Академия народного хозяйства
и государственной службы
при Президенте РФ

Кафедра
системного анализа и информатики

Метод группового учета аргументов
(часть 1)

М. Александров
РАНХиГС при Президенте РФ
Автономный Университет Барселоны
В. Степашко (консультант)
Международный научно-учебный центр
информационных технологий и систем, Украина

Москва 2012

Курс: Data Mining (вводный)

Раздел III. Индуктивное моделирование

Тема: Метод группового учета аргументов

Лекция: L3-2_part1

Версия: 3.2 (01.11.2012)

Содержание

Введение

Две задачи моделирования

Имеются данные наблюдений $Y_e(x)$

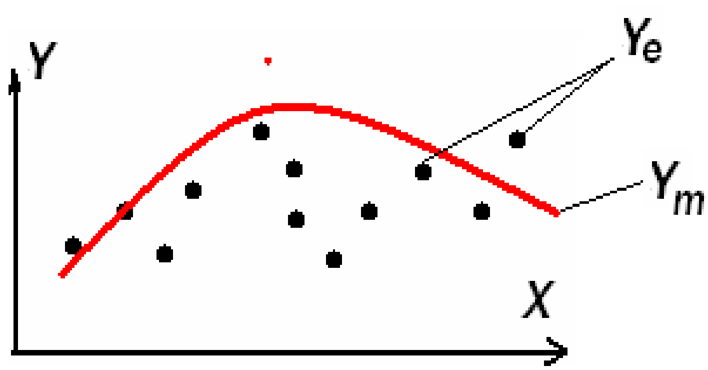
Надо построить модель для их описания $Y_m = F(a_1, a_2, \ldots x)$.

Обозначения: e = experiment, m = model

Здесь:

$(a_1, a_2, \ldots)$ – параметры модели

F – структура модели



Пусть наша модель – полином:

$Y_m = a_0 + a_1 x$

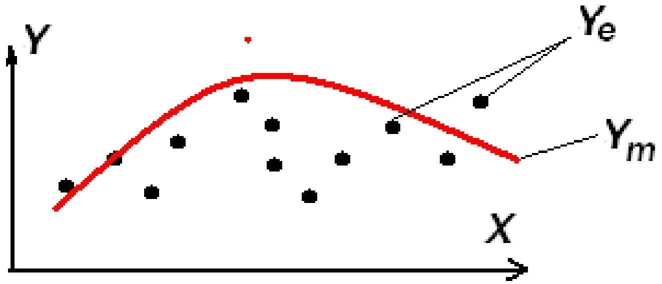
$Y_m = a_0 + a_1 x + a_2 x^2$

$Y_m = a_0 + a_3 x^3$

etc.

Введение

Две задачи моделирования



Ситуация 1. Модель задана

Найти (a_1, a_2, …) => $\min_{\boldsymbol{a}} \| Y_e - Y_m \|$

Имеем задачу параметрической оптимизации по ‘***a***’

Ситуация 2. Модель не задана

Найти F и (a_1, a_2, …) => $\min_{\boldsymbol{F}} K(Y_e, Y_m)$

Имеем задачу структурной оптимизации по ‘***F*** ’

Здесь: ***K*** – критерий, отражающий прогнозные свойства модели

Предмет рассмотрения: ситуация 2, т.е. поиск оптимальной модели F

Введение

История

Мы рассматриваем:
МГУА = Метод Группового Учета Аргументов

- Метод был разработан в 1968 году акад. А.Г. Ивахненко (Институт кибернетики АН Украины).
  Термин МГУА введен автором для предложенного им метода

- Классический МГУА принадлежит к числу эволюционных алгоритмов Искусственного Интеллекта (рассматривается во 2-й части лекции)

- В настоящее время МГУА развивается проф. В.С. Степашко (МНУЦ ИТС Национальной Академии Наук Украины).
  Проф. В.С. Степашко ученик и коллега акад. А.Г. Ивахненко
  Термин МГУА обозначает теперь уже несколько алгоритмов

Введение

История

С конца 90-х годов говорят не столько о МГУА
сколько об индуктивном подходе к процедуре моделирования

В научной литературе используются термины:

- индуктивное моделирование
- индуктивное порождение моделей

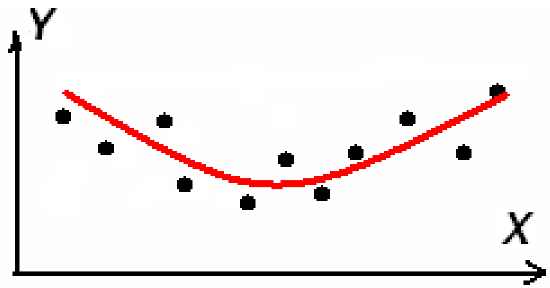
Они отражают развитие МГУА

По этой причине МГУА рассматривается как один из методов индуктивного моделирования

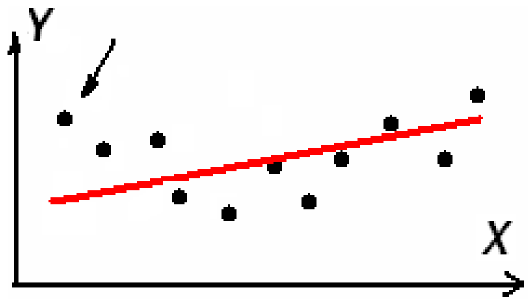
Введение

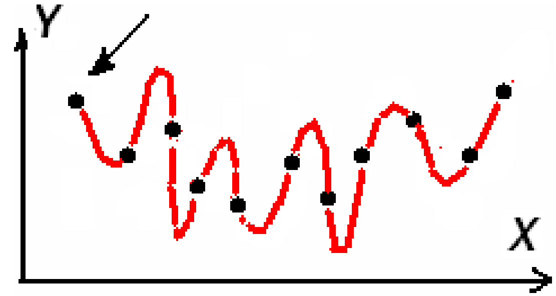
Сложность модели

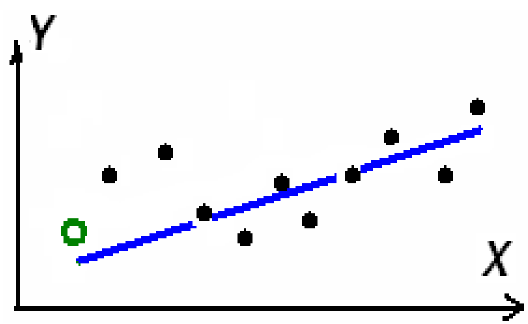
- Простая модель не реагирует на шум, но плохо отражает объект
- Сложная модель отражает объект, но чувствительна к шуму

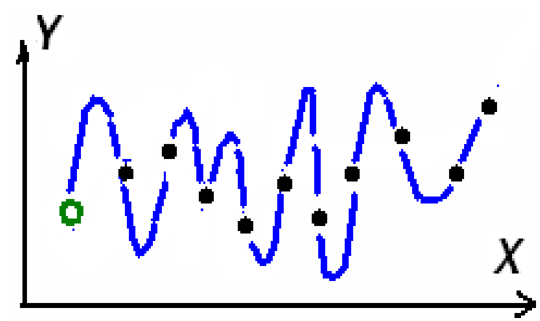


Исходные данные:
парабола + шум
(11 точек)









<u>Пример</u>.
Изменение модели
при изменении данных
(перемещаем одну точку)

Аппроксимация:
прямая линия
Почти нет изменений

Интерполяция:
полином 10-го порядка
Резкие изменения!

Введение

Назначение МГУА

МГУА позволяет выбрать модель оптимальной сложности из заданного класса моделей, чтобы описать имеющийся набор экспериментальных данных

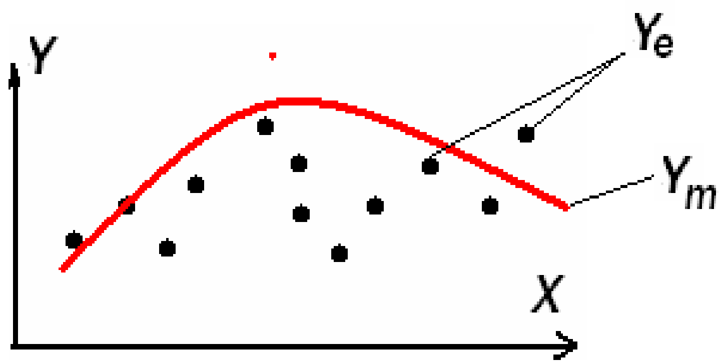
Построенная на этих данных модель им не противоречит, и поэтому может быть названа правдоподобной

Введение

Условия применения МГУА

МГУА обладает преимуществами, когда

(1) отсутствует или почти отсутствует априорная информация о структуре модели и распределении ее параметров



Пример:

Необходимо подобрать модель в классе тригонометрических полиномов:

$Y_m = a_o + \Sigma\, a_k \cos(kx) \quad k=1,\ldots,100$

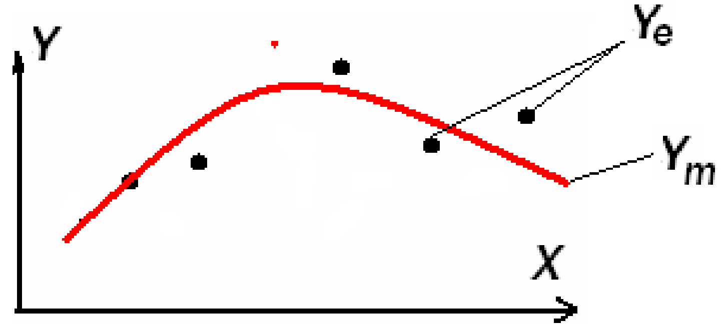
При этом спектр совсем неизвестен: какие частоты значимы, какие нет, и т.п.

Введение

Условия применения МГУА

МГУА обладает преимуществами, когда

(2) данных наблюдений крайне мало, вплоть до того, что параметров модели больше, чем число наблюдений



Пример:

Необходимо подобрать модель в классе тригонометрических полиномов

$Y_m = a_o + \Sigma\, a_k \cos(kx) \quad k=1,\ldots,100$

При этом число данных наблюдений $\{Y_e\}$ n=5,

то есть число неизвестных больше числа наблюдений

Введение

Условия применения МГУА

Однако, если:

- имеется априорная информация о структуре модели и/или распределении ее параметров
- имеется достаточно много данных наблюдений, чтобы обеспечить расчет параметров

тогда можно использовать другие подходы

Например,

- классическую теорию приближений (детерминированный подход)
- классическую математическую статистику (статистический подход)

и т.д.

Введение

Применение МГУА: задачи

- Аппроксимация таблично заданных функций
- Описание и прогнозирование временных рядов
- Выбор вычислительной схемы для решения дифф. уравнений
- Структурная идентификация объектов (модели вход-выход)
- Классификация / распознавание образов (поиск структур с учителем)
- Кластеризация / таксономия (поиск структур без учителя)
- Настройка структуры нейронных сетей (самоорганизация структуры)

и т.д.

