

# UM-SJTU Joint Institute Lab Course Guide

*For VC211 (Probably Not), VP141 and VP241. Detail specification may change subject to different instructors and Lab Report Rubrics.*

## 1 L<sup>A</sup>T<sub>E</sub>X Introduction

### 1.1 Overview

L<sup>A</sup>T<sub>E</sub>X is a free and open-source text processing program. It doesn't work like markdown or word which can preview outcome file in the real time. LaTeX has a rather complicated compiler system, which processes commands and compile your `.tex` source code file into other readable formats such as `.pdf`. So, why bother using such a complex approach to write one's lab report? It turns out that LaTeX has its own advantage over word or other softwares, especially in complicated projects.

L<sup>A</sup>T<sub>E</sub>X has the following advantages:

- Wide range of pre-defined styles and packages
- Better math mode, easier to type complex formula and equations
- Relatively stable format and outcome across devices and systems
- Freedom for organizing your contents in any way you want
- Macros and headers, reduce repeated work
- Focus on keyboard, complete your work without touching mouse

Since you can receive more detailed and systematic instruction on LaTeX workshops organized by SSTIA or JI-Tech, only some basic concepts and suggestions are provided here. Attending these workshops is recommended, or just exploring LaTeX like any other programming language would also be great.

### 1.2 Basic Concepts

For beginners, it is always good to introduce the methodology of LaTeX users, or the basic unit of LaTeX. There are 3 kind of basic units: plain text, comments and macros. Plain text is easy to understand, it's just text shown and not included as parameters of macros. For **comments**, you can make comments just like what you would do in MATLAB. Starting with a `%` sign and any text behind it in the same line will not be shown or processed by LaTeX. For example:

```
%Comments
%This is comment, won't be shown.
  This is plain text, will be shown.
```

For **macros**, you can call them like functions following the syntax below.

```
%Macros
\macroName[option]{parameter/argument}
%In beamer template (for slides), there is overlay (animation)
\macroName<overlay>[option]{parameter/argument}
```

Just like macros in C/C++, there also exist simple substitution macros used to represent some special characters or simply do not take any parameter (void macros).

```
%Substitution macro \{
%Indicates this bracket is plain text, should not be interpreted.
\LaTeX uses \{ and \} as brackets.
%Void macro \par, similar to Tab in word.
\par A new paragraph begins here.
```

You can **define your own macros** or **redefine existing commands** using the syntax below.

```
%New macro
\newcommand{macroName}[numberOfParameters][defaultParameters]{
  {
    Contents, use #n to refer to n-th parameter.
    First parameter: #1
  }
}
%Redefine macro
\renewcommand{macroName}[numberOfParameters][defaultParameters]{
  {
    Contents, use #n to refer to n-th parameter.
    First parameter: #1
  }
}
```

Then, we have a higher level of unit which is called **scope** or **environment**. The definition of scope here is very similar to the one in other programming language such as C/C++, but it doesn't start with a bracket. To declare an environment, you need to follow the syntax below.

```
%Enviroments and Scopes
\begin{environmentName}{parameters}[option]
  contents
\end{environmentName}
```

An exception of this syntax is **plain math environment**, where you may type some formula or equations. To declare such environment, follow the syntax below.

```
%Math environment
\par In-line math declares like this:  $E=mc^2$ ,
while dedicated line of math declare like this:

$$E=mc^2$$

```

Environments are widely used in LaTeX to indicate a pre-defined format or a local format of contents. For example, the very basic environment of LaTeX is `document` environment, out of which text won't be shown. It's similar to the `main` function of C/C++.

