

**AN AI-BASED FRAMEWORK AND DATA-DRIVEN METHODOLOGY  
FOR POST-PCR HIGH-RESOLUTION MELTING ANALYSIS.**

A MAJOR PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF COMPUTER  
APPLICATIONS, BHARATHIAR UNIVERSITY IN PARTIAL FULFILMENT OF THE  
REQUIREMENTS FOR THE AWARD OF THE DEGREE OF,

**MASTER OF SCIENCE IN DATA ANALYTICS**

Submitted By,

**RAJAGOPAL S**  
**(REG.NO: \*\*\*\*\*)**

Under the Guidance of,

**Prof. Dr. V. BHUVANESWARI, M.C.A., M.Phil., Ph.D.,**  
Professor, Department of Computer Applications.



**DEPARTMENT OF COMPUTER APPLICATIONS  
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING  
BHARATHIAR UNIVERSITY  
COIMBATORE-641046  
TAMIL NADU  
MAY – 2023**

Duplicate Copy

**CERTIFICATE**

# CERTIFICATE

This is to certify that the project titled “**AN AI-BASED FRAMEWORK AND DATA-DRIVEN METHODOLOGY FOR POST-PCR HIGH-RESOLUTION MELTING ANALYSIS.**” submitted to Bharathiar University in partial fulfilment of the requirement for the award of the degree of the **Master of Science in Data Analytics** is a record of the original work done by **RAJAGOPAL S** under my supervision and guidance and this project work has not formed the basis for the award of any Degree/Diploma/Associate ship/Fellowship or similar title to any candidate of any University.

Place: Coimbatore

Date:

Project Guide

Head of the Department

Submitted for the University Viva-voice Examination held on \_\_\_\_\_

Internal Examiner

External Examiner



# Microbiological Laboratory Research and Services India Private Limited

(An ISO 13485:2016 Certified Company)

0422-2425312/8098701010

E-mail: [sales@microserv.in](mailto:sales@microserv.in)

Webpage: [www.microserv.in](http://www.microserv.in)

2<sup>nd</sup> May 2023,  
Coimbatore

## To Whom it may concern

This is to certify that **Mr. Rajagopal S** has been our internship participant from 23<sup>rd</sup> January 2023 to 2<sup>nd</sup> May 2023. During this period, he has worked on developing "**An AI-based framework and data-driven methodology for post-PCR High-Resolution Melting Analysis**". This project aimed to develop software capable of interpreting molecular assays for diagnostic purposes, representing the first approach of its kind. Due to the interdisciplinary nature of the project, Mr. Rajagopal. S as a Data science specialist had to collaborate with clinicians and molecular biologists from different fields.

As required by the project, Mr. Rajagopal S also demonstrated a strong interest in learning rtPCR and related analysis protocols, which provided the foundation for the development of the software for automated interpretation of molecular assays.

In summary, we would like to thank Mr. Rajagopal. S for his contributions during his internship at Microbiological Laboratory Research and Services India Private Limited. We wish him the best for his future endeavours.

Regards, Dr. Rohit Radhakrishnan  
Ph.D. Director (Research and operations)  
Research and Operations



Website

Factory @ No. 2, Kings Colony, United Nagar, Veerakeralam Road, Vadavalli,  
Coimbatore - 641007



ISO 13485  
11-4 (Certification)

Duplicate Copy

**DECLARATION**

## DECLARATION

I hereby declare that this project work title “**AN AI-BASED FRAMEWORK AND DATA-DRIVEN METHODOLOGY FOR POST-PCR HIGH-RESOLUTION MELTING ANALYSIS**” submitted to Department of Computer Applications, Bharathiar University is a record of original work done by **RAJAGOPAL S** under the supervision and guidance of **Dr. V. BHUVANESHWARI., MCA., M.Phil., Ph.D.,** Professor Department of Computer Applications, Bharathiar University and that this project work has not formed the basis for the award of any Degree/ Diploma/ Associateship/ Fellowship or similar title to any candidate of any University.

Place: Coimbatore

Signature of the candidate

Date:

COUNTERSIGNED BY

Duplicate Copy

## **ACKNOWLEDGEMENT**

## ACKNOWLEDGEMENT

Union is Strength. It gives me a great pleasure to acknowledgement with gratitude to personalities, without their help the completion of this project work would not been possible.

I express my respectful thanks to **Prof. Dr. T. DEVI., M.C.A., M.Phil., Ph.D., (UK)**, Professor and Head, Department of Computer Applications, Bharathiar University, Coimbatore, for permitting me to carry out my project work.