Введение

Применение МГУА: предметные области

- Геофизика

  (описание пространственного распределения гео-параметров)
- Компьютерная лингвистика

  (проверка подобия слов, построение словарей)
- Бизнес приложения

  (прогноз временных рядов, группировка сходных эконом. объектов)

И т.д.

Введение

Применение МГУА: предметные области

Примеры бизнес-приложений

- Анализ эластичности и порогов цен (marketing mix метод)
- Изучение влияния цен и рекламы на объем продаж
- Сегментация (кластеризация) покупателей, магазинов, продуктов
- Определение наиболее востребованных характеристик продуктов

и т.д.

Содержание

Введение

Индуктивный подход

Внешние критерии

Алгоритмы МГУА

Комментарий к МГУА

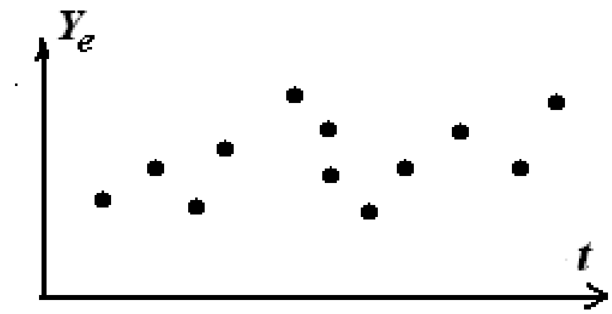
Материалы

Индуктивный подход

Типичная проблема

Описание временного ряда формулой

Пусть априорная информация задана ←



Экспериментальные данные

Заданная априорная информация

- Тип зависимости (формула)
- Серия моделей из заданного класса и уровень шума

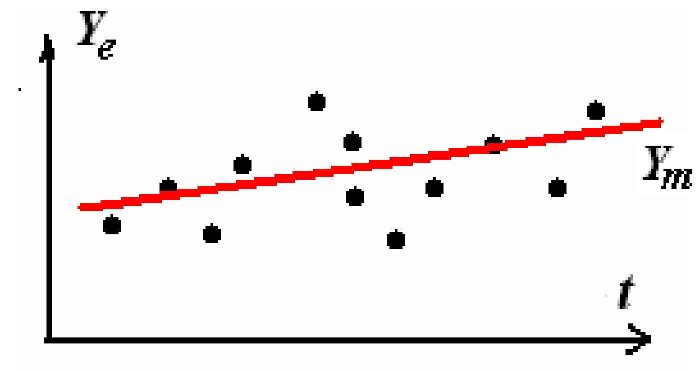
Индуктивный подход

Мы имеем априорную информацию

Заданная регрессионная модель

$Y_m = a_0 + a_1 t$ или $Y_m = a_0 + a_1 t + a_2 t^2$, *etc.*

$\| Y_m - Y_e \|$ => мин (используем МНК)



Точки – данные наблюдений

Красные линии – модели

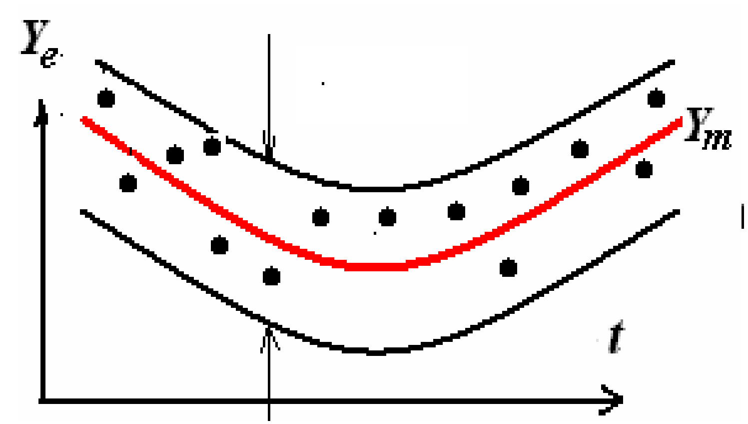
или

Серия моделей из заданного класса

с заданным уровнем шума

$Y_m = a_0 + a_1 t + a_2 t^2 + \ldots..$

$\| Y_m - Y_e \|$ => $\boldsymbol{\varepsilon^2}$ (используем МНК)



Коридор отражает ошибку **ε**

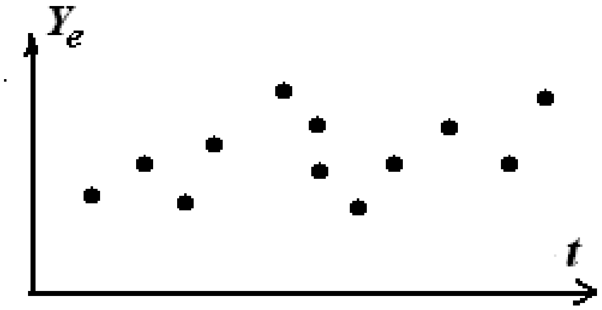
МНК = метод наименьших квадратов

Индуктивный подход

Типичная проблема

Описание временного ряда формулой

Пусть априорная информация отсутствует ←



Экспериментальные данные

У нас нет априорной информации

В этом случае мы используем Индуктивное Моделирование (ИМ)

Подход требует, чтобы мы, прежде всего, зафиксировали класс моделей, в котором будем искать лучшую модель

Замечание:

Класс моделей должен отражать возрастающую сложность модели

Индуктивный подход

Классы и сложность модели

ИМ имеет дело с заранее фиксированным классом моделей

Класс моделей зависит от рассматриваемой задачи

Это могут быть:

- полиномы и тригонометрические полиномы одной переменной
- линейные и нелинейные функции многих переменных
- кластеры объектов

и т.п.

Сложность модели также зависит от задачи

Это может быть:

- число оцениваемых параметров
- число кластеров

и т.п.

Индуктивный подход

Перебор моделей

ИМ не может найти самую оптимальную модель среди всех возможных! Он ищет оптимальную модель только в заданном классе.

Пример класса моделей: полиномы одной переменной *(t)*

$Y_0 = a_0$

$Y_1 = a_0+a_1t$ $\quad$ $Y_1 = a_0+a_2t^2$ $\quad$ …………… $\quad$ $Y_1 = a_0+a_{100}t^{100}$

$Y_2 = a_0+a_1t+ a_2t^2$ $\quad$ $Y_2 = a_0+a_1t + a_3t^3$ $\quad$ ……………... $\quad$ $Y_2 = a_0+a_{99}t^{99} + a_{100}t^{100}$

……………………

Пример класса моделей: линейные функции многих переменных $(x_1,x_2..)$

$Y_0 = a_0$

$Y_1 = a_0 + a_1x_1$ $\quad$ $Y_1 = a_0 + a_2x_2$ $\quad$ ………… $\quad$ $Y_1 = a_0 + a_{100}x_{100}$

$Y_2 = a_0 + a_1x_1 + a_2x_2$ $\quad$ $Y_2 = a_0 + a_1x_1 + a_3x_3$ …… $\quad$ $Y_2 = a_0+ a_{99}x_{99}+a_{100}x_{100}$

…………………………

Индуктивный подход

Расчет параметров данной модели

Когда модель задана, ее параметры подбираются по данным
Для этого составляется и решается система уравнений

Пусть ищем функцию 1-й переменной в форме полинома
Пусть имеются данные наблюдений в точках (0.1, 0.2, 0.3,...)
Пусть максимальная степень полинома ограничена n<=10

Пусть на данном этапе перебора моделей рассматриваются
только модели 2 порядка, тогда имеем

$Y_2 = a_0 + a_1 t + a_2 t^2$ $\quad$ $Y_2 = a_0 + a_1 t + a_3 t^3$ $\quad$ ………… $\quad$ $Y_2 = a_0 + a_9 t^9 + a_{10} t^{10}$

Индуктивный подход

Расчет параметров данной модели

Пример модели:

$Y_m = a_0 + a_1 t + a_2 t^2$

$Y_e(0.1) = a_0 + a_1\, 0.1 + a_2\, 0.1^2$

$Y_e(0.2) = a_0 + a_1\, 0.2 + a_2\, 0.2^2$

………..

………..

Пример модели:

$Y_m = a_0 + a_9 t^9 + a_{10} t^{10}$

$Y_e(0.1) = a_0 + a_9\, 0.1^9 + a_{10}\, 0.1^{10}$

$Y_e(0.2) = a_0 + a_9\, 0.2^9 + a_{10}\, 0.2^{10}$

…………

…………

Мы ищем параметры (a_0, a_1,...) из условия $\Sigma\,(Y_e - Y_m)^2 \Rightarrow \min$

Используются:

а) метод наименьших квадратов, если параметры входят линейно

$Y_m = a_0 + a_1 t + a_2 t^2 + \dots$

б) нелинейное программирование, если параметры входят нелинейно

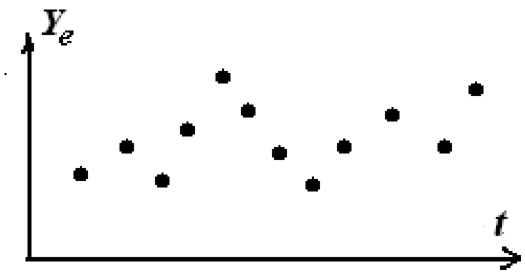
$Y_m = a_0 + a_1 \cos(b_1 t) + a_2 \cos(b_2 t) + \dots$

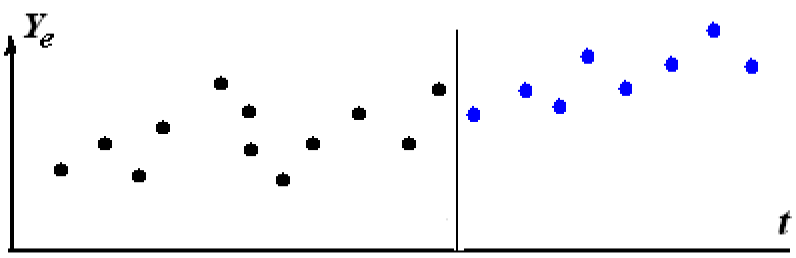
Индуктивный подход

Требование 1: устойчивость к новым данным

Мы должны обеспечить хорошее прогнозное свойство модели

то есть устойчивость к новым данным





Черные точки – данные эксперимента

Синие точки – возможные новые данные

Мы строим прогноз для синего набора точек

Индуктивный подход

Мера устойчивости модели, идея

Поскольку чувствительность модели должна проверяться на новых данных, то естественно разбить все множество значений на две части.