## 1.3 L<sup>A</sup>T<sub>E</sub>X Project Structure

Usually, several elements are needed to form a complete `.tex` file. In the beginning, we need to specify **document class** of our file. The following example shows how to declare document class and default settings for all lab reports.

```
\documentclass[12pt,a4paper]{article}
```

Specifying a document class actually sets some default format for the `document` environment we mentioned before. Another element is the **use of other pre-defined packages**. This is achieved by using `\usepackage` macro as following, which is similar to the `#include` instruction in C/C++.

```
\usepackage[option]{packageName}
```

The third thing is that we need a **document environment** as the "main function" of our LaTeX project, as what we mentioned before.

```
\begin{document}
    content
\end{document}
```

## 1.4 Useful Commands

Macros:

```
\title{Title} %Specify title, outside document scope
\author{Author} %Specify title, outside document scope
\date{Date} %Specify date, outside document scope
\section{sectionName} %Start a new section with section title
%Similar usage as \subsection, \subsubsection and \paragraph
```

Environments (graphicx and float package needed for figure):

```

%Include a table
\begin{table}[H]
  \centering %Alignment
  \begin{tabular}{cc}
    \hline %Horizontal line for table
    col11 & col21\\
    col12 & col22
  \end{tabular}
  \caption{A Table} %Title of the table
\end{table}

%Include a figure
\begin{figure}[H]
  \centering %Alignment
  %Usually set width of the figure to 0.6 to 0.8 text line width
  %This preserves the proportion of the figure
  \includegraphics[width=0.7\textwidth]{figure.png}
  \caption{A figure} %Title of the figure
\end{figure}

```

## 1.5 Suggestions

- Write your LaTeX project in VS Code
- Use `cSpell` extension to check spelling errors
- Use `LaTeX Workshop` extension would be enough
- Do not over sectioning your report
- Make sure your figures and tables are readable

# 2 Uncertainty and Statistics

Since detailed theories and explanation for most contents in this section is included in the PDF version of *Uncertainty Analysis Handbook*, this section only serves to better explain uncertainty propagation. The major problem is that some situations one would encounter in actual data processing of some labs cannot be explained using only the theories in the handbook, so we would like to provide extra information for you.

## 2.1 Introduction to Statistics and Linear Uncertainty Propagation

The concept of uncertainty is not well-defined without introducing some basics of statistics. Thus, we will first introduce some formal definitions of terms in statistics and then introduce their interpretations on data and analysis.

- ***Mathematical Expectation***

Mathematical expectation of a statistical variable is usually denoted by  $E(X)$  if  $X$  is the variable. You can interpret this as the **weighted mean of variable  $X$** . Let  $\Omega$  be the set where all possible value of  $X$  sits and  $P(x)$  be the probability of  $X = x$  in set  $\Omega$ , then the expectation is defined in the following way.

$$E(X) = \int_{\Omega} X dP$$

Since the integral sign is a linear operator, the expectation is also **linear**. This means that the following equation holds for any statistical variables.

$$E(a_1 X_1 + a_2 X_2) = a_1 E(X_1) + a_2 E(X_2)$$

- **Covariance**

Covariance describes **how 2 statistical variables correlates with each other statistically**. The covariance of 2 variables is usually denoted as  $\text{cov}(X, Y)$ , which is defined in the following way.

$$\text{cov}(X, Y) = E[(X - E(X))(Y - E(Y))] = E(XY) - E(X)E(Y)$$

If  $X$  and  $Y$  are statistically independent of each other, then the covariance is  $\text{cov}(X, Y) = 0$ .

- **Variance and Standard Deviation**

The standard deviation is often denoted as  $\sigma_X$  and variance is often denoted as  $\sigma_X^2 = \text{var}(X)$ . This immediately shows their relationship: standard deviation equals the square root of variance. The variance is a special case of covariance where  $X = Y$  in the formula. It describes **how discrete a variable distributes**, and it is defined in the following way.

$$\text{var}(X) = \text{cov}(X, X) = E[(X - E(X))^2] = E(X^2) - E(X)^2$$

We are interested at what is the **variance of a linear combination** of variables given the variance of these variables. Let  $X, Y$  be 2 statistical variables and linear combination  $f = aX + bY$ ,  $a, b = \text{const.}$  Let  $\sigma_X^2 = \text{var}(X)$  and  $\sigma_Y^2 = \text{var}(Y)$ ,  $\mu_X = E(X)$  and  $\mu_Y = E(Y)$ , then the variance of the linear combination  $f$  can be represented in the following way.

