессионного анализа и классификации. Этот метод принадлежит к семейству линейных классификаторов[23].

— **Дискриминант Правдоподобия** (Likelihood Discriminant). Примитивный (наивный) байесовский классификатор – простой вероятностный классификатор, основанный на применении теоремы Байеса со строгими (наивными) предположениями о независимости [24].

— **Ядерная Оценка Плотности** (Kernel Density Estimation). В статистике ядерная оценка плотности это непараметрический способ оценить функцию плотности вероятности случайной переменной. Ядерная оценка плотности – фундаментальная задача сглаживания данных, где выводы о генеральной совокупности основаны на выборочных данных [25].

— **Нейронные сети** (Neural Networks). Искусственные нейронные сети - математические модели, а также их программные или аппаратные реализации, построенные по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы. Впоследствии, после разработки алгоритмов обучения, получаемые модели стали использовать в практических целях:

в задачах прогнозирования, для распознавания образов, в задачах управления и др. [26]. Один из наиболее простых и, тем не менее, широко применяемых типов нейросетей - многослойный персептрон. Многослойными персептронами называют нейронные сети прямого распространения. Входной сигнал в таких сетях распространяется в прямом направлении, от слоя к слою. Многослойный персептрон в общем представлении состоит из следующих элементов:

- множества входных узлов, которые образуют входной слой;
- одного или нескольких скрытых слоев вычислительных нейронов;
- одного выходного слоя нейронов.

Многослойный персептрон представляет собой обобщение однослойного персептрона Розенблатта [27]. Основным элементом нейронной сети является нейрон, который имеет один или несколько входов и один выход, значение которого вычисляется через передаточную функцию (или функцию активации нейрона). Если два нейрона по обе стороны синапса (соединения) активизируются одновременно (то есть синхронно), то прочность этого соединения возрастает. Если два нейрона по обе стороны синапса активизируются асинхронно, то такой синапс ослабляется или вообще отмирает.

Важным свойством любой нейронной сети является способность к обучению. Классический метод обучения перцептрона – это метод коррекции ошибки. Он представляет собой вид обучения с учителем, при котором вес связи не изменяется до тех пор, пока текущая реакция перцептрона остается правильной. При появлении неправильной реакции вес изменяется на единицу, а знак (+/-) определяется противоположным от знака ошибки. Существует метод Розенблатта обучения без учителя, а также градиентные методы (например, метод обратного распространения ошибки [28], [29]).

— **Байесовские нейронные сети** (Bayesian Neural Networks). Если методы построения и обучения нейронных сетей являются Байесовскими, то и сети называются Байесовскими нейронными сетями. Существуют два подхода к Байесовскому обучению нейронных сетей – подход МакКея [30] и Нила [31]. В физике высоких энергий под Байесовскими сетями обычно понима-

ют сети с обучением по второму подходу. Байесовские нейросети имеют ряд черт – качественно отличающих их от нейронных сетей в традиционном подходе. Для них не осуществляется градиентный поиск минимума ошибки, то обучения в его классическом понимании не производится. Тем не менее, из-за использования в методе Монте Карло Марковских цепей, при применении Байесовских нейросетей необходимы значительные вычислительные ресурсы. Результатом обучения нейросети является не одна натренированная, то есть обученная, сеть, а ансамбль, обычно содержащий сотни сетей. Все эти сети имеют одинаково большое число нейронов в скрытом слое.

— **Деревья принятия решений** (Decision Trees). Деревья принятия решений обычно используются для решения задач классификации данных или, иначе говоря, для задачи аппроксимации заданной булевой функции [32].

— **Случайные Леса** (Random Forests}). Алгоритм машинного обучения заключается в использовании ансамбля решающих деревьев. Алгоритм сочетает в себе метод бэггинга (Bootstrap AGGregating – объединения выборок) и метод случайных подпространств. Алгоритм применяется для задач классификации, регрессии и кластеризации [33].

— **Генетические алгоритмы** (Genetic algoritms). Генетический алгоритм – это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, напоминающих биологическую эволюцию [34].

Можно также отметить Случайный Поиск по Сетке (Random Grid Search), который позволяет обойти трудности, возникающие при использовании регулярных сеточных алгоритмов.

Часть из перечисленных методов реализована в пакете программ [9].