- Первый набор данных будет использоваться для построения модели. Этот набор данных называется данными для обучения   (Training)

- Второй набор данных будет играть роль новых данных и он будет использоваться для проверки качества построенной модели. Этот набор данных называется данными для проверки (Checking)

Индуктивный подход

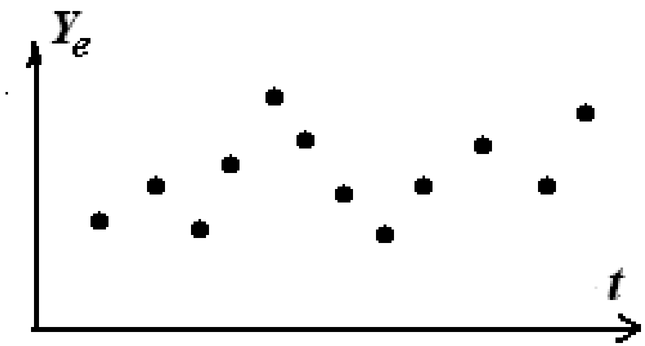
Реализация идеи

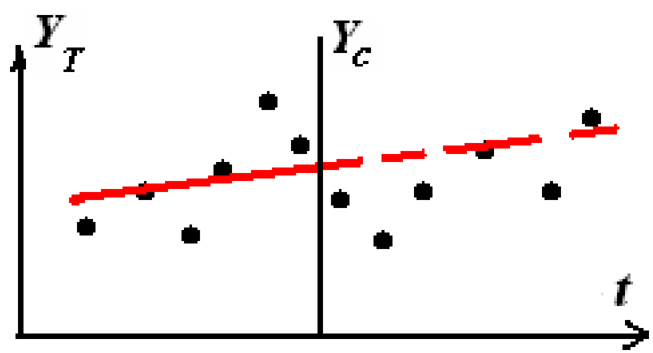
Для реализации идеи надо иметь критерий,

который бы оценивал качество модели на новых данных.

Этот критерий называется критерием регулярности

Он является внешним критерием качества модели





Точки – эксперимент

Красная линия – модель

Training

Левые половина

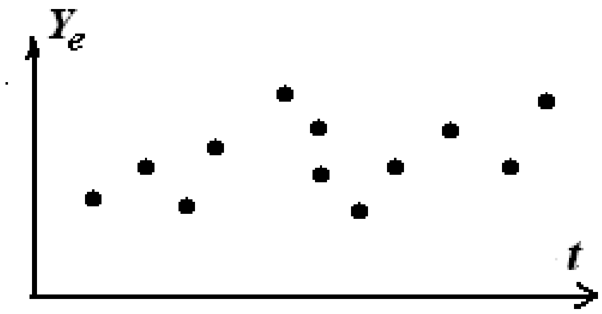
Checking

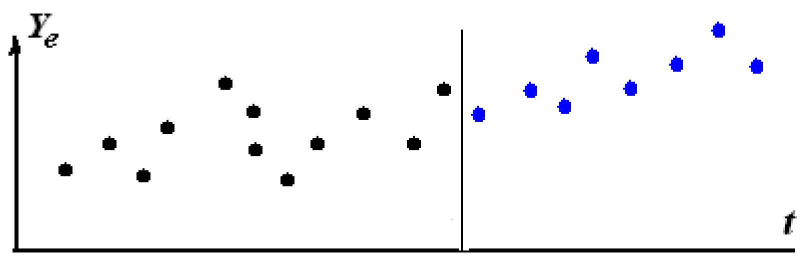
Правая половина

Индуктивный подход

Требование 2

Мы должны обеспечить хорошее описательное свойство модели, то есть независимость описания модели от данных





Черные точки – данные эксперимента

Синие точки – возможные новые данные

Мы должны построить модели для обоих наборов точек

Индуктивный подход

Мера независимости модели от данных - идея

Поскольку независимость от данных должна проверяться построением модели на разных данных, то естественно разбить все множество значений на две части.

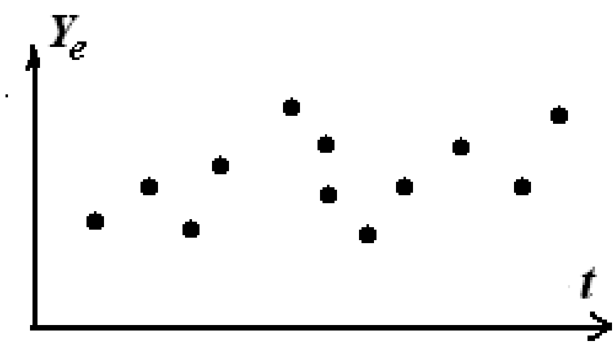
- Первый набор данных будет использоваться для построения одной модели. Второй набор будет использоваться для построения другой модели. Оба набора равноправны.

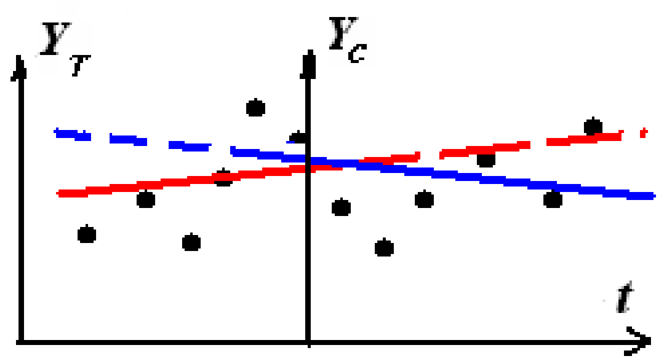
- В качестве первого набора можно взять данные Training.
  В качестве второго набора можно взять данные Checking.

Индуктивный подход

Реализация идеи

Для реализации идеи надо иметь критерий,
который бы позволил сравнить две модели
Этот критерий называется критерием несмещенности
Он является внешним критерием качества модели





Точки – эксперимент
Красная и фиолетовая
линии – модели

Первый набор — Второй набор
Левая половина — Правая половина

Индуктивный подход

Шаги МГУА

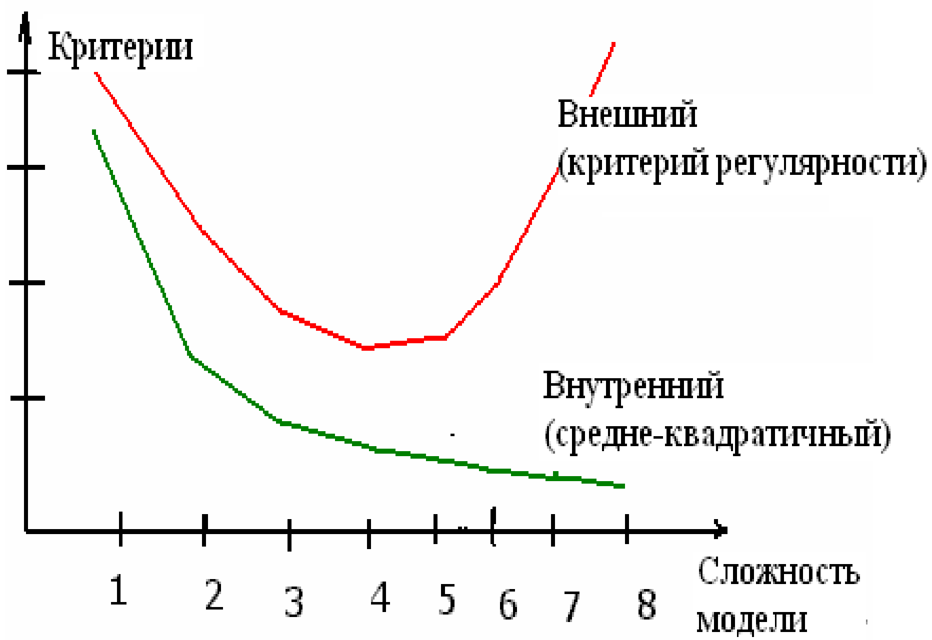
Пример с критерием регулярности

1. Определить серию моделей возрастающей сложности

2. Экспериментальные данные = *Набор данных для обучения + Набор данных для проверки*

3. Для заданного уровня сложности определяется оценка параметров модели на первом наборе данных. Используется *внутренний критерий*

4. Эта модель проверяется на втором наборе данных. Используется какой-либо *внешний критерий* (здесь регулярность)



5. Если внешний критерий достигает **минимума**, то STOP иначе увеличиваем сложность модели идем на шаг 3

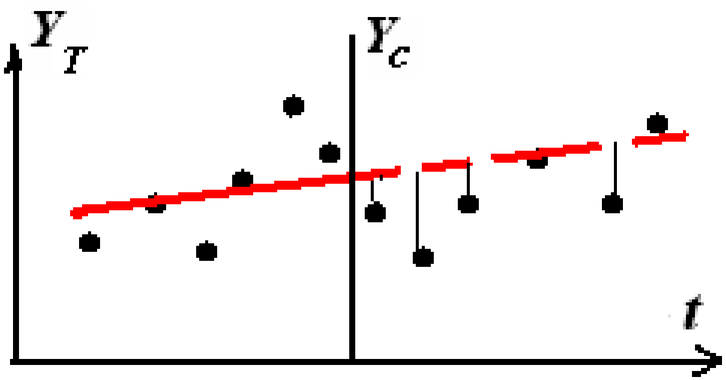
Содержание

Внешние критерии

Критерий регулярности

Оценивает устойчивость к новым данным

Модель, обученная на первом наборе данных, должна давать хорошие результаты на втором наборе данных (***T*** обучение, ***C*** проверка)

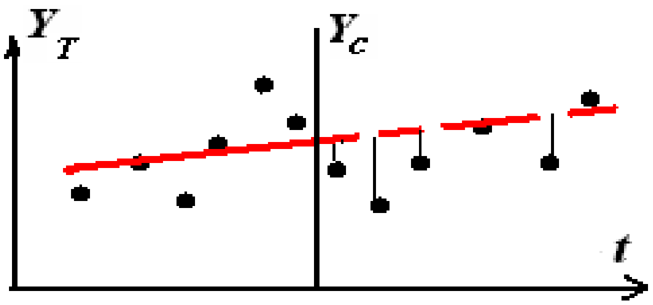


Обучение на Training

Проверка на Checking

Внешние критерии

Пример критерия регулярности



Training – левая половина

Checking – правая половина

$$Kreg = \sum_c (Y_e - Y_m(T))^2$$

Здесь: Y_e – данные наблюдений

$Y_m(T)$ - значения модели, обученной на наборе Training

$\sum_c$ - суммирование по точкам набора Checking

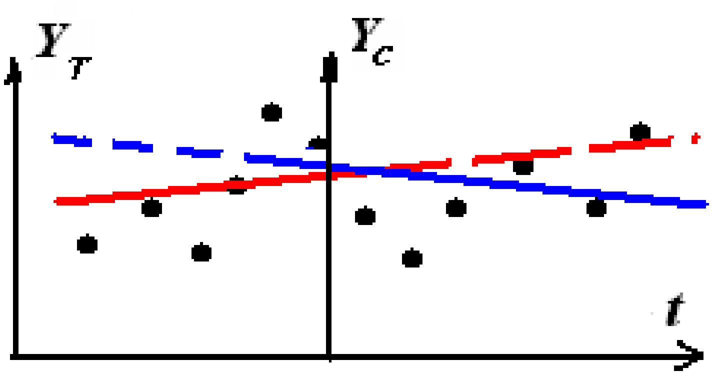
P.S. Можно пронормировать $Kreg$, разделив выражение на $\sum_c Y_e^2$