I really deem it a special privilege to convey my prodigious and everlasting thanks to my guide **Prof. Dr. V. BHUVANESWARI, M.C.A., M.Phil., Ph.D.**, Professor, Department of Computer Applications, Bharathiar University, for her valuable guidance and suggestions to this project work.

I extended my sincere thanks to **Dr. ROHIT RADHAKRISHNAN, Ph.D.**, Director (Research and Operations), Microbiological Laboratory Research and Services (I) PVT LTD, Coimbatore, for providing opportunity to work on their R&D project.

Finally, I express my thanks to my dear parents and my dear friends for their support and encouragement for the successful completion of this project. I am highly obliged to those who have helped me directly and indirectly in making this project a successful one.



Duplicate Copy

**ABSTRACT**

## ABSTRACT

During the onset of the COVID-19 epidemic, there was a significant investment globally by clinical laboratories and hospitals in improving their molecular-based diagnostics infrastructure. In the current post-epidemic scenario, the upgraded infrastructure is underused as there is a significant decrease in COVID-19 testing. This is primarily due to the unavailability of commercial molecular assays to diagnose different infectious and non-infectious diseases. Another gap in implementing the molecular assays in mainstream diagnosis is the unavailability of automated analysis and database management of real-time PCR results for both probe-based and **High-Resolution Melt Analysis (HRMA)** based assay. **Microbiological Laboratory, Coimbatore** has patented an HRMA-based identification of molecular targets which is critical for the treatment of fatal diseases such as septicemia. The unavailability of customized predictive analysis and reporting for this HRM increases the dependency on error-prone manual interpretation of such complex data.

This thesis is a foundation for developing a first-of-its-kind framework for automated analysis of HRM data which can be customized for different molecular targets. We have used advanced computational techniques like Machine Learning, Signal Processing and Deep Learning predictive analysis of HRM data of clinical samples tested. In this thesis, the team has designed and discussed the fundamental principle for processing, analysing and interpreting the HRM data for a representative set of molecular targets, which can aid the technicians and clinicians in reporting. We have also developed a database for structured storage and retrieval of HRM data which could help in linking this analysis software with the existing Laboratory Information Management Software used for reporting clinical re

# TABLE OF CONTENTS

CHAPTER NO	DESCRIPTION	PAGE NO
	ACKNOWLEDGEMENT	
	ABSTRACT	
	LIST OF FIGURES	
	LIST OF TABLES	
<b>1.</b>	<b>INTRODUCTION</b>	
	1.1 ORGANIZATION PROFILE	21
	1.2 PROBLEM DEFINITION	21
	1.3 BACKGROUND AND NEED	23
	1.4 PURPOSE OF THE STUDY	24
	1.5 DEFINITIONS	24
	1.6 ROLE OF DATA IN REAL-TIME PCR	34
	1.7 PREDICTIVE ANALYSIS IN DIAGNOSIS	35
	1.8 OVERALL RESEARCH AIM AND OBJECTIVES	37
<b>2.</b>	<b>EXISTING SYSTEM</b>	
	2.1 DATA ANALYSIS SOFTWARE	38
<b>3.</b>	<b>LITERATURE REVIEW</b>	50
<b>4.</b>	<b>PROPOSED METHODOLOGIES</b>	
	4.1 KEY ASPECTS	52
	4.2 PROPOSED METHODOLOGIES	53
<b>5.</b>	<b>APPROACH ON IMAGES OF DNA MELT SIGNALS</b>	
	5.1 IMAGE PROCESSING	54

	<b>RESULT AND DISCUSSION</b>	58
<b>6.</b>	<b>APPROACH ON COORDINATES OF RAW FLUORESCENCE SIGNAL</b>	
	RESULT AND DISCUSSION	73
	CONCLUSION	73
<b>7.</b>	<b>APPROACH ON COORDINATES OF DNA MELTING SIGNAL</b>	74
	7.1 MELT CONVERSION	76
	7.2 SPLINE AND SAVGOL FILTER	77
	7.3 BSPLINE	80
	7.4 BASELINE SUBTRACTION	81
	7.5 BACKGROUND CORRECTION	83
	7.6 SIGNAL PROCESSING ON DNA MELTING SIGNAL	89
	7.7 THRESHOLDING LOGIC	90
	RESULT AND DISCUSSION	90
	CONCLUSION	90
<b>8</b>	<b>COMBINATION OF APPROACH ON IMAGES AND THE COORDINATES OF DNA MELTING SIGNAL</b>	91
	8.1 CONVOLUTION NEURAL NETWORK	93
	8.2 GENERATING IMAGE DATASET	95
	8.3 MODEL ARCHITECTURE	99
	RESULT AND DISCUSSION	
<b>9</b>	<b>SYSTEM DESIGN AND DEVELOPMENTS</b>	100
	9.1 COMPONENTS	102
	9.2 EXTRACTOR	105
	9.3 PYHRM	110
	9.4 MELTCURVE INTERPRETER	

