УДК 519.65

А.Б. Иващенко (ассист.)**,**
В.Н. Беловодский (канд.техн.наук, доц.)
Донецкий национальный технический университет
alesya_iva@list.ru, belovodskiy@cs.dgtu.donetsk.ua

АНАЛИЗ МЕТОДИКИ ЭГЛАЙСА АППРОКСИМАЦИИ ТАБЛИЧНЫХ ДАННЫХ

Изложена перспективная методика аппроксимации данных, позволяющая эффективно восстанавливать математическую модель по экспериментальным данным. Приводятся особенности программной реализации алгоритма, а также результаты серии вычислительных экспериментов. Предложены варианты совершенствования методики.

аппроксимация, реконструкция уравнений, регрессионная модель, зависимость, банк функций, элиминация, синтез уравнения

Введение

Задача восстановления зависимостей по эмпирическим данным была и, вероятно, всегда будет одной из центральных в прикладном анализе. Эта задача является математической интерпретацией одной из основных проблем естествознания: как найти существующую закономерность по разрозненным фактам.

В наиболее простой постановке, проблема состоит в восстановлении функции по ее значениям в некоторых точках. Необходимо сформулировать общие принципы восстановления функциональных зависимостей, а затем в соответствии с ними построить алгоритмы восстановления.

Обычно, когда ищется общий принцип, предназначенный для решения широкого класса задач, выделяется наиболее простая, базовая задача. Эта задача подвергается тщательному теоретическому анализу, а полученная для нее схема решения распространяется на все задачи данного класса.

Существуют различные варианты конкретизации постановки этой задачи. Они основаны на разных моделях «измерения с ошибками». Однако каковы бы ни были эти модели, изучение базовой задачи приводит к использованию следующего классического принципа восстановления функциональных зависимостей по эмпирическим данным:

– из допустимого множества функций надлежит выбрать такую, которая наилучшим образом приближается к совокупности имеющихся эмпирических данных.

Этот принцип является достаточно общим. Он оставляет свободу в толковании того, что является мерой качества приближения функции к совокупности эмпирических данных. Существуют различные способы ее определения, например, величина среднеквадратичного уклонения значений

функции, величина среднего уклонения, величина наибольшего уклонения и т. д. Каждый способ определения меры порождает и свой подход к восстановлению зависимостей (метод наименьших квадратов, наименьших модулей и т. д.). Однако во всех случаях принцип отыскания решения – поиск функции, наилучшим образом приближающейся к эмпирическим данным, – остается неизменным.

Однако существует и другой, неклассический принцип восстановления зависимостей, хоть и не менее популярный:

– из допустимого множества функций необходимо выбрать такую, которая удовлетворяет определенному соотношению между величиной, характеризующей качество приближения функции к заданной совокупности эмпирических данных, и величиной, характеризующей «сложность» приближающей функции.

Этот принцип нуждается в пояснении. Дело в том, что с увеличением «сложности» приближающей функции удается получать все лучшие и лучшие приближения к имеющимся эмпирическим данным и даже, быть может, построить функцию, проходящую через все ее заданные точки.

Сформулированный принцип, в отличие от классического, утверждает, что не следует добиваться приближения к эмпирическим данным любыми средствами (т.е. за счет выбора чрезмерно «сложной» приближающей функции). Для каждого объема эмпирических данных существует свое соотношение между «сложностью» приближающей функции и достигнутым качеством приближения, при соблюдении которого восстановленная зависимость, в некотором смысле, лучше, т.е., другими словами, – адекватно описывает искомую. Дело в том, что стремление к дальнейшему приближению к эмпирическим данным за счет «усложнения» приближающей функции может привести к высокому качеству аппроксимации именно этих данных, однако, в целом, – невысокому качеству соответствия искомой функции.

Неклассический принцип восстановления отражает попытку учесть то обстоятельство, что зависимость восстанавливается в условиях ограниченного объема эмпирических данных.

Всякий раз, когда возникает проблема выбора функциональной зависимости, рассматривается одна и та же схема: среди множества возможных зависимостей выбрать такую, которая наилучшим образом удовлетворяет заданному критерию качества.

Общая постановка проблемы

Для проведения процедуры восстановления или идентификации модели по экспериментальным данным имеются различные подходы, которые базируются на принципиально разных методах анализа табличной информации.

Анализ литературных источников и Интернет-ресурсов позволяет выделить ряд основных наиболее развитых подходов в этом направлении, т.е., в так называемом, «интеллектуальном» анализе данных. К основным и наиболее популярным на сегодняшний день можно отнести:

1) регрессионный анализ;
2) нейронные сети;
3) системы рассуждений на основе аналогичных случаев и деревья решений;
4) эволюционные и генетические алгоритмы;
5) методы группового учета аргументов (МГУА).

Каждый из них имеет свои особенности, достоинства и недостатки. За последние десятилетия этим направлениям посвящено большое количество исследований, предложен ряд вариантов их развития и совершенствования. Но, следует отметить, что какой бы из описанных подходов не был принят при анализе данных, для подбора удачной модели экспериментатор, в конечном счете, должен руководствоваться субъективными предположениями о характере зависимости между входными величинами, своим опытом и интуицией. Поэтому качество решения конкретной задачи, полученного с помощью того или иного метода, и степень адекватности построенной модели, в значительной степени, определяются индивидуальными качествами пользователя.

В этой связи, в ходе выполнения информационного поиска при анализе литературных и интернет источников, среди обилия литературы, посвященной упомянутым методам, обратила на себя внимание методика синтеза аппроксимирующих функций латвийского ученого В.О.Эглайса [1, 2]. Предложенный им подход заслуживает уже внимания, хотя бы потому, что позволяет находить «золотую середину» в стремлении подобрать одновременно «красивую» и «точную» модель, даже в случае модели типа «черный ящик», формировать компромиссную модель, которая, в некотором здравом смысле, одновременно удовлетворяет двум взаимоисключающим требованиям качества: критерию точности и показателю эффективности.

Ввиду отсутствия в современной литературе достаточной информации, посвященной этому методу, ниже приведем его краткое описание, опишем особенности его собственной программной реализации, а также, результаты проведенных вычислительных экспериментов, направленных, как на изучение, так и дальнейшее развитие метода.

Постановка задачи

Пусть некоторая информация представлена в виде таблицы 1.

Таблица 1. Исходные данные

№п/п	x_1	x_2	…	x_n	y
1					
2					
…					
k					

Здесь $x_1,\ldots,x_n$ – параметры объекта (или факторы, независимые переменные); y – отклик (зависимая переменная); k – число экспериментов.

Требуется синтезировать соответствующее уравнение регрессии в виде:

$$y = A_0 + F(A_j, X_i), \tag{1}$$

где A_0 – свободный член (постоянная), A_j – набор коэффициентов уравнения регрессии, X_i – набор параметров объекта.

Описание метода

Суть метода заключается в следующем. Аппроксимирующая функция строится в классе степенных разложений и процесс ее нахождения включает следующие этапы. Первоначально, по заданной степени одночлена, формируется множество базисных функций, затем, из этого множества выбирается заданное количество перспективных функций, после чего, направленным перебором, который В.Эглайс называет элиминацией, осуществляется окончательный подбор функции адекватной исходной информации. Остановимся более подробно на описании указанных этапов.

1. Формирование банка элементарных функций. Создается ограниченный банк элементарных функций $\Phi:\{\varphi_1,\varphi_2,\ldots,\varphi_l\}$. Здесь $\varphi_1,\varphi_2,\ldots,\varphi_l$ – функции параметров объекта, не содержащие неопределенных коэффициентов. Такой банк дает возможность синтезировать большое число разнообразных уравнений в виде

$$y = A_0 + \sum_{i=1}^{m} A_i f_i(\bar{x}), \tag{2}$$

где $\{f_i(\bar{x})\}$ – набор элементарных функций из банка Φ с коэффициентами, которые вычисляются по методу наименьших квадратов.

