МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"
Інститут прикладної математики та фундаментальних наук

Кафедра прикладної математики

Звіт
про проходження практики за темою
бакалаврської кваліфікаційної роботи
"Моделювання функціональних залежностей
мінімаксними многочленними наближеннями"

Виконав:
ст. гр. ПМ-41
Левантович Богдан
Керівник БКР:
доц. каф. ПМ
Пізюр Я.В.
Прийняв:
Гладун В.Р. керівник
практики від
університету

Львів 2017

Зміст

Вступ

Необхідність моделювання функціональних залежностей виникає в багатьох галузях прикладної математики та інформатики. При розв'язуванні багатьох задач науково-технічного характеру доводиться використовувати функції задані таблицею. Проте часто необхідно мати значення функції в точках, яких немає в таблиці. Також виникає необхідність використання простої функції замість складної.

Багатьом із тих, хто стикається з науковими та інженерними розрахунками часто доводиться оперувати наборами значень, отриманих експериментальним шляхом чи методом випадкової вибірки. Як правило, на підставі цих наборів потрібно побудувати функцію, зі значеннями якої могли б з високою точністю збігатися інші отримувані значення. Така задача називається апроксимацією кривої. Інтерполяцією називають такий різновид апроксимації, при якій крива побудованої функції проходить точно через наявні точки даних.

1 Найкраще чебишовське наближення

За теоремою Вейєрштрасса для довільних неперервних на обмеженому проміжку $[a, b]$ функцій $f(x)$ та $w(x) > 0$ і довільного $\epsilon > 0$ можна знайти такий многочлен $P_m(x)$, що

$$|\rho(x)| = \frac{|f(x) - P_m(x)|}{w(x)} < \epsilon, \quad x \in [a, b].$$

Ясно, що найменше при цьому значення степеня m многочлена $P_m(x)$ суттєво залежить від способу наближення. Серед усіх способів наближення функцій найменшу похибку а, значить, і найменше m при заданому ϵ, дає найкраще чебишовське наближення.

Вираз $F(A, x) \in F(B, x)$, для якого максимальне значення абсолютної величини зваженої похибки досягає на проміжку $[a, b]$ найменшого значення

$$\min_{c \in B} \max_{x \in [a,b]} \frac{|f(x) - F(C, x)|}{w(x)} = \max_{x \in [a,b]} \frac{|f(x) - F(A, x)|}{w(x)}, \tag{1}$$

звемо найкращим чебишовським зваженим (з вагою $w(x)$) наближенням функції $f(x)$ за допомогою виразу виду $F(A, x)$ на проміжку $[a, b]$.

У цій курсові розглянуто лише найкращі чебишовські наближення. Слова "чебишовські" і "зважені" будемо часом пропускати. При $w(x) = 1$ маємо найкраще абсолютне наближення, при $w(x) = f(x)$ - найкраще відносне.

Величину (1) називатимемо мінімальним (зваженим) відхиленням і позначаємо $E(f, W) \equiv \mu_0$; $E(f, 1) \equiv E(f) \equiv \Delta_0$ - мінімальне абсолютне відхилення; $E(f, f) \equiv \delta_0$ - мінімальне відносне відхилення.

Далі розглянемо властивості найкращих наближень многочленом.

Теорема 1. Для будь-яких неперервних на проміжку $[a, b]$ функцій $f(x)$ та $w(x) > 0$ і довільного ϵ, існує єдиний многочлен $P_m(x)$ степеня m, що має найменше відхилення $E(f, w)$.

Теорема 2. Нехай на проміжку $[a, b]$ задано неперервні функції $f(x)$ та $w(x) > 0$. Тоді для того, щоб деякий многочлен $P_m(x)$ степеня не вище m був многочленом найкращого чебишовського зваженого наближення функції $f(x)$ на проміжку $[a, b]$ необхідно і достатньо, щоб на цьому проміжку

знайшлась принаймні одна система з $m+2$ точок
$T = \{t_k\}_{k=0}^{m+1} \quad a \leq t_0 < t_1 < t_2 < \ldots \leq t_{m+1}$, у яких зважена різниця почергово набувала значень різних знаків і досягала за модулем найбільшого на $[a, b]$ значення тобто:

$$\rho(t_0) = -\rho(t_1) = \rho(t_2) = \ldots = (-1)^{m+1}\rho(t_{m+1}) = \pm E(f, W). \tag{2}$$

Система точок T із теореми 2 зветься системою точок (чебишовського альтернансу). Для побудови многочлена найкращого наближення необхідно визначити ці точки. Точно визначити їх значення можна тільки у часткових випадках.