9.5 ER DIAGRAM 116

**10. TESTING AND RESULTS** 117

10.1 TEST DATA

CONCLUSION 120

REFERENCES 121

Duplicate Copy

## LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
1	Capabilities of Laboratory Information Management System	18
2	Amplification of DNA segments	25
3	Melting profile of a PCR product	26
4	Negative derivative plot of the Melting curve	28
5	PCR Amplification Curve	29
6	Theoretical plot of PCR, three phases	30
7	High-Resolution Melt Curve	31
8	Reference DNA melting signal – Meningitis Panel	32
9	A and B: Rotor-Gene Q-Rex Interface	39
10	Rotor-Gene Q Rex Interface	40
11	Data analysis performed in ScreenClust HRM Software	42
12	Interface of BIO-RAD CFX Manager	43
13	Melt Peak Spreadsheet – CFX Manager	44
14	Interface of melt curve graph in Bio Molecular System micPCR	45
15	Interface of Melt curve graph in ThermoFisher QuantStudio	46
16	Interface of Roche's LightCycler	47
17	uANALYZE Interface	48
18	uMELT Interface	49
19	DNA Melt data report generated by thermal cycler machine	54
20	Colour mask to track yellow colour in the input image	55
21	Melt signal images scanned and cropped from PCR reports	56
22	Image masking performed on melt signal images	56
23	Cropping the images to retain only melt signals	57
24	Raw fluorescence signal plotted using matplotlib	60
25	Properties of raw fluorescence signal	61
26	First 10 linear points of the fluorescence signal	63
27	Fitting a straight line along the fluorescence signal's linear	64
28	Fluorescence signals after performing normalization	65
29	Erroneous intersection spotted in the signals	66
30	Raw fluorescence signal – Temperature range (30°C to 90°C)	67
31	Normalized fluorescence signals	67
32	Extrapolating imaginary lines on both the ends	68
33	Mapping the take-off & touch-down points with Normalized	70
34	DNA melting signals with double peaks and its corresponding	72
35	Comparison for manually converted melting signal(A) to	75
36	Before (A) and After (B) applying smoothening filter.	78
37	Machine converted melting signal (A) and manually converted	79

38	Features can be extracted from a melt curve using signal	83
39	Detecting peaks in the DNA melting signal	84
40	Calculating the peak prominence of the DNA melting signal	85
41	Calculating the peak width of the DNA melting signal	86
42	Detecting all the features from a melting signal.	86
43	Negative(noise) signal with peaks detected	88
44	Measuring the prominences	89
45	Concept of Convolution Neural Network for classifying DNA	92
46	Training images of DNA melting signals of various class	93
47	Model architecture with layers	95
48	Model performance with Accuracy and Loss	96
49	Confusion matrix for the results of CNN model.	97
50	Classification of CNN model between genuine and non-genuine	98
51	AI-framework	101
52	User interface of Extractor	102
53	Framework of extractor	103
54	Type of data to extract	104
55	PyHRM installation	105
56	File stack of PyHRM	105
57	Input data format for PyHRM	107
58	Import PyHRM	107
59	Output of plot()	108
60	Features of HRM data using feature_detection()	108
61	Reports of features_detection	109
62	File stack of meltcurve interpreter	110
63	MCI Interface	111
64	MCI Home page	112
65	MCI file upload	112
66	MCI Melt curve visualisation	113
67	MCI amplification curvet visualisation	113
68	MCI feature detection panel	114
69	MCI Statistical measures	114
70	MCI Report Generation	115
71	MCI Final Report	115
72	ER diagram of MCI database component	116
73	Melt curve test data	117
74	Features of melt curve test data	117
75	Precision and recall for classification model	118
76	Confusion matrix	118
77	Accuracy and metrics	119

## LIST OF TABLES

FIGURE NO	TITLE	PAGE NO
1	DNA melting temperature standards for Meningitis Panel of the	31
2	Sample DNA melting temperature standards for sepsis panel.	33
3	Sample data – coordinates of raw fluorescence	60
4	Model performance metrics	97

Duplicate Copy



Duplicate Copy

## **INTRODUCTION**

# CHAPTER 1

## INTRODUCTION

Laboratory Information Management Systems (LIMS) transformed conventional laboratory operations into digitally-enabled infrastructure operations to attain high productivity and efficiency. LIMS aid a clinical laboratory to create an ecosystem for automating workflows, integrate instruments, manage samples, data management, real-time collaboration, perform data analytics, check quality control and patient reporting in a secured, user-friendly and polarized environment (Fig. 1). Thus, the software is crucial not only in a clinical lab but also in wide laboratories ranging from academic research, chemical labs, and manufacturing to agricultural testing, forensics, etc., [1, 2].