$$\begin{aligned} \sigma_f^2 &= E[(aX + bY) - E(aX + bY)]^2 = E[(aX + bY) - (a\mu_X + b\mu_Y)]^2 \\ &= E[a(X - \mu_X) + b(Y - \mu_Y)]^2 \\ &= E[a^2(X - \mu_X)^2 + b^2(Y - \mu_Y)^2 + 2ab(X - \mu_X)(Y - \mu_Y)] \\ &= a^2 E[(X - \mu_X)^2] + b^2 E[(Y - \mu_Y)^2] + 2ab E[(X - \mu_X)(Y - \mu_Y)] \\ &= a^2 \sigma_X^2 + b^2 \sigma_Y^2 + 2ab \text{cov}(X, Y) \end{aligned}$$

If variable  $X$  and  $Y$  are **independent** of each other, then  $\text{cov}(X, Y) = 0$  and the formula simplifies to the following form.

$$\text{var}(aX + bY) = \sigma_f^2 = a^2 \sigma_X^2 + b^2 \sigma_Y^2$$

Moreover, **variance cannot be changed by adding a constant to the variable**. Here, we can prove it using definition and linearity of expectation. Let  $a = \text{const.}$  and  $X$  be the variables.

$$\text{var}(X + a) = E[(X + a - E(X + a))^2] = E[(X + a - E(X) - E(a))^2]$$

Since  $a = \text{const.}$ , so  $E(a) = a$ . Thus, we have the following equality.

$$\text{var}(X + a) = E[(X - E(X))^2] = \text{var}(X)$$

- **Normal Distribution**

Normal distribution is a type of probability distribution for most natural variables and uncertainties. A distribution is characterized by **density function**  $f(x)$  and **cumulated density function (CDF)**  $F(x)$ . Let  $\text{Pr}(\cdot)$  be the operator denoting the probability under certain condition, then by definition, we have the following relationships.

$$f(x) = \frac{d}{dx}\text{Pr}(X \leq x), F(x) = \text{Pr}(X \leq x)$$

For normal distribution specifically, the density function and cumulated density function is of the following form.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}, F(x) = \frac{1}{2}(1 + \text{erf}(\frac{x-\mu}{\sqrt{2}\sigma}))$$

Here,  $\mu = E(x)$ ,  $\sigma^2 = \text{var}(x)$ , and  $\text{erf}(\cdot)$  refers to the Gaussian error function, which is an monotonic odd function. You can find more information for this function and its inverse in MATLAB where they are pre-defined.

- **Uncertainty and Confidence Interval under Normal Distribution**

Speaking of uncertainties, we are actually referring to a concept called **Confidence Interval Half Width**, or **CI Half Width**. For a symmetric distribution such as normal distribution, the confidence interval  $I[\xi]$  of certain confidence level  $\xi$  width CI Half Width of  $u$  is defined in the following way.

$$I[\xi] = [\mu - u, \mu + u], \text{ s.t. } \text{Pr}(\mu - u \leq X \leq \mu + u) = \xi$$

Here,  $\mu = E(X)$ . In engineering, uncertainty usually refers to the CI Half Width with confidence level  $\xi = 95\%$ . If the variable  $X$  satisfies a normal distribution, then the CI Half Width of  $95\%$  is approximately  $2\sigma$ .

$$\text{Pr}(\mu - u \leq X \leq \mu + u) = \xi \Rightarrow F(\mu + u) - F(\mu - u) = \xi$$

$$\text{erf}(\frac{u}{\sqrt{2}\sigma}) = \xi \Rightarrow u = \sqrt{2}\text{erf}^{-1}(\xi)\sigma \approx 2\sigma, \xi = 95\%$$