В [1] рекомендуется создавать банк Φ с таким расчетом, чтобы элементарные функции, входящие в него, по возможности меньше дублировали друг друга и, в некотором смысле, соответствовали классу исследуемого объекта. Для описания достаточно гладких многомерных зависимостей предлагается использовать банк элементарных функций вида:

$$\varphi_k(\bar{x}) = \prod_{i=1}^{n} x_i^{\alpha_{k,i}}, \tag{3}$$

где n – число параметров объекта, $\alpha_{k,i}$ –наборы целых чисел, каждый из которых определяет свою элементарную функцию банка.

Чтобы ограничить, в разумных пределах, число функций банка, вводится своего рода ограничитель в виде условия:

$$\sum_{i=1}^{n} \left| \alpha_{k,i} \right| \le K_n , \tag{4}$$

где K_n – максимально возможная степень для каждого набора параметров.

В работе [1] , в частности, рекомендуется выбирать значение K_n так, чтобы общее число функций банка не превышало 400. Следует отметить, что с ростом вычислительных мощностей современных ЭЦВМ и развитием скоростей обработки данных эта граница, при необходимости, может быть значительно увеличена. Обычно, для выбора K_n достаточно множества $\{1,2,3,4,5\}$.

2. Отбор из банка «перспективных» функций. После того, как банк функций сформирован, производится отбор перспективных функций. Для этого, с помощью метода наименьших квадратов, для каждой базовой функции вычисляются коэффициенты элементарного уравнения регрессии:

$$y_i = A_i + B_i \varphi_i(\bar{x}) , \tag{5}$$

а также суммарные квадратичные отклонения:

$$s_i = \sum_{j=1}^{k} (A_i + B_i \varphi_i(\bar{x}_j) - y_j)^2 . \tag{6}$$

По их величинам отбирается заданное число функций, обеспечивающих наименьшие значения s_i, которые и формируют множество перспективных функций. Далее, строится функция:

$$y_j^* = A_0 + \sum_{i=1}^{p} A_i f_i(\bar{x}_j) , \tag{7}$$

где p – назначенное количество перспективных функций, $f_i(\bar{x})$ – отобранные перспективные функции, y_j^* – значение отклика, рассчитанное по отобранным перспективным функциям; A_0, A_i – коэффициенты, определенные по методу наименьших квадратов из требования минимума величины:

$$\Delta = \sum_{j=1}^{k} (y_j^* - y_j)^2 . \tag{8}$$

3. Элиминация и синтез уравнения. После отбора перспективных функций проводится элиминация (исключение) наименее существенных элементарных функций из набора перспективных. Это необходимо из следующих соображений. Естественно считать, что среди функций, отобранных в регрессионную модель (7), только часть действительно необходима в формируемом уравнении регрессии. Остальные из этого уравнения можно безболезненно исключить.

Пусть отобрано всего p функций. Тогда имеется p вариантов первого исключения одной функции из уравнения регрессии. По методу наименьших

квадратов проверяются все варианты, и исключается функция, дающая минимальное σ_i по формуле:

$$\sigma = \sqrt{\frac{\Delta}{k-(p+1)}}, \tag{9}$$

где k – число точек в таблице, а $p+1$ – количество параметров модели (число коэффициентов при функциях плюс свободный член).

Далее, после исключения одной из функций, переменной p присваивают значение $(p-1)$ и следующий шаг процесса элиминации повторяется аналогичным образом.

То есть, для определения наименее существенной функции на каждом шаге элиминации проверяются все возможные варианты исключения, и исключается функция, без которой уравнение регрессии в форме (7) дает минимум среднеквадратичного отклонения по формуле (9). Подобным образом последовательно исключаются и все остальные отобранные функции. Выбор окончательного варианта уравнения регрессии проводится по диаграмме элиминации $\sigma = \sigma(p)$, показанной на рисунке 1. Такая схема исключения функций удобна для визуальной оценки степени «важности» исключаемых функций.

Пока из уравнения регрессии исключаются несущественные функции, величина σ меняется мало. Когда же в наличии остаются только существенные функции, исключение любой из них заметно увеличивает среднеквадратичное отклонение. Таким образом, излом в диаграмме элиминации свидетельствует о получении наиболее предпочтительного уравнения регрессии в смысле точности σ (Эглайс называет эту характеристику абсолютной точностью) и надежности, которая определяется числом коэффициентов в уравнении регрессии [1].

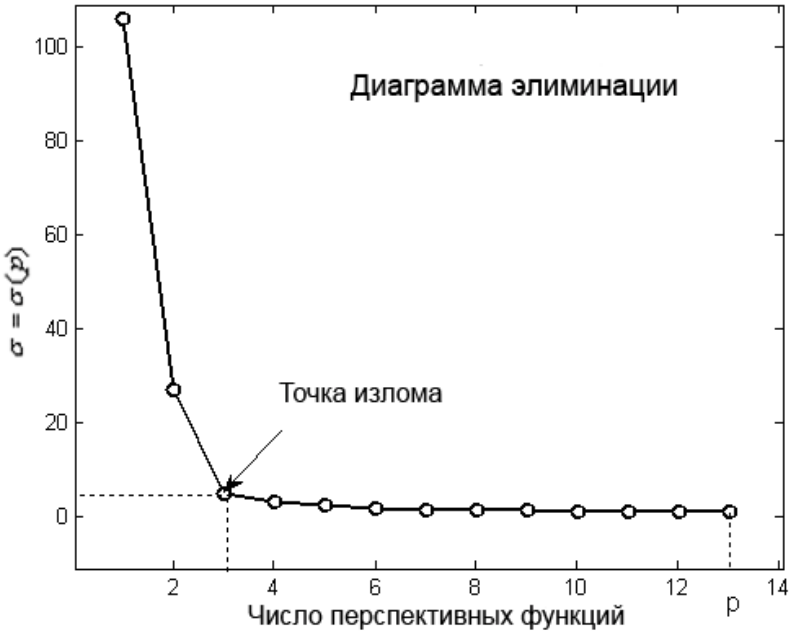


Рисунок 1 – Диаграмма элиминации (зависимость σ от p)

Точность уравнения регрессии можно также характеризовать относительным аналогом среднего квадрата отклонений σ – коэффициентом корреляции c (или, как она названа у Эглайса, относительная точность):

$$c = \left(1 - \frac{\sigma}{\sigma_0}\right) \cdot 100\%, \tag{10}$$

где σ_0 – среднеквадратичное отклонение откликов от среднего:

$$\sigma_0 = \sqrt{\frac{\sum_{i=1}^{k}\left(y_i - \frac{1}{k}\sum_{j=1}^{k} y_j\right)^2}{k-1}}. \tag{11}$$

Использование коэффициента c особенно оправдано в случаях, когда имеется необходимость наглядно продемонстрировать, визуально оценить или сравнить корреляционную зависимость между экспериментальным откликом и откликом, полученным с помощью синтезируемой модели, при варьировании количества членов аппроксимирующей функции.

То есть, в этом случае диаграмма элиминации $c = c(p)$ (приведена на рисунке 2) имеет более универсальный масштаб и позволяет пользователю установить, сколько функций необходимо сохранить в синтезированном уравнении, чтобы получить модель с достаточной или требуемой точностью.