1.1 Схема Ремеза побудови чебишовського наближення

У загальному випадку процес знаходження точок T побудовано на ітераційних методах. Найбільше практичне значення мають методи розроблені українським математиком Є.Я. Ремезом. Коротко розглянемо один з методів. Він складається з таких етапів.

1. З проміжку $[a, b]$ вибираємо початкове наближення T_0 до альтернансу
$$T : t_0^{(0)} < t_1^{(0)} < t_2^{(0)} < \ldots < t_{m+1}^{(0)}.$$
Можна, наприклад, прийняти $t_k^{(0)} = a + \frac{(b-a)k}{m+1}$.

2. Здійснюємо чебишовську інтерполяцію для множини точок
$T_j = \{t_k\}_{k=0}^{m+1}, t_k^{(j)} < t_{k+1}^{(j)}, k = \overline{0, m}$, тобто визначаємо коефіцієнти многочлена $P_m^i(x)$ і величину μ_j, для яких виконуються умови $\rho(t_k^{(j)}) = (-1)^k \mu_k \quad k = \overline{0, m+1}$. Для знаходження вказаних величин розв'язуємо систему рівнянь:

$$\begin{cases} f(t_0^{(j)}) - a_0 - a_1 t_0^{(j)} - \ldots - a_m (t_0^{(j)})^m = \mu_j w(t_0^{(j)}), \\ f(t_1^{(j)}) - a_0 - a_1 t_1^{(j)} - \ldots - a_m (t_1^{(j)})^m = -\mu_j w(t_1^{(j)}), \\ \ldots\ldots\ldots\ldots\ldots\ldots \qquad\qquad \ldots\ldots\ldots\ldots\ldots\ldots \\ f(t_{m+1}^{(j)}) - a_0 - a_1 t_{m+1}^{(j)} - \ldots - a_m (t_{m+1}^{(j)})^m = (-1)^{m+1} \mu_j w(t_{m+1}^{(j)}). \end{cases} \tag{3}$$

Система є системою $m+2$ алгебраїчних рівнянь з $m+2$ невідомими: $a_0, a_1, \ldots, a_m$ та μ.

3. Перевіряємо виконання рівності

$$|\mu_j| = \max_{x \in [a,b]} |f(x) - P_m^{(j)}(x)|/w(x) \equiv \rho_j. \tag{4}$$

Якщо рівність виконується, то у відповідності з теоремою 2 многочлен $P_m^{(j)}(x)$ і є шуканий многочлен найкращого наближення. При машинній реалізації алгоритму перевірку рівності заміняють перевіркою нерівності

$$\rho_j - |\mu_j| \leq \epsilon |\mu_j|, \tag{5}$$

де ϵ - допустима відносна помилка у визначенні похибки наближення. Можна, наприклад, прийняти $\epsilon = 10^{-2}$ чи $\epsilon = 10^{-3}$.

4. Якщо умова 4 чи 5 не виконується, то приймаємо $j := j+1$ і вибираємо наступне (уточнене) наближення до точок чебишовського альтернансу (наступний V-альтернанс). Далі виконання алгоритму повторюється починаючи з п.2.

   При обчисленнях на ЕОМ у цьому пункті іноді додатково перевіряються умови
   $$\left| t_k^{(j-1)} - t_k^j \right| < \eta, \quad k = \overline{0, m+1},$$
   де η - допустима помилка у визначенні точок альтернансу. Якщо остання нерівність справедлива для всіх точок $k = \overline{0, m+1}$, то вважаємо, що многочлен найкращого наближення знайдено.