And from the equation above we have also proved that the **uncertainty  $u$  is a linear term of  $\sigma$**  since for any confidence level,  $u$  always equals to a constant times  $\sigma$ . **This indicates that**

**uncertainty  $u$  inherits square-linearity of  $\sigma$ .** Let  $u[\cdot] = t\sigma$  denote the uncertainty of a variable, and let  $a, b = \text{const.}$ ,  $X, Y$  be variables, then we have the following equation.

$$u[aX + bY]^2 = a^2u[X]^2 + b^2u[Y]^2 + 2abt^2\text{cov}(X, Y)$$

In simpler cases where  $X, Y$  are independent, we have  $\text{cov}(X, Y) = 0$  and the formula like the one on the handbook appears.

$$u[aX + bY] = \sqrt{a^2u[X]^2 + b^2u[Y]^2}$$

This is the formula for **Uncertainty Propagation on Linear Combinations of Independent Variables**.

- **Advanced: Student's t-Distribution\***

The probability density function of t-distribution is characterized by a concept called **degree of freedom (DoF) of sample variance**. By definition, the variance of a set of discretely measured data involves  $n$  quantities, the measured values, and thus have an DoF of  $n$ . However, to evaluate variance, the expectation or mean value of this set of data must be a known constant calculated through these  $n$  quantities. This adds a restriction to these the quantities and thus the DoF becomes  $n - 1$ . The t-distribution's density function describes the distribution of the **error of mean value estimation of expectation**, or  $T = \frac{\bar{X} - \mu}{S_{\bar{X}}}$ , where  $\mu = E(X)$  and  $S_{\bar{X}} = S_X / \sqrt{n}$  is the **estimated mean value standard deviation**.

$$f_{\nu}(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

Based on the definition of  $T$  and properties of standard deviation,  $\sigma_T^2 = \sigma_X^2 / S_{\bar{X}}^2$ . Thus, the uncertainty, or CI Half Width of  $X$  (a linear term of standard deviation) follows  $u[X] = S_{\bar{X}}u[T]$ . Let  $\xi$  be the confidence level, the CI Half Width  $u[T] = t_{\xi}$  is found in the following way.

$$\Pr(-t_{\xi} \leq T \leq t_{\xi}) = \xi \Rightarrow \int_{-t_{\xi}}^{t_{\xi}} f_{\nu}(t)dt = \xi$$

$$\int_{-t_{\xi}}^{t_{\xi}} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}} dt = \frac{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}{\Gamma(\frac{\nu+1}{2})} \xi$$

This equation does not have an explicit analytical solution for  $t_{\xi}$ , so a feasible way is to solve it numerically. The uncertainty of certain confidence level  $\xi$  of  $X$  is  $u[X] = t_{\xi}S_{\bar{X}}$ .

## 2.2 Propagation on Non-Linear Functions

It is hard to formulate the propagation rule on non-linear relationships in the way above, so a simple approach is just applying linear approximation on these functions and evaluate them in the way above. Thus, there are some restrictions in this case.

- Relative uncertainty of each variable is small enough ( $u_r < 0.1$ )
- Variables are independent of each other ( $\frac{\partial x_i}{\partial x_j} = 0, i \neq j$ )
- Target function  $f(..., x_i, ...)$  has a well-defined gradient at the input of evaluation

Here,  $x_i$  refers to the  $i$ -th independent variable of the target function  $f$  with uncertainty  $u[x_i]$  and reference value  $x_{i_0}$ . Then, using the knowledge learned in calculus course, the linear approximation of non-linear multi-variable functions is

$$f(x_1, \dots, x_N) = f(x_{1_0}, \dots, x_{N_0}) + \nabla f \cdot (\Delta x_1, \dots, \Delta x_N)$$

Based on the properties of standard deviation and variance, we can deduce that adding or subtracting a constant does not affect uncertainty. Thus,  $u[\Delta x_i] = u[x_i]$ . Thus, the linear approximation of the function is a linear combination with following terms.

