

사랑은 언제까지나 떨어지지 아니하되 예언도 폐하고 방언도 그치고 지식도 폐하리라. (고린도전서 13:4)  
Love never fails. But where there are prophecies, they will cease; where there are tongues, the will be stilled; where there is knowledge, it will pass away. (1 Corinthians 13:4)



**NOTE:** The following materials have been compiled and adapted from the numerous sources including my own. Please help me to keep this tutorial up-to-date by reporting any issues or questions. Send any comments or criticisms to [idebtor@gmail.com](mailto:idebtor@gmail.com). Your assistances and comments will be appreciated.

## 제 1-3 강: 행렬 *Matrix*

### 학습 목표

- 행렬을 이해하고, 행렬 연산을 익힌다.

### 학습 내용

- 왜 행렬인가?
- 행렬에 관한 용어의 이해
- 행렬의 사칙 연산
- 행렬의 전치, 역행렬, 단위행렬

이번시간에는 행렬에 관해서 짚고 넘어가려합니다. 행렬을 배우신 분들도 있고 행렬을 배우지 않으신 분들도 있을 것이라 생각합니다. 행렬을 이미 공부하신 분이라면 복습을 하는 시간이라고 생각하셔도 좋습니다.

### 왜 행렬인가?

선형 대수학에서는 여러 개의 선형 방정식을 간결하게 표기할 수 있는 방법을 제공합니다. 예를 들면, 다음과 같은 두 개의 일차 방정식이 있다고 가정해 봅시다.

$$4x_1 - 5x_2 = -12 \quad (1)$$

$$-2x_1 + 3x_2 = 8 \quad (2)$$

이러한 2개의 변수에 2개의 방정식이 있으므로,  $x_1, x_2$ 에 관하여 해를 구할 수 있습니다. 이것을 행렬 방식으로 간결하게 표시하면 다음과 같습니다.

$$Ax = b \quad (3)$$

$$A = \begin{bmatrix} 4 & -5 \\ -2 & 3 \end{bmatrix}, b = \begin{bmatrix} -12 \\ 8 \end{bmatrix} \quad (4)$$

이와같이 행렬은 "연립방정식에서 계수와 변수를 분리하여 쓰면 어떨까?"라는 생각에서 탄생했습니다. 믿거나 말거나 이것은 역사적 사실입니다. 시간만 따지고 본다면 행렬보다 행렬식이 먼저 탄생한 셈입니다.

기계학습에서는 기본적으로 많은 학습자료를 다루어야 합니다. 이때 필수적으로 필요한 것이 행렬이며 나아가서는 텐서입니다. 행렬을 이용하면 여러가지 이점들이 있습니다.

- 첫째로 행렬은 이러한 모든 계산을 아주 간결하게 압축하여 표기가 가능하게 해줍니다. 우리같은 사람들에게는 아주 큰 이점이죠. 간결하니까 우리 같은 사람들이 실수를 할 수 있는 기회를 줄여주기 때문에 고마운거죠. 기계 학습 덕분에 선형 대수학, 행렬도 공부하면서 감사가 넘치고, 행복했으면 좋겠습니다.
- 둘째로는 프로그래밍에 적용하기 쉽습니다. 많은 컴퓨터 프로그래밍 언어가 행렬을 이해할 수 있으며, 실제 계산은 반복적이기 때문에 컴퓨터가 빠르고 효율적으로 해낼 수 있습니다. 더 감사한 일이죠, 이런 것을 여러분은 곧 경험할 것입니다. 감사에 또 감사하는 것이죠

위의 말들을 요약하자면 행렬은 우리가 해야 하는 작업을 간결하고 쉽게 표현하며, 컴퓨터의 입장에서는 빠르고 효율적으로 계산을 이해할 수 있다는 것입니다.

## 1. 용어의 이해

### 1.1 스칼라

크기와 방향을 가지는 벡터에 대비하는 개념으로, 크기만 있고 방향을 가지지 않는 양이 스칼라입니다. 사칙연산이 가능한 물리량입니다. 질량, 부피, 거리, 속도, 정수, 유리수, 실수, 등 모든 것들이 스칼라입니다.

### 1.2 벡터

수학에서 벡터는 크기와 방향을 가져야 합니다. 예를 들면, 속도, 가속도, 운동량, 전기장 등의 물리량이 여기에 속합니다. 기계학습에서 1차원 배열을 벡터로 간주합니다.

