

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
“ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

КАФЕДРА «ТЕХНИЧЕСКАЯ ТЕПЛОФИЗИКА»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по проведению практических, лабораторных занятий и
организации СРС по дисциплине
«Компьютерная обработка данных»

для студентов очной и заочной форм обучения направления подготовки
22.04.02 «Металлургия» магистерской программы
«Промышленная теплотехника»

Донецк – 2018

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
“ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

КАФЕДРА «ТЕХНИЧЕСКАЯ ТЕПЛОФИЗИКА»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по проведению практических, лабораторных занятий и
организации СРС по дисциплине
«Компьютерная обработка данных»

для студентов очной и заочной форм обучения направления подготовки
22.04.02 «Металлургия» магистерской программы
«Промышленная теплотехника»

РАССМОТРЕНО
на заседании кафедры
технической теплофизики
Протокол №12 от 31.05.2018 г.

УДК 669 04.1 (075)

Методические указания по проведению практических, лабораторных занятий и организации СРС по дисциплине «Компьютерная обработка данных» / Составитель: Кашаев В.В. – Донецк: ДонНТУ, 2018. – 38 с.

Эти методические указания по практическим и лабораторным занятиям научат обрабатывать различные массивы экспериментальных данных с помощью программного комплекса Mathcad используя интерполяцию, регрессию, сглаживание и фильтрацию, а также позволит закрепить и расширить магистрантами научно-технические знания, полученные по другим, ранее изученным дисциплинам.

Представлен перечень рекомендованной справочной литературы.

Составитель: доцент, к.т.н. Кашаев В.В.

Рецензент: доцент, к.т.н. Качура В.В.

СОДЕРЖАНИЕ

Введение	5
1. Общие сведения об обработке данных	6
2. Интерполяция	8
3. Регрессия	19
4. Сглаживание и фильтрация	30
Список литературы	37

ВВЕДЕНИЕ

Основной составной частью изучения дисциплины «Компьютерная обработка данных» является проведение со студентами практических и лабораторных занятий.

В современной промышленности математико-статистические методы исследования, моделирования и проектирования играют все возрастающую роль. Это вызвано в значительной мере быстрым развитием и усовершенствованием компьютерных технологий, благодаря которым непрерывно расширяются возможности их эффективного применения при решении конкретных проблем.

В распоряжении исследователя поступает огромное количество числовой информации, и средства, методы математики, статистики и информатики, столь обширные в своем прикладном значении, оказывают серьезную ему помощь в ее анализе, а также усовершенствовании учебно-тренировочного, организационно-управленческого и производственного процессов.

Методы обработки экспериментальных данных лежат в основе организации экспериментальных исследований, измерений и контроля качества продукции, а также эффективно используются в компьютерных измерительных технологиях.

Целью данной важной дисциплины – это знакомство с возможностями современных информационных технологий применимо к области профессиональных интересов магистрантов; формирование систематизированного представления о концепциях, моделях и принципах технологий компьютерной обработки информации и данных; знакомство с программным обеспечением для анализа экспериментальных данных и данных производственной деятельности; сформировать у магистрантов представление о навыках, необходимых при обработке информации на компьютере; способствовать развитию навыков обработки и анализа информации с помощью новых компьютерных технологий.

Цель методических указаний по практическим и лабораторным занятиям научить обрабатывать различные массивы экспериментальных данных с помощью программного комплекса Mathcad используя интерполяцию, регрессию, сглаживание и фильтрацию, а также позволяет закрепить и расширить студентами научно-технические знания, полученные по другим ранее изученным дисциплинам.

1. ОБЩИЕ СВЕДЕНИЯ ОБ ОБРАБОТКЕ ДАННЫХ

Когда мы имеем дело с выборкой экспериментальных данных, то они, чаще всего, представляются в виде массива, состоящего из пар чисел (x_i, y_i) . Поэтому возникает задача аппроксимации дискретной зависимости $y(x_i)$ непрерывной функцией $f(x)$. Функция $f(x)$, в зависимости от специфики задачи, может отвечать различным требованиям:

- ❖ $f(x)$ должна проходить через точки (x_i, y_i) , т.е. $f(x_i) = y_i, i=1 \dots n$. В этом случае говорят об интерполяции данных функцией $f(x)$ во внутренних точках между x_i или экстраполяции за пределами интервала, содержащего все x_i ;
- ❖ $f(x)$ должна некоторым образом (например в виде определенной аналитической зависимости) приближать $y(x_i)$, не обязательно проходя через точки (x_i, y_i) . Такова постановка задачи регрессии, которую во многих случаях также можно назвать сглаживанием данных;
- ❖ $f(x)$ должна приближать экспериментальную зависимость $y(x_i)$, учитывая к тому же, что данные (x_i, y_i) получены с некоторой погрешностью, выражающей шумовую компоненту измерений. При этом функция $f(x)$, с помощью того или иного алгоритма, уменьшает погрешность, присутствующую в данных (x_i, y_i) . Такого типа задачи называют задачами фильтрации. Сглаживание – частный случай фильтрации.

Различные виды построения аппроксимирующей зависимости $f(x)$ иллюстрирует рис. 1. На нем исходные данные обозначены кружками, интерполяция отрезками прямых линий – пунктиром, линейная регрессия – наклонной прямой линией, а фильтрация – жирной гладкой кривой. Эти зависимости приведены в качестве примера и отражают лишь малую часть возможностей Mathcad по обработке данных. Вообще говоря, в Mathcad имеется целый арсенал встроенных функций, позволяющий осуществлять самую различную регрессию, интерполяцию-экстраполяцию и сглаживание данных.

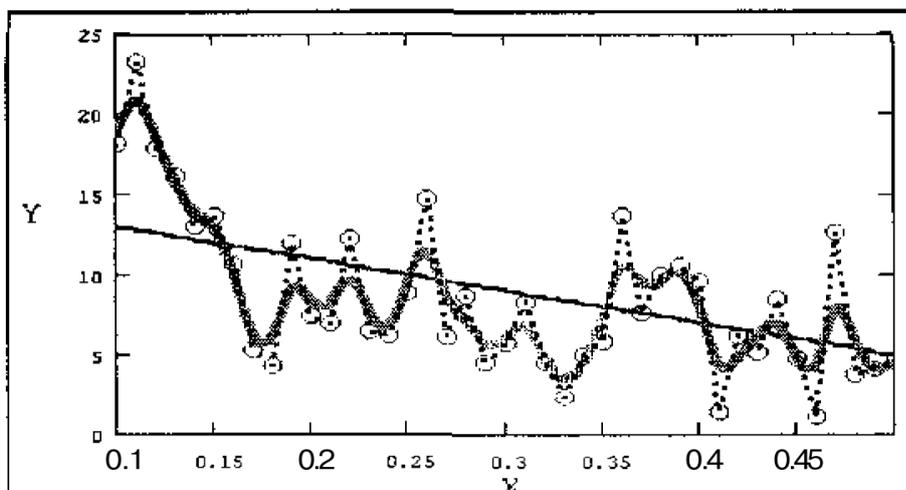


Рисунок 1 – Разные задачи аппроксимации данных

Как в целях подавления шума, так и для решения других проблем обработки данных, широко применяются различные интегральные преобразования.

Они ставят в соответствие всей совокупности данных $y(x)$ некоторую функцию другой координаты (или координат) $f(\omega)$. Примерами интегральных преобразований являются преобразование Фурье и вейвлетное преобразование. Напомним, что некоторые преобразования, например, Фурье и Лапласа, можно осуществить в режиме символьных вычислений. Каждое из интегральных преобразований эффективно для решения своего круга задач анализа данных.

2. ИНТЕРПОЛЯЦИЯ

Для построения интерполяции-экстраполяции в Mathcad имеются несколько встроенных функций, позволяющих «соединить» точки выборки данных (x_i, y_i) кривой разной степени гладкости.

По определению интерполяция означает построение функции $A(X)$, аппроксимирующей зависимость $y(x)$ в промежуточных точках (между x_i). Поэтому интерполяцию еще по-другому называют аппроксимацией. В точках x_i значения интерполяционной функции должны совпадать с исходными данными, т. е. $A(x_i) = y(x_i)$.

Примечание

Везде в этом разделе при рассказе о различных типах интерполяции будем использовать вместо обозначения $A(X)$ другое имя ее аргумента $A(t)$, чтобы не путать вектор данных x и скалярную переменную t .

Лабораторная работа №1 на тему «Линейная интерполяция»

Самый простой вид интерполяции – линейная, которая представляет искомую зависимость $A(X)$ в виде ломаной линии. Интерполирующая функция $A(X)$ состоит из отрезков прямых, соединяющих точки (рис. 2).

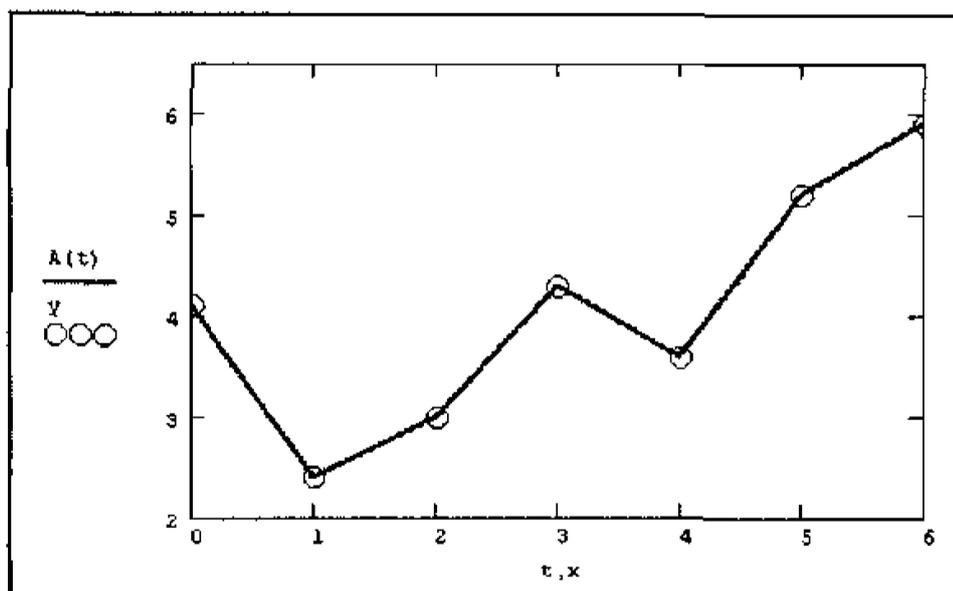


Рисунок 2 – Линейная интерполяция (листинг 1)

Для построения линейной интерполяции служит встроенная функция `linterp` (листинг 1).

