



Introduction To Scientific Computing

Basics of Latex

Dr. Fintan Healy
Room 1.31, Queens Building
fintan.healy@bristol.ac.uk

bristol.ac.uk

Lecture 4

Basics of Latex

Lecture Aims



Introduction to LaTeX

Basic commands for text, figures and mathematics

Introduction to Overleaf

Academic Dissemination

Academic Dissemination

- So far, we have learnt the basics of programming, including variables, loops, conditionals, functions, files, and arrays.
- And you've learnt programming in both Python and MATLAB
- But what's the best way to disseminate your results?

Academic Dissemination

The Challenge of Document Writing

- Technical and academic writing requires **precision, clarity, and consistency**.
- Common tasks include managing **equations, figures, references**, and formatting.
- Choosing the right tool impacts **productivity** and document **quality**.
 - Just like coding, you need to pick the right tool for the job

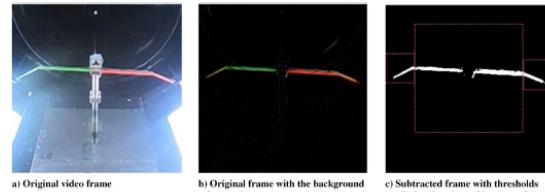


Fig. 4 An overview of the computer vision algorithm used to measure the roll and fold angles of the wind tunnel model.

Table 2 Test matrix for the transient release of the rolling model

Parameter	Value
Model configuration	Removed, fixed, free10, free30
Wind tunnel velocity, $m s^{-1}$	15, 20, 25, 30
Aileron deflection, deg	7, 14, 21

shown in Figs. 2 and 4a, the leading edge of the left and right sides of the model were painted green and red, respectively. By removing a background image from each frame (Fig. 4b) and applying a threshold for green and red pixels, a binary image could be created showing the location of the leading edge (Fig. 4c). From this image, a line segment from the center of the model could be selected to only show pixels that are part of the inner wing, allowing the roll angle to be calculated by fitting a line through these pixels. Two further regions of interest could then be selected from the binary image that represent the possible locations of the FFWTs, enabling the fold angles to be calculated in a similar fashion.

C. Methodology

The 7 ft by 5 ft low-speed closed-return wind tunnel at the University of Bristol was used to test each wing configuration at four airflow velocities⁴² and three aileron angles, as summarized in the test matrix shown in Table 2. At each test point the model was first locked using the electromechanical brake in the horizontal position, the aileron angle was set, then the brake was released, and the rig allowed to spin freely for 10 seconds.⁴³ The process was repeated three times at each test point. GoPro data, used to calculate the fold angles, were recorded for one min at each test point.

IV. Mathematical Modeling

A. Structural Model

A simplified three-degree-of-freedom (3-DoF) model, seen in Fig. 5, of a rectangular wing of span s and chord c , was developed to capture the pertinent dynamics as well as to further explore the design space. This mathematical model is a simplification, non-linear and includes the effect of gravity; it consists of an inner wing and two FFWTs that are represented as point masses with a specific mass and moment of inertia. The percentage of the span along the semichord line taken up by FFWTs is denoted as α , and the generalized coordinates of this system are the roll angle ϕ (about the center of the model) and the fold angle θ_1 , θ_2 of each FFWT:

⁴²Selected velocities were between 15 and 30 $m s^{-1}$, giving an approximate Reynolds number range of 70,000–140,000.

⁴³A video of an example test can be seen at <https://youtu.be/MX0KRM2uhk> (reviewed 16 April 2021).

