|  |  |  |
| --- | --- | --- |
| 4주차 선형시스템 보고서 | | |
| 제출일 : 2024년 04월 1일 | | 작성자 : 이준용 |
| 구분 | 내용 | |
| 학습 범위와 내용 | 4주차 온라인 강의 내용 | |
| 정리 내용 | **수식을 넣기가 힘들어 수식관련 개념은 개인적으로 공부하였고 보고서는 개념위주로 정리했습니다.**  **INTERPOLATION AND CURVE FITTING**  **3.1 INTERPOLATION BY LAGRANGE POLYNOMIAL**  라그랑지 다항식은 주어진 데이터 포인트들을 통해 보간(Interpolation)을 수행하는 방법 중 하나입니다. 이 방법은 주어진 데이터 포인트들을 정확하게 지나가는 다항식을 구성합니다. 주어진 데이터 포인트가 {(x0, y0), (x1, y1), . . . , (xN , yN )} 일 때, 라그랑지 보간 다항식은 다음과 같이 정의됩니다:    여기서 *Lk*​(*x*)는 라그랑지 기초 다항식(Lagrange basis polynomial)으로서 다음과 같이 정의됩니다:    라그랑지 보간 다항식은 각 데이터 포인트에서의 함수 값을 고려하여 주어진 데이터를 정확하게 통과하는 다항식을 생성합니다. 이를 통해 주어진 데이터 포인트 사이에서 임의의 값을 추정할 수 있습니다. 그러나 라그랑지 다항식은 보간에 사용되는 데이터 포인트의 수가 많을수록 계산량이 증가하고, 고차 다항식을 생성할 경우 오버피팅(overfitting)의 위험이 있습니다.  **3.2 INTERPOLATION BY NEWTON POLYNOMIAL**  뉴턴 보간법은 함수 값이 주어진 n+1개의 데이터 포인트에서 n차 다항식을 사용하여 함수를 근사화하는 방법입니다. 이는 분할된 차이(divided difference)를 사용하여 다항식 계수를 효율적으로 계산하는 것이 특징입니다.  이 방법은 주어진 데이터 포인트에 대한 항을 반복적으로 추가하여 다항식을 구성합니다. {(x0, y0), (x1, y1), . . . , (xN , yN )}와 같은 데이터 포인트 집합이 주어졌을 때, 뉴턴 보간 다항식은 다음과 같이 정의됩니다:    여기서 *f*[*x*0​],*f*[*x*0​,*x*1​],…는 주어진 데이터 포인트에서 함수의 분할 차이를 나타냅니다.  계산 과정이 비교적 간단합니다.  새로운 데이터 포인트가 추가될 때, 기존 계산 결과를 활용하여 다항식을 쉽게 업데이트할 수 있습니다.양식의 맨 위  단점으로는  고차 다항식의 경우 오차가 누적될 수 있습니다.  Runge 현상이 발생할 가능성이 있습니다.  **3.3 APPROXIMATION BY CHEBYSHEV POLYNOMIAL**  Chebyshev 다항식은 [−1,1] 구간에서 정의된 특수한 다항식 집합입니다. 이 다항식들은 특정한 성질을 가지고 있어 함수 근사에 유용하게 쓰입니다.  k차 Chebyshev 다항식은 일반적으로 T  k  ​  (x)로 표기되며, 다음과 같은 점들을 만족합니다.  [−1,1] 구간에서 절대값이 최대 k+1개의 근을 가지고, 이 근들은 모두 등간격으로 위치합니다.  Chebyshev 다항식은 [−1,1] 구간 밖에서는 발산합니다.  Chebyshev 근사 정리  Chebyshev 근사 정리는 연속이고 절대값이 특정 구간에서 bounded인 함수 f(x)에 대해 다음과 같은 사실을 보장합니다.  최대 k차 다항식 p\_k(x)를 사용하여 f(x)를 근사할 때, Chebyshev 다항식을 이용한 근사가 \*\*최악의 근사 오차 (worst-case error)\*\*를 만듭니다.  **근사 과정**  근사하려는 함수 f(x)와 근사 구간 [a, b]를 정의합니다.  [a, b] 구간을 [−1,1] 구간으로 변환합니다. (선형 변환 수행)  최대 근사 차수 k를 선택합니다.  k차 Chebyshev 다항식 T\_k(x)를 이용하여 근사 다항식 p\_k(x)를 생성합니다.  p\_k(x)를 다시 원래의 근사 구간 [a, b]로 변환합니다.  장점  특정 구간에서 최악의 근사 오차를 최소화하는 근사를 수행할 수 있습니다.  