- `linterp(x, y, t)` – функция, аппроксимирующая данные векторов x и y кусочно-линейной зависимостью;
- x – вектор действительных данных аргумента;
- y – вектор действительных данных значений того же размера;
- t – значение аргумента, при котором вычисляется интерполирующая функция.

Внимание!

Элементы вектора x должны быть определены в порядке возрастания, т. е.
 $x_1 < x_2 < x_3 < \dots < x_N$.

Листинг 1 Линейная интерполяция

```
x: = (0 1 2 3 4 5 6)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
A(t): = linterp (x, y, t)
```

Как видно из листинга, чтобы осуществить линейную интерполяцию, надо выполнить следующие действия:

1. Введите векторы данных x и y (первые две строки листинга).
2. Определите функцию $\text{linterp}(x, y, t)$.
3. Вычислите значения этой функции в требуемых точках, например $\text{linterp}(x, y, 2.4) = 3.52$ или $\text{linterp}(x, y, 6) = 5.9$, или постройте ее график, как показано на рис. 2.

Примечание

Обратите внимание, что функция $A(t)$ на графике имеет аргумент t , а не x . Это означает, что функция $A(t)$ вычисляется не только при значениях аргумента (т. е. в семи точках), а при гораздо большем числе аргументов в интервале $(0, 6)$, что автоматически обеспечивает **Mathcad**. Просто в данном случае эти различия незаметны, т. к. при обычном построении графика функции $A(x)$ от векторного аргумента x (рис. 15.3) **Mathcad**, по умолчанию, соединяет точки графика прямыми линиями (т. е. скрытым образом осуществляет их линейную интерполяцию).

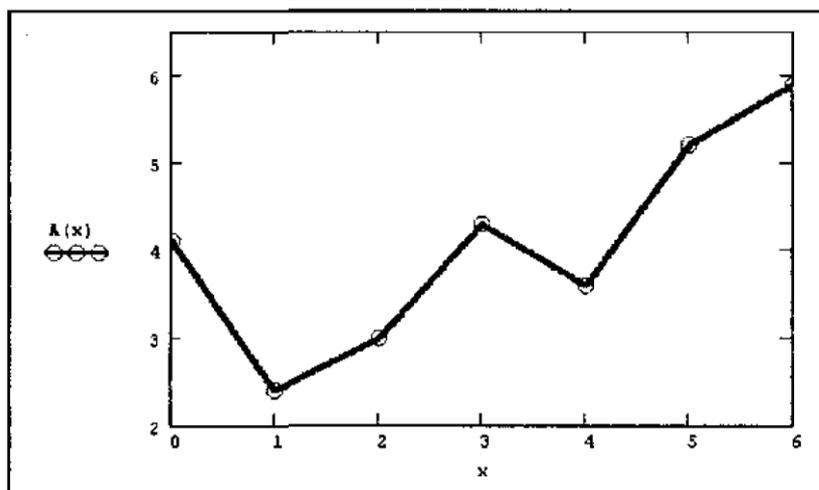


Рисунок 3 – Обычное построение графика функции от векторной переменной x (листинг 1)

Вопросы при текущем опросе к лабораторной работе №1:

1. Что такое интерполяция и в частности линейная интерполяция?

2. С помощью каких команд реализуется линейная интерполяция в программном комплексе Mathcad?

3. Какие надо осуществить действия чтобы выполнить линейную интерполяцию?

4. Как построить график функции от векторной переменной x ?

Лабораторная работа №2 на тему «Кубическая сплайн-интерполяция»

В большинстве практических приложений желательно соединить экспериментальные точки не ломаной линией, а гладкой кривой. Лучше всего для этих целей подходит интерполяция кубическими сплайнами, т. е. отрезками кубических парабол (рис. 4).

- $\text{interp}(s, x, y, t)$ – функция, аппроксимирующая данные векторов x и y кубическими сплайнами;
- s – вектор вторых производных, созданный одной из сопутствующих функций cspline , pspline или lspline ;
- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных данных значений того же размера;
- t – значение аргумента, при котором вычисляется интерполирующая функция.
-

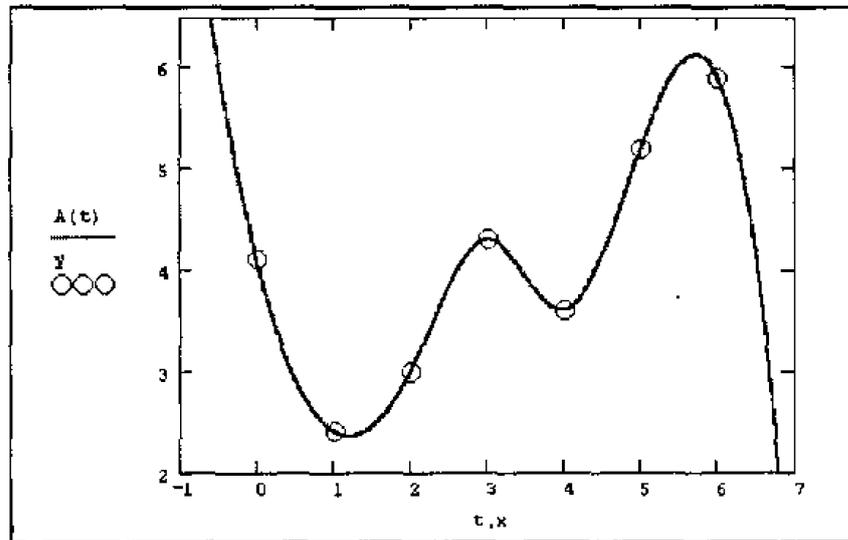


Рисунок 4 – Сплайн-интерполяция (листинг 2)

Сплайн-интерполяция в Mathcad реализована чуть сложнее линейной. Перед применением функции interp необходимо предварительно определить первый из ее аргументов – векторную переменную s . Делается это при помощи одной из трех встроенных функций тех же аргументов (x, y).

- $\text{lspline}(x, y)$ – вектор значений коэффициентов линейного сплайна;
- $\text{pspline}(x, y)$ – вектор значений коэффициентов квадратичного сплайна;
- $\text{cspline}(x, y)$ – вектор значений коэффициентов кубического сплайна;

- x, y – векторы данных.

Выбор конкретной функции сплайновых коэффициентов влияет на интерполяцию вблизи конечных точек интервала. Пример сплайн-интерполяции приведен в листинге 2.

Листинг 2 Кубическая сплайн-интерполяция

```
x: = (0 1 2 3 4 5 6)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
s: = cspline (x, y)
A(t): = interp (s, x, y, t)
```

Смысл сплайн-интерполяции заключается в том, что в промежутках между точками осуществляется аппроксимация в виде зависимости $A(t) = a \cdot t^3 + b \cdot t^2 + c \cdot t + d$. Коэффициенты a, b, c, d рассчитываются независимо для каждого промежутка, исходя из значений y_i в соседних точках. Этот процесс скрыт от пользователя, поскольку смысл задачи интерполяции состоит в выдаче значения $A(t)$ в любой точке t (рис. 4).

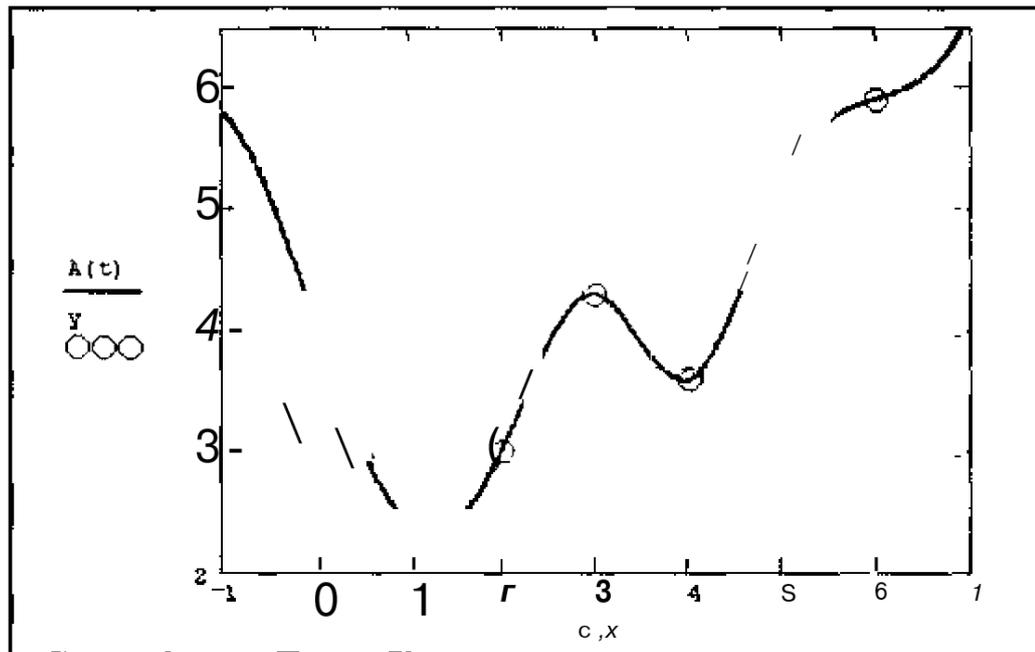


Рисунок 5 – Сплайн-интерполяция с выбором коэффициентов линейного сплайна lspline

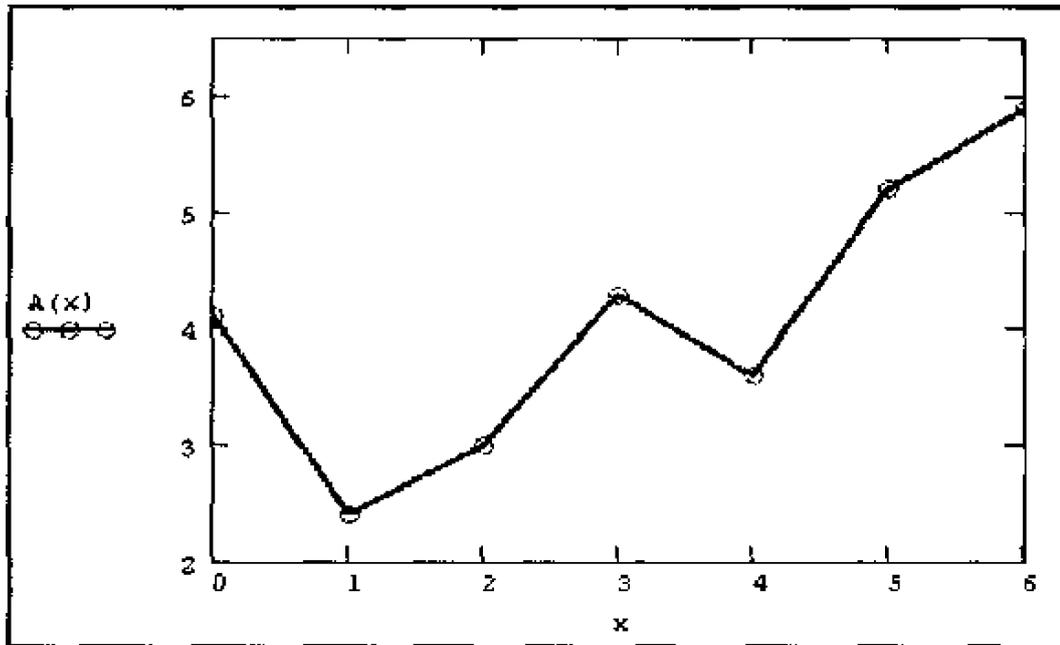


Рисунок 6 – Ошибочное построение графика сплайн-интерполяции (см. листинг 2)

Чтобы подчеркнуть различия, соответствующие разным вспомогательным функциям `cspline`, `pspline`, `lspline`, покажем результат действия листинга 2 при замене функции `cspline` в предпоследней строке на линейную `lspline` (рис. 5).