1.2 Алгоритм Валле-Пуссена заміни точок альтернансу

Існує кілька методів заміни точок альтернансу. Можлива заміна одної або кількох точок одночасно. Найпростішим алгоритмом є алгоритм Є.Я. Ремеза з одноточковою заміною (алгоритм Валлє-Пуссена). Опишемо цей алгоритм.

Нехай при виконанні п.3 знайдена точка $\tilde{x}$, для якої справедливо $\rho_j = |\rho(\tilde{x})|$. Можливі три випадки взаємного розміщення точок V-альтернансу та точки $\tilde{x}$:

1. $t_0^{(j)} < \tilde{x} < t_{m+1}^{(j)}$
2. $\tilde{x} < t_0^{(j)}$
3. $\tilde{x} > t_{m+1}^{(j)}$

Розглянемо спосіб заміни точок V-альтернансу у кожному випадку.

1. Знайдемо ціле число v таке, що $t_v^{(j)} < \tilde{x} < t_{v+1}^{(j)}$. Якщо $\operatorname{sign}(\rho(\tilde{x})) = \operatorname{sign}(\rho(t_{m+1}^{(j)}))$, то приймаємо $t_v^{(j+1)} := \tilde{x}$, у протилежному випадку $t_{v+1}^{(j+1)} := \tilde{x}$. Решту точок V-альтеранансу не змінюємо.

2. Якщо $\operatorname{sign} \rho(\tilde{x}) = \operatorname{sign} \rho(t_0^{(j)})$, то приймаємо $t_0^{(j+1)} := \tilde{x}$, а решту точок V-альтернансу не змінюємо. Якщо це не так, то заміняємо усі точки альтернансу за формулами:

$$t_0^{(j+1)} := \tilde{x}; \quad t_k^{(j+1)} := t_{k-1}^{(j)}, \quad k = \overline{1, m+1}.$$

   У цьому випадку із V-альтернансу виключається точка $t_{m+1}^{(j)}$

3. Якщо $\operatorname{sign} \rho(\tilde{x}) = \operatorname{sign} \rho(t_{m+1}^{(j)})$, то приймаємо $t_{m+1}^{(j)} := \tilde{x}$. і решту точок V-альтернансу не змінюємо. Якщо це не так, то замінюємо усі точки V-альтернансу за формулами:

$$t_k^{(j+1)} := t_{k+1}^{(j)}, \quad k = \overline{0, m}; \quad t_{m+1}^{(j+1)} := \tilde{x}.$$

   У цьому випадку із V-альтернансу виключається точка $t_0^{(j)}$.

Отже черговий V-альтернанс відрізняєтся від попереднього тим, що точка $\tilde{x}$, у якій досягається максимум абсолютної величини зваженої похибки, вводиться у V-альтернанс замість однієї із старих точок. Відомо, що алгоритм Валле-Пуссена для заміни точок альтернансу при знаходженні найкращого наближення попередньої функції многочленом на проміжку $[a, b]$ збігається незалежно від початкового наближення до точок альтернансу.

Більш того у цьому випадку цей алгоритм збігається зі швидкістю гометричної прогресії у тому сенсі, що знайдуться такі числа A та $0 < q < 1$, що відхилення $E^{(k)}(f, W)$ многочлена $P_m^{(k)}(x)$ від функції $f(x)$ будуть задовольняти нерівності

$$E^{(k)}(f, W) - E(f, W) \leq Aq^k; \quad k = 1, 2, \ldots$$

Фактична швидкість збіжності залежить від диференціальних властивостей функції та використовуваного алгоритму заміни точок альтернансу. Відомо, що коли $f(x) \in C^{m+1}[a, b], w(x) = 1$ або $w(x) = f(x)$ і $f^{(m+1)}(x)$ не змінює знак при $x \in [a, b]$, то граничні точки проміжку $[a, b]$ є точками альтернансу. Тому у цьому випадку алгоритм Валле-Пуссена для наближення многочленами невисоких степенів $m = \overline{0, 2}$ практично не програє у швидкості порівняно з іншими алгоритмами типу Є.Я. Ремеза. Зазначимо, що всі перелічені властивості найкращого чебишовського наближення неперервної при $x \in [a, b]$ функції $f(x)$ многочленом справедливі також і для наближення табличної функції. Більш того, при заміні неперервної функції її значенями в точках $x_k = a + \frac{(b-a)k}{N}$ різниця між відповідними відхиленнями при $N \to \infty$ прямує до нуля.