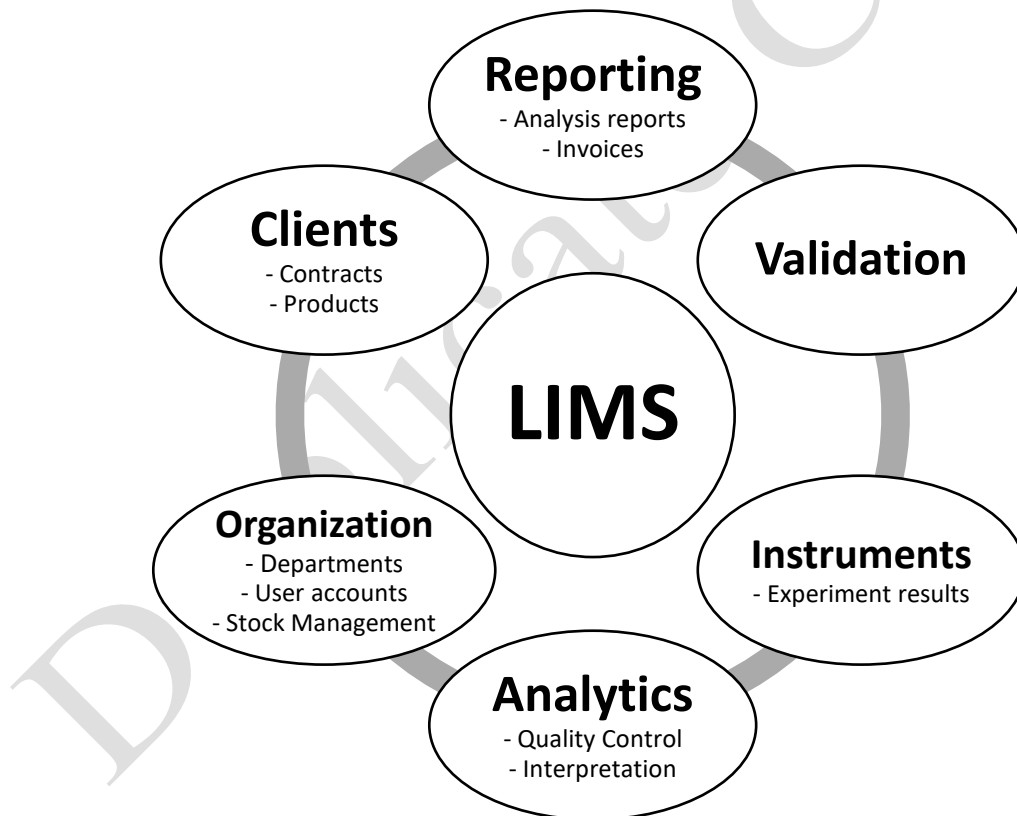


Figure 1: Capabilities of Laboratory Information Management System

Clinical laboratories are healthcare institutions that offer a variety of techniques to aid physicians with patient diagnostics, care, and management run by laboratory scientists [3]. With the rapid advancements in hardware and software technology, all the specialized clinical laboratories are modernized with state-of-the-art laboratory machines and instruments for

testing and diagnosing with high quality and accurate results, therefrom quick analysis and reports made in respective software.

Shirts et al. [4], stated the importance of analytics in clinical laboratories as ‘Clinical laboratory analytics is the systematic evaluation and communication of clinical laboratory testing data to improve healthcare operations and patient outcomes.’ [4, p. 9]. In looking at the system (Fig. 1), where the *analytics* part is the most demanding and includes complicated methods, such as analyzing and interpreting the test results and checking with quality control data to monitor the instrument's performance and accuracy. Along with these tasks, there are some limitations of data acquisition and integration phases in the analytics pipeline to integrate or acquire several other instruments with the respective data formats of their respective manufacturers' software/plugins in a general LIMS, taking into account large-sized labs. However, there are a few manufacturers/vendors that address this shortage by providing a solution for configuring instruments in the workflow [1, 2, 5], but this software might be expensive for mid to small-sized laboratories, or custom-built software can be a solution.