Внешние критерии

Критерий несмещенности

Обеспечивает независимость модели от данных

Модель, обученная на первом наборе данных, и модель обученная на втором наборе должны быть близки (***T*** первый набор, ***C*** второй набор)



Формы критерия несмещенности

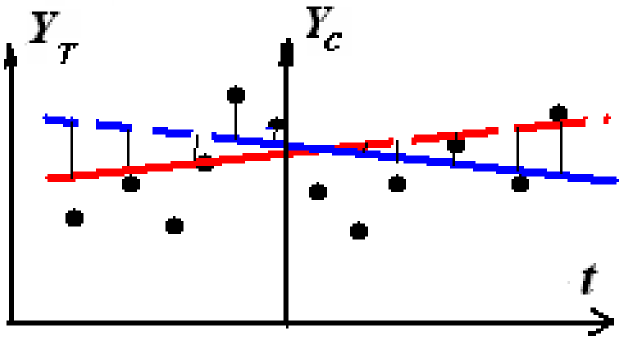
1) Критерий, ориентированный на
   анализ решений
2) Критерий, ориентированный на
   анализ параметров

Обучение на Training

Обучение на Checking

Внешние критерии

Пример критерия несмещенности, анализ решений



Различие моделей определяется
во всех точках наблюдений

$$K_{unbias} = \sum (Y_m(T) - Y_m(C))^2$$

Здесь: $Y_m(T)$ - значения модели, обученной на наборе Training

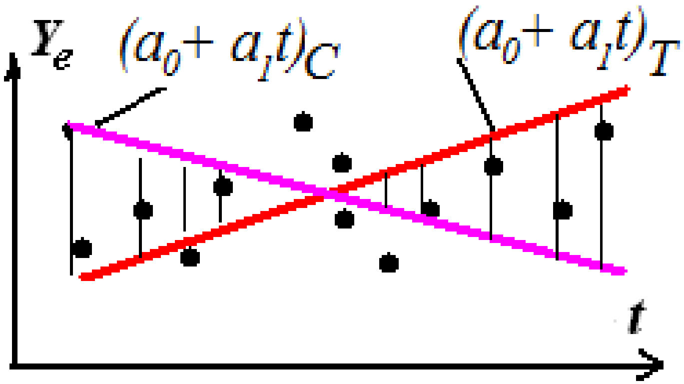
$Y_m(C)$ - значения модели, обученной на наборе Checking

$\sum$ - суммирование по всем точкам

P.S. Можно пронормировать K_{unbias}, разделив выражение на $\sum Y_e^2$

Внешние критерии

Пример критерия несмещенности, анализ параметров



Kunbias $=\sum a_i(T){*}a_i(C) \,/\, (|a(T)| * |a(C)|)$

Здесь:

$a_i(T)$ – модель, построенная на Training

$a_i(C)$ – модель, построенная на Checking

$Ym(T)$ - значения первой модели

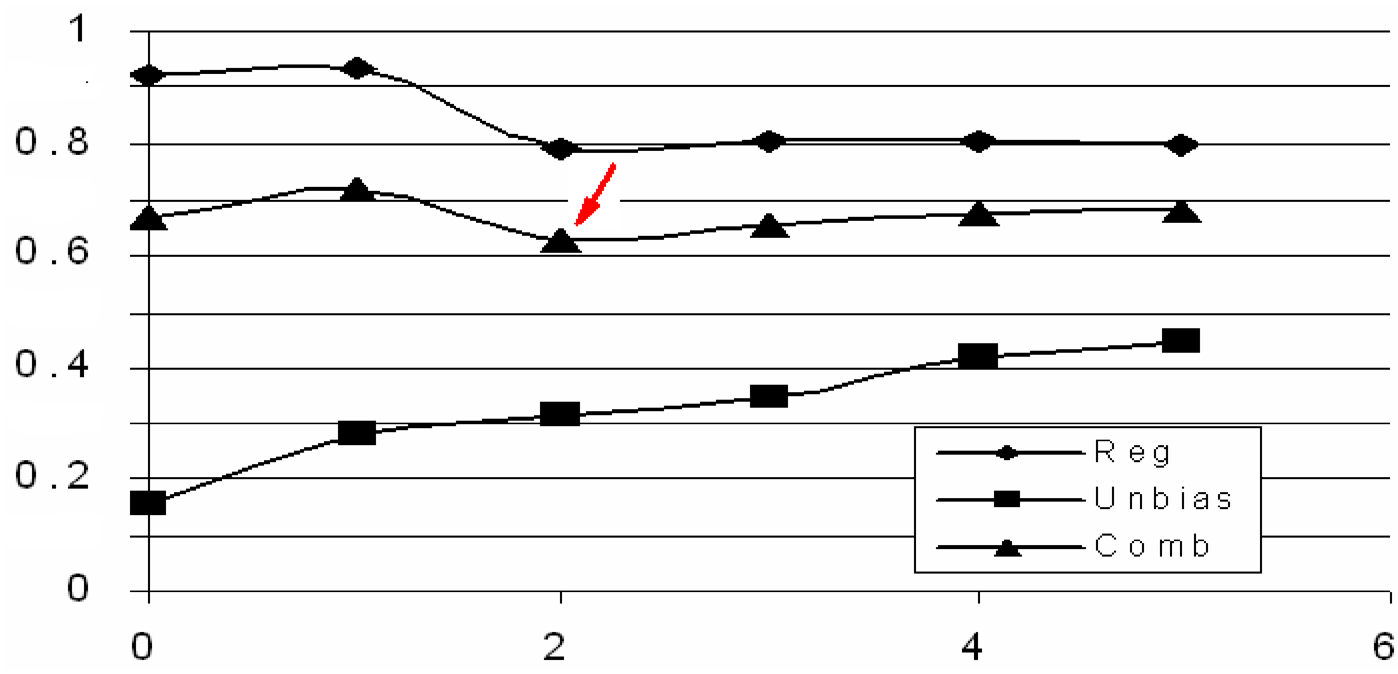
$Ym(C)$ - значения второй модели

$\sum$ - суммирование по всем точкам

Внешние критерии

Два критерия, правило свертки

- Назначаются веса λ_1, λ_2: $\lambda_1 + \lambda_2 = 1$ и рассчитывается комбинированный критерий $K = \lambda_1 K_{reg} + \lambda_2 K_{unbias}$
- Выбирается модель, лучшая по комбинированному критерию



Сложность модели

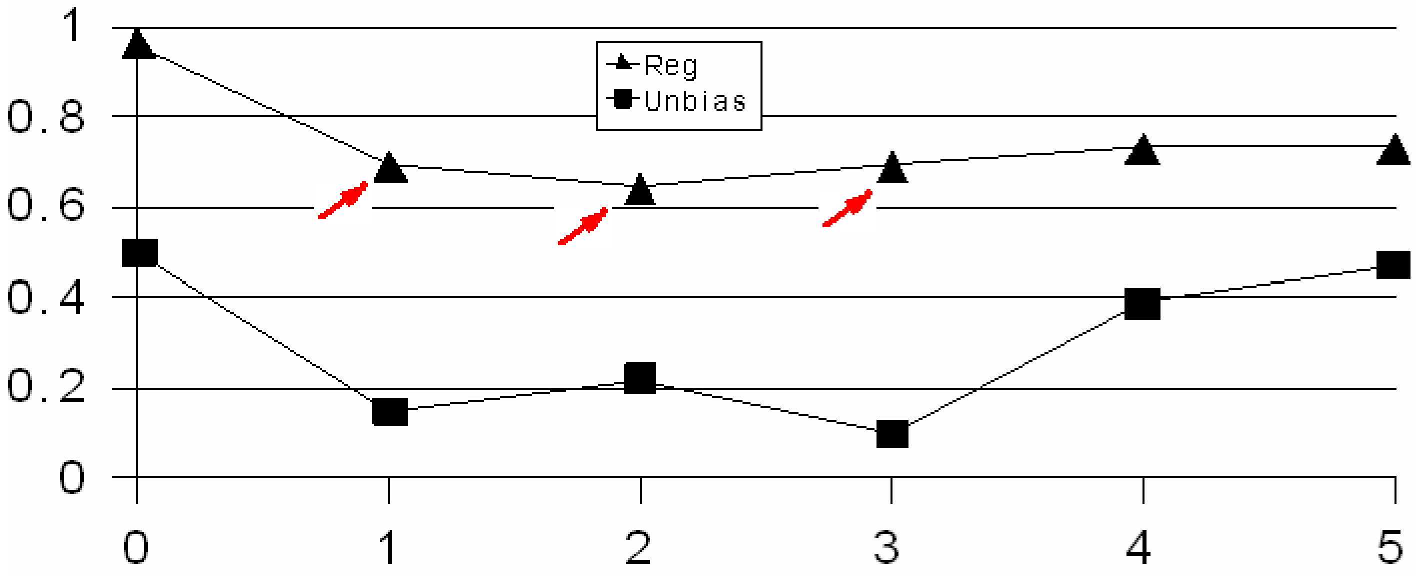
В данном примере лучшими по K_{reg} являются модели 2,3,4,5, по K_{unbias} - модель 0. Комбинированный критерий отбирает модель 2

Внешние критерии

Два критерия, последовательный отбор

Вместо отбора модели по комбинированному критерию $K = \lambda_1 K_{reg} + \lambda_2 K_{unbias}$ мы используем другую стратегию

- Выбираются лучшие модели по K_{reg}
- Из них выбираются лучшие по K_{unbias}



Сложность модели

В данном примере лучшими по K_{reg} являются модели 1,2 и 3

Критерий K_{unbias} отбирает модель 3

Внешние критерии

Верификация модели

Для реализации внешних критериев нужно иметь два набора данных

- В критерии регулярности они служат для построения и проверки модели, и называются Training Data и Checking Data
- В критерии несмещенности они служат для построения двух вариантов моделей, которые проверяются на полном наборе

Однако значения внешних критериев не дают объективную оценку качества модели, они служат только для отбора моделей.