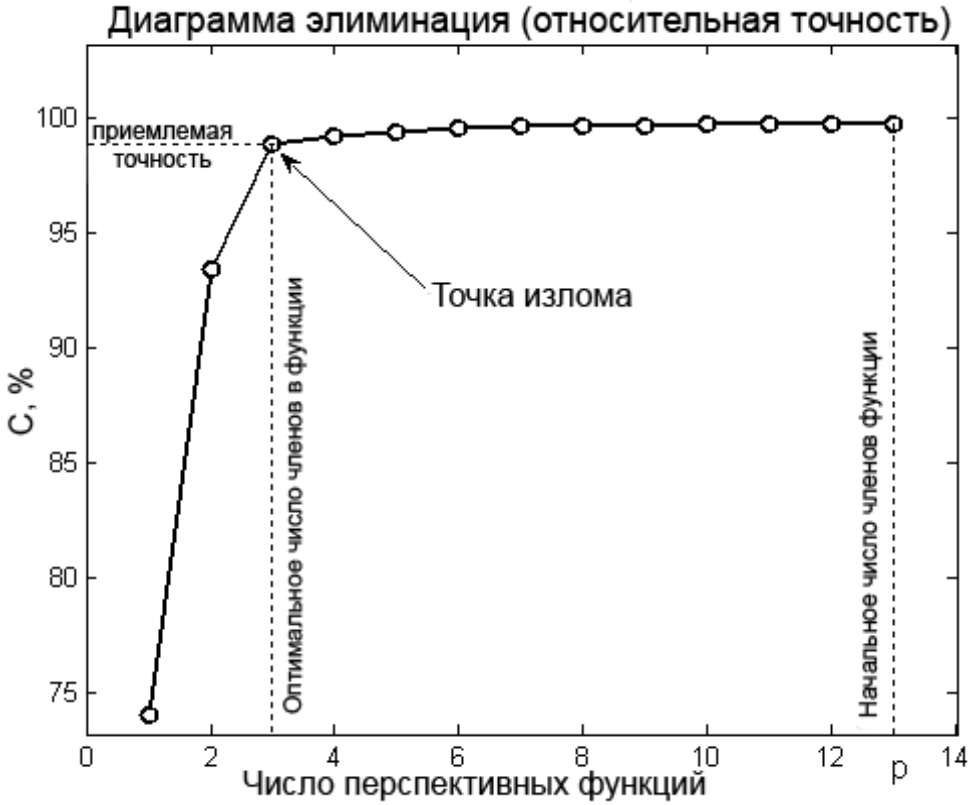


Рисунок 2 – Диаграмма элиминации (относительная точность модели)

Предполагается, что определение точки прекращения процесса элиминации (определение оптимального числа функций в регрессионной

модели) производится пользователем визуально, путем оценки одной из указанных выше диаграмм [1].

Особенности программной реализации

Описанная методика синтеза регрессионных уравнений была реализована в среде пакета Matlab. Основной задачей при проектировании программного обеспечения был анализ особенностей программной реализации отдельных этапов и проверка эффективности метода.

Разработка алгоритма генерации банка начальных функций была осуществлена с использованием комбинаторных соображений на базе перебора вариантов размещений с повторениями из k объектов по n местам. Вкратце идея реализации заключается в следующем. Пусть имеются n различных независимых параметра $x_1, x_2, \ldots, x_n$. Тогда формирование банка неповторяющихся функций вида (3), можно обеспечить путем перебора комбинаций при условии, что задана максимально возможная степень k (для выполнения условия (4)). Первоначально, перебираются все варианты размещений с повторениями из n по $2k+1$. Число $2k+1$ есть сумма количества возможных положительных и отрицательных степеней плюс нулевая степень. Тогда, результатом генерации всех возможных комбинаций размещений с повторениями будет массив:

$$\underbrace{\begin{pmatrix} 1 & 1 & \ldots & 1 & 1 \\ 1 & 1 & \ldots & 1 & 2 \\ & \ldots & & \ldots & \\ 1 & 1 & \ldots & & 2k+1 \\ 1 & 1 & \ldots & 2 & 1 \\ 1 & 1 & \ldots & 2 & 2 \\ & \ldots & & \ldots & \\ 1 & 1 & \ldots & 2 & 2k+1 \\ 1 & 1 & \ldots & 3 & 1 \\ & \ldots & & \ldots & \\ 2k+1 & 2k+1 & \ldots & 2k+1 & 2k \\ 2k+1 & 2k+1 & \ldots & 2k+1 & 2k+1 \end{pmatrix}}_{\text{Всего } n \text{ мест}} \qquad (12)$$

Используя формулу $\widetilde{A}_n^m = n^m$ для вычисления числа размещений с повторениями [3], составленных из n элементов по m, нетрудно заметить, что в нашем случае имеем $(2k+1)^n$ вариантов. Следует отметить, что на данный момент в базовом пакете MatLab отсутствуют специальные функции для генерации вариантов размещений с повторениями, как и в других математических средах моделирования. Тем не менее, на сайте корпорации

MathWorks™ [4], в сообществе разработчиков в среде MatLab, можно скачать готовую m-функцию *combinator* для генерации различных множеств и переборов комбинаций [5]. В нашем случае ее следует вызвать со следующими параметрами:

*combinator(2*k+1, n, 'p', 'r'),*

где первый параметр функции – число элементов множества $\{1, 2, \ldots, 2 \cdot k, 2 \cdot k + 1\}$, второй параметр – количество позиций (мест) для формирования необходимых комбинаций (перестановок или размещений), два последних параметра означают вычисление «размещения с повторениями» (permutations with repetition).

Поскольку, в процессе построения регрессионного уравнения, допускается возможность оперирования и с отрицательными степенями, далее, необходимо провести «центрирование» массива, то есть из каждого элемента полученного массива степеней вычесть $(k+1)$. После центрирования получим массив следующих наборов (строк) степеней:

$$\begin{pmatrix}
-k & -k & \ldots & -k & -k \\
-k & -k & \ldots & -k & -k+1 \\
 & \ldots & & \ldots & \\
-k & -k & \ldots & -k & k \\
-k & -k & \ldots & -k+1 & -k \\
-k & -k & \ldots & -k+1 & -k+1 \\
 & \ldots & & \ldots & \\
-k & -k & \ldots & -k+1 & k \\
-k & -k & \ldots & -k+2 & -k \\
 & \ldots & & \ldots & \\
k & k & \ldots & k & k-1 \\
k & k & \ldots & k & k
\end{pmatrix} \qquad (13)$$

В принципе, генерацию степеней на этом можно было бы и закончить. Однако, если точно следовать методике, то необходимо, кроме этого, перебрать строки полученного массива и оставить лишь те, сумма модулей элементов (то есть степеней в будущей функции) которых не превышает k – максимальное значение «общей» степени функции (для удовлетворения условия неравенства (4)). Обратим внимание, что среди сформированных таким способом наборов степеней будет и тот, в котором степени всех параметров равны нулю, то есть строка степеней $(0 \;\; 0 \;\; \ldots \;\; 0 \;\; 0)$. Такая функция будет постоянной для любой точки таблицы и будет равна единице. Учитывая, что мы ищем уравнение в виде (1), такая функция будет дублировать свободный член A_0, который изначально уже предусмотрен в

синтезируемом уравнении регрессии (2). Поэтому, для устранения нежелательного повтора функций, набор, в котором все степени равны нулю, тоже исключается.

Так, например, для числа параметров $n = 2$ и максимально возможной степени $k = 3$ получим банк из 24 элементарных функций. Для наглядности промежуточные результаты алгоритма и вид сгенерированных функций для данного примера приведен ниже (таблица 2).