According to Baron [4], clinical laboratory analytics should focus on improving *decision support* (i.e., the use of tools or systems to provide clinicians with relevant information, recommendations, and guidelines at the point of care when ordering and interpreting laboratory tests) during test ordering and result interpretation. This approach requires developing a strong decision support infrastructure that embodies both rule-based and *machine learning*-based algorithms into the clinical workflow. In addition to that the importance of using “offline” clinical laboratory analytics to analyze and enhance test utilization (e.g., identifying variations in test ordering patterns between clinicians that cannot be explained by clinical factors) was also discussed [4, p. 11].

Most of the third-party vendors provide generic LIMS, in which the analytics pipeline involves transferring data from the LIMS to the different environments/platforms for analyzing the data, thus making a quite complex situation for the technicians and clinicians to perform analysis and interpretation in intermediate-level and national reference labs with faster deliverance and accurate results, where they own different kinds of instruments [3], [4, p. 11]. Besides, mid to large-sized intermediate-level labs have laborious and time-consuming processes for doing analytics without LIMS. In concern to the lab budget, the specific or generic LIMS developed by various PCR manufacturers or software vendors may charge additional fees for updates or require ongoing maintenance for continued support and updates.

Molecular diagnostics is a high complexity clinical laboratory [3], which amplifies the genetic level (DNA or RNA) of cells or pathogens to detect mutations, gene expression, or infectious agents at the molecular level using PCR [6, Sec. 1.1]. Quantitative polymerase chain reaction (i.e., real-time PCR or rtPCR) data analysis is a highly significant process that includes many primarily different techniques such as experiment setup, data processing, normalization, amplification analysis, efficiency and performance of PCR reaction, visualization of results [7, 8], this technique allows DNA amplification in real-time accumulation of fluorescence in reaction. High-Resolution Melt Analysis (HRMA) is an advanced technique of conventional Melt Curve Analysis (MCA) with a rtPCR instrument or its specialized instrument to identify the melting temperatures ( $T_m$ ) of DNA, though melt curve analysis also gives reliable results however, HRMA gives more accurate results compared to MCA [7]. These techniques are done by the various commercialized PCR manufacturers' data analysis software/plugins. Though stipulated statistical analyses and mathematical algorithms which can be done easily by clinicians with this software then, after the analyses part, information and insights gained finally lead to interpreting the results require certain expertise in the field.

Wrong interpretations and analysis of PCR test data and post-PCR data, (i.e., DNA melting curve, melt peaks, and HRMA curve) lead to serious consequences both for the patients affected and for the laboratory itself. However, the interpretation of PCR test result data is manually done by the clinicians/microbiologists with the domain knowledge (e.g., DNA melting, amplification of DNA, high-resolution melt analysis, characteristics of molecular pathogens, and thermodynamics of PCR) and other major parameters (such as., components added in the PCR compound, primers, and so on) by visual inspections and gain detailed insights from the analysis software. This visual interpretation highly requires intense laborious and time-consuming processes for novice clinicians/researchers and for complex case data, hence it impacts the right time for results to be delivered.

Several clinical laboratories own different PCR manufacturers' instruments and their respective software/plugins for their unique features and accurate results to obtain and interpret the data is challenging for mid to large-sized intermediate-level labs, which run numerous PCR experiments on a day-to-day basis. In addition to that, various democratized software/plugins/web-based applications are available, and each of them has its advantages and limitations for doing analytics after the experiment is done but, this might or might not give accurate results, according to different types of analysis and techniques used in the software.

As defined by Shirts *et al.*, this project aims to improve microbiologists'/clinicians' decision support and assist them by leveraging Artificial Intelligence and Machine Learning with the advent of HRMA data during the conventional analytical process to improve the performance of the interpretation of results. Additionally, to address the absence in the analytics phase by developing an automated application to aid laboratorians/clinicians.

## 1.1 ORGANIZATION PROFILE



**Microbiological  
Laboratory**



The **Microbiological Laboratory Research and Services India Private Limited (Microserv)** is an ISO 13485:2016 certified medical manufacturing facility and a research institute. Microserv develops and produces diagnostic kits such as ready-to-use microbiological culture media for clinical and industrial use, molecular assay reagents and MLRS-STaTAST. MLRS-STaTAST is a novel patented antibiotic sensitivity testing technology jointly developed with Anna University. Microserv also provides training in molecular diagnosis jointly with Bharathiar University.

The **Microbiological Laboratory, Coimbatore** is a leading NABL-accredited clinical laboratory in India. It is the first clinical laboratory which has several molecular assays under the NABL scope since 2007. Microbiological Laboratory has developed and patented HRMA based molecular assay for infectious diseases which is currently being used for patient diagnosis. Microbiological Laboratory operates over 50 branches throughout India that are connected to a central server system, enabling the consistent delivery of high-quality reports across all locations.