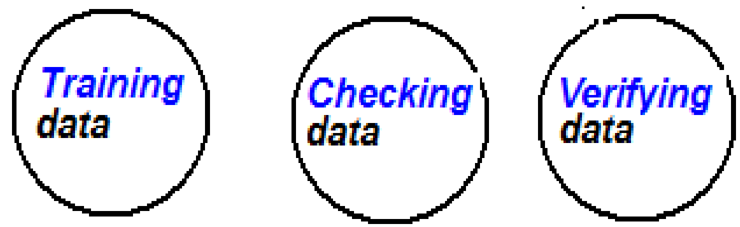
Для оценки качества нужно иметь еще один независимый набор данных, т.н. экзаменационную/контрольную выборку, Verifying Data. Именно, на этой выборке и проверяется качество/адекватность модели

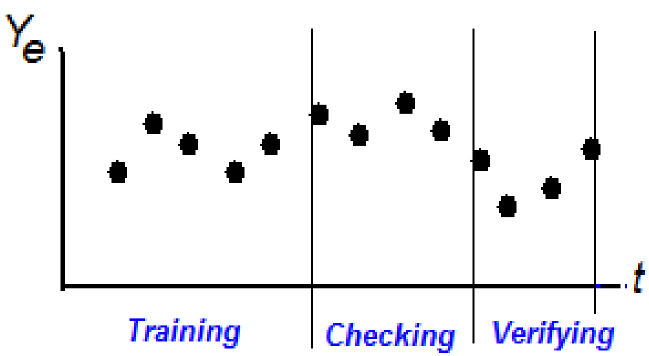
Внешние критерии

Верификация модели

Таким образом, чтобы проводить верификацию модели, в МГУА следует использовать не две, а три выборки из исходных данных

DATA =





Разбиение не обязательно должно быть равномерным

Типичный пример (В.С.Степашко):

Verifying 20%, Training:Checking = 2:1

<u>Примечание</u>: Learning Data = Training Data + Checking Data

В русском языке последнее «равенство» не отразишь, так как у нас Learning и Training переводятся одинаково

Содержание

Алгоритмы МГУА

Базовые алгоритмы

Имеется 4 базовых алгоритма МГУА и множество их модификаций

COMBI - комбинаторный алгоритм

MULTI - комбинаторно-селекционный алгоритм

MIA - многорядный итеративный алгоритм

RIA - релаксационный итеративный алгоритм

Рассмотрим базовые алгоритмы на примере линейной модели многих переменных ($x_1, x_2 \ldots\ldots x_{20}$)

$Y_0 = a_0$

$Y_1 = a_0 + a_1 x_1$ $\quad Y_1 = a_0 + a_2 x_2$ $\quad \ldots\ldots\ldots$ $\quad Y_1 = a_0 + a_{20} x_{20}$

$Y_2 = a_0 + a_1 x_1 + a_2 x_2$ $\quad Y_2 = a_0 + a_1 x_1 + a_3 x_3 \ldots\ldots\ldots$ $\quad Y_2 .= a_0 + a_{19} x_{19} + a_{20} x_{20}$

……………………….

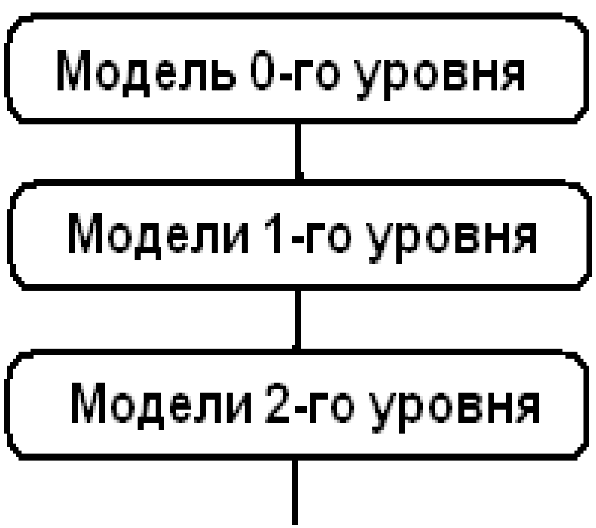
Алгоритмы МГУА

Базовые алгоритмы

Все алгоритмы являются многорядными.
Каждый ряд связан с одним уровнем сложности модели
Очевидно, в каждом ряду может находиться несколько моделей

Алгоритмы отличаются условиями формирования и отбора переменных при переходе от одного ряда к другому

Алгоритмы имеют параметры настройки, среди которых число лучших моделей каждого ряда. Параметры задаются пользователем



$Y_0 = a_0$

$Y_1 = a_0 + a_1 x_1$ $\quad$ $Y_1 = a_0 + a_2 x_2$ ..

$Y_2 = a_0 + a_1 x_1 + a_2 x_2$ $\quad$ $Y_2 = a_0 + a_1 x_1 + a_3 x_3$...

Алгоритмы МГУА - COMBI

COMBI – идея

COMBI = комбинаторный алгоритм

COMBI это простейший из базовых алгоритмов МГУА

Идея алгоритма: не пропустить ни одной из всевозможных моделей

Поэтому, на каждом уровне сложности:

- рассматриваются все модели
- не проводится селекция лучших комбинаций переменных

Если число переменных модели N, то число всех комбинаций $M=2^N$

Экспоненциальный рост ограничивает число переменных $N \leq (\sim)$ 20

Алгоритмы МГУА - COMBI

COMBI – реализация

На первом ряду используем все переменные x_i

$$Y_1 = a_0 + a_1x_1 \qquad Y_1 = a_0 + a_2x_2 \;\ldots\ldots\ldots \quad Y_1 = a_0 + a_{20}x_{20}$$

Во втором ряду используем все комбинации из 2-х переменных (x_i, x_j)

$$Y_2 = a_0 + a_1x_1 + a_2x_2 \quad Y_2 = a_0 + a_1x_1 + a_3x_3 \;\ldots Y_2 .= a_0 + a_{19}x_{19} + a_{20}x_{20}$$

В третьем ряду будут все комбинации из 3-х переменных (x_i, x_j, x_k)

И т.д.

Алгоритмы МГУА - COMBI

COMBI – реализация

Число моделей M на каждом ряду для случая N=20 (число переменных):