Таблица 2. Пример генерации базовых функций для n=2, k=3

№п/п	Шаг 1	Шаг 2	Шаг 3	Шаг 4
Описа-ние	Генерация размещений из $2k+1$ по n	«Центриро-вание» массива	Отбор строк, сумма модулей элементов которых не выше k	Формирование функций f по сгенерированным наборам степеней alfa
Опера-ции	a=combinator (2*k+1,n,'p','r')	a=a-(k+1)	for i=1:49 if bs(a(i,1))+abs(a(i,2))<= k alfa(j)=a(i); j=j+1; end;	for i=1:49 f(i)=x(1)^alfa(i,1)*x(2)^alf a(i,2)
Результат	a=	a=	alfa=	f=
1	1 1	-3 -3		
2	1 2	-3 -2		
3	1 3	-3 -1		
4	1 4	-3 0	-3 0	x_1^{-3}
5	1 5	-3 1		
6	1 6	-3 2		
7	1 7	-3 3		
8	2 1	-2 -3		
9	2 2	-2 -2		
10	2 3	-2 -1	-2 -1	$x_1^{-2} x_2^{-1}$
11	2 4	-2 0	-2 0	x_1^{-2}
12	2 5	-2 1	-2 1	$x_1^{-2} x_2^{1}$
13	2 6	-2 2		
14	2 7	-2 3		
15	3 1	-1 -3		
16	3 2	-1 -2	-1 -2	$x_1^{-1} x_2^{-2}$
17	3 3	-1 -1	-1 -1	$x_1^{-1} x_2^{-1}$
18	3 4	-1 0	-1 0	x_1^{-1}
19	3 5	-1 1	-1 1	$x_1^{-1} x_2^{1}$
20	3 6	-1 2	-1 2	$x_1^{-1} x_2^{2}$
21	3 7	-1 3		
22	4 1	0 -3	0 -3	x_2^{-3}
23	4 2	0 -2	0 -2	x_2^{-2}
24	4 3	0 -1	0 -1	x_2^{-1}
25	4 4	0 0		
26	4 5	0 1	0 1	x_2^{1}
27	4 6	0 2	0 2	x_2^{2}
28	4 7	0 3	0 3	x_2^{3}
29	5 1	1 -3		
30	5 2	1 -2	1 -2	$x_1^{1} x_2^{-2}$
31	5 3	1 -1	1 -1	$x_1^{1} x_2^{-1}$

Таблица 2. (продолжение)

32	5	4	1	0	1 0	x_1^1
33	5	5	1	1	1 1	$x_1^1 x_2^1$
34	5	6	1	2	1 2	$x_1^1 x_2^2$
35	5	7	1	3		
36	6	1	2	-3		
37	6	2	2	-2		
38	6	3	2	-1	2 -1	$x_1^2 x_2^{-1}$
39	6	4	2	0	2 0	x_1^2
40	6	5	2	1	2 1	$x_1^2 x_2^1$
41	6	6	2	2		
42	6	7	2	3		
43	7	1	3	-3		
44	7	2	3	-2		
45	7	3	3	-1		
46	7	4	3	0	1 0	x_1^3
47	7	5	3	1		
48	7	6	3	2		
49	7	7	3	3		

Таким способом, при $n = 2$ и $k = 3$ получается следующий банк функций:

$$\Phi = \left\{ \begin{array}{l} x_1^{-3}, x_1^{-2}x_2^{-1}, x_1^{-2}, x_1^{-2}x_2^{1}, \\ x_1^{-1}x_2^{-2}, x_1^{-1}x_2^{-1}, x_1^{-1}, \\ x_1^{-1}x_2^{1}, x_1^{-1}x_2^{2}, x_2^{-3}, x_2^{-2}, \\ x_2^{-1}, x_2^{1}, x_2^{2}, x_2^{3}, x_1^{1}x_2^{-2}, \\ x_1^{1}x_2^{-1}, x_1^{1}, x_1^{1}x_2^{1}, x_1^{1}x_2^{2}, \\ x_1^{2}x_2^{-1}, x_1^{2}, x_1^{2}x_2^{1}, x_1^{3} \end{array} \right\}. \qquad (13)$$

Обратим внимание, что с целью исключения деления на ноль в процессе вычислений, перед формированием банка функций рекомендуется предусмотреть линейную нормировку аргументов:

$$x^* = \frac{(x^*_{\max} - x^*_{\min})x + (x^*_{\min} x_{\max} - x^*_{\max} x_{\min})}{x_{\max} - x_{\min}} \qquad (14)$$

где x – натуральное, то есть исходное, значение независимой переменной, x^* – ее нормированное значение, $x_{\min}$, $x_{\max}$ – соответственно, минимальное и максимальное значение переменной x; $[x^*_{\min}, \quad x^*_{\max}]$ – диапазон изменения ее нормированного значения. Здесь предполагается, что $x^*_{\min} > 0, \quad x^*_{\max} > 0$.

Избежать ситуации деления на ноль можно также другим способом. Нормировку аргументов можно не проводить, предварительно проверяя входные данные на наличие нулевых элементов и запоминая каким-либо способом аргументы-столбцы, которые их содержат. Затем, описанным выше способом формировать массив наборов степеней и уже из него удалять те наборы, в которых для помеченных переменных присутствует отрицательная степень.

Вычислительные эксперименты

Для тестирования и верификации изложенного алгоритма использовались тестовые наборы «псевдоэкспериментальных» данных с различным числом точек и различным числом входных параметров. Под этим понималось искусственное задание значений параметров и соответствующих им откликов, аналитический вид зависимости между которыми задан заранее.

С целью наглядного представления эффективности работы программы приведем некоторые тестовые примеры.

Пример 1. Экспериментальные данные заданы функцией

$$y = x^2 + 2x + 5. \qquad (15)$$

Численные результаты построения аппроксимирующей зависимости приведены в таблице 3, в которой представлены входные данные (фактор x и отклик y), а также отклик y_exp, полученный в результате синтеза аппроксимирующей функции. Восстановление аппроксимирующей функции для этого примера производилось при условии, что $k = 2, \quad p = 3$.

Таблица 3. Входные данные и результаты аппроксимации для примера 1

№п/п	x	y	y_exp
1	1	8	8
2	2	15	15
3	3	26	26
4	4	41	41
5	5	60	60
6	6	83	83
7	7	110	110
8	8	141	141
9	9	176	176
10	10	215	215

На рисунке 3 представлена диаграмма элиминации для данного примера. Излом в точке $p^* = 2$ соответствует оптимальному числу членов в синтезируемой функции.

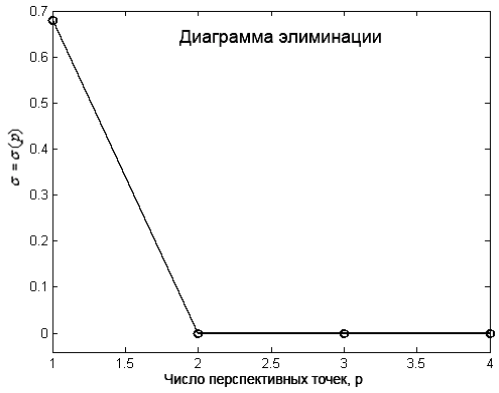


Рисунок 3 – Диаграмма элиминации (для примера 1)

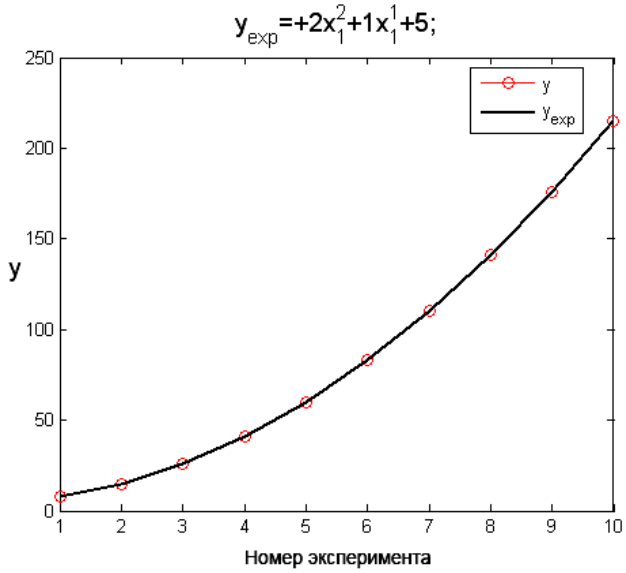


Рисунок 4 – График аппроксимирующей функции и значения отклика в экспериментальных точках для примера 1

Как видно по таблице 3 и из рисунка 4, аппроксимирующая функция, построенная при помощи исследуемой методики, восстановлена правильно.