TMVA – инструментарий для многофакторного анализа (the Toolkit for Multivariate Analysis)

Многофакторные методы классификации, основанные на методах машиного обучения, играют в настоящее время фундаментальную роль в исследованиях физики высоких энергий. При постоянно возрастающих информационных потоках и усложнении установок, производящих эти данные, становится важным использовать все возможные проявления в экспериментальных данных сигнальных и фоновых событий. Анализ событий становится многомерным с очень большим количеством размерностей, часто коррелированных. TMVA [9] – набор инструментов, которые позволяют реализовать большое разнообразие многофакторных алгоритмов классификации. В частности, в программе задействованы следующие классификаторы:

- оптимизация линейных обрезаний;
- проекционное оценивания методом максимального правдоподобия;
- многомерное оценивание методом максимального правдоподобия;
- линейный и нелинейный дискриминантный анализ;
- искусственные нейронные сети;
- метод опорных векторов;
- усиленные (boosted) деревья принятия решений и деревья с бэггингом (bagging)[4];
- прогнозирующее обучение через ансамбли правил (via rule ensembles).

Пакет также позволяет организовать решение следующих задач:

1) нахождение зависимостей между данными (задачи регрессии);
2) классификация множества классов;
3) автоматической настройки классификаторов и их проверка с помощью взаимотестирования;
4) усиление (бустинг) и бэггинг как собственные характеристики классификатора;
5) составные классификаторы для параллельного, но независимого обучения различных областей фазового пространства;

[4] Метод определения характеристик с помощью бутстреп размножения выборок и определения средних величин.

6) комбинированное преобразование входных данных;
7) параллельная (multi threaded) минимизация и обучение классификатора.

Пакет SPR (StatPatternRecognition)

Пакет SPR[10] обеспечивает пользователя доступом к методам многофакторного анализа. Анализ для выбранного классификатора состоит из обучения и тестирования. На стадии обучения пользователь создает тренируемый классификатор и обучает его за определенное число циклов или поставкой параметров для этого классификатора из работающего модуля в SPR, и ли при использовании соответствующего интерфейса (SprRootAdapter) из интерактивной сессии ROOT [1]. Для классификатора, который требует больше, чем один учебный цикл, можно контролировать поток ошибок классификатора, благодаря специальным тестовым данным. После того как обучение закончено, можно сохранить конфигурацию обученного классификатора в системе. В любое время можно восстановить сохраненную конфигурацию и либо продолжить обучение, либо использовать классификатор для задачи классификации данных.

Дерево принятия решений в пакете программ [35] состоит из двух составляющих - регулярное дерево принятия решений и нисходящее дерево решений, что позволяет ускорять процедуру принятия решения в задачах различной сложности.

Алгоритм охотника за пиками [36] позволяет оптимизировать некоторую меру (например, значимость превышения сигнала над фоном) и находить в многомерном пространстве многогранник с наилучшим потенциалом на открытие.

Усиление (boosting [37]) работает, добавляя много слабых классификаторов последовательно и увеличивая веса неправильно классифицированных событий на каждом шаге. Фокусируясь на событиях, которые являются неправильно классифицированными большую часть времени, усиление, как правило, достигает очень хорошей прогнозирующей степени.

Бэггинг (the bagging) алгоритм [38], который находит среднее по многим слабым классификаторам, построенным как бутстрап копии обучающего набора данных. SPR позволяет использовать бэггинг для произвольной последовательности классификаторов. Усредненные деревья решений используются в исследованиях с машинным обучением и применяются в физическом анализе. Случайный лес [39] как правило,

используется в соединении с бэггингом и представляет собой набор деревьев принятия решений. Каждое дерево построено, используя случайно отобранные входные переменные для каждого выбора решения. Случайное осуществление выборки входных переменных уменьшает корреляцию среди деревьев принятия решений и улучшает полную мощность классификатора. SPR осуществляет прямую связь с обратным переносом в нейронной сети с логистической функцией активации [26]. SPR также позволяет использовать ряд дополнительных инструментов для анализа данных [40].

Среда для принятия решений по выбору и уменьшению числа переменных PARADIGM

В физике высоких энергий выбор и уменьшение числа переменных в многофакторном анализе играет важную роль, поскольку, например, начальный выбор переменных часто приводит к наборам переменных с очень большим количеством составляющих. Иногда большим, чем количество степеней свободы основной модели. Это обуславливает потребность в обоснованном сокращении количества переменных. Примером одного из решений данной проблемы является среда для самосогласованного принятия решений по выбору и уменьшению число переменных PARADIGM [11]. Решение о судьбе переменной в данной системе программ основывается на некоторой мере, называемой глобальной функцией потерь. В системе также определяется количественное значение такого свойства, как важность каждой переменной, позволяющего выделить наиболее существенные переменные в применяемой модели.