Как видно, выбор вспомогательных функций существенно влияет на поведение $A(t)$ вблизи граничных точек рассматриваемого интервала $(0, 6)$ и особенно разительно меняет результат экстраполяции данных за его пределами.

В заключение остановимся на уже упоминавшийся распространенной ошибке при построении графиков интерполирующей функции (см. рис. 3). Если на графике, например, являющемся продолжением листинга 2, задать построение функции $A(X)$ вместо $A(t)$, то будет получено просто соединение исходных точек ломаной (рис. 6). Так происходит потому, что в промежутках между точками вычисления интерполирующей функции не производятся.

Вопросы при текущем опросе к лабораторной работе №2:

1. Что такое кубическая сплайн-интерполяция?
2. Как реализуется кубическая сплайн-интерполяция в программном комплексе Mathcad?
3. Приведите пример сплайн-интерполяции.
4. Объясните смысл сплайн-интерполяции.
5. Проиллюстрируйте возможные ошибки при построении графика сплайн-интерполяции на компьютере.

Лабораторная работа №3 на тему «Полиномиальная сплайн-интерполяция»

Более сложный тип интерполяции – так называемая интерполяция В-сплайнами. В отличие от обычной сплайн-интерполяции, сшивка элементарных В-сплайнов производится не в точках x_i , а в других точках u_i , координаты которых предлагается ввести пользователю.

Сплайны могут быть полиномами 1, 2 или 3 степени (линейные, квадратичные или кубические). Применяется интерполяция В-сплайнами точно так же, как и обычная сплайн-интерполяция, различие состоит только в определении вспомогательной функции коэффициентов сплайна.

- `interp (s, x, y, t)` – функция, аппроксимирующая данные векторов x и y с помощью В-сплайнов;
- `bspline (x, y, u, n)` – вектор значений коэффициентов В-сплайна;
- s – вектор вторых производных, созданный функцией `bspline`;
- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных данных значений того же размера;
- t – значение аргумента, при котором вычисляется интерполирующая функция;
- u – вектор значений аргумента, в которых производится сшивка В-сплайнов;
- n – порядок полиномов сплайновой интерполяции (1, 2 или 3)

Примечание

Размерность вектора u должна быть на 1, 2 или 3 меньше размерности векторов x и y . Первый элемент вектора u должен быть меньше или равен первому элементу вектора x , а последний элемент u – больше или равен последнему элементу x .

Интерполяция В-сплайнами иллюстрируется листингом 3 и рис. 7.

Листинг 3 Интерполяция В-сплайнами

```
x: = (0 1 2 3 4 5 6)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
u: = (0.5 2.2 3.3 4.1 5.5 7)T
s: = bspline (x, y, u, 2)
A(t): = linterp (s, x, y, t)
```

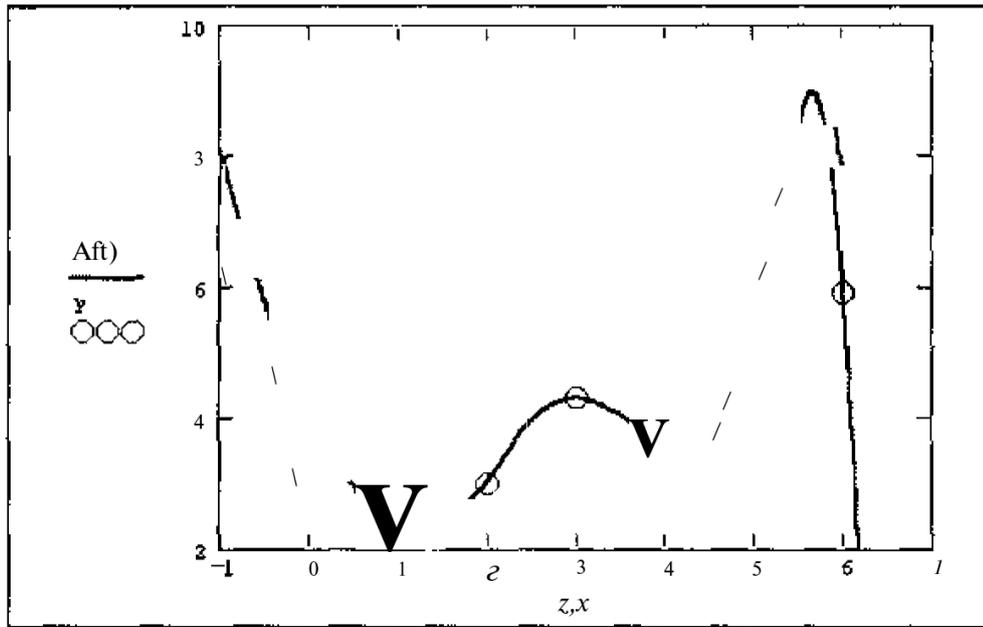


Рисунок 7 – B-сплайн-интерполяция (листинг 3)

Вопросы при текущем опросе к лабораторной работе №3:

1. Чем отличается обычная сплайн-интерполяция от полиномиальной сплайн-интерполяции?
2. Как реализуется полиномиальная сплайн-интерполяция в программном комплексе Mathcad?
3. Приведите пример полиномиальной сплайн-интерполяции.
4. Проиллюстрируйте B-сплайн-интерполяцию на компьютере.

Лабораторная работа №4 на тему «Экстраполяция функцией предсказания»

Все рассмотренные выше функции осуществляли экстраполяцию данных за пределами их интервала с помощью соответствующей зависимости, основанной на анализе расположения нескольких исходных точек на границах интервала.

В Mathcad имеется более развитый инструмент экстраполяции, который учитывает распределение данных вдоль всего интервала. В функцию `predict` встроен линейный алгоритм предсказания поведения функции, основанный на анализе, в том числе осцилляции.

- `predict (y, m, n)` – функция предсказания вектора, экстраполирующего выборку данных;
- y – вектор действительных значений, взятых через равные промежутки значений аргумента;
- m – количество последовательных элементов вектора y , согласно которым строится экстраполяция;
- n – количество элементов вектора предсказаний.

Пример использования функции предсказания на примере экстраполяции осциллирующих данных y_i с меняющейся амплитудой приведен в ли-

стинге 4. Полученный график экстраполяции, наряду с самой функцией, показан на рис. 8.

Аргументы и принцип действия функции predict отличаются от рассмотренных выше встроенных функций интерполяции-экстраполяции. Значений аргумента для данных не требуется, поскольку по определению функция действует на данные, идущие друг за другом с равномерным шагом. Обратите внимание, что результат функции predict вставляется «в хвост» исходных данных.

Листинг 4. Экстраполяция при помощи функции предсказания

```
k:=100                                j :=0 .. k
yj := e $\frac{-j}{100}$  · sin $\left(\frac{i}{10}\right)$ 
m:=50                                    n:=150
A := predict (y , m , n)
```

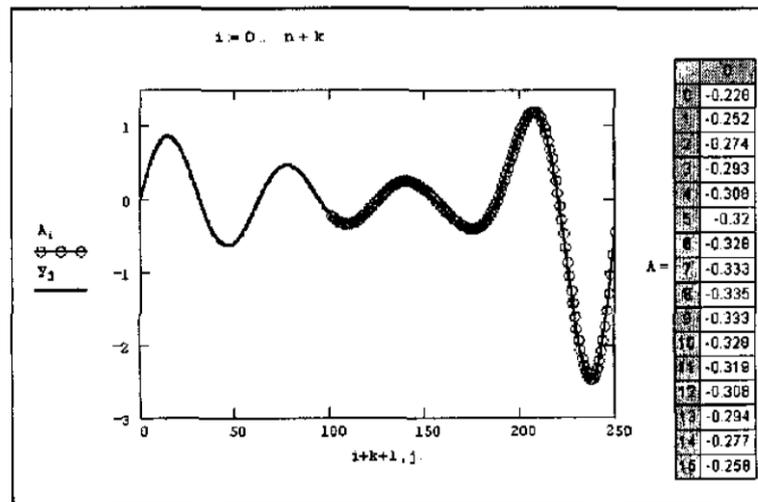


Рисунок 8 – Экстраполяция при помощи функции предсказания (листинг 4)

Как видно из рис. 9, функция предсказания может быть полезна при экстраполяции данных на небольшие расстояния. Вдали от исходных данных результат часто бывает неудовлетворительным. Кроме того, функция predict хорошо работает в задачах анализа подробных данных с четко прослеживаемой закономерностью (типа рис. 8), в основном осциллирующего характера.

Если данных мало, то предсказание может оказаться бесполезным. В листинге 5 приведена экстраполяция небольшой выборки данных (из примеров, рассмотренных в предыдущих разделах). Соответствующий результат показан на рис. 9 для различных крайних точек массива исходных данных, для которых строится экстраполяция.