Пример 2. Экспериментальные данные заданы функцией

$$y = 3x_1^3 + 2x_2^2 + x_1^{-1}x_2^2 + 6x_1x_2 + 7x_2 + 5. \qquad (16)$$

Создаем исходный файл, содержащий табличные данные, описываемые зависимостью (16). Задаем параметры синтеза уравнения $k = 3,\quad p = 15$. По диаграммам элиминации делаем вывод, что при пяти членах в синтезируемой функции на диаграмме происходит излом (рисунок 5, 6).

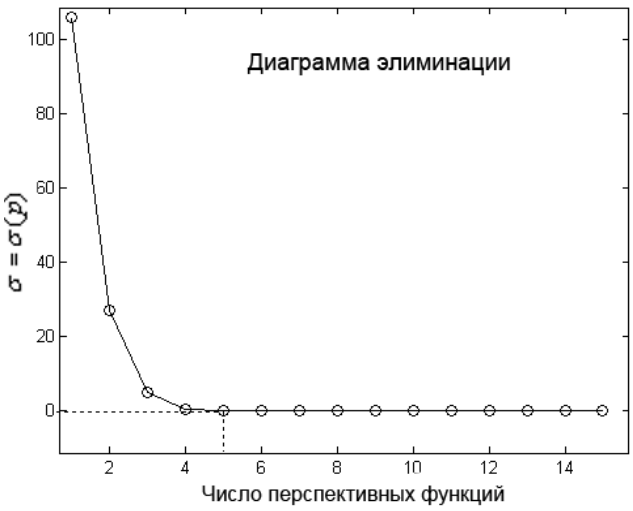


Рисунок 5 – Диаграмма элиминации для примера 2 (абсолютная точность)

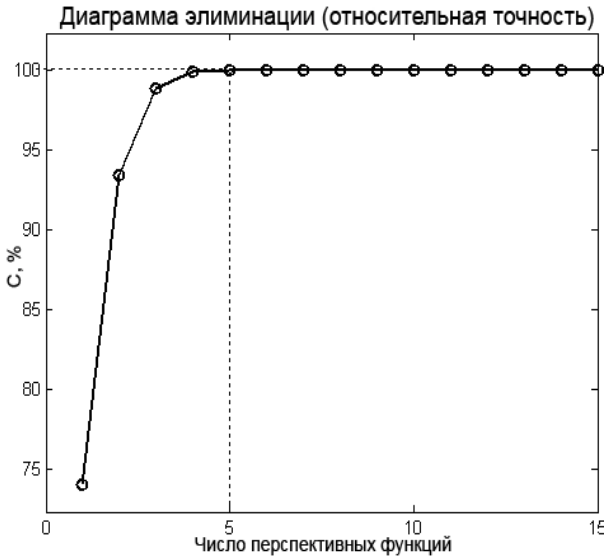


Рисунок 6 – Диаграмма элиминации для примера 2 (относительная точность)

Исходные данные и результаты восстановления откликов представлены в таблице 4.

Таблица 4. Входные данные и результаты аппроксимации для примера 2

№№	x1	x2	y	y_exp
1	2	1	50,5	50,5
2	1	2	46	46
3	2	3	108,5	108,5
4	3	1	113,33	113,33
5	1	2	46	46
6	5	3	510,8	510,8
7	6	1	698,17	698,17
8	7	6	1405,1	1405,1
9	6	7	1060,2	1060,2
10	2	1	50,5	50,5
11	5	5	620	620
12	2	0	29	29
13	-1	1	4	4
14	3	5	269,33	269,33
15	6	1	698,17	698,17
16	4	1	230,25	230,25
17	2	5	186,5	186,5
18	-1	5	32	32
19	2	6	233	233
20	3	7	375,33	375,33
21	1	6	194	194
22	2	2	77	77
23	3	5	269,33	269,33
24	7	0	1034	1034
25	6	-1	612,17	612,17
26	7	3	1200,3	1200,3
27	4	6	464	464

Таблица 4.(продолжение)

28	3	7	375,33	375,33
29	4	4	357	357
30	5	5	620	620
31	6	4	859,67	859,67
32	7	2	1140,6	1140,6
33	6	5	922,17	922,17
34	2	1	50,5	50,5
35	5	1	419,2	419,2
36	7	3	1200,3	1200,3
37	6	6	989	989
38	7	7	1482	1482
39	4	4	357	357
40	3	5	269,33	269,33
41	4	4	357	357
42	5	2	462,8	462,8
43	6	5	922,17	922,17
44	6	1	698,17	698,17
45	4	1	230,25	230,25
46	2	4	145	145
47	-2	5	-6,5	-6,5
48	2	4	145	145
49	2	2	77	77
50	1	5	148	148
51	2	1	50,5	50,5
52	3	8	435,33	435,33

Такой характер поведения элиминации приводит к окончательному варианту аппроксимирующей функции, полностью совпадающим с изначально заданной в тестовом наборе. На рисунке 7 представлен график аппроксимирующей функции как функции номера экспериментальной точки.

$$y_{exp}=+3x_1^3+6x_1^1x_2^1+1x_1^{-1}x_2^{-2}+2x_2^2+7x_2^1+5;$$

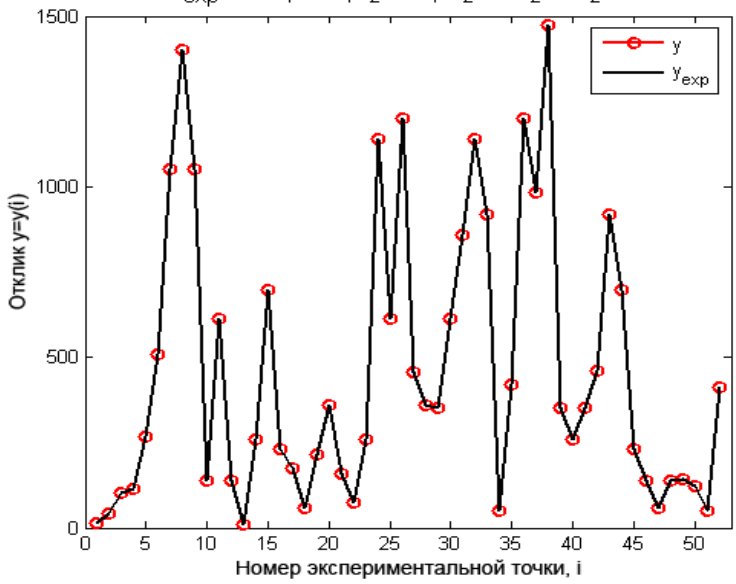


Рисунок 7 – Результаты построения аппроксимирующей зависимости для примера 2

Пример 3. Экспериментальные данные описываются функцией $y = \sin(x)$.

Соответствующие табличные данные и результаты аппроксимации (значения откликов, восстановленных по аппроксимирующей функции) представлены в таблице 5.

В этом случае были выбраны следующие параметры синтеза уравнения $k = 10,\ \ p = 10$. После отбора p перспективных функций было получено уравнение в виде:

$$\begin{aligned} y\_exp = &+0.9996x_1^1 + \\ &+0.0019623x_1^2 - 0.1709x_1^3 + \\ &+0.0048954x_1^4 + 0.0049728x_1^5 + \\ &+0.0014367x_1^6 - 0.00058386x_1^7 + \\ &+6.2042e-005x_1^8 - 2.2139e-006x_1^9 + \\ &+1.1862e-009x_1^{10} + 2.0878e-005; \end{aligned} \qquad (17)$$

После построения диаграмм элиминации (рисунки 8, 9) было принято решение об оставлении в аппроксимирующем выражении пяти перспективных функций.