$$y_0 = f(x_{1_0}, \dots, x_{N_0}) = \text{const}, y_i = \frac{\partial f}{\partial x_i} \Delta x_i, 1 \leq i \leq N$$

$$u[y_0] = 0, u[y_i] = \frac{\partial f}{\partial x_i} u[x_i], f \approx \sum_{i=0}^N y_i$$

Then, we directly apply the formula we deduced for linear combinations, we obtain the **General Formula for Uncertainty Propagation on Independent Variables**.

$$u[f] = \sqrt{\sum_{i=1}^N \left( \frac{\partial f}{\partial x_i} u[x_i] \right)^2}$$

## 2.3 Data Normalization

In many cases, we would like data to get rid of units so that we can put into some naturally pre-defined functions like  $\ln(x)$  and  $\sin(x)$  or compare with them for some relationship. The most common way of normalizing a data set is by dividing each data point by the maximum value of the data set. So the normalized data  $y_i$  in terms of original data  $x_i$  and maximum  $x_m$  is  $y_i = x_i/x_m$ . Then, what's the uncertainty of the normalized data? From my observation on some friends around me, most of them would answer that just applying the linear approximation method on each term. Unfortunately, it is not all correct. It is correct for terms where  $i \neq m$ , but for  $i = m$ , the story is different. Notice that there are 3 restrictions in the last section for the formula and one of them says: *Variables are independent of each other* ( $\frac{\partial x_i}{\partial x_j} = 0, i \neq j$ ). For  $i = m$ ,  $y_m = x_m/x_m = 1 = \text{const}$ . and the formula doesn't works since  $x_m$  and  $x_m$  are dependent. The case is much simpler that  $y_m = 1$  is a constant and it does not have uncertainty. Thus, remember that ***if you normalize the data in the way above, the uncertainty for the term with value 1 should always be 0.***

## 2.4 Singular Input



Singular inputs are cases where input data and non-linear function do not satisfy the restrictions in the section above. It is hard to cope with such inputs statistically because this requires advanced knowledge about continuous distributions such as Student's t-distribution. An approach to solve this problem is just simply avoiding it if not necessary, which include the following ways.

- Avoid singular data when doing experiment
- Discard directly when processing if not necessary

However, these singular data are at critical cases and it is necessary to account them sometimes. Thus, a systematic approach is needed for dealing with singular inputs. **Please keep in mind that it is quite complicated and it is better if you can locate the cause of singularity and separate it for analysis.** To simplify the problem, we will be discussing singularity for single variable functions. For most multi-variable functions, you may rewrite it into the form  $f(\vec{x}) = h(x_1, \dots, g(x_i), \dots, x_j, \dots, x_N)$ ,  $j \neq i$  and  $g(x_i)$  is the cause of singularity and  $h$  isn't. In this case, you can use linear approximation method to estimate  $h$  and after completing analysis of  $g(x_i)$ . Moreover, we will assume the variables satisfy normal distributions in terms of composite uncertainties (including type-A and type-B).

If a variable  $x$  satisfies normal distribution, then what can we say about the uncertainty of  $g(x)$  generally? The CDF of  $g(x)$  is of the following form.

$$F_g(y) = \Pr(g(X) \leq y) = \Pr(\{X | g(X) \leq y\})$$

This means that the CDF of  $g(x)$  can be represented using CDF of standard normal distribution. To better explain this idea, the following shows an example you may encounter in *Lab 4: Polarization of Light*. In one of the plot, we need to evaluate  $\sqrt{I/I_0}$  and one of the data point sits at  $I = 0$ ,  $I_0 = 0.5230$  with uncertainty  $u[I] = u[I_0] = 0.0001$ . Then, what's the uncertainty of  $\sqrt{I/I_0}$ ? It is obvious that  $I = 0$  is a singular point for function  $f(x) = \sqrt{x}$ , and moreover, since  $I$  refers to light intensity, it can never be negative. So the uncertainty here is actually a imaginary CI half width, and we need to make some changes to the density function of normal distribution. First, we calculate standard deviation  $\sigma$ .