### 1.3 행렬

행렬 $matrix$ 은 수 혹은 수식 등을 네모꼴로 배열한 것입니다. 2차원의 배열을 행렬이라 하며, 배열의 전체를 괄호로 묶어 표시합니다.

### 1.4 텐서

기계학습에서 텐서는 임의의 차원을 갖고 있는 배열이라는 의미로 사용합니다. 간단히 말하자면 텐서는 0차원인 스칼라, 1차원인 벡터, 2차원인 행렬, 혹은 N 차원, 고차원의 모든 행렬을 포함합니다.

## 2. 행렬의 정의

행렬(Matrix)는 숫자로 이루어진 사각형 모양의 테이블입니다. 행렬에 대해 이것보다 더 복잡한 것은 없습니다. (행렬에 대해 이것보다 더 간단한 것은 없습니다.) 여러분이 스프레드 시트(엑셀)를 사용해 보았다면, 격자에 배열된 숫자로 작업하는 데에 이미 익숙할 것입니다. 어떤 사람들은 테이블이라고 부르기도 하며, 행렬이라고도 부릅니다. 다음과 같이 행렬은 숫자로 이루어진 격자 혹은 테이블입니다.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad (5)$$

2 x 3 행렬의 예시

여기서  $A \in \mathbb{R}^{m \times n}$  이라고 표기하며, 이는 m 행과 n 열로 구성되어 있으며, 원소들은 실수(R, real number)집합에 속한다는 것을 의미합니다.

행렬(Matrix)은 1개 이상의 수나 식을 사각형의 배열로 나열한 것입니다. 가로줄을 행(row), 세로줄을 열(column)이라고 부릅니다. Column에는 기둥이라는 뜻도 있다는 것을 생각해본다면 세로줄이라는 뜻도 쉽게 이해할 수 있을 것입니다.

## 3. 행렬과 원소, 표기

- 행렬 A의 i번째 행과 j번째 열에 있는 원소는  $a_{ij}$ 로 표기합니다.

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \quad (6)$$

- 행렬 A의 j번째 열은  $a_j$ 로 표기합니다.

$$A_{m,n} = \begin{pmatrix} | & | & \cdots & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & \cdots & | \end{pmatrix} \quad (7)$$

- 행렬  $A$ 의  $i$ 번째 행은  $a_i^T$ 로 표기합니다.

$$A_{m,n} = \begin{pmatrix} - & a_1^T & - \\ - & a_2^T & - \\ & \vdots & \\ - & a_m^T & - \end{pmatrix} \quad (8)$$

## 4. 행 벡터와 열 벡터

선형 대수학에서, 행 벡터는  $1 \times m$  행렬,  $m$  원소들로 구성된 단일 행 행렬이며 다음과 같이 나타낼 수 있습니다.

$$\mathbf{w} = [w_1 \ w_2 \ \cdots \ w_m] \quad (9)$$

마찬가지로, 열 벡터는  $m \times 1$  행렬, 즉  $m$  원소들의 구성된 단일 열 행렬이며 다음과 같이 표시합니다.

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} \quad (10)$$

이렇게 하나의 행으로만 구성되어 있으면 행 벡터, 하나의 열로만 구성되어 있으면 열 벡터라고 부릅니다. 일반적으로 행 벡터인지 열 벡터인지 언급이 없다면, 대개의 경우 열 벡터를 의미합니다.

## 5. 행렬의 곱

두 행렬 즉  $A \in \mathbb{R}^{m \times n}$ ,  $B \in \mathbb{R}^{n \times p}$ 의 곱은 다음과 같이 정의 합니다.

$$C = AB \in \mathbb{R}^{m \times p} \quad \text{단, } C_{ij} = \sum_{k=1}^n a_{ik}b_{kj}. \quad (11)$$

행렬의 곱은 다른 행렬의 연산들과 같이 크기가 맞는 경우에만 계산할 수 있습니다.