Организационная база развития статистических методов и программных средств анализа экспериментальных данных

Достаточно долго существуют Рабочая группа по статистики в Коллаборации BaBar [41] и Комитет по статистики в коллаборации CDF [42]. Основной задачей этих комитетов является выработка рекомендаций и стандартов в использовании Коллаборациями статистических методов обработки данных.

Физиками было проведено несколько специальных Рабочих Совещаний и Конференций с целью унификации методов анализа и представления конечных данных с экспериментов:

Workshop on confidence limits, CERN, January, 2000 [43];

Workshop on confidence limits, Fermilab , March, 2000 [44];

Conference "Advanced Statistical Techniques in Particle Physics", Durham, March, 2002 [45];

Conference PHYSTAT2003, SLAC, September, 2003 [46];

Conference PHYSTAT2005, Oxford, September, 2005[47].

В связи с подготовкой к началу и с началом работы LHC заметно возросла исследовательская деятельность в области статистики. Были созданы специальные структуры: Статистический комитет CMS [48], Объединенный статистический форум ATLAS-CMS [49].

Было организовано несколько рабочих совещаний, посвященных запуску LHC, PhyStat-LHC Workshop, CERN, June, 2007 [50] и ACAT'2008, Are we ready for LHC era experiments?, Erice, November, 2008 [51], а также очередное совещание PhyStat'2011, CERN, January, 2011 [52].

Заключение

Рассмотрено состояние статистического программного обеспечения исследовательских работ в области физики высоких энергий. Важно отметить большой прогресс в развитии статистических методов анализа данных в последнее время и появление программных продуктов, обеспечивающих унифицированный подход к получению и представлению результатов различных экспериментов.

Работа поддержана грантом РФФИ 10-02-00468-а.

Список литературы

[1] R. Brun, F. Rademaker, Nucl.Instr.&Meth., **A389** 1(997), 81; см. также http://root.cern.ch/

[2] L. Moneta, I. Antcheva, R. Brun, A. Kreshuk, *ROOT Statistical Software*, Proceedings of PhyStat-LHC, CERN-2008-001, pp. 179-183.

[3] http://www.r-project.org/
[4] http://roofit.sourceforge.net
[5] https://twiki.cern.ch/twiki/bin/view/RooStats/
[6] http://www.mppmu.mpg.de/bat/
[7] https://indico.desy.de/conferenceDisplay.py?confId=1097
[8] http://indico.cern.ch/conferenceDisplay.py?confId=24781
[9] http://tmva.sourceforge.net/
[10] http://sourceforge.net/projects/statpatrec
[11] S.V. Gleyzer, H. Prosper, *PARADIGM, a Decision Making Framework for Variable Selection and Reduction in High Energy Ph*ysics, Proceedings of Science (ACAT08) 067, 2008.
[12] A. Caldwell, D. Kollar, K. Kroninger, *BAT- The Bayesian analysis toolkit*, Comp.Phys.Commun., **180** (2009) 2197.
[13] Дж. Дж. Кемени, Дж. Л. Снелл, Конечные цепи Маркова, Москва, "Наука", 1970.
[14] E.T. Jaynes, Probability Theory: The Logic of Science: Principles and Elementary Applications, v.1, Cambridge University Press, 2003.
[15] S. Bityukov, N. Krasnikov, S. Nadarajah, V. Smirnova, *Confidence distributions in statistical inference*. Bayesian Inference and Maximum Entropy Methods in Science and Engineering, 30-th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Chamonix, France, 4-9 July 2010, Eds: A. Mohammad-Djafari, J-F. Bercher; С.И. Битюков, Н.В. Красников, Препринт ИФВЭ 2008-10, 2010.
[16] G. D'Agostini, Bayesian Reasoning in Data Analysis, a Critical Introduction. World Scientific, Hackensack, NJ, 2003.
[17] Markov Chain Monte Carlo in Practice, eds W.R. Gilks, S. Richardson, D.~Spigelhalter, Chapman and Hall, 1996.
[18] H.B. Prosper, *Multivariate Methods in Particle Physics Today and Tomorrow*, Proceedings of Science (ACAT08) 010, 2008.
[19] Как пример, D0 Collaboration (V.M. Abazov et al.), Evidence for production of single top quarks, Phys.Rev. D78 (2008) 012005; FERMILAB-PUB-08-056-E, 2008.
[20] O'Hagan, Kendall's Advanced Theory of Statistics, vol. **2B**,Bayesian Inference, Oxford University Press, NY, 2002.
[21] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, 2007.
[22] R.J. Barlow, *Events Classification Using Weighting Methods*, J. Comput. Phys., **72** (1987) 1.
[23] Corinna Cortes and V. Vapnik, Support-Vector Networks, Machine Learning, 20, 1995.