Ряд 0: $M_0 = C_{20}^{0} = 1$

Ряд 1: $M_1 = C_{20}^{1} = 20$

Ряд 2: $M_2 = C_{20}^{2} = 190$

Ряд 3: $M_3 = C_{20}^{3} = 1140$

Ряд k: $M_k = C_{20}^{k}$

……………………

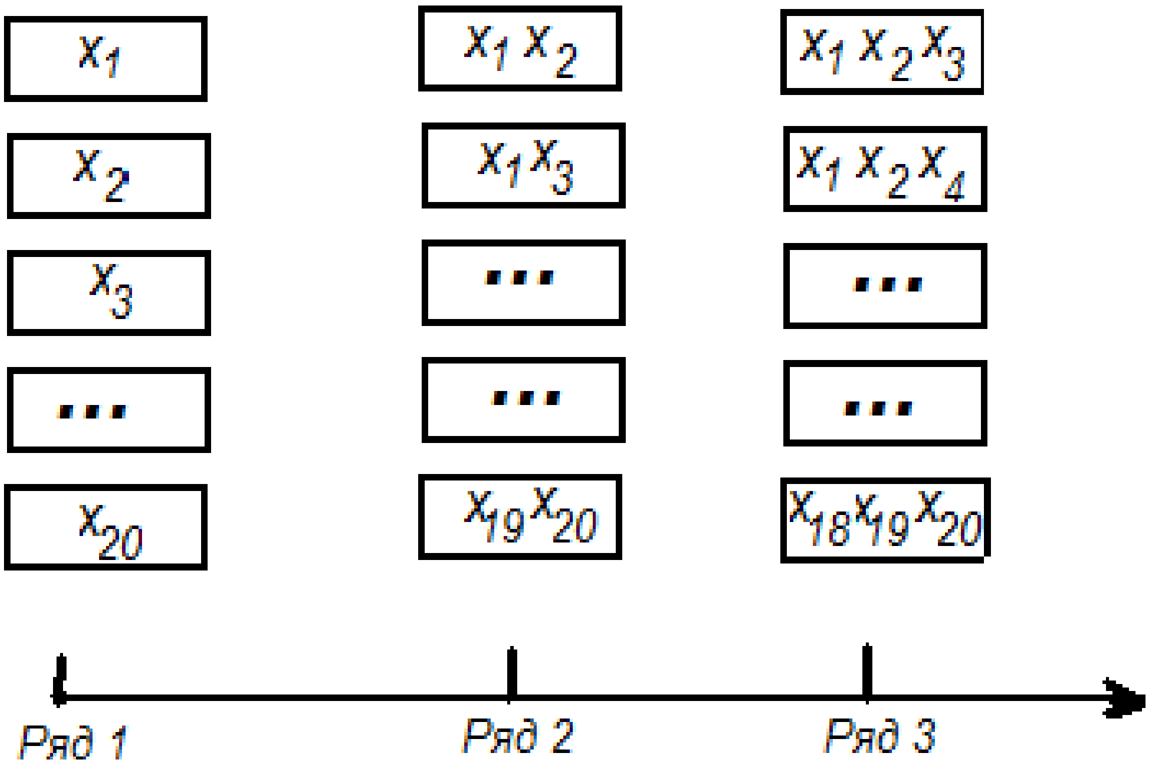
Ряд N: $M_N = C_{20}^{20} = 1$

Всего моделей (число всех комбинаций) $M_\Sigma = \Sigma_k C_N^k = 2^N$

При N=20 имеем $M_\Sigma = 2^{20} \sim 10^6 =$ 1 000 000

Алгоритмы МГУА - COMBI

COMBI – иллюстрация

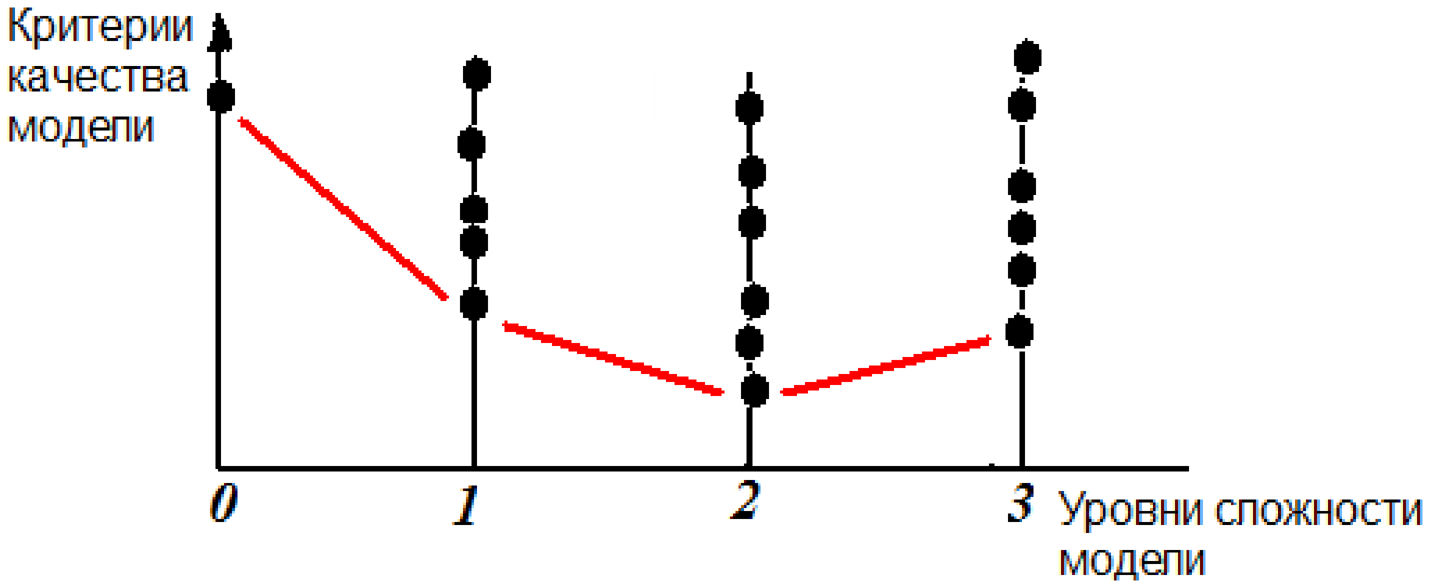


Полный перебор комбинаций параметров в каждом ряду

Номер ряда отражает уровень сложности модели

Алгоритмы МГУА - COMBI

COMBI – процедура перебора



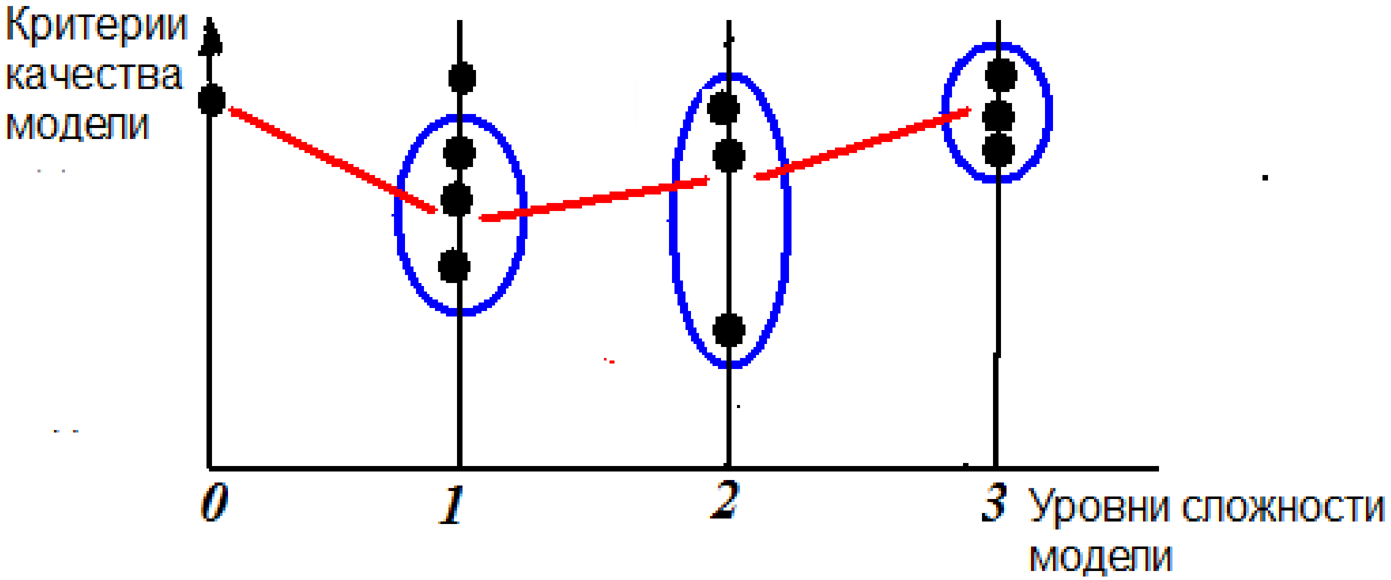
Имеем 1 модель (уровень 0), 20 моделей (уровень 1), и т.д.

<u>Правило перебора моделей</u>

- Рассчитывается качество всех моделей заданного ряда. Среди этих моделей выбирается лучшая модель, называемая рекордом
- Если текущий рекорд хуже рекорда предыдущего ряда, то процесс усложнения модели прекращается

Алгоритмы МГУА - COMBI

COMBI – модифицированная процедура перебора



Имеем 1 модель (уровень 0), 20 моделей (уровень 1), и т.д.

Модифицированное правило перебора (ансамблирование)

Рекорд определяется не по одной лучшей модели, а как среднее по *m*-лучшим моделям. Значение *m* - это параметр алгоритма, его задает пользователь. В нашем примере *m*=3

Алгоритмы МГУА - MULTI

MULTI – идея

MULTI = комбинаторно-селекционный алгоритм

MULTI представляет собой развитие COMBI

Основная идея MULTI:

уменьшить количество моделей, рассматриваемых на каждом ряду, и при этом по возможности не потерять лучшую комбинацию переменных

Поэтому, на каждом уровне сложности:

- отбирается фиксированное число лучших сочетаний переменных модели
- эти лучшие сочетания комбинируются со всеми остальными переменными (по одной) при переходе на следующий уровень

Алгоритмы МГУА - MULTI

MULTI – реализация

Параметр алгоритма p=2 (например), это число лучших моделей в ряду
Этот параметр задает пользователь

В первом ряду рассматриваем все модели:

$$Y_1 = a_0 + a_1 x_1 \qquad Y_1 = a_0 + a_2 x_2 \; \ldots\ldots \qquad Y_1 . = a_0 + a_{20} x_{20}$$

Из них выбираем переменные 2-х лучших моделей (т.к. p=2), пусть
это будут x_3, и x_7

Во втором ряду эти переменные x_3, и x_7 комбинируются со всеми другими переменными, то есть имеем различные пары (x_3, x_k), и (x_7, x_k), k=1,2,…20:

$$Y_2 = a_0 + a_3 x_3 + a_1 x_1 \quad Y_2 = a_0 + a_3 x_3 + a_2 x_2 \ldots Y_2 . = a_0 + a_7 x_7 + a_{20} x_{20}$$

Из них выбираем переменные 2-х лучших моделей (т.к. p=2), пусть

это будут (x_3, x_2) и (x_7, x_{15})

Алгоритмы МГУА - MULTI

MULTI – реализация

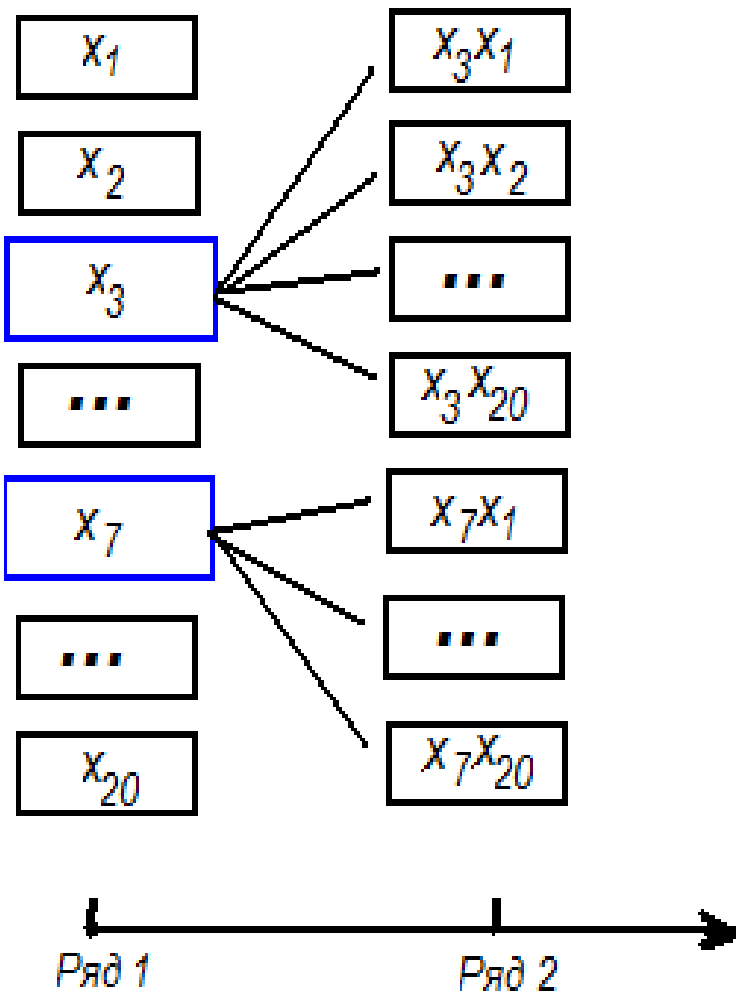
На третьем ряду процесс повторяется: эти пары переменных (x_3, x_2) и (x_7, x_{15}) комбинируются со всеми отсутствующими в них переменными по одной

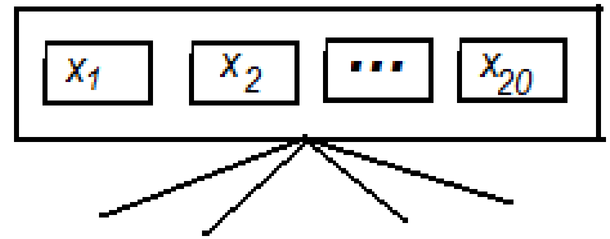
Число моделей M в каждом ряду: $M \leq pN = 2*20 = 40$

Всего моделей, рассматриваемых на всех шагах алгоритма (число всех комбинаций) $M_\Sigma \leq pN*N = pN^2 = 800$

Алгоритмы МГУА - MULTI

MULTI – иллюстрация





Ограниченный перебор возможных комбинаций переменных
Номер ряда соответствует уровню сложности модели

<u>Пример</u>

- Число лучших моделей p=2
- Переменные лучших моделей
  в первом ряду X_3, X_7

Алгоритмы МГУА - MIA

MIA – идея

MIA = многорядный итеративный алгоритм

MIA это исторически первый из алгоритмов МГУА

Основные идеи MIA:

- уменьшить количество моделей, рассматриваемых на каждом ряду
- уменьшить количество рядов, и, тем самым ускорить выход на оптимальный уровень сложности

Поэтому, на каждом ряду:

- отбирается фиксированное число лучших моделей, (каждая модель рассматривается, как переменная !)
- каждая пара лучших переменных порождает новую переменную при переходе на следующий уровень

Примечание. MIA несет в себе идею генетических алгоритмов и идею полиномиальных нейронных сетей. Однако MIA появился на 10-20 лет раньше !!!

Алгоритмы МГУА - MIA

MIA – реализация

- число лучших моделей в каждом ряду - p

  Например, пусть $p=5$

- функция преобразования пары переменных $t,s,$ одного ряда

  в переменные следующего ряда

  Примеры:

  $Z(t,s) = a_0 + a_t\, t + a_s\, s$

  $Z(t,s) = a_0 + a_t t + a_s s + a_{ts}\, t*s$

  $Z(t,s) = a_0 + a_t t + a_s s + a_{tt}\, t^2 + a_{ss} s^2$

  Можно использовать и др. функции

Число лучших моделей и функцию преобразования задает пользователь

Алгоритмы МГУА - MIA

MIA – реализация

В первом ряду имеем:

$$Y_1^{(1)} = a_0 + a_1x_1 \qquad Y_1^{(2)} = a_0 + a_2x_2 \;\ldots\ldots \qquad Y_1^{(20)} = a_0 + a_{20}x_{20}$$

Выбираем 5 лучших моделей (т.к. $p = 5$), пусть это будут, например, $Y_1^{(3)}, Y_1^{(5)}, Y_1^{(7)}, Y_1^{(9)}, Y_1^{(11)}$. Эти модели рассматриваются, как переменные

Во втором ряду рассматриваются функции от всех пар выбранных переменных $Y_1^{(3)}, Y_1^{(5)}, Y_1^{(7)}, Y_1^{(9)}, Y_1^{(11)}$ между собой, то есть модели вида: $Z(Y_1^{(i)}, Y_1^{(j)})$.