여기서 크기가 맞다는 것은 앞의 행렬의 열(column)의 수와 뒤의 행렬의 행(row)의 수가 같다는 것을 말합니다. 곱셈의 결과 나오는 행렬의 크기는 (앞의 행렬의 행(row)의 수) x (뒤의 행렬의 열(column)의 수)가 됩니다. 즉 앞의 행렬이  $M \times N$  크기이고 뒤의 행렬이  $N \times P$  크기인 경우에 결과는  $M \times P$  크기의 행렬이 됩니다.

행렬  $A$ 와  $B$ 의 곱( $AB = C$ )

$$\begin{array}{ccc} A & B & = C \\ (M \times N) & (N \times P) & = (M \times P) \end{array}$$

일치

그림 1: 행렬의 곱에서 대응하는 차원의 원소 수의 일치

다음은 행렬이 서로 곱해진 것을 보여주는 식입니다.

$$AB = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix} \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \end{bmatrix} \quad (12)$$

$$= \begin{bmatrix} x_{11}y_{11} + x_{12}y_{21} & x_{11}y_{12} + x_{12}y_{22} & x_{11}y_{13} + x_{12}y_{23} \\ x_{21}y_{11} + x_{22}y_{21} & x_{21}y_{12} + x_{22}y_{22} & x_{21}y_{13} + x_{22}y_{23} \\ x_{31}y_{11} + x_{32}y_{21} & x_{31}y_{12} + x_{32}y_{22} & x_{31}y_{13} + x_{32}y_{23} \\ x_{41}y_{11} + x_{42}y_{21} & x_{41}y_{12} + x_{42}y_{22} & x_{41}y_{13} + x_{42}y_{23} \end{bmatrix} \quad (13)$$

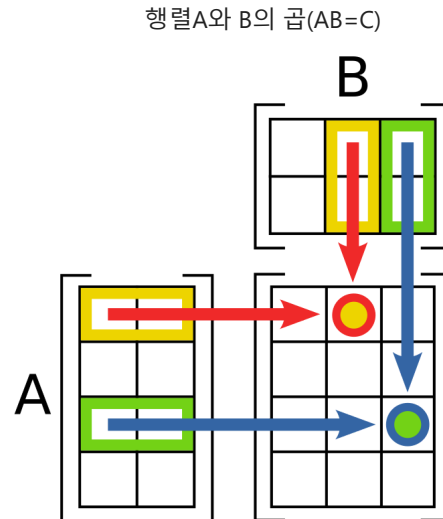


그림 2: 행렬A와 B의 곱(AB=C)

출처:[https://en.wikipedia.org/wiki/Matrix\\_multiplication](https://en.wikipedia.org/wiki/Matrix_multiplication)

식을 보면 앞의 행렬의 행(row)과 뒤의 행렬의 열(column)끼리 곱하고 더한 것을 알 수 있습니다. 일반적으로 말한다면 앞행렬의 앞에서부터 N번째 원소와 뒷행렬의 위에서부터 N번째 행렬들끼리 곱한 것을 더한 것입니다. 행렬의 크기가 커진다면 아래와 같이 됩니다.

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{j1} & a_{j2} & \dots & a_{jn} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1k} & \dots & b_{1q} \\ b_{21} & b_{22} & \dots & b_{2k} & \dots & b_{2q} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nk} & \dots & b_{nq} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & \dots & c_{1k} & \dots & c_{1q} \\ c_{21} & c_{22} & \dots & c_{2k} & \dots & c_{2q} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{j1} & c_{j2} & \dots & c_{jk} & \dots & c_{jp} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{m1} & c_{m2} & \dots & c_{mk} & \dots & c_{mq} \end{pmatrix} \quad (14)$$

크기가 큰 행렬의 곱셈 계산

여기서  $c_{jk}$ 의 값은  $c_{jk} = a_{j1} \times b_{1k} + a_{j2} \times b_{2k} + \dots + a_{jn} \times b_{nk}$ 가 됩니다.



## 5.1 두 벡터의 곱(내적)

열 벡터로 표현되는 두 개의 벡터  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ 가 주어졌을 때,  $\mathbf{x}^T \mathbf{y}$ 를 내적(inner product, dot product)이라고 합니다.

$$\mathbf{x}^T \mathbf{y} \in \mathbb{R} = [x_1 \ x_2 \ \dots \ x_n] \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad (15)$$

$$= x_1y_1 + x_2y_2 + \dots + x_ny_n \quad (16)$$

$$= \sum_{i=1}^n x_iy_i \quad (17)$$

이와 같은 행렬의 특별한 경우에 해당합니다. 이 때는

- 항상  $\mathbf{x}^T \mathbf{y} = \mathbf{y}^T \mathbf{x}$ 가 성립합니다.
- 두 벡터의 크기가 같습니다.
- 결과 값은 스칼라입니다.