[24] Субботин С.В., Большаков Д.Ю., *Применение байесовского классификатора для распознавания классов целей*, Журнал Радиоэлектроники, **4**, 2006.

[25] А.И. Орлов, Прикладная статистика, Москва, Издательство «Экзамен», 2004.

[26] С. Хайкин, Нейронные сети: полный курс, 2-е издание, Москва, Издательский дом «Вильямс», 2006.

[27] F. Rosenblatt. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, Cornell Aeronautical Laboratory, Psychological Review, **65** (1958) 386-408.

[28] А.И. Галушкин, Синтез многослойных систем распознавания образов, Москва, «Энергия», 1974.

[29] D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Learning Internal Representations by Error Propagation*, Parallel Distributed Processing, **11** (1986) 318-362, Cambridge, MA, MIT Press.

[30] D.J.C. MacKay, A practical Bayesian Framework for Backprop Networks, Neural Comp., **4** (1991) 448-472.

[31] R.M. Neal, Bayesian Learning for Neural Networks, PhD thesis, Dept. of Computer Science, Univ. of Toronto, 1995.

[32] А.В. Левитин, Алгоритмы: введение в разработку и анализ (Introduction to The Design and Analysis of Algorithms), Москва, Издательский дом ``Вильямс", 2004, стр. 409-417.

[33] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed., Springer-Verlag, 2009.

[34] J. H. Holland, Adaptation in natural and artificial systems, University of Michigan Press, Ann Arbor, 1975.

[35] L. Breiman et al., Classification and Regression Trees, Waldsworth International, 1984.

[36] J. Friedman, N. Fisher, *Bump hunting in high dimensional data*, Statistics and Computing, **9** (1999) 123-143.

[37] Y. Freund and R.E. Schapire, *A decision-theoretic generalization of on-line learning and an application to boosting*, J. of Computer and System Sciences 55 (1997) 119-139.

[38] L. Breiman, *Bagging Predictors*, Machine Learning **26** (1996) 123-140.

[39] L. Breiman, *Random Forests*, Machine Learning **45** (2001) 5-32.

[40] Narsky, *StatPatternRecognition in Analysis of HEP and Astrophysics Data*, Proceedings of PhyStat-LHC, CERN-2008-001, pp. 188-191.

[41] http://www.slac.stanford.edu/BFROOT/www/Statistics/

[42] http://www-cdf.fnal.gov/physics/statistics/statistics_home.html

[43] http://preprints.cern.ch/cernrep/2000/2000-005/2000-005.html

[44] http://conferences.fnal.gov/cl2k/

[45] http://www.ippp.dur.ac.uk/old/Workshops/02/statistics/Welcome.shtml

[46] http://www-conf.slac.stanford.edu/phystat2003/

[47] http://www.physics.ox.ac.uk/phystat05/
[48] https://twiki.cern.ch/twiki/bin/view/CMS/StatisticsCommittee
[49] http://indico.cern.ch/categoryDisplay.py?categId=1579
[50] http://phystat-lhc.web.cern.ch/phystat-lhc/
[51] http://acat2008.cern.ch/
[52] http://indico.cern.ch/conferenceOtherViews.py?view=standard&confId=107747

Рукопись поступила 3 июня 2011 г.

И.В. Ажиненко, С.И. Битюков, Н.В. Красников, В.В. Смирнова

Программное обеспечение статистической обработки информационных потоков в задачах физики высоких энергий.

Препринт отпечатан с оригинала-макета, подготовленного авторами.

Подписано к печати 27.10.2011. Формат 60 × 84/16. Офсетная печать.
Печ.л. 1,68. Уч.- изд.л. 2,59. Тираж 80. Заказ 28. Индекс 3649.

ГНЦ РФ Институт физики высоких энергий
142281, Протвино Московской обл

Индекс 3649

П Р Е П Р И Н Т 2011-23, ИФВЭ, 2011