1.3 Опис програми

Мета програми: знаходження найкращого чебишовського наближення для заданої функції.
Програма написана на мові програмування Python з використанням таких бібліотек: Sympy, Numpy, Plotly.

1.3.1 Вхідні дані

1. Початок інтервалу.
2. Кінець інтервалу.
3. Степінь многочлена.
4. Функція для апроксимації.
5. Точність (за замовчуванням 10^{-2}).

1.3.2 Вихідні дані

1. Коефіцієнти многочлена.
2. Графіки похибок на кожній ітерації.
3. Графік многочлена і функції.

2 Метод найменших квадратів

2.1 Опис алгоритму

Нехай в результаті вимірювань величини, яка описується функцією $y(x)$ при $x = x_1, x = x_2, \ldots, x = x_n, x_i \in [a, b], \quad i = \overline{1, n}$ отримаємо таблицю значень $y_i, i = \overline{1, n}$. За даними таблиці треба побудувати аналітичну формулу

$$\overline{y}(x) = f(x, a_1, \ldots, a_m), \tag{6}$$

яка залежить від $m \quad (m < n)$ параметрів $a_i, i = \overline{1, m}$, причому функція $\overline{y}(x)$ має "досить добре"наближувати функцію $y(x)$ на всьому проміжку $[a, b]$. Вигляд функції f і кількість параметрів у деяких випадках відомі на основі додаткових міркувань. В інших випадках вони визначаються за графіком, побудованим за відомими значеннями $y(x_i)$ так, щоб залежність (6) була досить простою і добре відображала результати спостережень. Якщо система рівнянь

$$\begin{cases} y_1 = f(x_1, a_1, \ldots, a_m), \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ y_n = f(x_n, a_1, \ldots, a_m) \end{cases} \tag{7}$$

має єдиний розв'язок, то він може бути знайдений з яких-небудь m рівнянь системи (7). Однак, у загальному випадку значення $yi, xi, i = \overline{1, n}$ є наближеними і точний вигляд залежності $\overline{y}(x)$ невідомий і через це система (7) переважно є несумісною. Тому визначимо параметри $a_1, \ldots, a_m$ так, щоб у деякому розумінні всі рівняння системи (7) задовольнялися з найменшою похибкою, точніше, щоб мінімізувати функцію

$$S(a_1, \ldots, a_m) = \sum_{i=1}^{n} [y_i - f(x_i, a_1, \ldots, a_m)]^2.$$

Такий метод розв'язання системи (7) називається методом найменших квадратів. Якщо функція $S(a_1, \ldots, a_m)$ досягає абсолютного мінімуму в області зміни параметрів $a_1, \ldots, a_m$, то, розв'язуючи систему

$$\frac{\partial S}{\partial a_k} = -2 \sum_{i=1}^{n} [y_i - f(x, a_1, \ldots, a_m)] \frac{\partial f(x, a_1, \ldots, a_m)}{\partial a_k}, \quad k = \overline{1, m},$$

знаходимо точки, в яких може бути екстремум. Вибравши той розв'язок, який належить області зміни параметрів $a_1, \ldots, a_m$ і в якому функція $S(a_1, \ldots, a_m)$ має абсолютний мінімум, знаходимо незалежні значення $a_1, \ldots, a_m$ Якщо $f(x, a_1, \ldots, a_m)$ лінійно залежить від параметрів $a_1, \ldots, a_m$, тобто

$$f(x, a_1, \ldots, a_m) = \sum_{j=1}^{m} f_j(x) a_j,$$

то система (7) набуває вигляду

$$y_i = \sum_{j=1}^{m} f_j(x) a_j, \quad i = \overline{1, n}. \tag{8}$$

Метод найменших квадратів розв'язування системи (8) полягає у тому, щоб визначити невідомі, які мінімізують суму квадратів нев'язок, тобто суму вигляду

$$S(a_1, \ldots, a_m) = \sum_{i=1}^{n} \left[y_i - \sum_{j=1}^{m} f_j(x) a_j \right]^2.$$

З умови мінімуму величини S як функції від $a_1, \ldots, a_m$ отримаємо систему лінійних алгебраїчних рівнянь

$$\frac{\partial S}{\partial a_k} = -2 \sum_{i=1}^{n} \left[y_i - \sum_{j=1}^{m} f_j(x) a_j \right] f_k(x_i) = 0, \quad k = \overline{1, m},$$

або

$$\sum_{i=1}^{n} \left[\sum_{j=1}^{m} f_j(x_i) a_j \right] f_k(x_i) = \sum_{i=1}^{n} f_k(x_i) y_i, \quad k = \overline{1, m}.$$

Розв'язок системи m лінійних алгебраїчних рівнянь з m невідомими вважаємо наближеним розв'язком системи.

2.2 Опис програми

Програму для пошуку апроксимації для функції методом найменших квадратів я написав на мові Python. Також зробив web сайт який можна переглянути за адресою http://least-squares.herokuapp.com/.

2.2.1 Вхідні дані

Користувачу пропонується ввести функцію яку він хоче апроксимувати методом найменших квадратів. Також степінь многочлена, інтервал на якому апроксимується функція, кількість точок які будуть використовуватися в методі найменших квадратів і кількість цифр після коми.

2.2.2 Вихідні дані

Після того як користувач натисне кнопку «Знайти», на екрані браузера появиться вигляд многочлена, максимальна похибка, та значення x в якому ця похибка досягається.

Висновки

У цій роботі я розглянув найкраще чебишовське наближення многочленами. Написав програму для знаходження коефіцієнтів такого многочлена. Також в програмі реалізув побудову графіків похибок на кожній ітерації, вивід максимальної похибки та значення аргументу при якому ця похибка досягається.

Також розглянув метод найменших квадратів. Написав програму яка реалізує алгоритм МНК на мові Python. Зробив web сайт для зручного пошуку апроксимації для функції.

Список використаної літератури

1. Демьянов В.Ф., Малоземов В.Н. Введение в минимакс. -М.: Наука, 1972. - 368 с.

2. Попов Б.А. Равномерное приближение сплайнами. -Киев: Наук. думка, 1989. - 272 с.

3. Попов Б.А., Теслер Г.С. Приближение функций для технических приложений. - Киев: Наук. думка, 1980. - 352 с.

4. Ремез Е.Я. Основы численных методов чебышовского приближения. Киев: Наук. думка, 1969, - 623 с.

5. Попов Б.О. Чисельні методи рівномірного наближення сплайнами. Конспект лекцій. -Львів: ЛДУ, 1992. - 92 с.

6. Самарский А.А., Гулин А.В. Численные методы. -М.: Наука, 1989. - 432 с.

7. Самарский А.А., Гулин А.В. Численные методы. -М.: Наука, 1989. - 432 с.

8. Кутнів М.В. Чисельні методи: Навчальний посібник.— Львів, 2010. –286 с.

9. https://plot.ly/ - для побудови графіків

10. http://www.sympy.org/ - для розв'язування систем

11. http://www.numpy.org/ - для наукових розрахунків

3 Додатки

3.1 Приклади виконання програми(чебишовське наближення)

Приклад 1

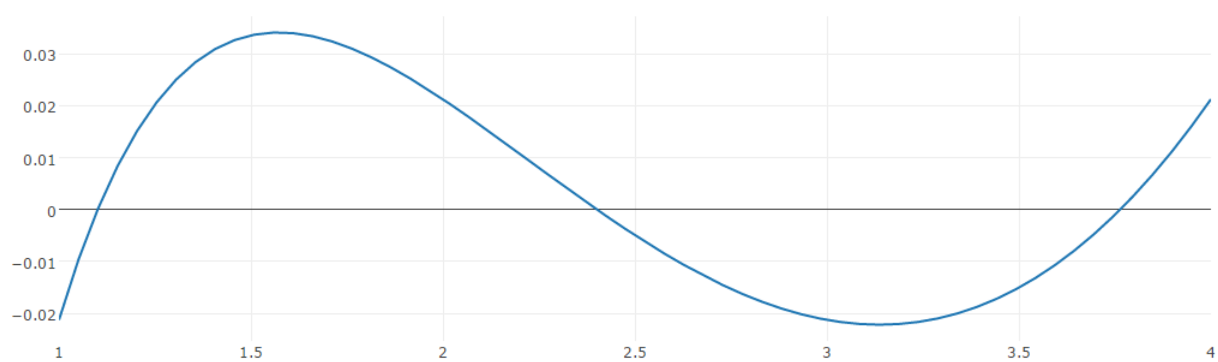
Знайдемо чебишовське наближення поліномом степеня 2 для функції $f(x) = ln(x)$ на проміжку $[1, 4]$. Точність ($\epsilon = 0.01$)

Ітерація 1

Точка альтернансу	1.0000000	2.0000000	3.0000000	4.0000000
Похибка	-0.0212374	0.0212374	-0.0212374	0.0212374

Максимальна похибка	0.0341106
Значення *x* в якому досягається максимальна похибка	1.5723191
Продовжуємо алгоритм бо	0.6061591 > 0.01
Аналітичний вигляд многочлена	$-\mathbf{0.1013x^2 + 0.9547x - 0.8321}$

Графік функції похибки

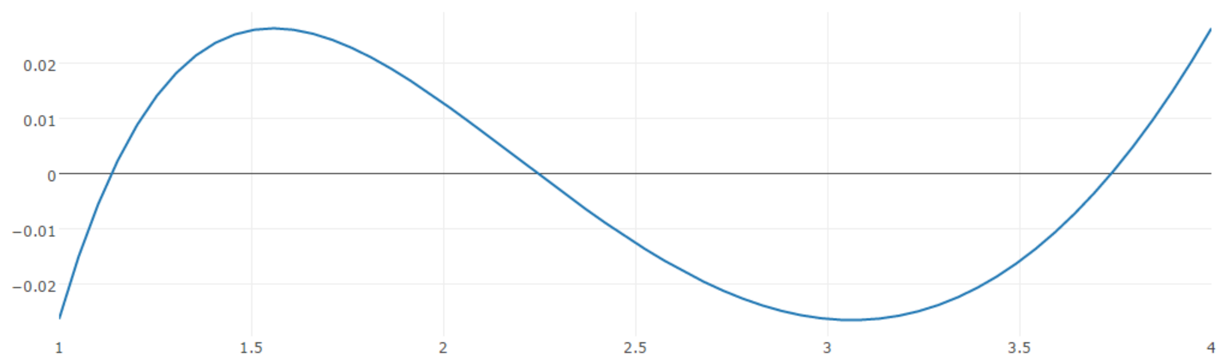


Ітерація 2

Точка альтернансу	1.0000000	1.5723191	3.0000000	4.0000000
Похибка	-0.0262962	0.0262962	-0.0262962	0.0262962

Максимальна похибка	-0.0265072
Значення x в якому досягається максимальна похибка	3.0644688
Алгоритм закінчено бо	0.0080264 < 0.01
Аналітичний вигляд многочлена	$\mathbf{-0.1047x^2 + 0.9682x - 0.8372}$

Графік функції похибки



Перевіримо умову завершення алгоритму: $\frac{\rho_j - |\mu_j|}{|\mu_j|} \leq \epsilon$.

$$\frac{0.0265072 - 0.0262962}{0.0262962} \approx 0.00802 < \epsilon = 0.01$$

Оскільки умова виконується то многочлен чебишовського наближення знайдено. Його вигляд:

$$P_2(x) = -0.10474x^2 + 0.96826x - 0.83723$$

3.2 Приклади виконання програми (МНК)

Функція, яку апроксимуємо

ln(x)

Степінь многочлена, яким апроксимуємо

3

Початок інтервалу

0.2

Кінець інтервалу

2.5

Кількість точок розбиття інтервалу

7

Кількість знаків після коми в коефіцієнтах

4

Результати роботи програми

$$0.3219x^3 - 1.8075x^2 + 3.7891x - 2.2697$$

Максимальна похибка: -0.106

x в якому макс. похибка: 0.401