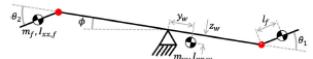


Fig. 5 Diagrammatic representation of the 3-DoF mathematical model, as viewed from downstream of the model.

$$\mathbf{q} = \begin{bmatrix} \phi \\ \theta_1 \\ \theta_2 \end{bmatrix} \quad (8)$$

All motion is assumed to occur in a two-dimensional plane; hence the moment arm of each FFWT with respect to the hinge (I_z) is corrected by the cosine of the fold angle to account for the angle between the spanwise direction and the direction perpendicular to the hinge. Additionally, the experimental model was not balanced about the center of rotation due to differences in material densities between 3D printed parts and the addition of the control mechanism. Hence, as shown from Fig. 5 the center of mass (CoM) of the main wing (m_w) is offset from the center of rotation by y_w and z_w in the yaw and z direction, respectively. However, as indicated in Table 1, the magnitude of this offset is in the order of 2 mm.

The equations of motion of this system were found using the Euler–Lagrange method using the python package sympy⁴⁴, and they were of the form

$$M(\mathbf{q}, \dot{\mathbf{q}})\ddot{\mathbf{q}} - f(\mathbf{q}, \dot{\mathbf{q}}) = 0 \quad (9)$$

where M is the mass matrix and f is a matrix of additional forces.

B. Aerodynamic Model

The aerodynamic forces acting on the wing were modeled using quasi-steady modified strip theory at a discrete number of panels, with the main wing being split into 20 uniform panels and each FFWT being split into 10. At each panel the local AoA is calculated as

$$\alpha(y) = \alpha_{geom} + \frac{w(p, y)}{V} \quad (10)$$

where α_{geom} represents the geometric AoA. For the inner wing α_{geom} is assumed zero, and for each FFWT it is modeled using Eq. (1).

To account for 3D effects, the local lift-curve slope at each panel was interpolated from a set of lift distributions that were precalculated using lifting line theory. The purpose of this paper is not to accurately predict this lift distribution, as it is likely to vary as a function of not only velocity and root AoA but also the fold angle of each wingtip. Therefore, a set of lift distributions were calculated to best represent

LaTeX versus Microsoft Word

Microsoft Word

- Drag and Drop user interface
- **WYSIWYG** (What you see is what you get) interface
- Familiar to most people (hopefully!)
- Equations, figures and citations possible but get tricky on large documents
- Large documents misbehave...
- Expensive*

LaTeX

- A “programming language”
- Allows you to **focus on the content, not the formatting**
- you write your document semantically e.g.
 - this is a section
 - this is a figure
 - this is a citation
- and LaTeX takes care of how it looks ensuring
 - Consistency
 - Elegance
 - typographic quality.
- Robust handling of figures, equations and citations etc...
- You must ‘compile the code’ to view the output
- Free*

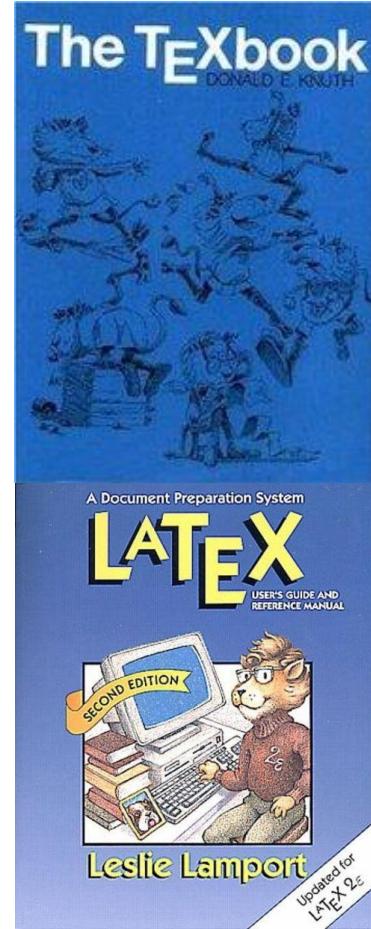
Latex versus Microsoft Word

	Word	LaTeX
Ease of Use	Easy	Steep learning curve
Mathematic Support	Okay -> Good	Excellent
Collaboration	Good	Excellent (with overleaf)
Figures + Numbering	Okay	Excellent
Version Control	Poor	Good
Bibliography	Okay	Excellent
Formatting Control	Okay	Good

LaTeX Basics

bristol.ac.uk

"tex", from the Greek word τέχνη (techne, meaning "art" or "craft").