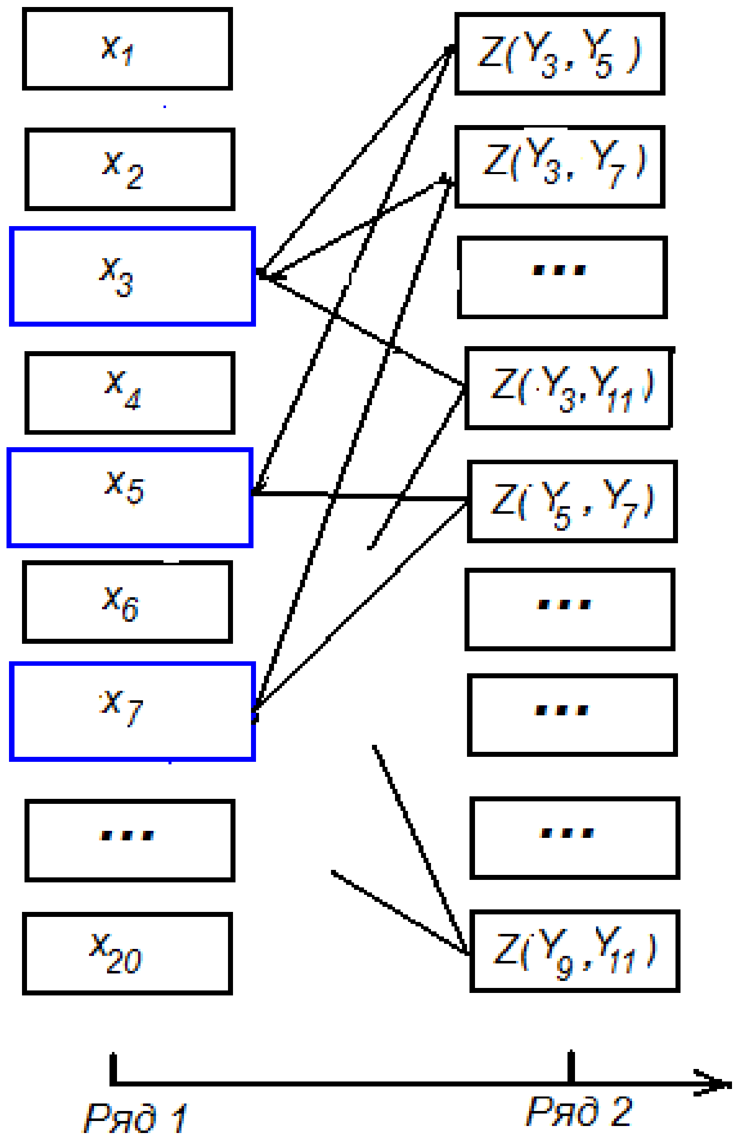
Пример одной (k-ой) из переменных второго ряда:

$$Z_k\,(Y_1^{(3)}, Y_1^{(5)}) = a_{0,k} + a_{3,k}\,Y_1^{(3)} + a_{5,k}Y_1^{(5)} + a_{3,5,k}Y_1^{(3)}\,Y_1^{(5)}$$

Число переменных M в каждом ряду: $M = C_p^2 = C_5^2 = 10$

Алгоритмы МГУА - MIA

MIA – иллюстрация



Формирование новых переменных
из лучших переменных текущего ряда

Пример

- Число лучших моделей p=5
- Переменные лучших моделей
  в первом ряду $Y_3, Y_5, Y_7, Y_9, Y_{11}$

Алгоритмы МГУА - RIA

RIA – идея

RIA = релаксационный итеративный алгоритм

RIA представляет собой развитие MIA

Основные идеи RIA:

- уменьшить количество моделей, рассматриваемых на каждом ряду, но при этом не потерять удачное сочетание переменных
- уменьшить количество рядов, и, тем самым ускорить выход на оптимальный уровень сложности

Поэтому, на каждом ряду:

- отбирается фиксированное число лучших моделей, (каждая модель рассматривается, как переменная !)
- каждая лучшая модель комбинируется с одной переменной исходного набора и эта пара порождает новую переменную при переходе на следующий уровень

Алгоритмы МГУА - RIA

RIA – реализация

Алгоритм имеет два параметра:

- число лучших моделей в каждом ряду - *p*

  Например, пусть *p=5*

- функция преобразования пары переменных *t*,*s*, взятых по одной из текущего ряда и из исходного ряда, в переменные следующего ряда

  Примеры:

  $Z(t,s) = a_0 + a_t t + a_s s$

  $Z(t,s) = a_0 + a_t t + a_s s + a_{ts} t*s$

  $Z(t,s) = a_0 + a_t t + a_s s + a_{tt} t^2 + a_{ss} s^2$

  Можно использовать и др. функции

Число лучших моделей и функцию преобразования задает пользователь

Алгоритмы МГУА - RIA

RIA – реализация

На первом ряду имеем:

$$Y_1^{(1)} = a_0 + a_1 x_1 \qquad Y_1^{(2)} = a_0 + a_2 x_2 \;\ldots\ldots \qquad Y_1^{(20)} = a_0 + a_{20} x_{20}$$

Выбираются 5 лучших моделей (т.к. p=5), пусть это будут, например, $Y_1^{(3)}, Y_1^{(5)}, Y_1^{(7)}, Y_1^{(9)}, Y_1^{(11)}$. Эти модели рассматриваются, как переменные

Во втором ряду рассматриваются функции от всех комбинаций выбранных переменных и исходных переменных, то есть модели вида $Z(Y_1^{(k)}, X_i)$, где $X_i = \{x_1, x_2, \ldots x_{20}\}$.

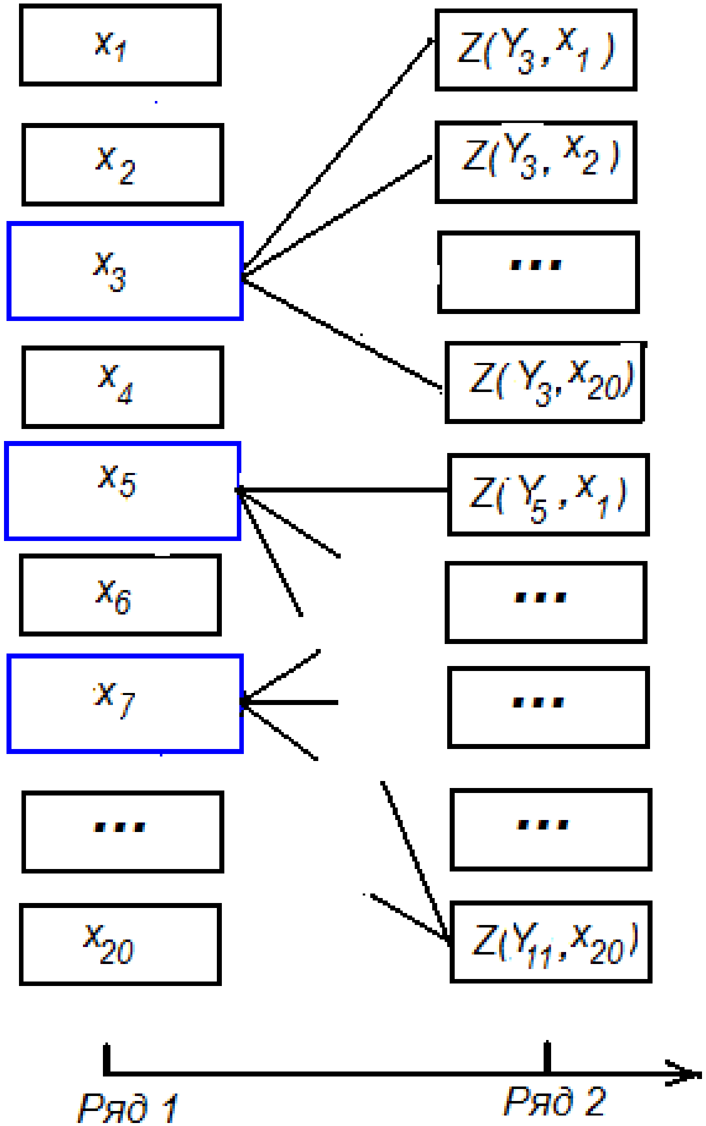
Пример одной (k-ой) из переменных второго ряда:

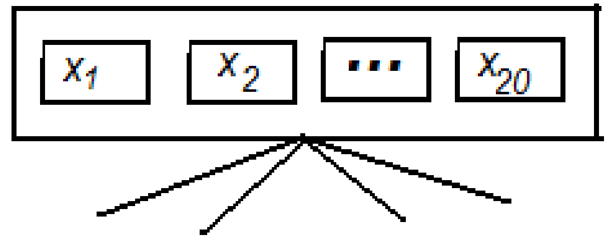
$$Z_k\,(Y_1^{(3)}, X_5) = a_{0k} + a_{3,k} Y_1^{(3)} + a_{5,k} X_5 + a_{3,5,k} Y_1^{(3)} X_5$$

Число переменных M в каждом ряду: $M = pN = 100$

Алгоритмы МГУА - RIA

RIA – иллюстрация





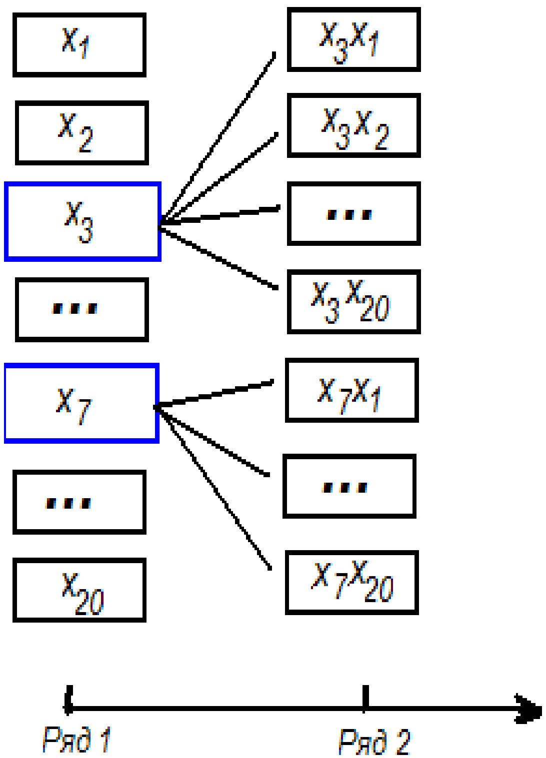
Формирование новых переменных
из лучших переменных текущего ряда
и исходного набора переменных