$$\sigma \approx \frac{1}{2}u[I] = 0.0005$$

The expectation of  $I$  is given, which is 0. Thus, the new density function can be seen as doubling the part greater than 0 and eliminate the negative part.

$$f_{\text{new}}(x) = 2f(x) = \frac{2}{\sigma\sqrt{2\pi}}e^{-\frac{x^2}{2\sigma^2}}, x \geq 0, \sigma = 0.0005$$

The corresponding CDF of this distribution can be expressed in terms of standard normal distribution CDF.

$$F_{\text{new}}(x) = 2[F(x) - F(0)] = \text{erf}\left(\frac{x}{\sqrt{2}\sigma}\right), x \geq 0, \sigma = 0.0005$$

The CDF for  $\sqrt{x}$  can be simplified since  $\sqrt{x}$  is monotonically increasing in the domain and thus an inverse function exists.

$$F_T(t) = \Pr(\sqrt{x} \leq t) = \Pr(x \leq t^2) = F_{\text{new}}(t^2) = \text{erf}\left(\frac{t^2}{\sqrt{2}\sigma}\right)$$

Since it is not a symmetric standard normal distribution, to get uncertainty, we solve the equation  $F_T(t) = 0.95$ . Thus, the solution can be written as following.

$$u[\sqrt{I}] = t = \sqrt{\sqrt{2}\sigma \text{erf}^{-1}(0.95)} \approx 0.03$$

Then, we apply the general formula for function  $h(\sqrt{I}, I_0) = \sqrt{I}/\sqrt{I_0}$ , which is  $h(x, y) = xy^{\frac{1}{2}}$ .

$$u[\sqrt{I/I_0}] = \sqrt{\left(\frac{\partial h}{\partial x}u[\sqrt{I}]\right)^2 + \left(\frac{\partial h}{\partial y}u[I_0]\right)^2} \approx 0.04$$

## 3 Data Processing in MATLAB

### 3.1 Matrix Operations

Data to be processed in MATLAB usually exists in the form of an array or a matrix, where one can manipulate them as a whole. They can go through the same operations, functions for each element in the collection. For arithmetic operators, add a `.` in front of them indicates that they act element-wise. For most mathematical functions, they are also element-wise. For functions that act on vectors and matrix particularly, they tend to treat it as a whole and may not act element-wise.

### 3.2 Uncertainty Evaluation

For non-constant type-B uncertainties, function handles are quite helpful in this scenario. For example, you may meet some cases where the labeled uncertainty on the multimeter is  $\pm(0.6\%V + 0.001)$ . This means that the uncertainty depends on the measured value. When you have multiple array of voltage data, then how to generate corresponding uncertainties in forms of array? The following MATLAB code gives an elegant answer.

```
%V represents the voltage data array
uncertaintyFunction=@(V)(0.006.*V+0.001);
uncertaintyOfV=uncertaintyFunction(V);
```

Moreover, function handles, unlike functions that need a dedicated file, can be stored in the workspace file as a variable, you can take care of the rest any time later. Of course, for linear uncertainties, you can also use matrix to represent such relationship, but it would be redundant. Using function handles to process uncertainties is efficient and convenient.

For multiple measurements and composite uncertainties (including type-A uncertainties), some functions such as `std` and `mean` are helpful. `std` function returns the ***sample standard deviation of an array*** while `mean` returns the ***mean of an array***.

In terms of MATLAB code, the propagation formula can be deduced from general formula for uncertainty propagation on independent variables using **symbolic math toolbox**. A sample code is shown below.

```
%Define Variables
%syms x y z ...;
%Construct Expression using Variables
%expression=...;
%rawData is the initial data array
%rawUncertainties is the initial uncertainty array
uExpression=norm.gradient(expression)'.*rawUncertainties);
%... refers to symbolic variables
%uncertainty=double(subs(uExpression,{...},num2cell(rawData))));
```