제공된 코드는 DAMP (Discord Aware Matrix Profile) 알고리즘의 MATLAB 구현입니다. DAMP는 시계열에서 디스코드를 식별하기 위해 설계되었습니다. 디스코드란 큰 시계열 내에서 동일한 길이의 모든 다른 부분 시퀀스와 최대한 다른 부분 시퀀스를 의미합니다. 다시 말해, 나머지 데이터와는 맞지 않는 패턴입니다.

코드의 주요 구성 요소를 분석해보겠습니다:

**1. 함수 시그니처**

function [discord\_score, position] = DAMP\_2\_0(T, SubsequenceLength, location\_to\_start\_processing, varargin)

주요 함수 `DAMP\_2\_0`는 다음을 입력으로 받습니다:

- `T`: 시계열 데이터.

- `SubsequenceLength`: 디스코드를 찾기 위해 고려할 부분 시퀀스의 길이.

- `location\_to\_start\_processing`: 시계열에서 처리를 시작할 위치.

- `varargin`: 추가 선택적 매개변수를 허용합니다.

**2. 입력 파싱:**

함수는 선택적 매개변수를 처리하기 위해 MATLAB의 `inputParser`를 사용합니다. 두 개의 선택적 매개변수가 정의되어 있습니다:

- `lookahead`: 알고리즘이 얼마나 지연되는지를 정의합니다. 기본적으로 로 설정됩니다.

- `enable\_output`: 함수가 출력 메시지를 표시할지 결정하는 부울 값입니다. 기본적으로 활성화되어 있습니다.

**3. 초기 출력:**

`enable\_output`이 true인 경우 함수는 시계열, 사용 중인 DAMP 버전 및 사용자에게 몇 가지 힌트에 대한 초기 정보를 출력합니다.

**4. 오류 처리:**

함수는 입력 시계열에서 특정 조건을 확인하고 경고 또는 오류를 제공합니다. 예를 들어, 시계열에 일정한 영역이 포함되어 있는지, `location\_to\_start\_processing`이 유효한지, `SubsequenceLength`가 합리적인 범위 내에 있는지 확인합니다.

**5. 초기화:**

몇몇 초기 변수와 배열이 설정됩니다. 여기에는 `Left\_MP`가 포함되며, 이는 왼쪽(시간으로 거꾸로)만 보는 특별한 Matrix Profile입니다.

**6. DAMP 알고리즘:**

DAMP 알고리즘의 핵심은 두 개의 주요 루프에서 구현됩니다:

- 첫 번째 루프는 상대적으로 높은 디스코드 점수를 얻기 위해 시계열의 접두사를 처리합니다.

- 두 번째 루프는 접두사를 제외한 나머지 테스트 데이터를 처리합니다.

알고리즘은 무차별 대입 방식과 MASS 알고리즘을 사용하여 Matrix Profile 값을 계산합니다. 또한 불필요한 계산을 제거하고 프로세스를 가속화하기 위해 룩어헤드 메커니즘을 사용합니다.

**7. 결과:**

함수는 가지치기 비율을 계산하고 시계열에서 최상위 디스코드를 식별합니다. `enable\_output`이 true인 경우 결과를 표시하고 Left Matrix Profile과 정규화된 시계열을 플롯합니다.

**8. 도우미 함수:**

`MASS\_V2`: 이 함수는 FFT 기반 MASS 알고리즘을 사용하여 쿼리와 시계열 사이의 거리 프로필을 계산합니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

- `contains\_constant\_regions`: 이 함수는 시계열에 일정하거나 거의 일정한 영역이 포함되어 있는지 확인합니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

요약

DAMP는 시계열에서 디스코드를 효율적으로 식별하기 위해 설계된 알고리즘입니다. 제공된 구현은 무차별 대입, MASS 알고리즘 및 룩어헤드 메커니즘의 조합을 사용하여 검색을 최적화합니다. 함수는 또한 적절한 매개변수를 선택하는 데 도움이 되도록 광범위한 오류 처리와 사용자 피드백을 제공합니다.