<u>Пример</u>

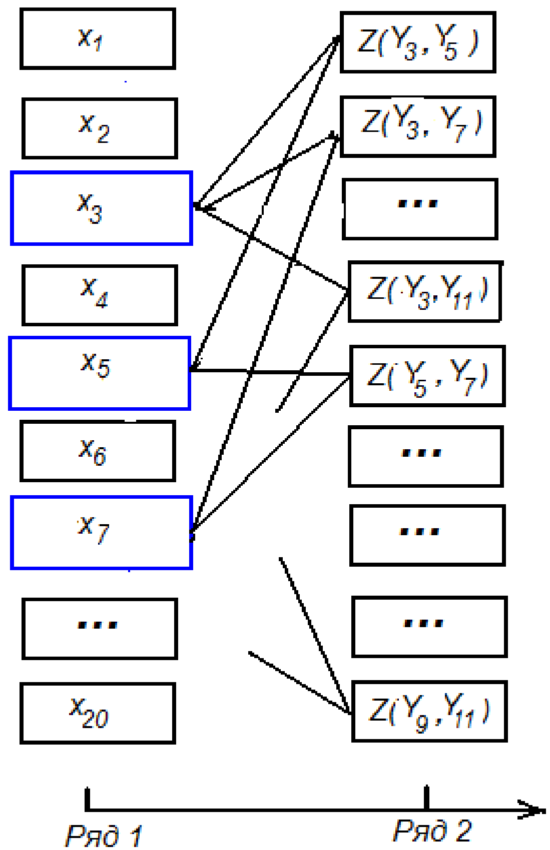
- Число лучших моделей p=5
- Переменные лучших моделей
  в первом ряду $Y_3, Y_5, Y_7, Y_9, Y_{11}$

Алгоритмы МГУА

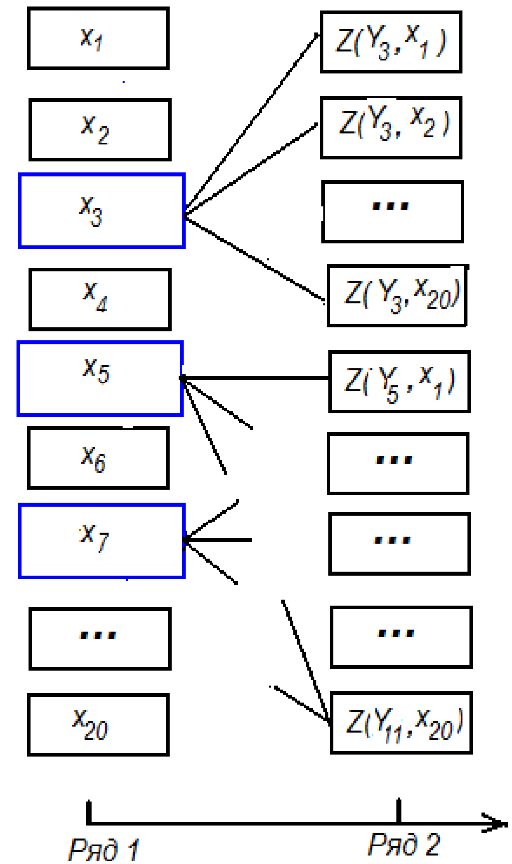
Свобода выбора в алгоритмах



Алгоритм MULTI



Алгоритм MIA



Алгоритм RIA

В указанных алгоритмах число лучших моделей, отбираемых на каждом ряду, может быть постоянным или варьироваться. Задает пользователь

Содержание

Комментарий к МГУА

Еще раз о терминологии

МГУА сейчас рассматривается как один из методов индуктивного моделирования наряду с другими методами

МГУА = метод группового учета аргументов
GMDH = group method of data handling (англ.)

Именно эти термины используются чаще всего в российской и зарубежной литературе

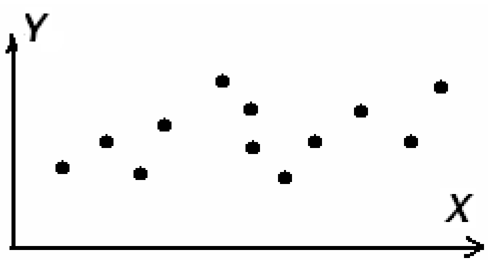
Автор и его ученики в своих ранних работах называли свой метод также индуктивным методом само-организации моделей
Однако эта терминология использовалось, скорее, для пояснения общего подхода, а не названия конкретного метода

Комментарий к МГУА

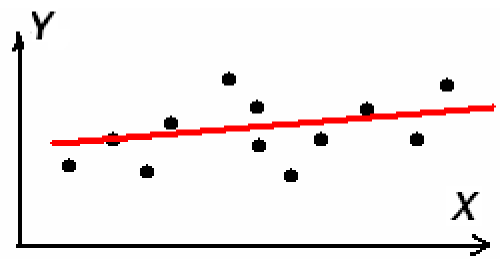
Что означает слово «модель» ?

Содержание модели зависит от задачи.

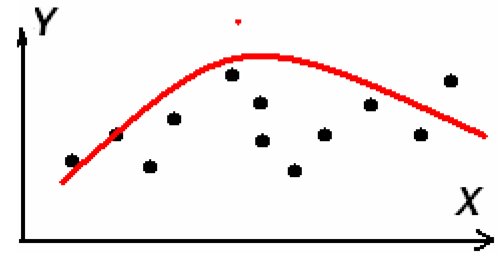
Пример. Аппроксимация временного ряда. Какая формула лучшая?



Эксперимент

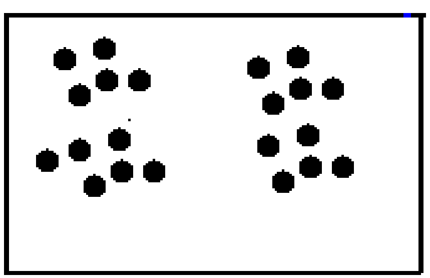


Линейная модель

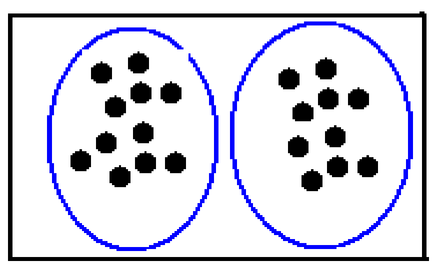


Квадратичная модель

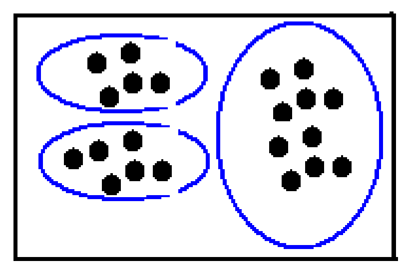
Пример. Группировка данных. Какая структура лучшая?



Эксперимент



Два кластера



Три кластера

Комментарий к МГУА

Что значит «групповой учет аргументов»?

На каждой итерации происходит селекция наиболее перспективных сочетаний переменных

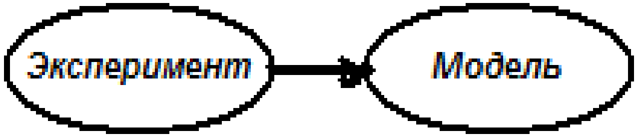
- В комбинаторном варианте МГУА в каждом ряду рассматриваются все комбинации переменных (то есть, все модели данного уровня сложности). Лучшая комбинация в ряду определяет условие перехода от текущего ряда к следующему

- В других вариантах МГУА в каждом ряду рассматривается не только лучшая модель, но и отбираются лучшие сочетания переменных. Они и задают модели следующего ряда

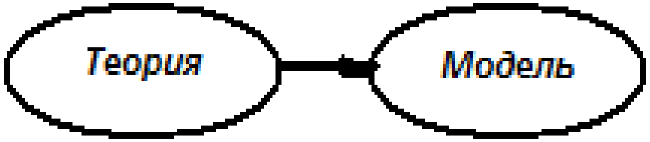
Комментарий к МГУА

В чем состоит «индуктивность» ?

Индукция = от частного к общему



Дедукция = из общего к частному



Индуктивность в нашем случае это получение моделей возрастающей сложности из ограниченного числа наблюдений

Примеры моделей:

а) в случае одной входной переменной это может быть - прямая, парабола, полином 3-го порядка, и т.д., это может быть ряд Фурье с одной, двумя, тремя гармониками, и т.д.

б) в случае многих входных переменных это могут быть линейные и нелинейные функции двух и более переменных

Напомним ограничение: класс моделей заранее определен

Комментарий к МГУА

В чем состоит «самоорганизация» ?

Самоорганизация системы – это изменение ее структуры/параметров под влиянием внешних условий

Самоорганизация у нас состоит в том, что модель меняется от простой к сложной в процессе автоматического перебора моделей, пока она не достигнет оптимальной сложности

Внешние условия в нашем случае – это данные наблюдений
Они отражают как поведение объекта, так и шум

Тогда очевидно, что:
- Простая модель не реагирует на шум, но плохо отражает объект
- Сложная модель отражает объект, но чувствительна к шуму

Есть некоторый оптимум, который достигается в процессе перебора

Содержание

Материалы

Литература

1. Ивахненко А.Г. *Индуктивный метод самоорганизации сложных систем*. Киев, «Наукова думка», 1982, 296 сс.
2. Ивахненко А.Г., Карпинский А.М. *Самоорганизация моделей на ЭВМ в терминах общей теории связи*. Автоматика, 1982, N_4, сс. 7-26
3. Ивахненко А.Г., Степашко В.С. *Помехоустойчивость моделирования*. Киев, «Наукова думка», 1985, 216 сс.
4. Степашко В.С. *Метод критических дисперсий как аналитический аппарат теории индуктивного моделирования*. ПУИ,2008,N_2,сс.8-26

Журналы

- Проблемы управления и информатики (ПУИ). Изд. НАН Украины, Киев
- Сборники трудов МНУЦ ИТС, Изд. НАН и МОН Украины, Киев
- Сборники трудов ВЦ РАН, Москва

Материалы

Международные конференции, семинары

- Львов, Украина, конференция, 2002
- Киев, Украина, семинар, 2005
- Прага, Чехия, семинар, 2007
- Киев, Украина, конференция, 2008
- Криница, Польша, семинар, 2009
- Евпатория, Украина, конференция, 2010
- Киев, Украина, семинар, 2011
- Киев, Украина, конференция, 2012

Будущая конференция по Индуктивному Моделированию будет посвящена 100-летию со Дня Рождения акад. А.Г. Ивахненко

Место проведения: Киев, Украина

Время проведения: лето 2013 года

Публикации: сборник «Индуктивное Моделирование. Теория и практика»

возможное изд-во CRC, Шпрингер, и т.п.

Материалы

Ежегодная летняя школа по ИМ

Школу проводит МНУЦ (отдел ИМ), июль, г. Жукин (около Киева)
Участие и проживание – бесплатно (с 2013 г за символическую плату)

Участники:

Приглашенные лектора (3-5)

Молодые и опытные исследователи

Докторанты (претенденты на степень Dr. Sci.)

Аспиранты (претенденты на степень Ph.D.)

Магистранты

Публикации: Труды МНУЦ НАН Украины

Языки: Русский, Украинский

Метод группового учета аргументов
(часть 1)

Конец лекции

Лекция: L3-2_part1

Версия: 3.2 (1.11.2012)