위의 식 끝에 나오는  $\sum$ 은 "summation, 더하기"는 수학 기호이며, 밑첨자는  $i=1$ 부터  $i$ 을 하나씩 증가시키며 윗첨자  $n$ 이 될 때까지 나타나는 항목을 모두 더하라는 표시입니다.

## 5.2 두 벡터의 곱(외적)

열 벡터로 표현되는 두 개의 벡터  $\mathbf{x} \in \mathbb{R}^m, \mathbf{y} \in \mathbb{R}^n$ 가 주어질 때, (두 벡터의 크기가 반드시 같을 필요가 없음), 두 벡터를  $\mathbf{x}\mathbf{y}^T$  외적(outer product)하면 행렬을 얻게 됩니다.

$$\mathbf{x}^T \mathbf{y} \in \mathbb{R}^{m \times n} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} (y_1 \ y_2 \ \cdots \ y_n) = \begin{pmatrix} x_1 y_1 & x_1 y_2 & \cdots & x_1 y_n \\ x_2 y_1 & x_2 y_2 & \cdots & x_2 y_n \\ \cdots & & \cdots & \cdots \\ x_m y_1 & x_m y_2 & \cdots & x_m y_n \end{pmatrix} \quad (18)$$

외적이 유용할 때가 있습니다. 모든 원소가 1인 열 벡터를  $\mathbf{I} \in \mathbb{R}^n$ 라고 합시다. 또한 행렬  $A \in \mathbb{R}^{m \times n}$ 의 모든 열이 동일한 벡터  $\mathbf{x} \in \mathbb{R}^m$ 라고 하면, 이를 다음과 같이 표기할 수 있습니다.

$$A_{m,n} = \begin{pmatrix} | & | & \cdots & | \\ x & x & \cdots & x \\ | & | & & | \end{pmatrix} = \begin{pmatrix} x_1 & x_1 & \cdots & x_1 \\ x_2 & x_2 & \cdots & x_2 \\ \vdots & \vdots & \ddots & \vdots \\ x_m & x_m & \cdots & x_m \end{pmatrix} \quad (19)$$

$$= \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} [1 \ 1 \ \cdots \ 1] = \mathbf{x}\mathbf{I}^T \quad (20)$$

## 5.3 행렬 곱의 특징

$k$ 를 스칼라,  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 를 각 공식에서 연산이 정의되도록 크기를 정한 행렬이라고 하면 다음이 성립합니다.

- 결합법칙  
 $\mathbf{A}(\mathbf{B}\mathbf{C}) = (\mathbf{A}\mathbf{B})\mathbf{C}$
- 분배법칙  
 $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{A}\mathbf{B} + \mathbf{A}\mathbf{C}$   
 $(\mathbf{B} + \mathbf{C})\mathbf{A} = \mathbf{B}\mathbf{A} + \mathbf{C}\mathbf{A}$
- $k(\mathbf{A}\mathbf{B}) = (k\mathbf{A})\mathbf{B} = \mathbf{A}(k\mathbf{B})$
- 교환법칙  
 $\mathbf{A} \cdot \mathbf{B} \neq \mathbf{B} \cdot \mathbf{A}$

앞에서 말했듯이 행렬의 곱은 앞의 행렬의 행과 뒤의 행렬의 열이 대응되는 특성이 있기 때문에 일반적으로 교환법칙이 성립하지 않습니다.