## 1.2 PROBLEM DEFINITION

The High-Resolution Melting Analysis (HRMA) involves monitoring the disassociation characteristics of double-stranded DNA during denaturation heating. Mutations and sequence variations in the DNA cause changes in the melting temperature and curve shape, allowing for sensitive and rapid detection of genetic variations without the need for expensive

probes or post-PCR processing. HRMA is aided by commercially available thermal-cycler (PCR) machines and respective analysis plugins/software for generating the melting curve graphs based on the raw fluorescence data.

### **Visual interpretation**

The interpretation of HRM data is crucial and it requires a clear understanding of the melting temperatures of every DNA target. HRMA software provide visualization (graphs) of melting signals against temperature, which comprises both perfect and imperfect (noisy) signals. The result interpretation usually involves visual observations and analysis by technicians. Experts who perform interpretation must focus on removing such noisy signals from their analysis by following some thresholding metrics. As the number of samples scales up, the interpretation also requires scaling, and doing it manually is challenging and time-consuming. Typically, experts would have much experience, and the perception they have in interpretation is huge and impeccable. In practice, experts alone cannot perform interpretation at all times, and several other beginners and junior technicians are also often required to perform interpretation, considering the productivity.

### **Versatile software**

Various PCR instrument manufacturers have their proprietary data analysis software and algorithms, which calculate and process the HRMA data with stipulated steps. Hence, different software can produce different melting curves for the same sample in HRMA analysis. This can occur due to differences in the algorithms used for data analysis and curve fitting, variations in the baseline correction and normalization methods, and other factors related to data processing and interpretation. To ensure accurate and reliable results, it is important to use standardized melting curve plotting protocol and software validated for the specific HRMA application.

### **Predictive analysis**

Most of the analysis software of HRMA uses the fluorescence response vs temperature data for the samples tested to plot the melting curve. The software uses statistical techniques and mathematical algorithms to analyze the raw fluorescence signals to melt signals. This software has various features such as identifying mutations of the pathogens, genotyping and detecting SNPs. Predictive analysis techniques are unavailable in this software for studying the unique features (e.g., Melting temperature ( $T_m$ ), melting peak height, curve shape, curve width,

inflection point, and area under the curve) of melt signals which can be used for creating digital signatures unique for each target.

### **1.3 BACKGROUND AND NEED**

The analysis, interpretation, and reporting of HRMA in the context of enhancing the decision support of the clinicians with the interoperable customized predictive analysis and reporting will decrease the dependency on laborious visual interpretation of such complex data. With the existing software available, such have their limitations of producing melt curve with the standard statistical and mathematical methods, and to the extent there are few commercial software which utilizes some of the machine learning algorithms such as principal component analysis and k-means for dimensionality reduction and clustering the data for other applications such as genotyping and mutation scanning. However, this software is outdated with support only for Windows 7 platform.

Currently, HRMA data analysis and interpretation require technical expertise, which can result in variability and errors in the results. An AI-based framework can standardize the interpretation and analysis of HRMA data, leading to more accurate and reliable results. This framework can provide automated data management, making the process more efficient and less time-consuming. With the integration of machine learning algorithms, the framework can learn from past data and adapt to new data sets, improving its accuracy and efficiency over time. The AI-based framework can also reduce human error and increase the speed of analysis, making it possible to analyze more clinical samples in a shorter time frame. The predictive analysis of HRMA data allows for rapid identification of the pathogen causing the disease.

The implementation of an AI-based framework for HRMA data management, interpretation, and reporting can also facilitate the sharing of data between laboratories and clinics, improving collaboration and accelerating the development of new diagnostic tools. The framework can also be used to track the evolution of genetic variations in pathogens, enabling early detection of emerging pathogens and their drug resistance patterns.

Overall, an AI-based framework for HRMA data management, interpretation, and reporting has the potential to revolutionize the clinical diagnosis of infectious diseases, making it more accurate, reliable, and efficient. It can provide a standardized approach to HRMA data analysis, allowing for better comparison of results across different laboratories, and can ultimately lead to the development of more effective diagnostic tools and treatment strategies.

## **1.4 PURPOSE OF THE STUDY**

The purpose of the study aims to develop and implement an AI framework to analyze and interpret High-Resolution DNA melt data with faster deliverance of accurate results and reports to aid clinicians/laboratorians in an intermediate-level laboratory.