LaTeX

- LaTeX is a markup language used to create scientific documents.
- Similar to writing and compiling a C program
 - A LaTeX document (“.tex”) is a text file with a series of commands
- As with other programming languages, there is a *reasonable* learning curve
- We will only scratch the surface today...
 - But there are lots of examples online
 - And LLMs are remarkably good at LaTeX

LaTeX (*.tex file)

```
\documentclass{article}
```

```
\usepackage{amsmath}  
\usepackage{graphicx}
```

```
\title{My First Document}  
\author{Fintan Healy}  
\date{October 2025}
```

```
\begin{document}
```

```
\maketitle
```

```
\section{Introduction}
```

An example document to show you the basics of creating documents in \LaTeX.

```
\subsection{Mathematics}
```

It is simple to create complex mathematical equations such as

```
\begin{equation}  
\label{eq:myEq}  
y = \int^b_a \frac{1}{2} \sin(x) * (1 + \tan(x^2)) dx  
\end{equation}
```

It is then easy to directly reference Eq.~\ref{eq:myEq} in the text.

```
\end{document}
```

LaTeX Anatomy

The type of document

Load some packages

Preamble

Start the main body of the document

Body

Custom functions

Equations

Cross reference

End main body

LaTeX (*.tex file)

```
\documentclass{article}
```

```
\usepackage{amsmath}  
\usepackage{graphicx}
```

```
\title{My First Document}  
\author{Fintan Healy}  
\date{October 2025}
```

```
\begin{document}
```

```
\maketitle
```

```
\section{Introduction}
```

An example document to show you the basics of creating documents in **\LaTeX**.

```
\subsection{Mathematics}
```

It is simple to create complex mathematical equations such as

```
\begin{equation}  
\label{eq:myEq}  
y = \int^b_a \frac{1}{2} \sin(x) * (1 + \tan(x^2)) dx  
\end{equation}
```

It is then easy to directly reference Eq.~\ref{eq:myEq} in the text.

```
\end{document}
```

LaTeX Anatomy

My First Document

Fintan Healy

October 2025

1 Introduction

An example document to show you the basics of creating documents in L^AT_EX.

1.1 Mathematics

It is simple to create complex mathematical equations such as

$$z = \int_a^b \frac{1}{2} \sin(x) * (1 + \tan(x^2)) dx \quad (1)$$

It is then easy to directly reference Eq. 1 in the text.

LaTeX (*.tex file)

```
\documentclass{article}
```

```
\usepackage{amsmath}  
\usepackage{graphicx}
```

```
\title{My First Document}  
\author{Fintan Healy}  
\date{October 2025}
```

```
\begin{document}
```

```
\maketitle
```

```
\section{Introduction}
```

An example document to show you the basics of creating documents in L^AT_EX.

```
\subsection{Mathematics}
```

It is simple to create complex mathematical equations such as

```
\begin{equation}  
 \label{eq:myEq}  
 y = \int_a^b \frac{1}{2} \sin(x) * (1 + \tan(x^2)) dx  
\end{equation}
```

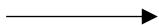
It is then easy to directly reference Eq.~\ref{eq:myEq} in the text.

```
\end{document}
```

LaTeX Anatomy

Latex

```
\documentclass{article}
```



- LATEX has different basic types of documents

- article (most common)
- letter
- report
- beamer (create slide decks)
- Custom classes for journals etc...

Latex

```
\usepackage{}
```



- To use various functionalities, we need to load packages (like in Python)
- Common packages are for loading images, entering mathematical formulae or citations

Latex

```
\begin{document}  
% ----- the body -----  
\end{document}
```



- Defines the start and end of the main body of the document.
- Generally, all text should be written inside these comments.