Таблица 5. Исходные данные и результаты для примера 3

№п/п	x	y	y_exp
1	0,017453	0,017452	0,020252
2	0,034907	0,034899	0,037583
3	0,05236	0,052336	0,054903
4	0,069813	0,069756	0,072208
5	0,087266	0,087156	0,089492
6	0,10472	0,10453	0,10675
7	0,12217	0,12187	0,12398
8	0,13963	0,13917	0,14117
9	0,15708	0,15643	0,15831
10	0,17453	0,17365	0,17541
11	0,19199	0,19081	0,19246
12	0,20944	0,20791	0,20945
13	0,22689	0,22495	0,22638
14	0,24435	0,24192	0,24324
15	0,2618	0,25882	0,26003
16	0,27925	0,27564	0,27674
17	0,29671	0,29237	0,29337
18	0,31416	0,30902	0,30991
19	0,33161	0,32557	0,32636
20	0,34907	0,34202	0,34271
21	0,36652	0,35837	0,35895
22	0,38397	0,37461	0,37509

Таблица 5. (продолжение)

23	0,40143	0,39073	0,39112
24	0,41888	0,40674	0,40703
25	0,43633	0,42262	0,42281
26	0,45379	0,43837	0,43847
27	0,47124	0,45399	0,454
28	0,48869	0,46947	0,46939
29	0,50615	0,48481	0,48464
30	0,5236	0,5	0,49975
31	0,54105	0,51504	0,5147
	…	…	…
176	3,0718	0,069756	0,070812
177	3,0892	0,052336	0,053322
178	3,1067	0,034899	0,035815
179	3,1241	0,017452	0,018295
180	3,1416	0,0	0,000768
181	3,159	-0,01745	-0,01676
182	3,1765	-0,0349	-0,03429
	…	…	…
353	6,161	-0,12187	-0,12078
354	6,1785	-0,10453	-0,10255
355	6,1959	-0,08716	-0,08419
356	6,2134	-0,06976	-0,0657
357	6,2308	-0,05234	-0,04706
358	6,2483	-0,0349	-0,02829
359	6,2657	-0,01745	-0,00938

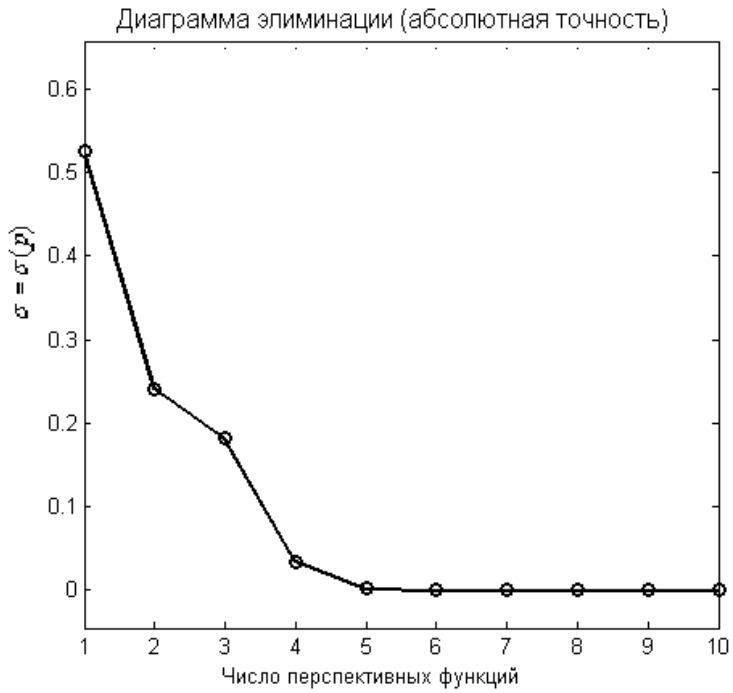


Рисунок 8 – Диаграмма элиминации (абсолютная точность)

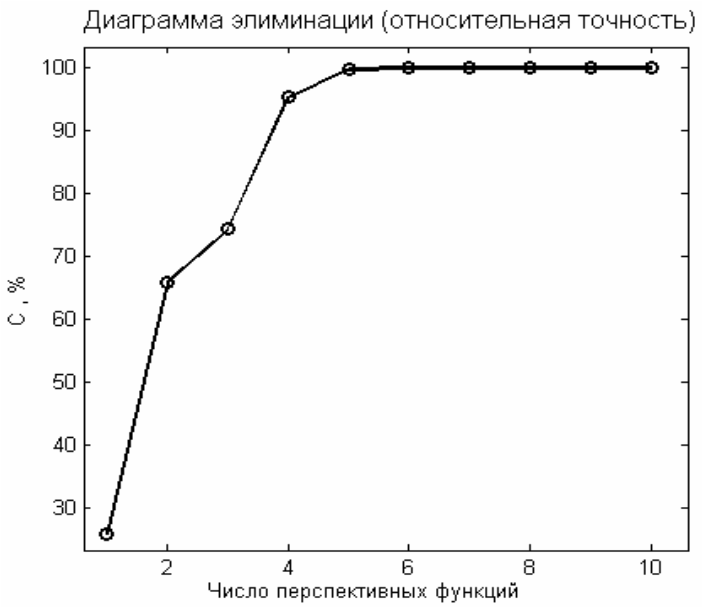


Рисунок 9 – Диаграмма элиминации (относительная точность)

В результате получен следующий окончательный вид аппроксимирующей функции:

$$\begin{aligned} y\_exp = {} & +0.99335x_1^1 - 0.16441x_1^3 + \\ & + 0.0081094x_1^5 - 0.00021947x_1^7 + \\ & + 1.6461e-005x_1^9 + 0.0029153 \end{aligned} \qquad (18)$$

На рисунке 10 построен график функции (18) по исходным точкам. Для сравнения на рисунке 11 представлен график функции, которая получилась бы, если бы на этапе элиминации были удалены также и некоторые существенные функции.

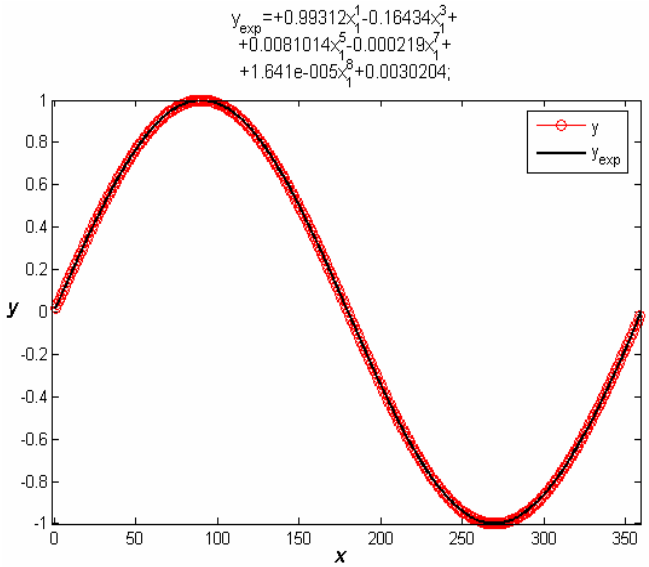


Рисунок 10 – Графики экспериментальной и аппроксимирующей функции

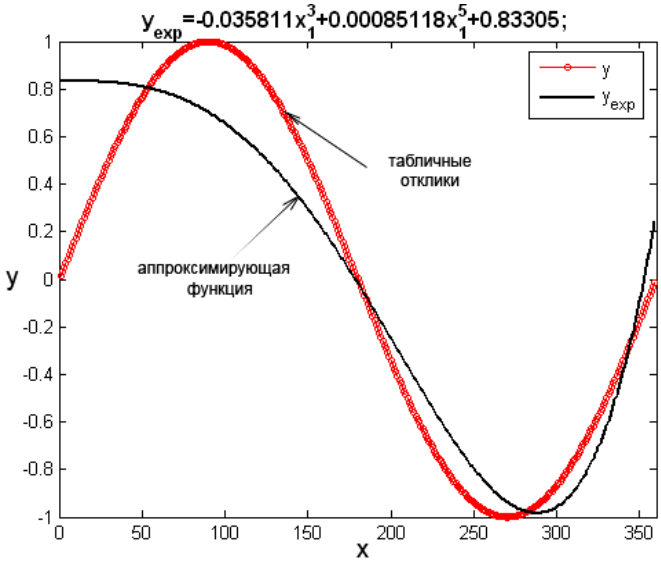


Рисунок 11 – Графики экспериментальной и аппроксимирующей функции с недостаточным числом членов

В части интерпретации полученных выражений возникают следующие соображения. Известно, что синус представим в виде степенного ряда:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \ldots. \qquad (19)$$

Тогда, из сравнения правых частей выражений (18) и (19) видно, что результат, полученный с помощью рассматриваемого алгоритма, близок к ряду (19) качественно, а это, на наш взгляд, свидетельствует о том, что рассматриваемая методика синтеза аппроксимирующей функции позволяет улавливать структуру модели по экспериментальным данным.