Листинг 5 Экстраполяция при помощи функции предсказания

```
y = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
```

$k := \text{rows}(y) - 1$
 $m := 5 \quad n := 10$
 $A := \text{predict}(y, m, n)$

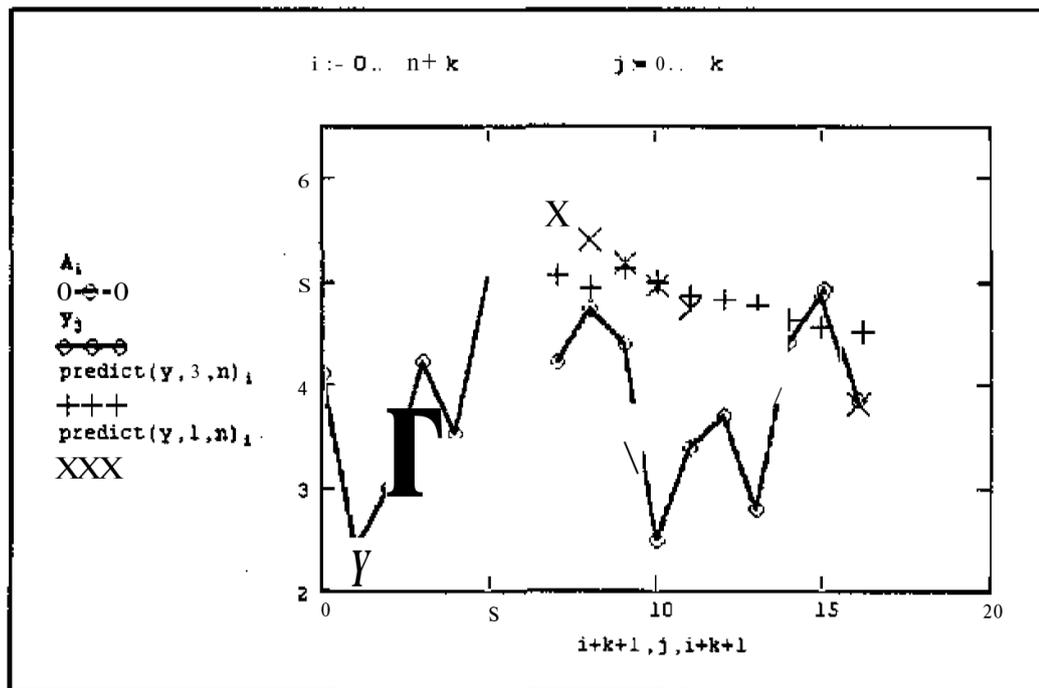


Рисунок 9 – Работа функции предсказания в случае малого количества данных (листинг 5)

Вопросы при текущем опросе к лабораторной работе №4:

1. Как реализуется экстраполяция данных функцией предсказания в программном комплексе Mathcad?
2. Приведите линейный алгоритм предсказания поведения функции.
3. Приведите пример использования функции предсказания на примере экстраполяции осциллирующих данных y_i с меняющейся амплитудой.
4. Чем аргументы и принцип действия функции `predict` отличаются от рассмотренных выше встроенных функций интерполяции-экстраполяции?
5. Что будет при экстраполяции при помощи функции предсказания, если данных окажется недостаточно или мало?

Лабораторная работа №5 на тему «Многомерная интерполяция»

Двумерная сплайн-интерполяция приводит к построению поверхности $z(x, y)$, проходящей через массив точек, описывающий сетку на координатной плоскости (x, y) .

Поверхность создается участками двумерных кубических сплайнов, являющихся функциями (x, y) и имеющих непрерывные первые и вторые производные по обеим координатам.

Многомерная интерполяция строится с помощью тех же встроенных функций, что и одномерная, но имеет в качестве аргументов не векторы, а соответствующие матрицы.

Существует одно важное ограничение, связанное с возможностью интерполяции только квадратных $N \times N$ массивов данных.

- `interp (s, x, z, v)` – скалярная функция, аппроксимирующая данные выборки двумерного поля по координатам x и y кубическими сплайнами;
- s – вектор вторых производных, созданный одной из сопутствующих функций `cspline`, `pspline` или `lspline`;
- x – матрица размерности $N \times 2$, определяющая диагональ сетки значений аргумента (элементы обоих столбцов соответствуют меткам x и y и расположены в порядке возрастания);
- z – матрица действительных данных размерности $N \times N$;
- v – вектор из двух элементов, содержащий значения аргументов x и y , для которых вычисляется интерполяция.

Примечание

Вспомогательные функции построения вторых производных имеют те же матричные аргументы, что и `interp`: `lspline(X,Y)`, `pspline(X,Y)`, `cspline(X,Y)`.

Листинг 6. Двумерная интерполяция

```

x:=
(
0  0
1 10
2 20
3 30
4 40
)
y:=
(
1  1  0  1.1  1.2
1  2  3  2.1  1.5
1.3 3.3 5  1.7  2
1.3  3  3.7 2.1  2.9
1.5  2  2.5 2.8  4
)
S:=cspline (X, Y)
v:=
(
3.7
2.2
)
k:=30
interp (S , X, Y, V) =1.636

```

```

i := 0 .. k
j := 0 .. k
Aij := interp (S.X.Y,
(
i/k * 4
j/k * 40
)
)

```

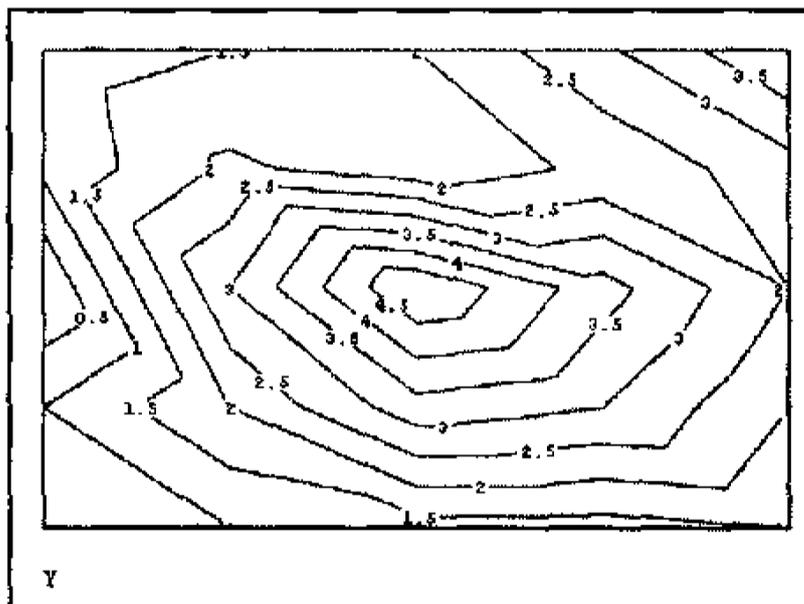


Рисунок 10 – Исходное двумерное поле данных (листинг 6)

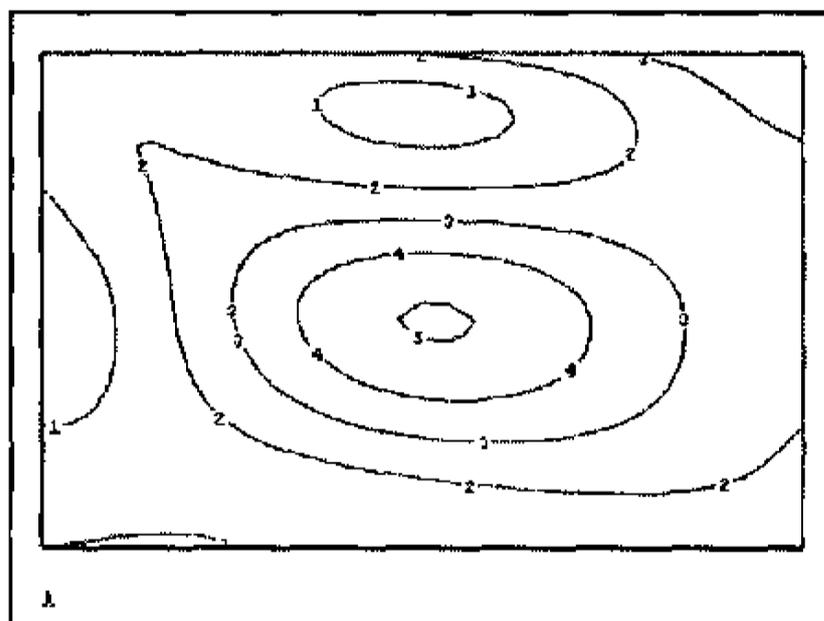


Рисунок 11 – Результат двумерной интерполяции (листинг 6)

Вопросы при текущем опросе к лабораторной работе №5:

1. Как строится многомерная интерполяция в программном комплексе Mathcad?
2. Какое существует важное ограничение при многомерной интерполяции данных?
3. Приведите пример вспомогательных функций при многомерной интерполяции в программном комплексе Mathcad.
4. Как выглядит исходное двумерное поле данных?
5. Проиллюстрируйте на компьютере результат двумерной интерполяции?

3. РЕГРЕССИЯ

Задачи математической регрессии имеют смысл приближения выборки данных (x_i, y_i) некоторой функцией $f(x)$, определенным образом минимизирующей совокупность ошибок $|f(x_i) - y_i|$.

Регрессия сводится к подбору неизвестных коэффициентов, определяющих аналитическую зависимость $f(x)$. В силу производимого действия большинство задач регрессии являются частным случаем более общей проблемы сглаживания данных.

Как правило, регрессия очень эффективна, когда заранее известен (или, по крайней мере, хорошо угадывается) закон распределения данных (x_i, y_i) .

Лабораторная работа №6 на тему «Линейная регрессия»

Самый простой и наиболее часто используемый вид регрессии – линейная. Приближение данных (x_i, y_i) осуществляется линейной функцией $y(x) = b + a \cdot x$. На координатной плоскости (x, y) линейная функция, как известно, представляется прямой линией (рис. 12).

Еще линейную регрессию часто называют методом наименьших квадратов, поскольку коэффициенты a и b вычисляются из условия минимизации суммы квадратов ошибок $|b + a \cdot x_i - y_i|$.

Примечание

Чаще всего такое же условие ставится и в других задачах регрессии, т. е. приближения массива данных (x_i, y_i) другими зависимостями $y(x)$. Исключение рассмотрено в листинге 9.