간단한 행렬로 확인해볼 수 있습니다.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix} \quad (21)$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 4 \\ 1 & 2 \end{bmatrix} \quad (22)$$

## 6. 행렬의 덧셈과 뺄셈

행렬의 덧셈과 뺄셈은 대응하는 원소끼리 즉 같은 위치에 있는 원소끼리 가능합니다. 행렬의 크기가 서로 같은 경우에만 할 수 있습니다. 대응하는 원소끼리 계산하고 대응하는 자리에 위치시키면 됩니다. 행렬의 덧셈과 뺄셈은 다음과 같습니다.

$$\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} + \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix} = \begin{pmatrix} x_{11} + y_{11} & x_{12} + y_{12} \\ x_{21} + y_{21} & x_{22} + y_{22} \end{pmatrix} \quad (23)$$

$$\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} - \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix} = \begin{pmatrix} x_{11} - y_{11} & x_{12} - y_{12} \\ x_{21} - y_{21} & x_{22} - y_{22} \end{pmatrix} \quad (24)$$

행렬의 덧셈과 뺄셈

행렬의 상수배

행렬의 상수배는 더 어렵지 않습니다. 모든 원소에 상수배를 해주면 됩니다.

$$c \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix} = \begin{pmatrix} cx_{11} & cx_{12} \\ cx_{21} & cx_{22} \end{pmatrix} \quad (25)$$

행렬의 상수배

## 7. 행렬의 전치<sup>transpose</sup>

전치는 행렬 내의 원소를 왼쪽 위에서 오른쪽 아래를 가로지르는 대각선을 기준으로 서로 위치를 바꾼 것입니다. 다른 말로, 행과 열의 인덱스를 바꾸는 것을 전치<sup>transpose</sup>라고 합니다.

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

다음의 동영상 파일을 보면 이해에 도움이 될 것입니다.

그림 3: 행렬의 전치(출처:<https://en.wikipedia.org/wiki/Transpose>)

다음과 같이  $M \times N$  행렬  $\mathbf{A}^{m \times n}$ 의 전치 행렬  $\mathbf{A}^T$ 는  $N \times M$  행렬이 됩니다. (T는 transpose를 뜻합니다.)

$$A_{m,n} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mj} & \dots & a_{mn} \end{pmatrix} \quad (26)$$

$$(27)$$

$$A^T = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{i1} & \dots & a_{m1} \\ a_{12} & a_{22} & \dots & a_{i2} & \dots & a_{m2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{1j} & a_{2j} & \dots & a_{ij} & \dots & a_{mj} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{1n} & a_{2n} & \dots & a_{in} & \dots & a_{nm} \end{pmatrix} \quad (28)$$

행렬의 전치

$$A_{m,n} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2j} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mj} & \dots & a_{mn} \end{pmatrix} \quad (29)$$

$$(30)$$

$$A^T = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{i1} & \dots & a_{m1} \\ a_{12} & a_{22} & \dots & a_{i2} & \dots & a_{m2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{1j} & a_{2j} & \dots & a_{ij} & \dots & a_{mj} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{1n} & a_{2n} & \dots & a_{in} & \dots & a_{nm} \end{pmatrix} \quad (31)$$

행렬의 전치

아래는 간단한 예입니다.

$$B = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \quad B^T = \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} \quad (32)$$

전치행렬의 예시

간단히 설명하자면 열을 왼쪽에서부터 순서대로 행으로 옮기면 됩니다

전치의 특성

$$(A^T)^T = A$$

$$(AB)^T = B^T A^T$$

$$(A + B)^T = A^T + B^T$$

행/열 벡터의 전치<sup>transpose</sup>

행 벡터의 전치 행렬은 열 벡터이며, 열 벡터의 전치 행렬은 행 벡터입니다.

$$[x_1 \ x_2 \ \cdots \ x_m]^T = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \quad (33)$$

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}^T = [x_1 \ x_2 \ \cdots \ x_m] \quad (34)$$

## 8. 단위행렬

다음으로는 단위행렬에 대해 알아보도록 하겠습니다. 단위행렬이란 주대각선은 모두 1이고 나머지 성분은 모두 0인 행렬로  $I$ 로 적습니다. 여기서  $I$ 는 Identity를 의미합니다. 행렬에 단위행렬을 곱하여도 행렬에는 변화가 없습니다. 즉  $A \cdot I = A$ 입니다. 따라서 곱셈(내적)의 항등원이 됩니다.

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdots \quad (35)$$

단위행렬

---

## 참고자료

- [Jupyter Notebook 행렬 표기](#)
- [Udacity Deep Learning Nanodegree](#)

## 학습 정리

- 행렬의 사용 목적
- 행렬에 관한 용어의 이해
- 행렬의 사칙 연산
- 행렬의 전치, 역행렬, 단위행렬

---

In [ ]: