

# 인공지능을 활용한 운영분석 프로젝트

## 3학년 전공 역량 강화교육

권용현

육군사관학교 수학과

2026년 2월

# 3학년 전공 역량 강화 수업 계획

- ▶ 2/4(수) : Latex 기초 학습
  - ▶ Overleaf 계정 생성
- ▶ 2/6(금) : 인공지능을 활용한 운영분석 프로젝트
  - ▶ 운영분석 관련 자유주제 (수학·통계 개념 소개, 문제 풀이, 데이터 분석 등)
  - ▶ 개인 프로젝트. 단, 최종 발표자료는 Latex beamer로 만들어져야함.
  - ▶ 생성형 AI(Gemini pro 등) 이용 가능.
- ▶ 2/9(월) : 프로젝트 발표
  - ▶ 발표 5분 + 질문 2분
  - ▶ 보고서 불필요. PDF 발표자료만 이용.

# An Interactive Introduction to L<sup>A</sup>T<sub>E</sub>X

## 1부: 기초

Dr John D. Lees-Miller

February 3, 2026



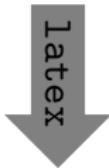
# 왜 LATEX인가?

- ▶ 문서를 아주 예쁘게 만들 수 있다
  - ▶ 특히 수학식(수식 조판)에 강하다
- ▶ 과학자들이 과학자를 위해 만들었다
  - ▶ 크고 활발한 커뮤니티가 있다
- ▶ 강력하고 확장 가능하다 — 기능을 얼마든지 늘릴 수 있다
  - ▶ 논문, 발표자료, 스프레드시트, ... 등 다양한 목적의 패키지가 있다

# 어떻게 작동하나요?

- ▶ 문서를 일반 텍스트(plain text)로 작성하고, 구조와 의미를 나타내는 **명령어**(commands)를 함께 적습니다.
- ▶ latex 프로그램이 이 텍스트와 명령어를 처리(컴파일)해서 보기 좋은 결과물을 만들어 줍니다.

스페인의 비는 \emph{주로} 평야에 내린다.



스페인의 비는 주로 평야에 내린다.

# 명령어와 출력 결과의 예시...

```
\begin{itemize}  
\item 차(Tea)  
\item 우유(Milk)  
\item 비스킷(Biscuits)  
\end{itemize}
```

- ▶ 차(Tea)
- ▶ 우유(Milk)
- ▶ 비스킷(Biscuits)

```
\begin{figure}  
\includegraphics{gerbil}  
\end{figure}
```



```
\begin{equation}  
\alpha + \beta + 1  
\end{equation}
```

$$\alpha + \beta + 1 \quad (1)$$

## 관점(태도) 바꾸기

- ▶ “어떻게 보이게 할지”가 아니라 “그게 무엇인지”를 명령어로 표현하세요.
- ▶ 내용(콘텐츠)에 집중하세요.
- ▶ 서식(디자인)은 L<sup>A</sup>T<sub>E</sub>X에게 맡기세요.

# 시작하기

- ▶ 최소한의 L<sup>A</sup>T<sub>E</sub>X 문서:

```
\documentclass{article}  
\begin{document}  
Hello World! % your content goes here...  
\end{document}
```

- ▶ 명령어는 백슬래시(*backslash*) \ 로 시작합니다.
- ▶ 모든 문서는 \documentclass 명령으로 시작합니다.
- ▶ 중괄호 {} 안의 인자(*argument*)는 만들 문서의 종류를 뜻합니다: 예를 들어 article.
- ▶ 퍼센트 기호 % 는 주석(*comment*)의 시작입니다 — L<sup>A</sup>T<sub>E</sub>X는 그 줄의 나머지를 무시합니다.

# 시작하기: Overleaf 사용하기

- ▶ Overleaf는  $\text{\LaTeX}$  문서를 작성할 수 있는 웹사이트입니다.
- ▶  $\text{\LaTeX}$ 를 자동으로 ‘컴파일’해서 결과를 바로 보여줍니다.

예제 문서를 **Overleaf**에서 열려면 여기를 클릭

원활한 사용을 위해 Google Chrome 또는 최신 FireFox 사용을 권장합니다.

- ▶ 다음 슬라이드들을 보면서, Overleaf 예제 문서에 직접 타이핑해 보세요.
- ▶ 진짜로요! 따라 치면서 익히는 게 제일 빠릅니다.

# 텍스트 조판하기

- ▶ `\begin{document}` 와 `\end{document}` 사이에 본문 텍스트를 입력합니다.
- ▶ 대부분은 그냥 평소처럼 텍스트를 입력하면 됩니다.

단어는 한 개 이상의  
공백으로 구분됩니다.

문단은 한 줄 이상의  
빈 줄로 구분됩니다.

단어는 한 개 이상의  
공백으로 구분됩니다.

문단은 한 줄 이상의 빈 줄로  
구분됩니다.

## ▶ 소스

파일에서의 공백(스페이스)은 출력에서는 대부분 하나로 정리됩니다.

스페인의      비는  
주로              평야에  
내린다.

스페인의 비는 주로 평야에  
내린다.

# 텍스트 조판하기: 주의사항

- ▶ 따옴표(quotation mark)는 약간 까다롭습니다:  
왼쪽은 백틱 `'`, 오른쪽은 아포스트로피 `'` 를 씁니다.

작은따옴표: `'text'`.

큰따옴표: ```text''`.

작은따옴표: `'text'`.

큰따옴표: `“text”`.

- ▶ L<sup>A</sup>T<sub>E</sub>X에서 몇몇 문자는 특별한 의미를 가집니다:



퍼센트 기호(percent sign)



샵/해시 기호(hash sign)



앰퍼샌드(ampersand)



달러 기호(dollar sign)

- ▶ 이런 문자를 그냥 입력하면 에러가 납니다. 출력에 그대로 보여주고 싶다면, 백슬래시로 이스케이프(escape) 해야 합니다.

`\$ \% \& \# !`

`$ % & # !`

# 오류(에러) 다루기

- ▶  $\text{\LaTeX}$ 가 문서를 컴파일하는 과정에서 해석이 꼬이면, 에러와 함께 멈춥니다. 출력이 나오려면 에러를 먼저 고쳐야 합니다.
- ▶ 예를 들어 `\emph` 를 `\meph` 라고 잘못 쓰면,  $\text{\LaTeX}$ 는 “meph”라는 명령을 모르기 때문에 “undefined control sequence” 에러가 납니다.

## 에러 대처 팁

1. 당황하지 마세요! 에러는 누구나 납니다.
2. 에러가 생기면 바로 고치세요 — 방금 입력한 부분이 원인인 경우가 많습니다.
3. 에러가 여러 개면, **첫 번째 에러부터** 해결하세요 — 원인이 그 위에 있을 수도 있습니다.

# 조판 연습 1

다음을 L<sup>A</sup>T<sub>E</sub>X로 조판해 보세요: <sup>1</sup>

2006년 3월, 의회는 그 한도를 추가로 \$0.79 조(Trillion) 달러 올려 \$8.97 조 달러로 만들었는데, 이는 GDP의 약 68%입니다. 2008년 10월 4일 기준으로, “Emergency Economic Stabilization Act of 2008”은 현재의 부채 한도를 \$11.3 조 달러로 올렸습니다.

연습 파일을 Overleaf에서 열려면 클릭

- ▶ 힌트: 특별한 의미를 가진 문자들(\$, %, ...)을 조심하세요!
- ▶ 시도해 본 뒤, 해설(해답)을 보려면 클릭.

---

<sup>1</sup>[http://en.wikipedia.org/wiki/Economy\\_of\\_the\\_United\\_States](http://en.wikipedia.org/wiki/Economy_of_the_United_States)

# 수학식 조판하기: 달러 기호(\$)

- ▶ 왜 달러 기호  $\$$  가 특별할까요? 텍스트 안에서 수학 모드 (*math mode*)를 표시하는 데 쓰기 때문입니다.

% 별로 좋지 않음:

서로 다른 양의 정수  $a$ ,  $b$ 에 대하여  
 $c = a - b + 1$  이라 하자.

% 훨씬 좋음:

서로 다른 양의 정수  $\$a\$$ ,  $\$b\$$ 에 대하여  
 $\$c = a - b + 1\$$  이라 하자.

서로 다른 양의 정수  $a$ ,  $b$ 에 대하여  $c = a - b + 1$  이라 하자.

서로 다른 양의 정수  $a$ ,  $b$ 에 대하여  $c = a - b + 1$  이라 하자.

- ▶ 달러 기호는 항상 쌍으로 써야 합니다 — 시작 1개, 끝 1개.
- ▶ L<sup>A</sup>T<sub>E</sub>X는 수학에서의 간격(spacing)을 자동으로 처리하며, 사용자가 친 공백은 대부분 무시합니다.

Let  $\$y=mx+b\$$  be  $\backslash ldots$

Let  $\$y = m x + b\$$  be  $\backslash ldots$

Let  $y = mx + b$  be ...

Let  $y = mx + b$  be ...

# 수학식 조판하기: 표기법

- ▶ 위첨자(superscript)는 캐럿  $\wedge$ , 아래첨자(subscript)는 언더스코어  $\_$  를 씁니다.

```
$y = c_2 x^2 + c_1 x + c_0$
```

$$y = c_2 x^2 + c_1 x + c_0$$

- ▶ 위/아래첨자가 길면, 중괄호  $\{ \}$  로 묶어서 “한 덩어리”로 만들어야 합니다.

```
$F_n = F_{n-1} + F_{n-2}$ % 이런 실수 자 $F_n$  합 $F_{n-1} + F_{n-2}$ 
```

```
$F_n = F_{\{n-1\}} + F_{\{n-2\}}$ % 이렇게 써야 한 $F_{\{n-1\}} + F_{\{n-2\}}$ 
```

- ▶ 그리

스 문자나 자주 쓰는 표기는 명령어로 쉽게 입력할 수 있습니다.

```
$\mu = A e^{Q/RT}$
```

$$\mu = A e^{Q/RT}$$

```
$\Omega = \sum_{k=1}^n \omega_k$
```

$$\Omega = \sum_{k=1}^n \omega_k$$

# 수학식 조판하기: 디스플레이 수식(Displayed Equations)

- ▶ 길고 복잡한 수식은 본문 안에 얹지로 넣기보다, `\begin{equation}` 과 `\end{equation}` 으로 한 줄을 따로 써서 디스플레이(display) 합니다.

이차방정식의 근은 다음과 같이 주어진다:

```
\begin{equation}
x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}
\end{equation}
```

여기서 `$a$`, `$b$`, `$c$`는 `\ldots`

이차방정식의 근은 다음과  
같이 주어진다:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (2)$$

여기서  $a, b, c$ 는 ...

주의: L<sup>A</sup>T<sub>E</sub>X는 수학 모드에서 공백을 대부분 무시하지만, 수식 안에 “빈 줄(blank line)”은 처리하지 못합니다 — 수식 안에는 빈 줄을 넣지 마세요.

# 잠깐: 환경(Environments)

- ▶ `equation`은 환경(*environment*)입니다 — “문맥/상황(*context*)”라고 생각하면 됩니다.
- ▶ 같은 명령어라도 환경이 달라지면 출력이 달라질 수 있습니다.

본문에서는

```
$ \Omega = \sum_{k=1}^n \omega_k $
```

처럼 쓸 수 있고, 또는

```
\begin{equation}
\Omega = \sum_{k=1}^n \omega_k
\end{equation}
```

처럼 따로 표시할 수도 있습니다.

본문에서는  $\Omega = \sum_{k=1}^n \omega_k$   
처럼 쓸 수 있고, 또는

$$\Omega = \sum_{k=1}^n \omega_k \quad (3)$$

처럼 따로 표시할 수도 있습니다.

- ▶ `equation` 환경에서는  $\Sigma$ 가 더 크게 나오고, 아래첨자/위첨자의 위치도 달라집니다. 그런데도 우리는 같은 명령어를 썼습니다.

사실 `...$` 는 `\begin{math} ... \end{math}` 로도 쓸 수 있습니다.

# 잠깐: 환경(Environments)

- ▶ `\begin` 과 `\end`로 다양한 환경을 만들 수 있습니다.
- ▶ `itemize` 와 `enumerate` 환경은 목록(list)을 만들어 줍니다.

```
\begin{itemize} % 글머리표(bullet)
\item 비스킷
\item 차
\end{itemize}
```

▶ 비스킷

▶ 차

```
\begin{enumerate} % 번호(number)
\item 비스킷
\item 차
\end{enumerate}
```

1. 비스킷

2. 차

## 잠깐: 패키지(Packages)

- ▶ 지금까지 사용한 명령어와 환경은 LATEX에 기본으로 포함된 기능입니다.
- ▶ 패키지(package)는 추가 명령어/환경을 제공하는 라이브러리입니다. 무료로 사용 가능한 패키지가 수천 개나 있습니다.
- ▶ 쓰고 싶은 패키지는 프리앰블(preamble)에 `\usepackage` 로 불러와야 합니다.
- ▶ 예: American Mathematical Society(미국수학회)의 amsmath 패키지.

```
\documentclass{article}  
\usepackage{amsmath} % 프리앰블  
\begin{document}  
% 이제 여기서 amsmath의 기능을 사용할 수 있습니다...  
\end{document}
```