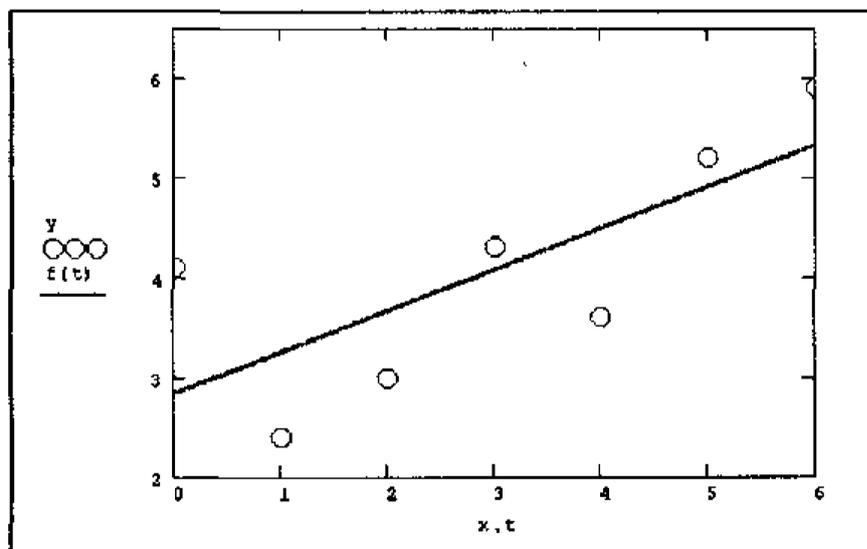


Рисунок 12 – Линейная регрессия (листинг 7 или 8)

Для расчета линейной регрессии в Mathcad имеются два дублирующих друг друга способа.

Правила их применения представлены в листингах 7 и 8. Результат

обоих листингов получается одинаковым (рис. 12).

- `line (x, y)` – вектор из двух элементов (b, a) коэффициентов линейной регрессии $b + a \cdot x$;
- `intercept (x, y)` – коэффициент b линейной регрессии;
- `slope (x, y)` – коэффициент a линейной регрессии;
- `x` – вектор действительных данных аргумента;
- `y` – вектор действительных данных значений того же размера.

Листинг 7. Линейная регрессия

```
x := (0 1 2 3 4 5 6)T
y := (4.1 2.4 3 4.3 3.6 5.2 5.9)T
line (x, y) =  $\begin{pmatrix} 2.829 \\ 0.414 \end{pmatrix}$ 
f (t) := line (x, y) 0 + line (x, y) 1 · t
```

Листинг 8. Другая форма записи линейной регрессии

```
x := (0 1 2 3 4 5 6)T
y := (4.1 2.4 3 4.3 3.6 5.2 5.9)T
intercept (x, y) = 2.829
slope (x, y) = 0.414
f (t) := intercept (x, y) + slope (x, y) · t
```

В Mathcad имеется альтернативный алгоритм, реализующий не минимизацию суммы квадратов ошибок, а медиан-медианную линейную регрессию для расчета коэффициентов a и b (листинг 9).

- `medfit (x, y)` – вектор из двух элементов (b, a) коэффициентов линейной медиан-медианной регрессии $b + a \cdot x$;
- `x, y` – векторы действительных данных одинакового размера.

Листинг 9. Построение линейной регрессии двумя разными методами (продолжение листинга 7)

```
medfit (x, y) =  $\begin{pmatrix} 2.517 \\ 0.55 \end{pmatrix}$ 
g (t) := medfit (x, y) 0 + medfit (x, y) 1 · t
```

Различие результатов среднеквадратичной и медиан-медианной регрессии иллюстрируется рис. 13.

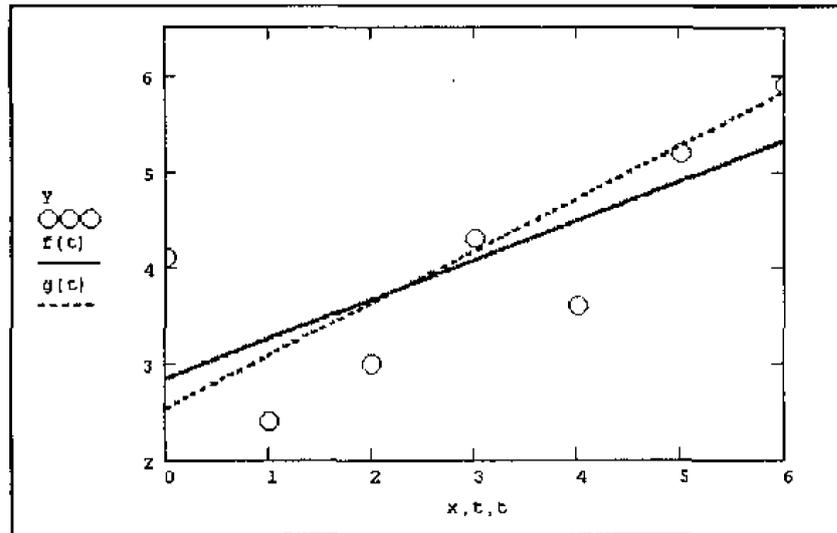


Рисунок 13 – Линейная регрессия по методу наименьших квадратов и методу медиан (листинги 7 и 9)

Вопросы при текущем опросе к лабораторной работе №6:

1. Что такое регрессия?
2. Объясните суть линейной регрессии.
3. Опишите два дублирующих друг друга способа для расчета линейной регрессии в программном комплексе Mathcad.
4. Проиллюстрируйте на компьютере различие результатов средне-квадратичной и медиан-медианной регрессии?

Лабораторная работа №7 на тему «Полиномиальная регрессия»

В Mathcad реализована регрессия одним полиномом, отрезками нескольких полиномов, а также двумерная регрессия массива данных.

Полиномиальная регрессия означает приближение данных (x_i, y_i) полиномом k -й степени $A(x) = a + b \cdot x + c \cdot x^2 + d \cdot x^3 + \dots + h \cdot x^k$ (рис. 14). При $k = 1$ полином является прямой линией, при $k = 2$ – параболой, при $k = 3$ – кубической параболой и т. д. Как правило, на практике применяются $k < 5$.

Примечание

Для построения регрессии полиномом k -й степени необходимо наличие по крайней мере $(k+1)$ точек данных.

В Mathcad полиномиальная регрессия осуществляется комбинацией встроенной функции `regress` и полиномиальной интерполяции.

- `regress (x, y, k)` – вектор коэффициентов для построения полиномиальной регрессии данных;
- `interp (s, x, y, t)` – результат полиномиальной регрессии;
- `s := regress (x, y, k)`;
- `x` – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- `y` – вектор действительных данных значений того же размера;

- k – степень полинома регрессии (целое положительное число);
- t – значение аргумента полинома регрессии.

Внимание!

Для построения полиномиальной регрессии после функции `regress` Вы обязаны использовать функцию `interp`.

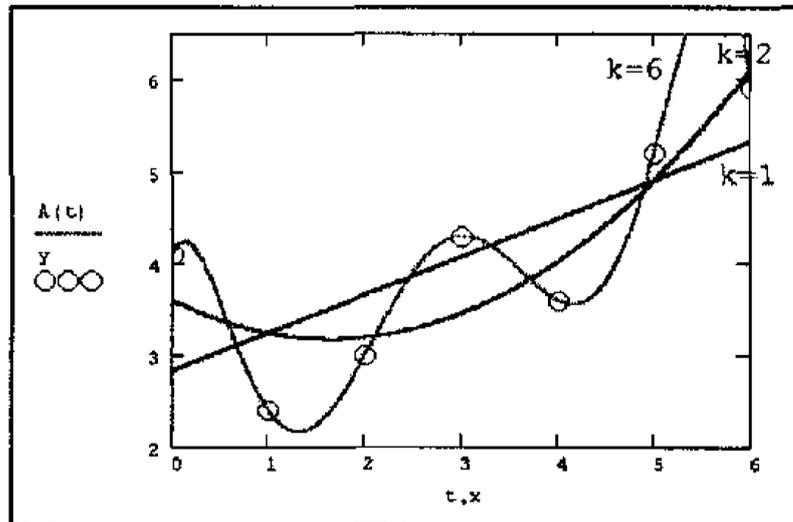


Рисунок 14 – Регрессия полиномами разной степени (коллаж результатов листинга 10 для разных k)

Пример полиномиальной регрессии квадратичной параболой приведен в листинге 10.

Листинг 10. Полиномиальная регрессия

```
x: = (0 1 2 3 4 5 6)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
k: = 2
s: = regress (x, y, k)
A (t): = interp (s, x, y, t)
```

Вопросы при текущем опросе к лабораторной работе №7:

1. Как в Mathcad реализуется регрессия данных?
2. Объясните суть полиномиальной регрессии.
3. Как осуществляется полиномиальная регрессия в программном комплексе Mathcad?
4. Проиллюстрируйте на компьютере пример полиномиальной регрессии квадратичной параболой с учетом функции `interp` в Mathcad?

Лабораторная работа №8 на тему «Регрессия отрезками полиномов»

Помимо приближения массива данных одним полиномом имеется воз-

возможность осуществить регрессию сшивкой отрезков (точнее говоря, участков, т.к. они имеют криволинейную форму) нескольких полиномов.

Для этого имеется встроенная функция loess, применение которой аналогично функции regress (листинг 11 и рис. 15).

- loess (x, y, span) – вектор коэффициентов для построения регрессии данных отрезками полиномов;
- interp (s, x, y, t) – результат полиномиальной регрессии;
- s: = loess (x, y, span);
- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных данных значений того же размера;
- span – параметр, определяющий размер отрезков полиномов (положительное число, хорошие результаты дает значение порядка span = 0.75).

Параметр span задает степень сглаженности данных. При больших значениях span регрессия практически не отличается от регрессии одним полиномом (например, span = 2 дает почти тот же результат, что и приближение точек параболой).

Листинг 11. Регрессия отрезками полиномов

```
x: = (0 1 2 3 4 5 6)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9)T
s: = regress (x, y, 0.9)
A (t): = interp (s, x, y, t)
```

Совет

Регрессия одним полиномом эффективна, когда множество точек выглядит как полином, а регрессия отрезками полиномов оказывается полезной в противоположном случае.

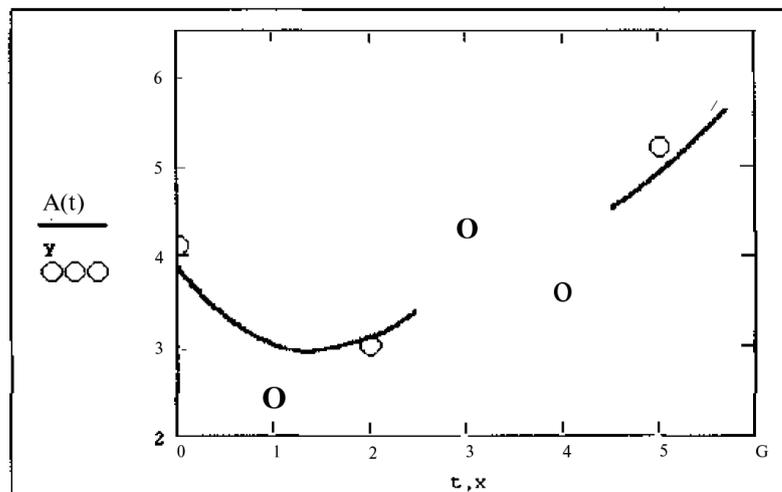


Рисунок 15 – Регрессия отрезками полиномов (листинг 11)

Вопросы при текущем опросе к лабораторной работе №8:

1. Как в Mathcad можно осуществить регрессию сшивкой отрезков (участков)?
2. С помощью каких команд осуществляется регрессия сшивкой отрезков (участков) в программном комплексе Mathcad?
3. Что задает параметр *span*?
4. Приведите условие эффективности применения регрессии одним полиномом.
5. Проиллюстрируйте на компьютере пример регрессии отрезками полиномов в Mathcad?