알고리즘 분석

**1. MASS 알고리즘:**

MASS\_V2 함수를 통해 구현됩니다. 주어진 시계열과 쿼리 사이의 거리 프로필을 빠르게 계산하기 위해 사용됩니다. FFT (Fast Fourier Transform)를 활용하여 시계열 데이터와 쿼리 사이의 유사도를 효율적으로 계산합니다.

**2. 룩어헤드 메커니즘:**

알고리즘의 성능을 향상시키기 위해 미래의 데이터를 미리 검토합니다. 현재 부분 시퀀스를 기준으로 미래의 데이터를 룩어헤드 범위만큼 검토하고, "best so far" 디스코드 점수보다 낮은 값을 가진 부분 시퀀스를 프루닝합니다.

**3. 부분 시퀀스 검색:**

시계열 내의 각 부분 시퀀스에 대해 디스코드 점수를 계산합니다. 룩어헤드 메커니즘과 결합하여 불필요한 부분 시퀀스 검색을 최소화합니다.

**4. 에러 및 경고 처리:**

시계열에 일정한 영역이 포함되어 있는지, 시작 위치가 유효한지, 부분 시퀀스의 길이가 합리적인 범위 내에 있는지 등의 조건을 확인합니다. 사용자에게 경고나 오류 메시지를 제공하여 적절한 매개변수 선택을 도와줍니다.

**5. 결과 출력 및 시각화:**

enable\_output 매개변수가 활성화된 경우, 알고리즘의 결과와 함께 유용한 정보와 힌트를 출력합니다. Left Matrix Profile과 정규화된 시계열을 함께 플롯하여 시각적으로 결과를 표시합니다.

**6. 디스코드 점수 계산:**

시계열 내의 모든 부분 시퀀스에 대한 디스코드 점수를 계산하고, 가장 높은 점수를 가진 부분 시퀀스의 위치를 찾습니다.

이러한 주요 알고리즘 및 메커니즘을 통해 DAMP는 시계열 데이터 내에서 가장 일치하지 않는 부분 시퀀스, 즉 디스코드를 효과적으로 식별할 수 있습니다.

1. MASS 알고리즘

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

2. 룩어헤드(Lookahead) 매커니즘

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

우리가 사용할 DAMP\_multidim

전반적인 리뷰

이 코드는 시계열 데이터에 대한 다차원 디스코드 탐색을 수행하는 MATLAB 함수로 구성되어 있습니다. 디스코드는 시계열 데이터에서 특이한 패턴을 찾는 데 사용되는 기술입니다. 코드를 분석하면 다음과 같은 주요 부분들을 확인할 수 있습니다:

**1. DAMP\_Multidim 함수:**

입력: T (시계열 데이터), dimension\_num (데이터의 차원 수) 출력: 없음. 그러나 함수는 결과를 그래프로 표시하고, 디스코드의 점수와 위치를 출력합니다.

**2. MASS\_V2 함수:**

입력: x (데이터), y (쿼리) 출력: dist (거리)

**3. DAMP\_Multidim 함수의 주요 부분:**

데이터 T의 차원이 dimension\_num과 일치하는지 확인하고, 필요한 경우 데이터를 전치합니다. xcorr 함수를 사용하여 첫 번째 차원의 자기 상관을 계산합니다. findpeaks 함수를 사용하여 주요 피크를 찾아서 부분 시퀀스의 길이를 결정합니다. 다양한 초기화 작업을 수행합니다. 데이터의 접두사 부분에 대한 처리를 수행합니다. 나머지 데이터에 대한 처리를 수행합니다. 결과를 그래프로 표시하고, 디스코드의 점수와 위치를 출력합니다.

**4. MASS\_V2 함수:**

이 함수는 시계열 데이터 x와 쿼리 y 사이의 거리 프로파일을 계산합니다. movmean 및 movstd 함수를 사용하여 데이터의 이동 평균 및 이동 표준 편차를 계산합니다. FFT (Fast Fourier Transform)를 사용하여 x와 y의 점곱을 효율적으로 계산합니다. 결과 거리 프로파일 dist를 반환합니다.