Development Environments

- You can install LaTeX locally and compile documents offline
 - I recommend “TeX Live” coupled with “VS Code” (+ LaTeX Workshop extension)
- The easiest IDE is Overleaf. It is an online platform where:
 - Developing a Latex document is easy
 - Sharing documents is easy
 - Collaboratively working on documents is easy
- The final style of a LaTeX document can be customised
- Overleaf provides templates that you can use to get started.
 - <https://www.overleaf.com/latex/templates>



Basic Syntax

Writing Text

- To write text:
 - separate words with spaces
 - separate sentences using periods
 - separate paragraphs by blank lines.
- When compiling, if multiple spaces are together, the compiler accepts these as just one continuous space.

LaTeX

We write a latex document as we would in any normal document . Of course the main difference is that the compiler takes care of the formatting for us.

Leave a blank line to start a new paragraph.

But sentences.

Even short.

will be wrapped automatically, with a blank line.

`\noindent` can be used to start a new paragraph without indentation.

We write a latex document as we would in any normal document . Of course the main difference is that the compiler takes care of the formatting for us.

Leave a blank line to start a new paragraph. But sentences. Even short. Will be wrapped automatically, without a blank line. `\noindent` can be used to start a new paragraph without indentation.

Text Formatting

- To **bold** and *italic* pieces of text we can

Use the “textbf” macro to `\textbf{bolden text}`, and the “textit” macro to `\textit{italicise text}`.

- The default font size in the article class is 10pt. The font size can also be changed from the default using:

`\huge{This text}` is huge. This is normal sized, and this is `\tiny{tiny}`.

Command	Example text	Font size
<code>\tiny</code>	Some text	5pt
<code>\scriptsize</code>	Some text	7pt
<code>\footnotesize</code>	Some text	8pt
<code>\small</code>	Some text	9pt
<code>\normalsize</code>	Some text	10pt
<code>\large</code>	Some text	12pt
<code>\Large</code>	Some text	14.4pt
<code>\LARGE</code>	Some text	17.28pt
<code>\huge</code>	Some text	20.74pt
<code>\Huge</code>	Some text	24.88pt

Sectioning and Structure

- Scientific reports are usually structured in a logical numbering fashion, where there are sections and sub-sections within those sections:
- LaTeX automatically handles numbering. You just need to use
 - `\section{}`
 - `\subsection{}`
 - `\subsubsection{}`

LaTeX

```
\section{Introduction}
An example section.

\subsection{Aims}
With not one.

\subsubsection{Objectives}
But two sub sections. Some with their own sub sections
\subsubsubsection{Objective 1}
This is the first objective.
\subsubsubsection{Objective 2}
This is the second objective.

\section{Conclusion}
In conclusion, we have presented a document with sections and
subsections.
```

1 Introduction

An example section.

1.1 Aims

With not one.

1.2 Objectives

But two sub sections. Some with their own sub sections

1.2.1 Objective 1

This is the first objective.

1.2.2 Objective 2

This is the second objective.

2 Conclusion

In conclusion, we have presented a document with sections and subsections.

Lists

- Numbered lists or Bullet points are simple in LaTeX
 - `\itemize` – bullet points
 - `\enumerate` – numbered lists
- Each element starts with the keyword `\item`
- You can nest lists
 - You can mix and match

LaTeX

```
\begin{itemize}
  \item First element.
  \item Second element.
  \begin{itemize}
    \item A sub element.
    \item Another sub element.
  \end{itemize}
  \item Third element.
\end{itemize}
```

- First element.
- Second element.
 - A sub element.
 - Another sub element.
- Third element.