Interpretation of HRMA data is handled by clinicians with keen visual observations and inspections with high domain expertise, which is a time-consuming and laborious process, and the lack of data acquisition and extraction pipeline makes a bit tangled situation at intermediate-level and national reference laboratories. These factors influence the speed and accuracy of deliverance and providing responsive reports to physicians and patients. Harnessing state-of-the-art AI and ML techniques and algorithms to analyze, interpret, and report without human intervention in a web-based platform.

To encounter the interpretation of HRMA data in concern with rtPCR data analysis techniques, the team explored the HRMA data of various pathogens by performing pre-processing, and feature engineering, with appropriate statistical analyses to gain insights. Implemented some of the existing methodologies beginning from the bare method of examining the images of DNA melt signal, to the approach of unravelling the co-ordinates of raw-fluorescence signal, DNA melt signal, and lastly the combination of images and co-ordinates of DNA melt signals. Upon the researched methodologies, ultimately the team devised a feasible and practical solution of developing a Python-based library with custom-trained Deep Learning models for the interpretation, and finally measured and validated the results with the expert clinicians. Along with these research methods, the team developed automated software for data extraction from the PCR data analysis plugin/software.

The goal of the study is to implement the web-based AI framework for analysis, interpretation, and reporting to assist technicians and clinicians. Another goal of the study is to develop automated software for data extraction.

## **1.5 DEFINITIONS**

This section provides a major list of concepts in detail such as PCR reaction, the processes behind the reaction, DNA melting behaviour, amplification analysis, and so on.



## 1.5.1 POLYMERASE CHAIN REACTION

The Polymerase Chain Reaction (PCR) is a molecular technique used for DNA quantification, biomarker identification, genotyping, and mutation detection. The technique is based on the amplification of a specific segment of DNA into several copies, using a DNA polymerase enzyme (Fig. 2). PCR involves using short synthetic DNA fragments called *primers* designed based on sequences specific to each target. The segment of the DNA complementing the sequences of the primer will be amplified for multiple PCR cycles until it reaches the limit of detection [9, 10].

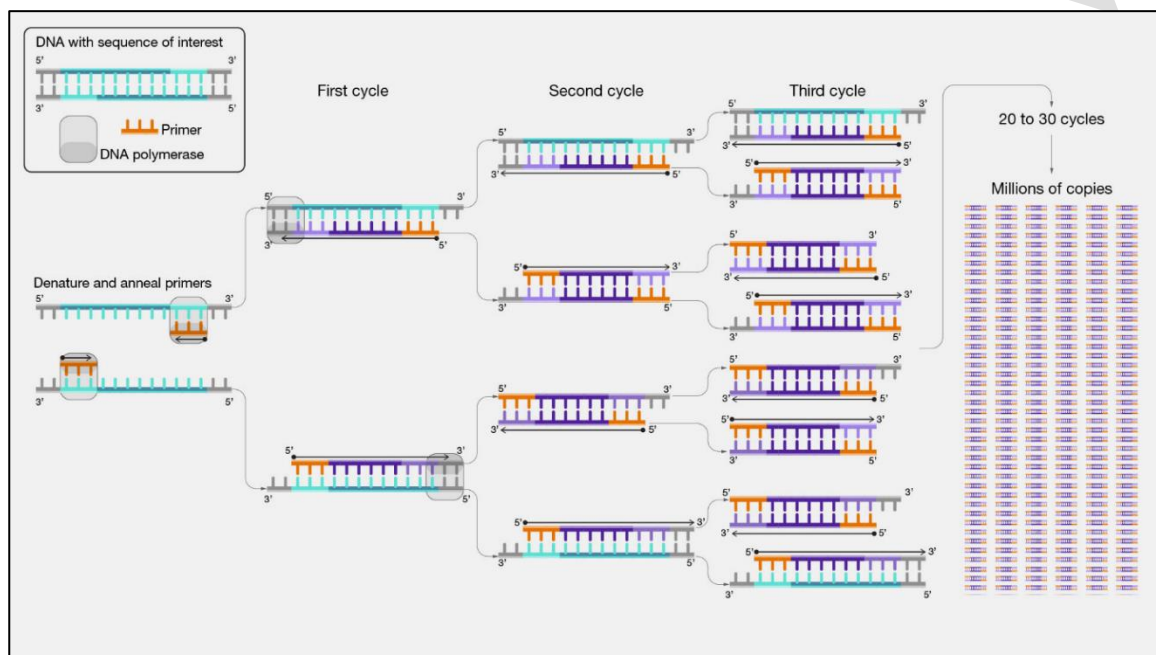


Figure 2: Amplification of DNA segments  
Source: National Human Genome Research Institute

Traditional PCR demand that the product be examined following the completion of the reaction, this procedure is frequently referred to as “*end point*” analysis [7]. Due to the improvements in hardware and software, real-time PCR (also known as quantitative PCR or qPCR or rtPCR) occurred as a new variation that constantly accumulates fluorescent signals from several polymerase reactions and permits to detect the DNA amplification at the right moment [11].