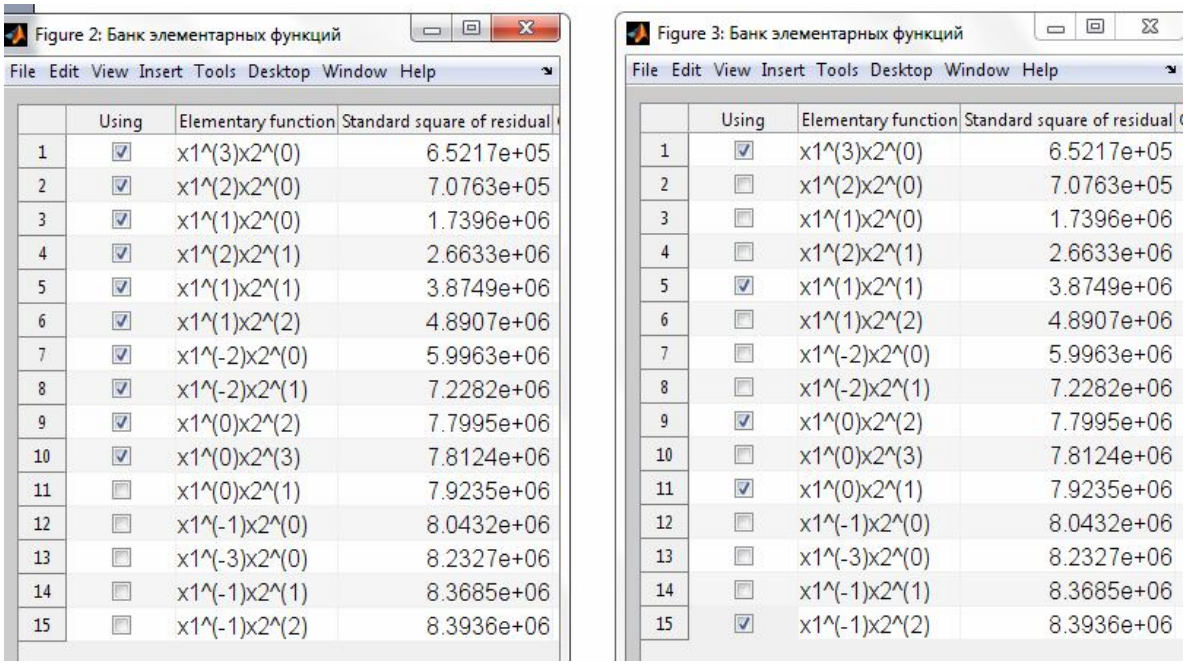
Анализ результатов

Полученные результаты, в целом, подтверждают эффективность рассматриваемого подхода и демонстрирует соответствие заявленным требованиям. Испытание алгоритма и эксплуатация его программной разработки дает возможность говорить как о ряде положительных сторон: универсальности, точности и надежности полученных уравнений, простоте и удобстве пользования методикой, так и о некоторых недостатках. В частности, не во всех случаях алгоритм оправдывает ожидания при восстановлении относительно простых псевдоэкспериментальных зависимостей, в ходе тестирования имели место некоторые сбои, то есть случаи «непопадания» функций, заведомо присутствующих в структуре регрессионной модели, в число перспективных функций. Причины этого, на данный момент, однозначно выяснить не удалось и пока открыты для обсуждения. Наиболее правдоподобной представляется версия, связанная с объемом и количественными особенностями экспериментальных данных.

Для наглядности продемонстрируем такую ситуацию на одном из рассмотренных выше примеров (пример 2). Для $k = 3, n = 2$ и учитывая, что

вторая переменная содержит нулевые значения, программа генерирует пятнадцать различных наборов степеней и соответственно столько же функций-предикторов. Если задать значение параметра $p = 15$, т.е. максимально возможное его значение, то синтез регрессионного уравнения проходит вполне успешно. Если же значение p уменьшить, то уже не все «полезные» функции оказываются в числе перспективных. Такая ситуация изображена на рисунке 12, где приведены две таблицы с информацией, упорядоченной по возрастанию погрешности, о функциях, содержащихся в банке. Слева, на рисунке 12а галочками отмечены те функции, которые попадают в число перспективных при автоматическом режиме отбора (в данном случае $p = 10$), а на рисунке 12б галочками показаны функции, которые в действительности должны входить в ожидаемое уравнение регрессии.

Как видно из 12б «нужные» функции разбросаны по всей таблице, а не сосредоточены, как хотелось бы, в верхней ее части (одна из них вообще находится в самом конце списка, т. е. расценивается как наименее перспективная, хотя в действительности это не так).



	Using	Elementary function	Standard square of residual
1	☑	x1^(3)x2^(0)	6.5217e+05
2	☑	x1^(2)x2^(0)	7.0763e+05
3	☑	x1^(1)x2^(0)	1.7396e+06
4	☑	x1^(2)x2^(1)	2.6633e+06
5	☑	x1^(1)x2^(1)	3.8749e+06
6	☑	x1^(1)x2^(2)	4.8907e+06
7	☑	x1^(-2)x2^(0)	5.9963e+06
8	☑	x1^(-2)x2^(1)	7.2282e+06
9	☑	x1^(0)x2^(2)	7.7995e+06
10	☑	x1^(0)x2^(3)	7.8124e+06
11	☐	x1^(0)x2^(1)	7.9235e+06
12	☐	x1^(-1)x2^(0)	8.0432e+06
13	☐	x1^(-3)x2^(0)	8.2327e+06
14	☐	x1^(-1)x2^(1)	8.3685e+06
15	☐	x1^(-1)x2^(2)	8.3936e+06

	Using	Elementary function	Standard square of residual
1	☑	x1^(3)x2^(0)	6.5217e+05
2	☐	x1^(2)x2^(0)	7.0763e+05
3	☐	x1^(1)x2^(0)	1.7396e+06
4	☐	x1^(2)x2^(1)	2.6633e+06
5	☑	x1^(1)x2^(1)	3.8749e+06
6	☐	x1^(1)x2^(2)	4.8907e+06
7	☐	x1^(-2)x2^(0)	5.9963e+06
8	☐	x1^(-2)x2^(1)	7.2282e+06
9	☑	x1^(0)x2^(2)	7.7995e+06
10	☐	x1^(0)x2^(3)	7.8124e+06
11	☑	x1^(0)x2^(1)	7.9235e+06
12	☐	x1^(-1)x2^(0)	8.0432e+06
13	☐	x1^(-3)x2^(0)	8.2327e+06
14	☐	x1^(-1)x2^(1)	8.3685e+06
15	☑	x1^(-1)x2^(2)	8.3936e+06

а) б)

Рисунок 12 – Демонстрация эффективности отбора перспективных функций

Кроме этого, в ходе вычислительного эксперимента были замечены некоторые неудобства для пользователя при эксплуатации программы и в связи с этим были предприняты попытки их устранения. Так, в частности, тестирование показало, что нормирование исходных данных не всегда обязательно и, возможно даже, не всегда желательно (поскольку способствует, в целом, усложнению внешнего вида регрессионного уравнения). Поэтому в программе была реализована возможность выбора

режима работы алгоритма по усмотрению пользователя: с нормировкой или же без нее (то есть с автоматической проверкой входных данных и генерацией отрицательных степеней лишь для переменных, не содержащих нулевые значения).