단점  Chebyshev 다항식은 [−1,1] 구간 밖에서는 발산하기 때문에 다른 구간에 직접적으로 적용하기 어렵습니다.  계산 과정이 다소 복잡할 수 있습니다.  함수 f(x)=e  x  를 [−1,1] 구간에서 2차 Chebyshev 다항식을 이용하여 근사하는 예시입니다.  근사하려는 함수와 근사 구간 정의  f(x) = exp(x)  a = -1  b = 1  [a, b] 구간을 [-1, 1] 구간으로 변환  x = (2\*x - a - b) / (b - a)  최대 근사 차수 선택k = 2  k차 Chebyshev 다항식을 이용하여 근사 다항식 생성  p\_k(x) = T\_k(x)  p\_k(x)를 다시 원래의 근사 구간 [a, b]로 변환  p\_k(x) = exp((b - a) \* x / 2 + (a + b) / 2)  결과  p\_k(x) = 1.469535  **3.4 PADE APPROXIMATION BY RATIONAL FUNCTION**  Pade 근사는 특정 함수 f(x)를 주어진 차수의 분수 함수 (rational function)으로 근사하는 방법입니다. 이 근사는 특정 조건 하에 원 함수 f(x)의 테일러 급수 전개와 동일한 초기 항을 가지도록 만들어집니다.  함수 f(x)를 x=a 근처에서 m차 Pade 근사로 나타내면 다음과 같이 표현됩니다.    f(x)의 테일러 급수 전개와 처음 (m+n+1)개 항이 일치합니다.  특히, a ≠ 0 인 경우 f(a), f'(a), ..., f^(m+n)(a) 까지의 도함수 값과 일치합니다.  적절한 조건 하에, Pade 근사는 원 함수 f(x)보다 특정 구간에서 더 나은 근사값을 제공할 수 있습니다.  **3.5 INTERPOLATION BY CUBIC SPLINE**  Cubic Spline Interpolation: 보간을 위한 3차 스플라인  보간(interpolation)은 주어진 데이터 포인트 사이에 매끄러운 함수를 그려내는 방법입니다. Cubic Spline Interpolation은 이러한 보간 과정에서 3차 다항식 (cubic polynomial)을 사용하여 각 데이터 포인트를 연결하는 방법  \상대적으로 간단한 계산 과정  데이터 포인트를 모두 거쳐가는 (interpolation) 성질  매끄러운 (smooth) 함수를 생성  기본 원리  Cubic Spline Interpolation은 다음과 같은 세 가지 조건을 만족하는 3차 다항식 조각 (piecewise polynomial)으로 함수를 근사합니다.  통과 조건 (Interpolation Condition): 모든 스플라인 세그먼트 (spline segment)는 각 데이터 포인트를 지나야 합니다.  연속성 조건 (Continuity Condition): 두 스플라인 세그먼트가 만나는 지점에서 함수 값 (f(x))과 1차 미분 값 (f'(x))이 연속적이어야 합니다.  매끄러움 조건 (Smoothness Condition): 두 스플라인 세그먼트가 만나는 지점에서 2차 미분 값 (f''(x))이 연속적이어야 합니다. (경우에 따라 자연 스플라인 (natural spline)에서는 2차 미분 값이 0이 되도록 제약함)  **3.6 HERMITE INTERPOLATING POLYNOMIAL**  Hermite 보간은 단순히 데이터 포인트를 통과하는 다항식을 만드는 것 이상을 하는 보간 기법입니다. 데이터 포인트의 도함수 값 (또는 고차 도함수 값)까지 일치시키도록 다항식을 구성합니다. 이를 통해 결과 다항식은 함수의 값뿐만 아니라 국소적인 행동까지 포착할 수 있습니다.  데이터 포인트 (*x*0​,*y*0​), (*x*1​,*y*1​)과 그에 해당하는 1차 도함수 (*y*0′​,*y*1′​)을 고려합니다.    **3.7 TWO-DIMENSIONAL INTERPOLATION**  2차원 보간 (Two-Dimensional Interpolation)  2차원 보간은 불규칙한격자에서 얻은 데이터 points를 기반으로 해당 영역 안의 임의의 지점에서의 값을 근사하는 방법입니다.  보통 데이터는 격자 상의 매듭점 (grid nodes)에서만 측정되지만, 우리는 꼭 그 지점에서만 값을 알 필요가 없을 수도 있습니다. 2차원 보간을 이용하면 이러한 격자 사이의 값을 추정하여 전체적인 함수 또는 데이터의 거동을 더 세밀하게 이해할 수 있습니다.  쌍 선형 보간 (Bilinear Interpolation): 가장 간단한 2차원 보간 방법 중 하나입니다. 주변하는 4개의 격자 점과 그 값을 이용하여 가중 평균을 통해 중간 지점의 값을 근사합니다.  **3.8 CURVE FITTING**  선형 회귀: 이 방법은 데이터 포인트를 가장 잘 근사하는 직선을 설정합니다. 선형 관계를 모델링하는 데 적합합니다.  다항 회귀: 이 기술은 특정 차수의 다항 함수를 데이터에 맞춥니다. 선형 회귀에 비해 더 복잡한 관계를 포착할 수 있습니다.  지수 회귀: 이 방법은 지수 함수를 사용하여 지수 성장 또는 감소 추세를 나타내는 데이터를 모델링합니다.  과적합: 복잡한 함수를 선택하거나 Fitting process가 정규화되지 않으면 결과 함수는 훈련 데이터에 밀접하게 접근하지만 보이지 않는 데이터에서는 성능이 저하될 수 있습니다 (과적합).  모델 선택: 데이터에 가장 적합한 함수를 선택하는 것은 매우 중요합니다. 부적절한 함수 선택은 부정확한 결과를 초래할 수 있습니다.  **3.9 FOURIER TRANSFORM**  주기 함수를 더 간단한 사인파 함수들의 합으로 분해하는 푸리에 해석의 개념을 확장하여 일반적인 함수 f(t)를 복소수 사인파 함수들의 무한 합으로 변환하는 것입니다.      종류  연속 푸리에 변환 (Continuous Fourier Transform, CTFT): 이는 앞서 설명한 수식과 같은 형태이며, 이론적인 배경을 설명하는 데 주로 사용됩니다. 실제 응용에서는 아래와 같은 이산 푸리에 변환이 더 많이 사용됩니다.  이산 푸리에 변환 (Discrete Fourier Transform, DFT): 주기적인 이산 신호를 분석하는 데 사용되며, 컴퓨터 과학 분야에서 널리 활용됩니다.  빠른 푸리에 변환 (Fast Fourier Transform, FFT): DFT 계산을 효율적으로 수행하는 알고리즘입니다. 컴퓨터 응용 프로그램에서 푸리에 변환을 수행할 때는 대부분 FFT가 사용됩니다.    **3.9.3 Interpolation by Using DFS**  DFT (Discrete Fourier Transform):  신호 x[n]에 대한 N-point DFT를 수행하여 X(k)를 얻습니다. DFT는 신호를 주파수 성분으로 분해합니다.  수정된 역 DFT:  다음 공식 (3.9.5)을 사용하여 보간된 신호를 재구성합니다.  N은 신호의 총 샘플 수입니다.  합은 주파수 성분의 절반 (|k| < N/2)에 대해 반복됩니다. 이는 실제 신호의 경우 음수 주파수 성분이 양수 주파수 성분과 중복되기 때문입니다.  X(k)는 DFT에서 얻은 주파수 성분을 나타냅니다.  j는 허수 단위 (√(-1))입니다.  t는 보간 값을 추정하고자 하는 연속 시간 변수입니다.  T는 샘플링 주기 (샘플 사이의 시간 간격)입니다.  Real{X(k)ej2πkt/NT}: 이 항은 X(k)의 크기와 위상 정보를 사용하여 각 주파수 성분을 재구성합니다.  합은 스펙트럼의 양수 절반 (|k| < N/2)만 고려합니다.  k = 0과 k = N/2에 대한 특수 경우는 각각 상수 성분과 나이키스트 주파수 성분을 고려하기 위해 포함됩니다. | |
| 질문 내용 | 1. **DFT를 이용한 보간 방법의 정확도를 평가하기 위한 지표는 무엇이며, 제일 많이 사용되는 지표가 있다면 어떤 장점 때문에 많이 쓰이나요?** 2. **스플라인 보간은 데이터를 부드럽게 보간하면서도 과적합을 방지하는 데 유용한 반면, 다항식 보간은 주어진 데이터를 정확하게 지나가는 고차 다항식을 사용하여 데이터를 보간한다고 이해했습니다.**   **그렇다면 스플라인 보간을 사용하여 주어진 데이터를 추정하는 데 어떤 영역에서 가장 유용한가요?** | |