순천향대학교
SOON CHUN HYANG UNIVERSITY

2024년 2학기

선형대수

8-2장

Chapter 08

- SECTION 08-1 직교 행렬
- SECTION 08-2 그람-슈미트 과정
- SECTION 08-3 QR 분해

SCH

SECTION 08-3 QR 분해(1)

- QR 분해의 작동 방식
  - GS는 행렬을 직교 행렬 **Q**로 변환
  - **Q**는 원래 행렬과 당연히 달라짐(원래 행렬이 직교 행렬이 아니라고 가정할 때) 따라서 원래 행렬에 대한 정보가 손실
  - 이 '손실된' 정보는 **Q**에 곱하는 다른 행렬 **R**에 쉽게 복구해서 저장할 수 있음

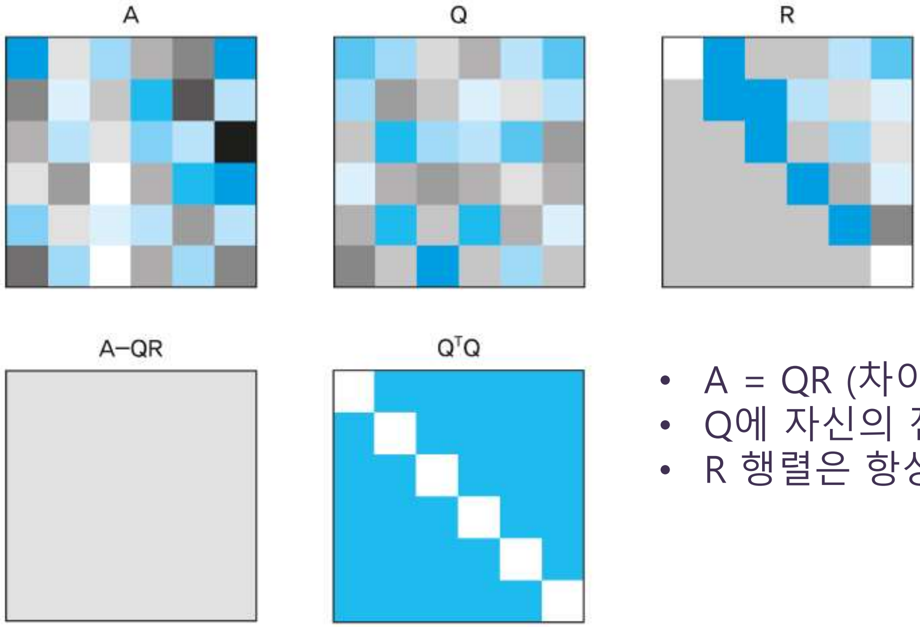
  - 직교 행렬의 이점 - 역을 계산할 필요 없이 행렬 방정식을 풀 수 있음
- Q: Orthonormal matrix, R: upper triangular matrix

$$A = QR$$
$$Q^T A = Q^T QR$$
$$Q^T A = R$$

SECTION 08-3 QR 분해(2)

- 정방 행렬의 QR 분해를 계산하는 파이썬 코드

```
A = np.random.randn(6,6)
Q,R = np.linalg.qr(A)
```



- A = QR (차이는 영행렬)
- Q에 자신의 전치를 곱하면 단위 행렬이 됨
- R 행렬은 항상 상삼각

SECTION 08-3 QR 분해(3)

8.3.1 Q와 R의 크기

- 경제형 또는 축소 - 높은 행렬($M > N$)에서 열이 N 개인 **Q** 행렬을 생성
- 완전형 또는 전체 - 열이 M 개인 **Q** 행렬 생성

	A	Q	Q^TQ	QQ^T	R
정사각 완전 계수	$M \times M$ $r=M$	$M \times M$ $r=M$	I_M	I_M	$M \times M$ $r=M$
정사각 특이	$M \times M$ $r=k<M$	$M \times M$ $r=M$	I_M	I_M	$M \times M$ $r=k$
높은 완전형	$M>N$ $r=k$	$M \times M$ $r=M$	I_M	I_M	$M \times N$ $r=k$
높은 경제형	$M>N$ $r=k$	$M \times N$ $r=N$	I_N	?	$M \times N$ $r=k$
넓은	$M<N$ $r=k$	$M \times M$ $r=M$	I_M	I_M	$M \times N$ $r=k$

A의 크기에 따른 Q와 R의 크기: '?'는 행렬 원소가 A의 값에 따라 달라짐을 나타내며, 즉 행렬이 단위 행렬이 아님을 의미함

SECTION 08-3 QR 분해(4)

- 행렬 **A**가 높을 때 **Q**가 정방 행렬

```
A = np.array([ [1,-1] ]).T
Q,R = np.linalg.qr(A,'complete')
Q*np.sqrt(2) # 정수 값을 얻기 위해 sqrt(2)로 크기 조정
>> array([[-1., 1.],
        [ 1., 1.]])
```

- 선택적인 두 번째 인수에 'complete'을 넣으면 전체 QR 분해가 됨
- 이를 기본값인 'reduced'로 설정하면 **Q**와 **A**가 같은 크기인 경제형 QR 분해가 됨
- N 개의 열을 가진 행렬에서 $M > N$ 개 이상의 직교벡터를 만들 수 있기 때문에 **Q**의 계수는 항상 최대한으로 가능한 계수, 따라서 모든 정방 **Q** 행렬에서는 M, 경제형 **Q**에서는 N
- **R**의 계수는 **A**의 계수와 동일
- 직교화로 인한 **Q**와 **A**의 계수 차이는 **A**의 열 공간이 $\mathbb{R}^M$ 의 저차원 하위 공간일지라도 **Q**는 $\mathbb{R}^M$ 전체를 생성한다는 것을 의미

> - QR 분해는 모든 행렬의 크기와 계수에 대해 고유하지 않음
>   즉, $\mathbf{Q}_1 \neq \mathbf{Q}_2$인 A = $\mathbf{Q}_1\mathbf{R}_1$ 과 $\mathbf{A} = \mathbf{Q}_2\mathbf{R}_2$를 구할 수 있음
> - 그러나 모든 QR 분해 결과는 이 절에서 설명한 것과 동일한 특성을 가짐
> - 추가 제약 조건(예를 들어 R 의 모든 대각선의 값이 양수)이 주어지면 QR 분해를 고유하게 만들 수 있지만, 대부분의 경우 불필요하며 파이썬이나 MATLAB에서 구현되어 있지 않음

SECTION 08-3 QR 분해(6)

8.3.2 QR과 역

- QR 분해를 사용하면 역행렬을 수치적으로 더 안정적으로 계산
- QR 분해 공식을 작성하고 방정식의 양쪽을 반전
  - **A**의 역행렬은 **R**의 역행렬에 **Q**의 전치를 곱하여 구할 수 있음
  - **Q**는 하우스홀더 변환 알고리즘 덕분에 수치적으로 안정적
  - **R**은 단순히 행렬 곱셈의 결과이기 때문에 역시 수치적으로 안정적

$$
\begin{aligned}
A &= QR \\
A^{-1} &= (QR)^{-1} \\
A^{-1} &= R^{-1}Q^{-1} \\
A^{-1} &= R^{-1}Q^{T}
\end{aligned}
$$