# 수학식 조판하기: amsmath 예시

- ▶ equation\* (equation-star)는 번호 없는 수식을 만들 때 씁니다.

```
\begin{equation*}
\Omega = \sum_{k=1}^n \omega_k
\end{equation*}
```

$$\Omega = \sum_{k=1}^n \omega_k$$

- ▶ LATEX는 붙어 있는 글자를 “변수의 곱”으로 해석하는데, 그게 항상 우리가 원하는 의미는 아닙니다. amsmath에는 흔한 수학 연산자를 위한 명령어들이 정의되어 있습니다.

```
\begin{equation*} % 나쁨!
\min_{x,y} (1-x)^2 + 100(y-x^2)^2
\end{equation*}
\begin{equation*} % 좋음!
\min_{x,y} [(1-x)^2 + 100(y-x^2)^2]
\end{equation*}
```

$$\min_{x,y} (1-x)^2 + 100(y-x^2)^2$$

$$\min_{x,y} (1-x)^2 + 100(y-x^2)^2$$

- ▶ 다른 연산자는 \operatorname{}로 직접 만들 수도 있습니다.

```
\begin{equation*}
\beta_i =
\frac{\operatorname{Cov}(R_i, R_m)}{\operatorname{Var}(R_m)}
\end{equation*}
```

$$\beta_i = \frac{\text{Cov}(R_i, R_m)}{\text{Var}(R_m)}$$

## 수학식 조판하기: amsmath 예시

- ▶ 등호(=) 기준으로 여러 줄 수식을 정렬하려면 `align*` 환경을 사용합니다.

$$\begin{aligned}(x + 1)^3 &= (x + 1)(x + 1)(x + 1) \\&= (x + 1)(x^2 + 2x + 1) \\&= x^3 + 3x^2 + 3x + 1\end{aligned}$$

```
\begin{align*}
(x+1)^3 &= (x+1)(x+1)(x+1) \\
&= (x+1)(x^2 + 2x + 1) \\
&= x^3 + 3x^2 + 3x + 1
\end{align*}
```

- ▶ 앤퍼샌드 `&` 는 왼쪽 열(= 앞)과 오른쪽 열(= 뒤)을 구분합니다.
- ▶ 백슬래시 두 번 `\\"` 는 줄바꿈(새 줄)을 뜻합니다.

## 조판 연습 2

다음을  $\text{\LaTeX}$ 로 조판해 보세요:

서로 독립이고 동일분포(i.i.d.)인 확률변수열  $X_1, X_2, \dots, X_n$ 이 있고,  $E[X_i] = \mu$ ,  $\text{Var}[X_i] = \sigma^2 < \infty$ 라고 하자. 또한

$$S_n = \frac{1}{n} \sum_{i=1}^n X_i$$

를 이들의 평균(표본평균)이라 하자. 그러면  $n \rightarrow \infty$ 일 때, 확률변수  $\sqrt{n}(S_n - \mu)$ 은 분포수렴하여 정규분포  $N(0, \sigma^2)$ 가 된다.

연습 파일을 **Overleaf**에서 열려면 클릭

- ▶ 힌트:  $\infty$ 의 명령어는 `\infty`입니다.
- ▶ 시도해 본 뒤, 해설(해답)을 보려면 클릭.

# 1부 마무리

- ▶ 축하합니다! 여러분은 이미 다음을 배웠습니다 . . .
  - ▶ LATEX에서 텍스트를 조판하는 법
  - ▶ 다양한 명령어를 사용하는 법
  - ▶ 여러가 났을 때 대처하는 법
  - ▶ 예쁜 수학식을 조판하는 법
  - ▶ 여러 가지 환경(environment)을 사용하는 법
  - ▶ 패키지를 불러오는 법
- ▶ 정말 대단합니다!
- ▶ 2부에서는 LATEX로 더 “구조적인 문서”를 작성하는 법을 다룹니다: 섹션, 상호참조, 그림, 표, 참고문헌(bibliography) 등. 다음 시간에 만나요!