Как отмечалось выше, в некоторых тестовых наборах, функции, заведомо присутствующие в структуре модели, не попадали в число перспективных, т.к. оказывались в конце списка функций-претендентов, упорядоченного по убыванию «критерия перспективности», т.е. суммы квадратов отклонений. По эти причинам было принято решение предусмотреть возможность отбора функций в число перспективных самому пользователю, который, при желании, переключает «автоматический» режим отбора перспективных функций на режим отбора функций «вручную». Эта опция в программной реализации может оказаться действительно полезным средством в тех случаях, когда пользователь – эксперт, то есть имеет определенный опыт работы с конкретными экспериментальными данными и обладает базовыми знаниями или интуитивными соображениями о характере зависимостей в исследуемой модели. В таком случае при проверке собственных предположений и гипотез исследователь может отбирать в список перспективных лишь те функции, которые его интересуют, и связь между которыми он пытается восстановить.

Окончательный вариант пользовательского интерфейса с учетом всех опций представлен на рисунке 13.

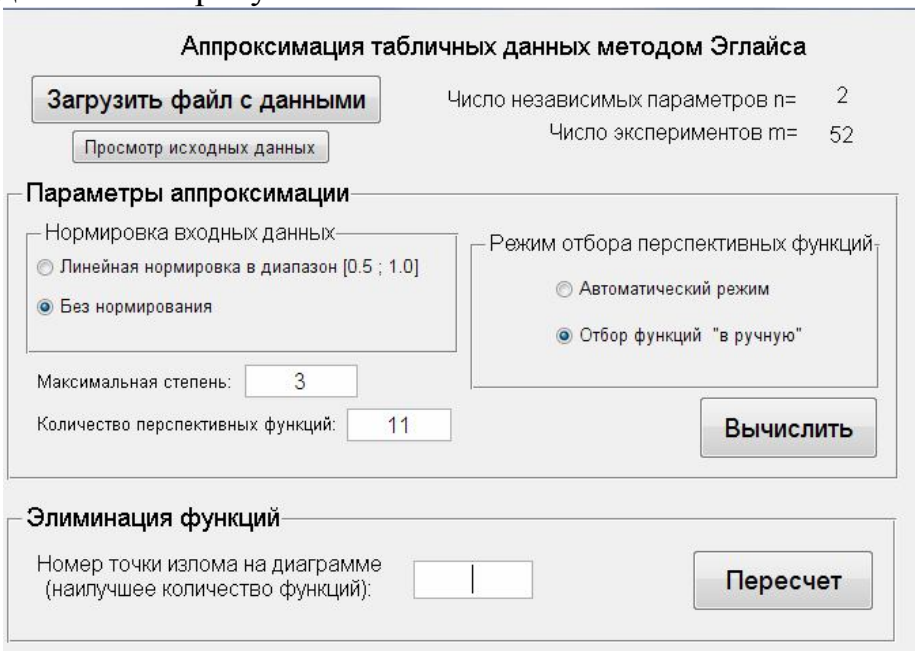


Рисунок 13 – Пользовательский интерфейс программы

Кроме этого был сделан вывод о том, что для получения более качественной модели параметр p, задаваемый пользователем, то есть число перспективных функций, должен быть, по возможности, достаточно большим. Объективного и точного способа определения хотя бы границ изменения параметра p, а, тем более, явно выраженной зависимости p от

максимальной степени регрессионного уравнения или его сложности выявить пока не удалось.

Еще один вопрос, который не остался без внимания – критерии эффективности модели, используемые на этапе элиминации функций. Как указывалось ранее, в базовом алгоритме используются формулы (9) или (10). Возникают вопросы: почему именно эти критерии и насколько они обоснованы, будет ли более эффективным использование других, пусть и относительно схожих – ответы на эти вопросы невозможны без вычислительного эксперимента и последующего сравнительного анализа.

Заключение и перспективы дальнейших исследований

С учетом изложенного представляется целесообразным развитие, реализация и проверка этих соображений в части модификации алгоритма и совершенствования как этапа отбора перспективных функций, так и процесса элиминации функций.

Результаты, полученные в ходе исследования, в целом, подтверждают эффективность рассматриваемого подхода и демонстрирует соответствие заявленным требованиям. Испытание алгоритма и эксплуатация его программной разработки дает возможность говорить о ряде положительных сторон: универсальности, точности и надежности полученных уравнений, простоте и удобстве пользования методикой.

Рассмотренный метод синтеза аппроксимирующей зависимости экспериментальных данных обладает множеством привлекательных характеристик:

- не требует предварительных знаний о структуре модели;
- данную методику можно считать относительно универсальной, поскольку в качестве базовых функций возможно использование различных по структуре элементарных функций;
- позволяет синтезировать аппроксимирующую функцию из различных комбинаций базовых функций;
- восстанавливаемая модель, в некотором смысле компромиссная, стремится одновременно удовлетворить двум взаимоисключающим оценкам качества модели: критерию точности и показателю эффективности модели, так как позволяет добиться высокой точности и параллельно свести к минимуму число функций-признаков, необходимых для описания отклика и восстановления адекватной регрессионной модели.

Конечно, этой методике еще предстоит пройти испытание на прочность при восстановлении зависимостей в реальных экологических, социальных и экономических данных. Но результаты, полученные в ходе исследования методики, подтверждают целесообразность ее использования для решения реальных задач и дальнейшего совершенствования как самой методики, так и развития области ее применения.

Список литературы

1. Эглайс В.О. Аппроксимация табличных данных многомерным уравнением регрессии / В.О. Эглайс // Вопросы динамики и прочности. - 1981. - Вып. 39. – С. 120–125.
2. Эглайс В.О. Синтез регрессионной модели объекта на основе табличных данных / В.О. Эглайс // Изв. АН Латв. ССР. Сер. физ. и тех. наук. - 1980. – № 4. – С. 109–112.
3. Виленкин Н.Я. Комбинаторика / Н.Я. Виленкин. – М.: Наука, 1969. – 323 с.
4. The MathWorks. MatLab and Simulink for technical computing [Электронный ресурс] Режим доступа – http://www.mathworks.com.
5. Matlab Central – File detail. Combinator: Combinations and permutations [Электронный ресурс] Режим доступа –http://www.mathworks.com/matlabcentral/fileexchange/24325-combinator-combinations-and-permutations.

Надійшла до редакції 4.10.2010 *Рецензент: канд.техн.наук, доц. Климко Г.Т.*

А.Б. Іващенко, В.М. Бєловодський
Донецький національний технічний університет

Аналіз методики Еглайса апроксимації табличних даних. Викладено перспективну методику апроксимації даних, що дозволяє ефективно відновлювати математичну модель за експериментальними даними. Приводяться особливості програмної реалізації алгоритму, а також результати серії обчислювальних експериментів. Запропоновано варіанти вдосконалювання методики.

апроксимація, реконструкція рівнянь, регресійна модель, залежність, банк функцій, елімінація, синтез рівняння

A.B. Ivashchenko, V.N. Belovodskiy
Donetsk National Technical University

An Analysis of Eglais Technique for Approximation of the Tabular Data. Perspective technique for approximation of the data, which allows reconstructing a mathematical model on the experimental data is presented. The features of software realization of the algorithm and also the results of a series of tests and computing experiments are discussed. The ways of technique improvements are offered.

approximation, reconstruction of the equations, regressive model, dependence, bank of functions, elimination, equation synthesis