LaTeX

```
\begin{enumerate}
  \item First element.
  \item Second element.
  \begin{itemize}
    \item A sub element.
    \item Another sub element.
  \end{itemize}
  \item Third element.
\end{enumerate}
```

1. First element.
2. Second element.
 - A sub element.
 - Another sub element.
3. Third element.

Tables

- To create a table, we use the **tabular** environment.
- The specification of the columns (i.e. justification) is then done using entries in a {} entry
 - **l**, **c**, or **r** for left, centre or right justification
 - **|** for vertical lines
- Use **&** to separate entries in a row
- Use **\\"** to create a new line
- Online tools such as tablesgenerator.com are great to help you quickly make tables

LaTeX

```
\begin{tabular}{|c|c c|}\hline Variable & Value & Units \\\hline $x$ & 5 & m \\\$y$ & 10 & s \\\hline\end{tabular}
```

Variable	Value	Units
x	5	m
y	10	s

Figures

- To load figures, we need to load the package **graphic**

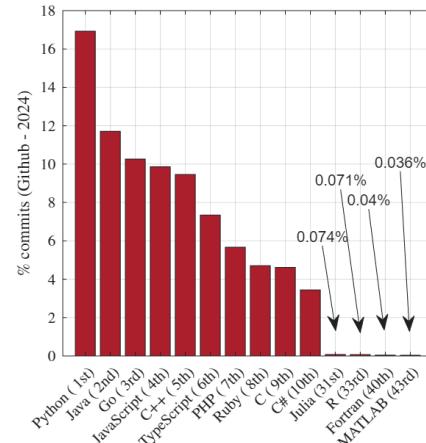
```
\usepackage{graphicx}
```

- To include a figure
 - The file must be in the same directory
 - Use the command
- where '*langChart.png*' is the filename
- The optional argument '**width=0.5\linewidth**' stretches the figure to half the total width of the page.
 - But if you've made your figures the right size, you don't need this!

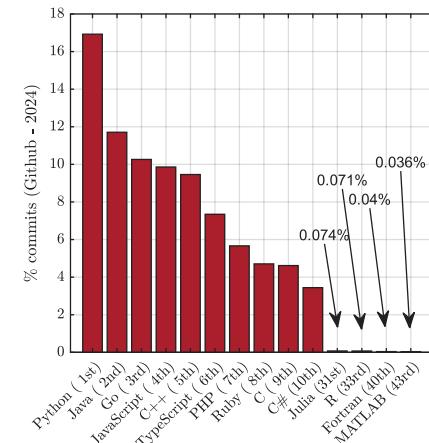
Raster Versus Vector Graphics

- When you save a figure, you can generally save it as either a raster or a vector graphic

Raster (*png*)



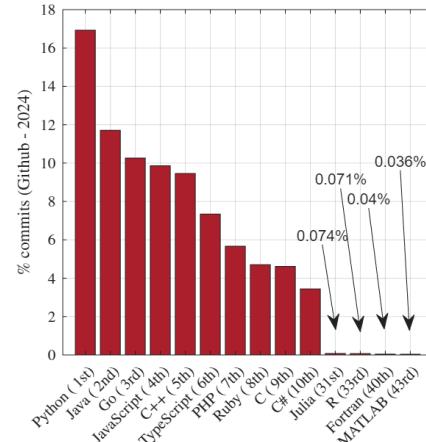
Vector (*pdf*)



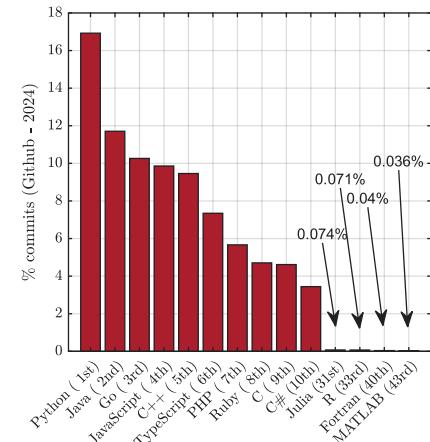
Raster Versus Vector Graphics

- When you save a figure, you can generally save it as either a raster or a vector graphic
 - **Raster graphics** are pixel-based images best suited for detailed visuals like photographs or complex surface plots
 - `.png` `.jpeg`
 - **Vector graphics** use mathematical paths to create scalable and resolution-independent illustrations
 - `.svg` `.pdf`
- If your report will be read digitally, vector graphics give it a more professional feel...
 - However, they don't always play nice...