Лабораторная работа №9 на тему «Двумерная полиномиальная регрессия»

По аналогии с одномерной полиномиальной регрессией и двумерной интерполяцией Mathcad позволяет приблизить множество точек $z_{ij}(x_i, y_i)$ поверхностью, которая определяется многомерной полиномиальной зависимостью.

В качестве аргументов встроенных функций для построения полиномиальной регрессии должны стоять в этом случае не векторы, а соответствующие матрицы.

- *regress* (*x*, *z*, *k*) – вектор коэффициентов для построения полиномиальной регрессии данных;
- *loess* (*x*, *z*, *span*) – вектор коэффициентов для построения регрессии данных отрезками полиномов;
- *interp* (*s*, *x*, *z*, *v*) – скалярная функция, аппроксимирующая данные выборки двумерного поля по координатам *x* и *y* кубическими сплайнами;
- *s* – вектор вторых производных, созданный одной из сопутствующих функций *loess* или *regress*;
- *x* – матрица размерности $N \times 2$, определяющая пары значений аргумента (столбцы соответствуют меткам *x* и *y*);
- *z* – вектор действительных данных размерности *N*;
- *span* – параметр, определяющий размер отрезков полиномов;
- *k* – степень полинома регрессии (целое положительное число);
- *v* – вектор из двух элементов, содержащий значения аргументов *x* и *y*, для которых вычисляется интерполяция.

Внимание!

Для построения регрессии не предполагается никакого предварительного упорядочивания данных (как, например, для двумерной интерполяции, которая требует их представления в виде матрицы $N \times N$). В связи с этим данные представляются как вектор.

Двумерная полиномиальная регрессия иллюстрируется листингом 12 и рис. 16. Сравните стиль представления данных для двумерной регрессии с представлением тех же данных для двумерной сплайн-интерполяции (см. ли-

стинг 6) и ее результаты с исходными данными (см. рис. 10) и их сплайн-интерполяцией (см. рис. 11).

Листинг 12. Двумерная полиномиальная регрессия

```

X:= $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 & 3 & 4 & 4 & 4 & 4 & 4 \end{pmatrix}^T$ 
Z:=(1 1 0 1.11.21 2 3 2.11.51.33.35 1.72 1.33 3.72.12.91.52 2.52.84)^T
S:=regress(X,Z,3)
k:=30
i:=0..k
j:=0..k
Ai,j:=interp[S,X,Z, $\begin{pmatrix} \frac{i}{k}-4 \\ \frac{j}{k}\cdot 40 \end{pmatrix}$ ]

```

Примечание

Обратите внимание на знаки транспонирования в листинге. Они применены для корректного представления аргументов (например: z в качестве вектора, а не строки).

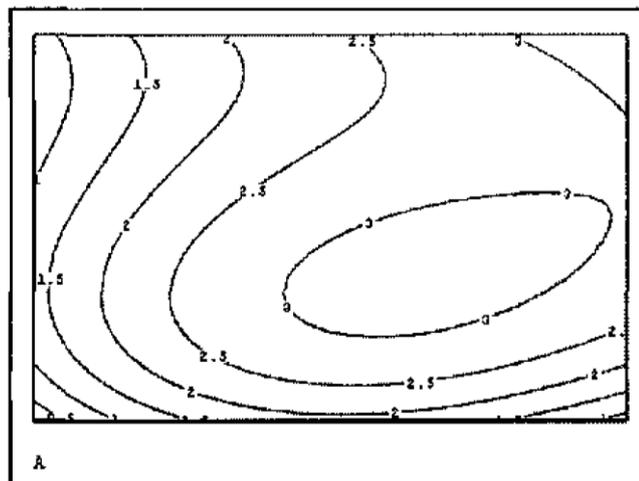


Рисунок 16 – Двумерная полиномиальная регрессия (листинг 12)

Вопросы при текущем опросе к лабораторной работе №9:

1. Что такое двумерная полиномиальная регрессия?
2. С помощью каких команд осуществляется двумерная полиномиальная регрессия в программном комплексе Mathcad?
3. Сравните стиль представления данных для двумерной регрессии с представлением тех же данных для двумерной сплайн-интерполяции и ее результаты с исходными данными и их сплайн-интерполяцией на компьютере.

4. Что такое знаки транспонирования?

Лабораторная работа №10 на тему «Регрессия специального вида»

Кроме рассмотренных, в Mathcad встроено еще несколько видов трехпараметрической регрессии. Их реализация несколько отличается от приведенных выше вариантов регрессии тем, что для них, помимо массива данных, требуется задать некоторые начальные значения коэффициентов a , b , c .

Используйте соответствующий вид регрессии, если хорошо представляете себе, какой зависимостью описывается массив данных. Когда тип регрессии плохо отражает последовательность данных, то ее результат часто бывает неудовлетворительным и даже сильно различающимся в зависимости от выбора начальных значений.

Каждая из функций выдает вектор уточненных параметров a , b , c .

- $\text{expfit}(x, y, g)$ – регрессия экспонентой $f(x) = a + e^{bx} + c$;
- $\text{lgsfit}(x, y, g)$ – регрессия логистической функцией $f(x) = a / (1 + b \cdot e^{-cx})$;
- $\text{sinfit}(x, y, g)$ – регрессия синусоидой $f(x) = a \cdot \sin(x+b) + c$;
- $\text{pwfit}(x, y, g)$ – регрессия степенной функцией $f(x) = a \cdot x^b + c$;
- $\text{logfit}(x, y, g)$ – регрессия логарифмической функцией $f(x) = a \cdot \ln(x+b) + c$;
- $\text{Infit}(x, y)$ – регрессия двухпараметрической логарифмической функцией $f(x) = a \cdot \ln(x) + b$;
- x – вектор действительных данных аргумента;
- y – вектор действительных значений того же размера;
- g – вектор из трех элементов, задающий начальные значения a , b , c .

Примечание

Правильность выбора начальных значений можно оценить по результату регрессии — если функция, выданная Mathcad, хорошо приближает зависимость $y(x)$, значит они были подобраны удачно.

Пример расчета одного из видов трехпараметрической регрессии (экспоненциальной) приведен в листинге 13 и на рис. 17. В предпоследней строке листинга выведены в виде вектора вычисленные коэффициенты a , b , c , а в последней строке через эти коэффициенты определена искомая функция $f(x)$.

Листинг 13. Экспоненциальная регрессия

```
x := (0 1 2 3 4 5 6)T
y := (4.1 2.4 3 4.3 3.6 5.2 5.9)T
g :=  $\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ 
C := expfit(x, y, g)
```

$$C = \begin{pmatrix} 0.111 \\ 0.544 \\ 3.099 \end{pmatrix}$$

$$f(t) = C_0 \cdot \exp(C_1 \cdot t) + C_2$$

Примечание

Многие задачи регрессии данных различными двухпараметрическими зависимостями $y(x)$ можно свести к более надежной, с вычислительной точки зрения, линейной регрессии. Делается это, с помощью, соответствующей замены переменных.

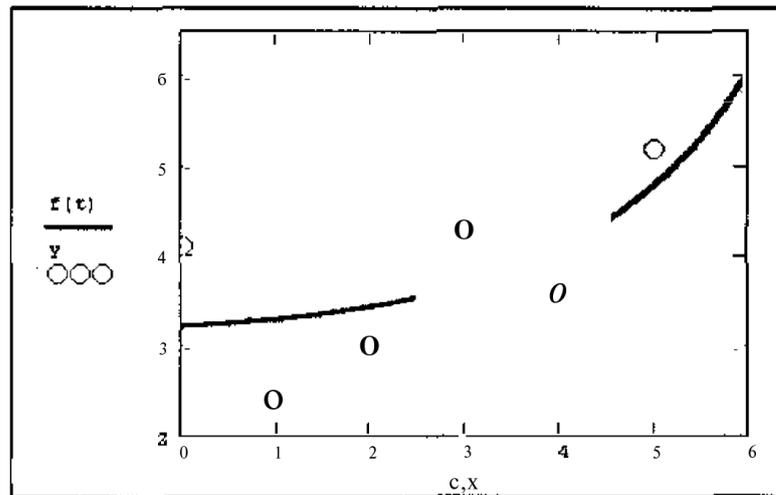


Рисунок 17 – Экспоненциальная регрессия (листинг 13)

Вопросы при текущем опросе к лабораторной работе №10:

1. Что такое регрессия специального вида и какие виды регрессий существуют?
2. С помощью каких команд осуществляется регрессия специального вида в программном комплексе Mathcad?
3. Проиллюстрируйте на компьютере пример расчета одного из видов трехпараметрической регрессии (экспоненциальной).
4. Как можно свести к более надежной, с вычислительной точки зрения, линейной регрессии многие задачи регрессии данных?

Лабораторная работа №11 на тему «Регрессия общего вида»

В Mathcad можно осуществить регрессию в виде линейной комбинации $C_1 f_1(x) + C_2 f_2(x) + \dots$, где $f_i(x)$ – любые функции пользователя, а C_i – подлежащие определению коэффициенты.

Кроме того, имеется путь проведения регрессии более общего вида, когда комбинацию функций и искомым коэффициентов задает сам пользователь.

Приведем встроенные функции для регрессии общего вида и примеры их использования (листинги 14 и 15).

При необходимости можно найти более подробную информацию об этих специальных возможностях в справочной системе и Mathcad Resources.

- $\text{linfit}(x, y, F)$ – вектор параметров линейной комбинации функций пользователя, осуществляющей регрессию данных;
- $\text{genfit}(x, y, g, G)$ – вектор параметров, реализующих регрессию данных с помощью функций пользователя общего вида;
- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных значений того же размера;
- $F(X)$ – пользовательская векторная функция скалярного аргумента;
- g – вектор начальных значений параметров регрессии размерности N ;
- $G(x, c)$ – векторная функция размерности $N+1$, составленная из функции пользователя и ее N частных производных по каждому из параметров c .

Листинг 14. Регрессия линейной комбинацией функций пользователя

$$x := (0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6)^T$$

$$y := (4.1 \ 2.4 \ 3 \ 4.3 \ 3.6 \ 5.2 \ 5.9)^T$$

$$F(x) := \begin{pmatrix} 1 \\ x + 1 \\ x \\ e^x \end{pmatrix}$$

$$C := \text{linfit}(x, y, F)$$

$$C = \begin{pmatrix} 3.957 \\ 0.854 \\ 5.605 \times 10^{-4} \end{pmatrix}$$

$$f(x) := [C_0 / (x + 1)] + C_1 \cdot x + C_2 \cdot e^x$$

Листинг 15. Регрессия общего вида

$$x := (0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6)^T$$

$$y := (4.1 \ 2.4 \ 3 \ 4.3 \ 3.6 \ 5.2 \ 5.9)^T$$

$$g := \begin{pmatrix} 0.1 \\ 1 \end{pmatrix}$$

$$G(x, C) := \begin{pmatrix} C_0 \cdot x^{C_1 \cdot x} \\ x^{C_1 \cdot x} \\ C_0 \cdot x^{C_1 \cdot x + 1} \end{pmatrix}$$

$$C := \text{genfit}(x, y, g, G)$$

$$C = \begin{pmatrix} 3.168 \\ 0.057 \end{pmatrix}$$
$$f(x) := C_0 \cdot x^{C_1 \cdot x}$$

Вопросы при текущем опросе к лабораторной работе №11:

1. Что такое регрессия регрессия общего вида?
2. С помощью каких команд осуществляется регрессия общего вида в программном комплексе Mathcad?
3. Проиллюстрируйте на компьютере пример регрессии линейной комбинацией функций пользователя.
4. Проиллюстрируйте на компьютере пример регрессии общего вида.

4. СГЛАЖИВАНИЕ И ФИЛЬТРАЦИЯ

При анализе данных часто возникает задача их фильтрации, заключающаяся в устранении одной из составляющих зависимости $y(x_i)$. Наиболее часто целью фильтрации является подавление быстрых вариаций $y(x_i)$, которые чаще всего обусловлены шумом. В результате из быстроосциллирующей зависимости $y(x_i)$ получается другая, сглаженная зависимость, в которой доминирует более низкочастотная составляющая.

Наиболее простыми и эффективными рецептами сглаживания (smoothing) можно считать регрессию различного вида. Однако регрессия часто уничтожает информативную составляющую данных, оставляя лишь наперед заданную пользователем зависимость.

Часто рассматривают противоположную задачу фильтрации – устранение медленно меняющихся вариаций в целях исследования высокочастотной составляющей.

В этом случае говорят о задаче устранения тренда. Иногда интерес представляют смешанные задачи выделения сред немасштабных вариаций путем подавления как более быстрых, так и более медленных вариаций. Одна из возможностей решения связана с применением полосовой фильтрации.

Несколько примеров программной реализации различных вариантов фильтрации приведены в данном разделе.

Лабораторная работа №12 на тему «Встроенные функции для сглаживания»

В Mathcad имеется несколько встроенных функций, реализующих различные алгоритмы сглаживания данных.

- `medsmooth (y, b)` – сглаживание алгоритмом «бегущих медиан»;
- `ksmooth (x, y, b)` – сглаживание на основе функции Гаусса;
- `supsmooth (x, y)` – локальное сглаживание адаптивным алгоритмом, основанное на анализе ближайших соседей каждой пары данных;
- x – вектор действительных данных аргумента (для `supsmooth` его элементы должны быть расположены в порядке возрастания);
- y – вектор действительных значений того же размера, что и x ;
- b – ширина окна сглаживания.

Все функции имеют в качестве аргумента векторы, составленные из массива данных, и выдают в качестве результата вектор сглаженных данных того же размера. Функция `medsmooth` предполагает, что данные расположены равномерно.

Примечание

Подробную информацию об алгоритмах, заложенных в функции сглаживания, Вы найдете в справочной системе Mathcad в статье **Smoothing** (Сглаживание), находящейся в разделе **Statistics** (Статистика)..

Часто бывает полезным совместить сглаживание с последующей интерполяцией или регрессией. Соответствующий пример приведен в листинге 16 для функции `supsmooth`.

Результат работы листинга показан на рис. 18 (кружки обозначают исходные данные, крестики – сглаженные, пунктирная кривая – результат сплайн-интерполяции). Сглаживание тех же данных при помощи «бегущих медиан» и функции Гаусса с разным значением ширины окна пропускания показаны на рис. 19 и 20, соответственно.

Листинг 16. Сглаживание с последующей сплайн-интерполяцией

```
x: = (0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9 5 4.7 4 3.5 3.9 3 2.7 3.7 4.8 5.4)T
z: = supsmooth (x, y)
s: = cspline (x, z)
A(t): = interp (s, x, z, t)
```

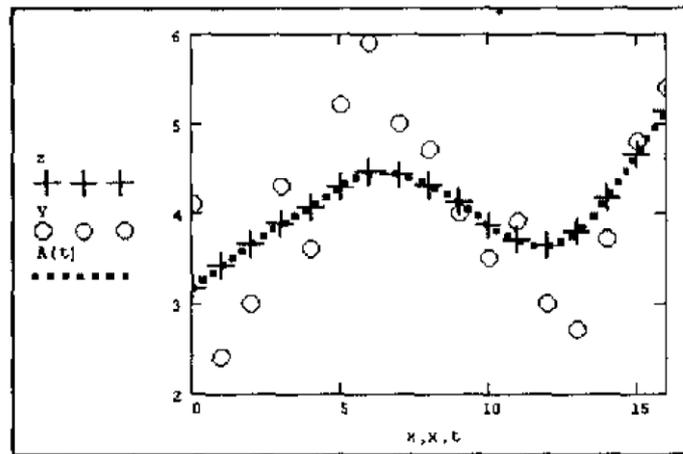


Рисунок 18 – Адаптивное сглаживание (листинг 16)

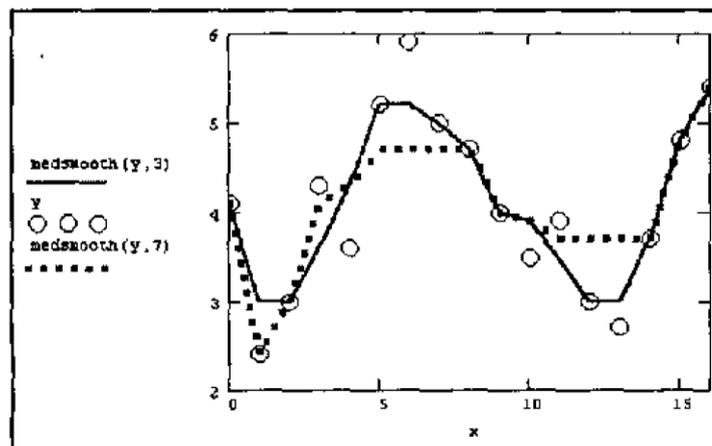


Рисунок 19 – Сглаживание «бегущими медианами»

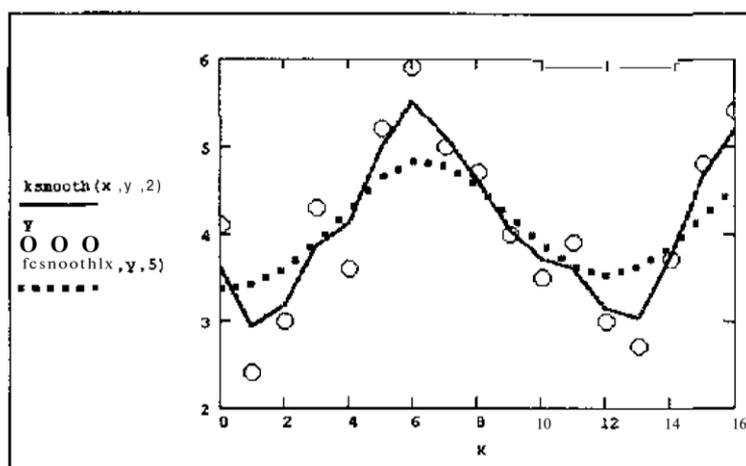


Рисунок 20 – Сглаживание при помощи функции ksmooth

Вопросы при текущем опросе к лабораторной работе №12:

1. Что такое сглаживание данных?
2. С помощью каких команд реализуются различные алгоритмы сглаживания данных в программном комплексе Mathcad?
3. Для чего нужна функция medsmooth?
4. Проиллюстрируйте на компьютере пример сглаживания с последующей сплайн-интерполяцией.

Лабораторная работа №13 на тему «Скользящее усреднение»

Помимо встроенных в Mathcad, существует несколько популярных алгоритмов сглаживания, на одном из которых хочется остановиться особо. Самый простой и очень эффективный метод – это скользящее усреднение. Его суть состоит в расчете для каждого значения аргумента среднего значения по соседним w данным. Число w называют окном скользящего усреднения; чем оно больше, тем больше данных участвуют в расчете среднего, тем более сглаженная кривая получается.

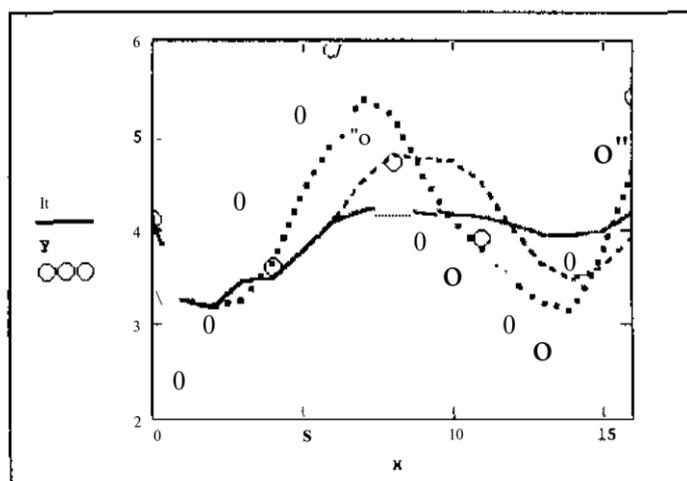


Рисунок 21 – Скользящее усреднение с разными $w = 3, 5, 15$
(листинг 17, коллаж трех графиков)

На рис. 21 показан результат скользящего усреднения одних и тех же данных (кружки) с разным окном $w = 3$ (пунктир), $w = 5$ (штрихованная кривая) и $w = 15$ (сплошная кривая). Видно, что при малых w сглаженные кривые практически повторяют ход изменения данных, а при больших w – отражают лишь закономерность их медленных вариаций.