**5. 코드의 목적:**

이 코드는 주어진 다차원 시계열 데이터에서 디스코드를 탐색하려는 것입니다. 디스코드는 데이터에서 가장 특이한 부분 시퀀스를 나타냅니다. 이 코드는 디스코드를 찾기 위해 다양한 최적화 및 가지치기 기술을 사용합니다.

알고리즘

이 코드는 다차원 시계열 데이터에서 디스코드를 탐색하는 알고리즘을 구현하고 있습니다. 디스코드는 시계열 데이터에서 가장 특이한 부분 시퀀스를 의미합니다. 주요 알고리즘 및 사용된 기술은 다음과 같습니다:

**1. 자기 상관 (Autocorrelation):**

xcorr 함수를 사용하여 데이터의 첫 번째 차원에 대한 자기 상관을 계산합니다. 이 자기 상관은 데이터 내에서 패턴의 반복성을 파악하는 데 도움이 됩니다.

**2. 피크 탐색 (Peak Detection):**

findpeaks 함수를 사용하여 자기 상관 결과에서 주요 피크를 탐색합니다. 이 피크는 부분 시퀀스의 길이를 결정하는 데 사용됩니다.

**3. 거리 프로파일 계산 (Distance Profile Computation):**

MASS\_V2 함수를 사용하여 주어진 쿼리와 데이터 사이의 거리 프로파일을 계산합니다. 이 거리 프로파일은 디스코드 탐색에 필요한 핵심 요소입니다.

**4. 프루닝 (Pruning):**

알고리즘은 "lookahead" 기술을 사용하여 불필요한 계산을 줄입니다. 특정 부분 시퀀스가 현재까지의 최고 디스코드 점수보다 낮은 점수를 가질 가능성이 있으면 해당 부분 시퀀스의 계산을 건너뜁니다.

**5. DAMP (Dynamic Algorithm for Matrix Profile):**

DAMP는 Matrix Profile을 계산하는 동적 알고리즘입니다. 이 코드에서는 DAMP의 변형을 사용하여 디스코드를 탐색합니다. 알고리즘은 특정 부분 시퀀스의 거리 프로파일을 계산하고, 이 거리 프로파일의 최소값을 사용하여 Matrix Profile을 업데이트합니다.

**6. FFT (Fast Fourier Transform):**

MASS\_V2 함수 내에서 FFT를 사용하여 시계열 데이터와 쿼리 사이의 점곱을 효율적으로 계산합니다. FFT는 시계열 데이터의 처리 속도를 크게 향상시킵니다.

요약하면, 이 코드는 다차원 시계열 데이터에서 디스코드를 효율적으로 탐색하기 위해 여러 최적화 및 가지치기 기술을 사용하는 알고리즘을 구현하고 있습니다.

결론: 딥러닝이 아닌 통계로 구하는 이상 탐지이다.

\

PHM 2010 data challenge에 사용할 수 있는가?

스크린샷, 텍스트, 라인, 직사각형이(가) 표시된 사진

자동 생성된 설명

각 열에 대한 plot

스크린샷, 다채로움, 예술이(가) 표시된 사진

자동 생성된 설명스크린샷, 라인, 직사각형, 다채로움이(가) 표시된 사진

자동 생성된 설명

스크린샷, 예술이(가) 표시된 사진

자동 생성된 설명스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷, 디자인이(가) 표시된 사진

자동 생성된 설명스크린샷, 직사각형, 라인이(가) 표시된 사진

자동 생성된 설명

스크린샷, 그래프, 라인, 텍스트이(가) 표시된 사진

자동 생성된 설명

코드 실행

텍스트, 폰트, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

결과

텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 라인, 폰트이(가) 표시된 사진

자동 생성된 설명

값은 나오는데 이게 정확히 무엇을 의미하는지는 파악하기 어려움 – 좀더 공부가 필요함.

간단한 EDA

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 다채로움이(가) 표시된 사진

자동 생성된 설명

텍스트, 도표, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

라인, 도표, 평행, 그래프이(가) 표시된 사진

자동 생성된 설명

도표, 라인, 평행, 직사각형이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 평행, 사각형이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 패턴, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