Raster (`png`)

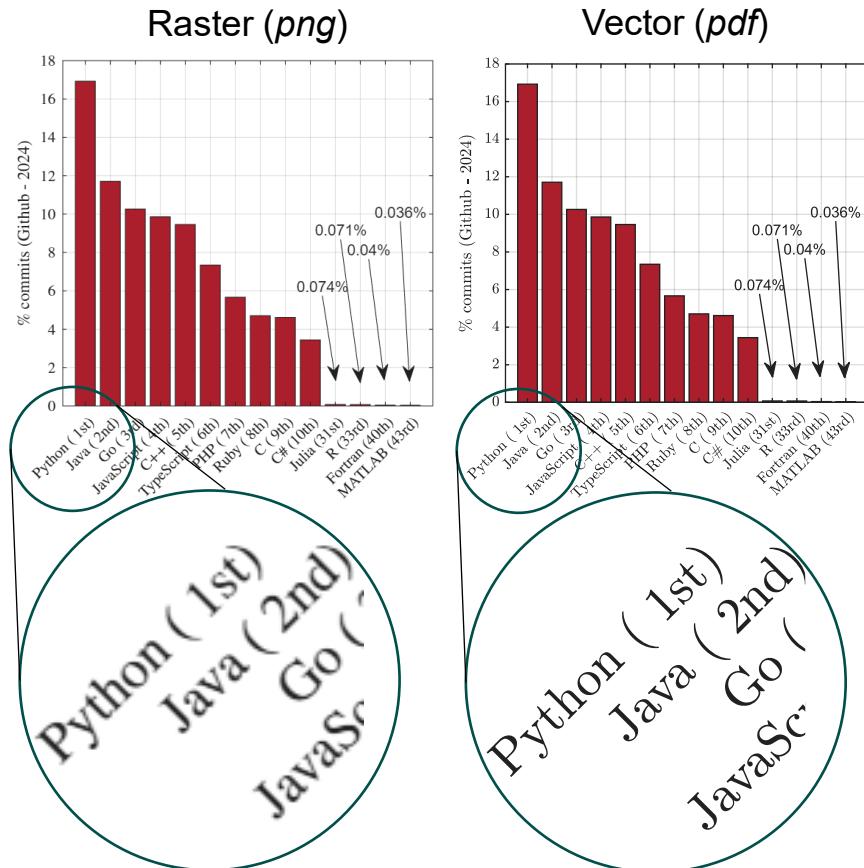


Vector (`pdf`)



Raster Versus Vector Graphics

- When you save a figure, you can generally save it as either a raster or a vector graphic
 - **Raster graphics** are pixel-based images best suited for detailed visuals like photographs or complex surface plots
 - `.png` `.jpeg`
 - **Vector graphics** use mathematical paths to create scalable and resolution-independent illustrations
 - `.svg` `.pdf`
- If your report will be read digitally, vector graphics give it a more professional feel...
 - However, they don't always play nice...



Floats, Captions and Placement

- We can encapsulate figures and tables within a floating environment
- This lets LaTeX decide where to place the element
 - You “***focus on the content, not the formatting***”...
- For figures, we can use a **figure** environment
 - Use **\caption** to add a ... caption
 - Use **\label** to be able to refer to the figure in the text
- The optional placement specifier (e.g., [h!]) suggests preferred float positioning
 - E.g., h means 'here' and ! means "try harder". There is also t – top, b – bottom, etc....
- You can constrain the movement of floats with the macro **\FloatBarrier**

```
\usepackage{placeins}  
\FloatBarrier
```

LaTeX

```
\begin{figure}  
  \centering  
  \includegraphics[width =0.5\linewidth]{LangChart.png}  
  \caption{A simple figure with a caption.}  
  \label{fig:lang}  
\end{figure}
```

A reference to Fig.~\ref{fig:lang} in the text.

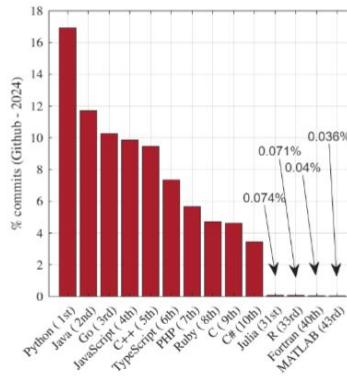


Figure 1: A simple figure with a caption.

A reference to Fig. 1 in the text.

Floats, Captions and Placement

- Tables can be encapsulated in a ‘tabular’ environment
- By convention, captions should be shown
 - Before tables
 - After figures
- **\centering** centre justifies content in the ‘environment’

LaTeX

```
\begin{table}
  \centering
  \caption{A simple table with three columns.}
  \label{tab:simple}
  \begin{tabular}{|c|c c|}
    \hline
    Variable & Value & Units \\
    \hline
    $x$ & 5 & m \\
    $y$ & 10 & s \\
    \hline
  \end{tabular}
\end{table}
```

Table 1: A simple table with three columns.

Variable	Value	Units
x	5	m
y	10	s

Sub-figures

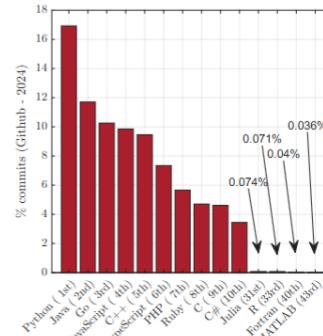
- In many circumstances, we want a figure to have multiple sub-figures.
- To create subfigures, each with its own caption, load the package

```
\usepackage{subcaption}
```

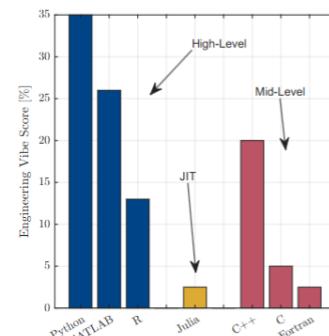
- Then have multiple subfigure environments in one figure environment
- The size of each figure drives the layout

LaTeX

```
\begin{figure}
  \centering
  \begin{subfigure}{0.45\linewidth}
    \includegraphics[width=\linewidth]{LangChart.pdf}
    \caption{A subfigure.}
    \label{fig:lang_sub}
  \end{subfigure}
  \hfill
  \begin{subfigure}{0.45\linewidth}
    \includegraphics[width=\linewidth]{LangVibe.pdf}
    \caption{Another subfigure.}
    \label{fig:lang_sub2}
  \end{subfigure}
  \caption{A figure with subfigures.}
  \label{fig:lang_main}
\end{figure}
```



(a) A subfigure.



(b) Another subfigure.

Maths

- One of the great strengths of LaTeX is equation writing
- We will only cover the basics here, but more details can be found in the references.
- It's best to load the package **amsmath**
- Equations are written in a maths environment, use:
 - $\$ \dots \$$ for inline mathematics
 - $\begin{equation} \dots \end{equation}$ for number equations
- You can also label and cross-reference equations

LaTeX

Simple equations in the text: $y = mx + c$ or in a separate line

```
\begin{equation}
L = \frac{1}{2} \rho v^2 S C_L
\label{eq:lift}
\end{equation}
```

which is known as the lift equation (Eq.~(\ref{eq:lift})).

Simple equations in the text: $y = mx + c$ or in a separate line

$$L = \frac{1}{2} \rho v^2 S C_L \quad (1)$$

which is known as the lift equation (Eq. (1)).

Maths

Feature	LaTeX	Output
Sub- or superscripts	<code>f(x)=a_{0}+ a_{1} x+a_{2} x^{2}+\dots</code>	$f(x) = a_0 + a_1 x + a_2 x^2 + \dots$
Greek Letters	<code>\alpha , \beta , \gamma , \Gamma , \pi , \Pi , \phi</code>	$\alpha, \beta, \gamma, \Gamma, \pi, \Pi, \phi$
Bold font (e.g for vectors)	<code>\mathbf{a} \cdot \mathbf{b} = \mathbf{a} \mathbf{b} \cos \theta</code>	$\mathbf{a} \cdot \mathbf{b} = \mathbf{a} \mathbf{b} \cos \theta$
Fractions	<code>R = \frac{\rho V}{L}</code>	$R = \frac{\rho V}{L}$
Large Brackets “ <code>\left(... \right)</code> ”	<code>\left(\frac{V_1}{V_2}\right)^\gamma</code>	$\left(\frac{V_1}{V_2}\right)^\gamma$
Summation and integration	<code>\int_a^b f(x) dx \approx \sum_{i=1}^N \Delta x f(x_i)</code>	$\int_a^b f(x) dx \approx \sum_{i=1}^N \Delta x f(x_i)$

More Documentation

- We have only scratched the surface of LaTeX
- For further details, please start with Overleaf's 30-minute guide
https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes
- And the overleaf documentation in general is excellent
- LaTeX also has an online wiki
<https://en.wikibooks.org/wiki/LaTeX>
- And LLMs are generally very good at LaTeX

Templates

- Don't forget to start with a template you like the look of
- These *control* the general styling of your document.

<https://www.overleaf.com/latex/templates>



Journal articles

Select from an array of journal templates and submit directly from Overleaf.



Bibliographies

Create bibliographies in LaTeX quickly and easily using packages like bibtex, natbib, and bibtex.



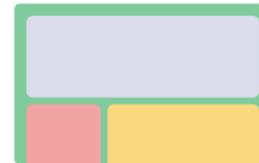
Books

Book templates to write your next best seller—whether you're writing a short story or a textbook.



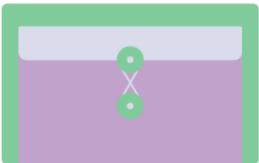
Calendars

Keep organized with this handy collection of LaTeX calendar templates.



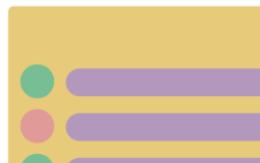
CVs and résumés

Make the right first impression with our popular range of CV templates.



Formal letters

Format formal letters fast with our range of LaTeX letter templates.



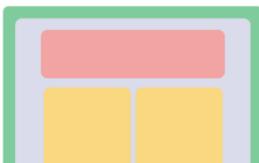
Assignments

LaTeX templates for homework assignments to use at school, college, and university.



Newsletters

Keep colleagues or contacts informed in style with one of our LaTeX newsletter templates.



Posters

Showcase your work with these eye-catching LaTeX poster templates.



Presentations

Communicate more effectively with our selection of engaging presentation templates.



Reports

Templates for producing project and lab reports, with guidelines to help you in the writing process.



Theses

Templates to help you tackle the most important piece of work you'll produce as a student.

SUMMARY

Summary

- Have introduced software for producing professional scientific reports (LaTeX)
- Looked at:
 - Basic formatting
 - Import figures and tables
 - Writing equations



<https://i2sc.fintanhealy.co.uk/>

Moving Forward

- **No more labs – but your journey is not over!**
 - Attempt the rest of the workbooks; they will save you time when it comes to your coursework!
- I will release solutions at the start of week 5
- **Don't neglect Python**
 - Being 'conversational' in multiple programming languages is a good skill for the 21st century
- Have a go creating a report with LaTeX