Чтобы реализовать в Mathcad скользящее усреднение, достаточно очень простой программы, приведенной в листинге 17. Она использует только значения y , оформленные в виде вектора, неявно предполагая, что они соответствуют значениям аргумента x , расположенным через одинаковые промежутки.

Вектор x применялся лишь для построения графика результата (рис. 21).

Листинг 17. Сглаживание скользящим усреднением

```
x: = (0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)T
y: = (4.1 2.4 3 4.3 3.6 5.2 5.9 5 4.7 4 3.5 3.9 3 2.7 3.7 4.8 5.4)T
w: = 15
N: = rows (y)                N = 17
i: = 0.....N-1
```

$$m_i := \text{if} \left(i < w, \frac{\sum_{j=0}^i y_j}{i+1}, \frac{\sum_{j=i-w+1}^i y_j}{w} \right)$$

Примечание

Приведенная программная реализация скользящего усреднения самая простая, но не самая лучшая. Возможно, Вы обратили внимание, что все кривые скользящего среднего на рис. 21 слегка «обгоняют» исходные данные. Почему так происходит, понятно: согласно алгоритму, заложенному в последнюю строку листинга 17, скользящее среднее для каждой точки вычисляется путем усреднения значений предыдущих w точек.

Чтобы результат скользящего усреднения был более адекватным, лучше применить центрированный алгоритм расчета по $w/2$ предыдущим и $w/2$ последующим значениям. Он будет немного сложнее, поскольку придется учитывать недостаток точек не только в начале (как это сделано в программе с помощью функции условия if), но и в конце массива исходных данных.

Вопросы при текущем опросе к лабораторной работе №13:

1. Что такое скользящее усреднение данных?
2. С помощью каких команд реализовать алгоритм скользящего усреднения данных в программном комплексе Mathcad?
3. Что нужно предпринять, чтобы результат скользящего усреднения был более адекватным?

4. Проиллюстрируйте на компьютере пример сглаживания скользящим усреднением.

Лабораторная работа №14 на тему «Устранение тренда»

Еще одна типичная задача возникает, когда интерес исследований заключается не в анализе медленных (или низкочастотных) вариаций сигнала $y(x)$ (для чего применяется сглаживание данных), а в анализе быстрых его изменений.

Часто бывает, что быстрые (или высокочастотные) вариации накладываются определенным образом на медленные, которые обычно называют трендом. Часто тренд имеет заранее предсказуемый вид, например, линейный. Чтобы устранить тренд, можно предложить последовательность действий, реализованную в листинге 18.

1. Вычислить регрессию $f(x)$, например, линейную, исходя из априорной информации о тренде (предпоследняя строка листинга).

2. Вычесть из данных $y(x)$ тренд $f(x)$ (последняя строка листинга).

Листинг 18. Устранение тренда

```
x: = (0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)T
y: = (5.1 4.4 5.7 5.3 5.6 5.2 5.9 7.7 6.7 7 6.5 5.9 8.1 8.7 9.7 9.8
10.4)T
w: = 15
N: = rows (y)           N = 17
i: = 0.....N-1
f(t): = line (x, y)0 + line (x, y)1 · t
mi: = yi - f(xi)
```

На рис. 22 показаны исходные данные (кружками), выделенный с помощью регрессии линейный тренд (сплошной прямой линией) и результат устранения тренда (пунктир, соединяющий крестики).

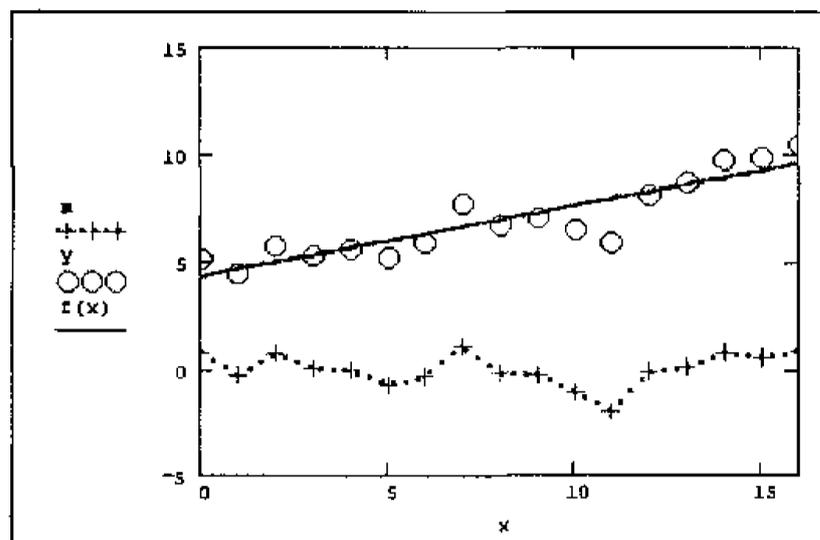


Рисунок 22 – Устранение тренда (листинг 18)

Вопросы при текущем опросе к лабораторной работе №14:

1. Что такое устранение тренда?
2. С помощью каких команд реализуется алгоритм устранения тренда в программном комплексе Mathcad?
3. Что показывает предпоследняя строка листинга?
4. Проиллюстрируйте на компьютере пример устранения тренда.

Лабораторная работа №15 на тему «Полосовая фильтрация»

В предыдущих разделах была рассмотрена фильтрация быстрых вариаций сигнала (сглаживание) и его медленных вариаций (снятие тренда). Иногда требуется выделить среднemasштабную составляющую сигнала, уменьшив как более быстрые, так и более медленные его компоненты.

Одна из возможностей решения этой задачи связана с применением полосовой фильтрации на основе последовательного скользящего усреднения.

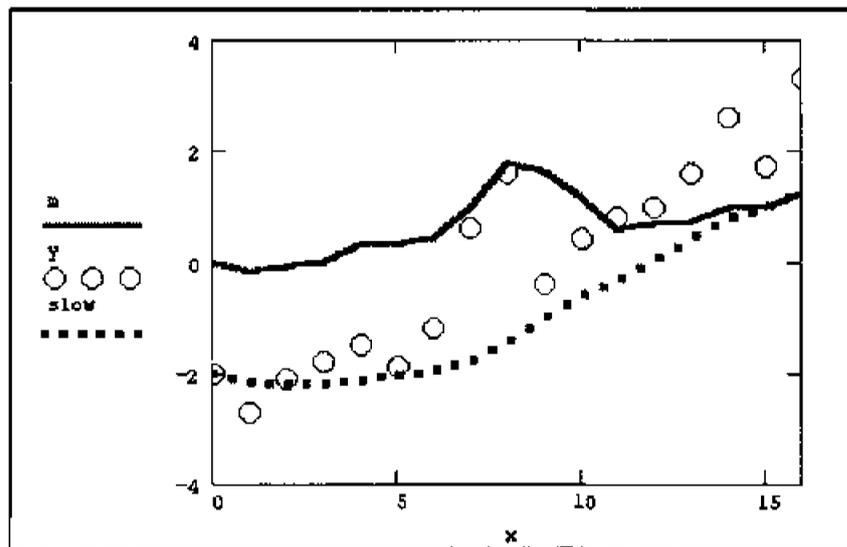


Рисунок 23 – Результат полосовой фильтрации (листинг 19)

Алгоритм полосовой фильтрации приведен в листинге 19, а результат его применения показан на рис. 23 сплошной кривой.

Алгоритм реализует такую последовательность операций:

1. Приведение массива данных y к нулевому среднему значению путем его вычитания из каждого элемента y (третья и четвертая строки листинга).
2. Устранение из сигнала y высокочастотной составляющей, имеющее целью получить сглаженный сигнал middle , например, с помощью скользящего усреднения с малым окном w (в листинге 19 $w = 3$).
3. Выделение из сигнала middle низкочастотной составляющей slow , например, путем скользящего усреднения с большим окном w (в листинге 19 $w = 7$) либо, с помощью снятия тренда.
4. Вычитание из сигнала middle тренда slow (последняя строка листинга), тем самым выделяя среднemasштабную составляющую исходного сигнала y .

Листинг 19. Полосовая фильтрация

$x := (0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10 \ 11 \ 12 \ 13 \ 14 \ 15 \ 16)^T$
 $y := (5.1 \ 4.4 \ 5.7 \ 5.3 \ 5.6 \ 5.2 \ 5.9 \ 7.7 \ 6.7 \ 7 \ 6.5 \ 5.9 \ 8.1 \ 8.7 \ 9.7 \ 9.8$
 10.4)^T
 meanY := mean (y)
 y := y – meanY
 N := rows (y) N = 17
 i := 0.....N-1
 w := 3

$$\text{middle}_i := \text{if} \left(i < w, \frac{\sum_{j=0}^i y_j}{i+1}, \frac{\sum_{j=i-w+1}^i y_j}{w} \right)$$

w := 7

$$\text{slow}_i := \text{if} \left(i < w, \frac{\sum_{j=0}^i \text{middle}_j}{i+1}, \frac{\sum_{j=i-w+1}^i \text{middle}_j}{w} \right)$$

m := middle – slow

Вопросы при текущем опросе к лабораторной работе №14:

1. Что такое полосовая фильтрация?
2. Приведите алгоритм полосовой фильтрации в программном комплексе Mathcad?
3. Что показывает последняя строка листинга?
4. Проиллюстрируйте на компьютере пример реализации полосовой фильтрации в программном комплексе Mathcad.

СПИСОК ЛИТЕРАТУРЫ

1. Фигурнов, В.Э. Анализ данных на компьютере / В.Э. Фигурнов [и др.]; под ред. В.Э. Фигурнова. – М.: ИНФРА – М.: Финансы и статистика, 1995. – 384 с.
2. Захарова, И.Г. Информационные технологии в образовании: учеб. пособие / И.Г. Захарова. – Изд. 2-е., стер. – М.: Академия, 2005. – 192 с.
3. Боровиков, В.П. STATISTICA – статистический анализ и обработка данных в среде Windows / В.П. Боровиков, И.П. Боровиков. – М.: Информационно – издательский дом «Филинь», 1998. – 608 с.
4. Кирьянов, Д.В. Самоучитель Mathcad 13 / Д.В. Кирьянов. – СПб.: БХВ-Петербург, 2006. – 528 с.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по проведению практических, лабораторных занятий и
организации СРС по дисциплине
«Компьютерная обработка данных»

для студентов очной и заочной форм обучения направления подготовки
22.04.02 «Металлургия» магистерской программы
«Промышленная теплотехника»

Составитель: доцент, к.т.н. Кашаев Виталий Валерьевич