Real-time PCR uses commercially available fluorescence-detecting thermocyclers to amplify specific nucleic-acid sequences tagged with different types of fluorescent dyes (probes and SYBR™ green dye) and measure their concentration simultaneously. Target sequences are amplified and quantified simultaneously in the same PCR machine. Hence, the PCR amplification of the target sequence can be monitored in real-time thus eliminating

quantification steps such as agarose gel electrophoresis [11]. PCR was widely used during the recent COVID-19 outbreak to manage the epidemic across the world and remains the gold-standard method for COVID-19 diagnosis. The COVID-19 diagnosis is the latest application that has popularised this qPCR technique in recent times, and the application of PCR in the diagnosis of pathogens (targets) such as bacteria, viruses, fungi, and other non-culture biomarkers has been available for several years [12, 13].

### 1.5.2 DNA MELTING

The dissolution of the double-stranded DNA (dsDNA) helix into single coils is referred to as DNA melting. It can be accomplished by simply heating double-stranded DNA. The temperature at which the DNA strands dissociate into single coils depends on the number of hydrogen bonds holding the complementary strands. The most commonly used method to determine the melting temperature of a PCR product is to subject the product to a temperature gradient in the presence of intercalating dye. The intercalating dyes are chemicals that only emit light when bound to double-stranded DNA.

In a typical melting experiment, a PCR product is mixed with an intercalating dye, and fluorescence emitted by this mix is monitored as the sample is slowly heated (subjected to a temperature gradient). The outcome of the analysis is a curve displaying fluorescence changes emitted by the sample over the range of temperatures that the sample was subjected to, commonly referred to as a melting profile (Fig. 3).

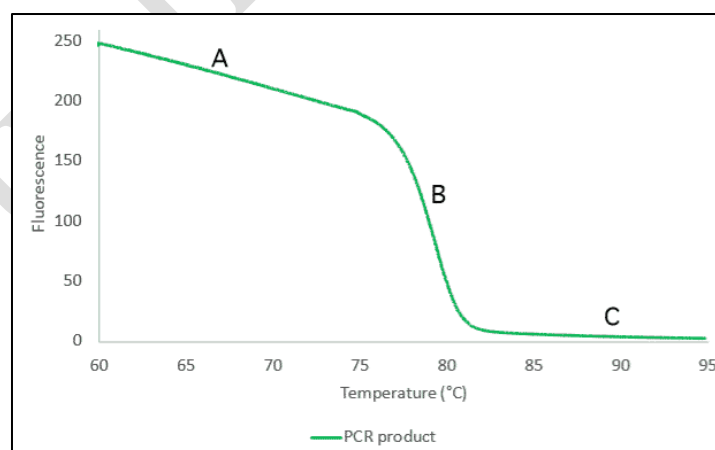


Figure 3: Melting profile of a PCR product

Source: MethylDetect

At the beginning of the experiment, the temperature is low and all PCR product in the sample is double-stranded. Thus, the fluorescence level is high in the sample (Fig. 3-A). Observing high levels of fluorescence, as the temperature increases up to the point, where all

hydrogen bonds within the PCR fragment are broken and the amount of double-stranded PCR product drastically decreases. Consequently, a sharp decrease in the detected fluorescence level (Fig. 3-B). At a high temperature, there is no double-stranded PCR product in the sample and the fluorescence levels are close to 0 (Fig. 3-C). The temperature at which the sharp drop in the fluorescence depends on the number of hydrogen bonds in the analyzed PCR product and hence is specific to the analyzed fragment.

Duplicate Copy

### 1.5.3 MELT CURVE ANALYSIS

The Melt Curve is derived from the raw fluorescence data, by getting the first negative derivative ( $-dF/dT$ ) of Fluorescence intensity and Temperature (fig. 4). In Melt curve analysis, the data comes as a result of HRM (in the case of specialized instrument used) being analyzed further, to determine the melting characteristics of several DNA in a more precise way. In this stage, the derivative of fluorescence intensity captured in real-time will be plotted against the temperature so, the temperature at which the dsDNA began to denature into ssDNA, the point at the melt peaks which resemble the melting temperature of the DNA. A threshold is manually set by the clinicians through keen visual inspections and observations with the help of commercial PCR manufacturers' analysis software.

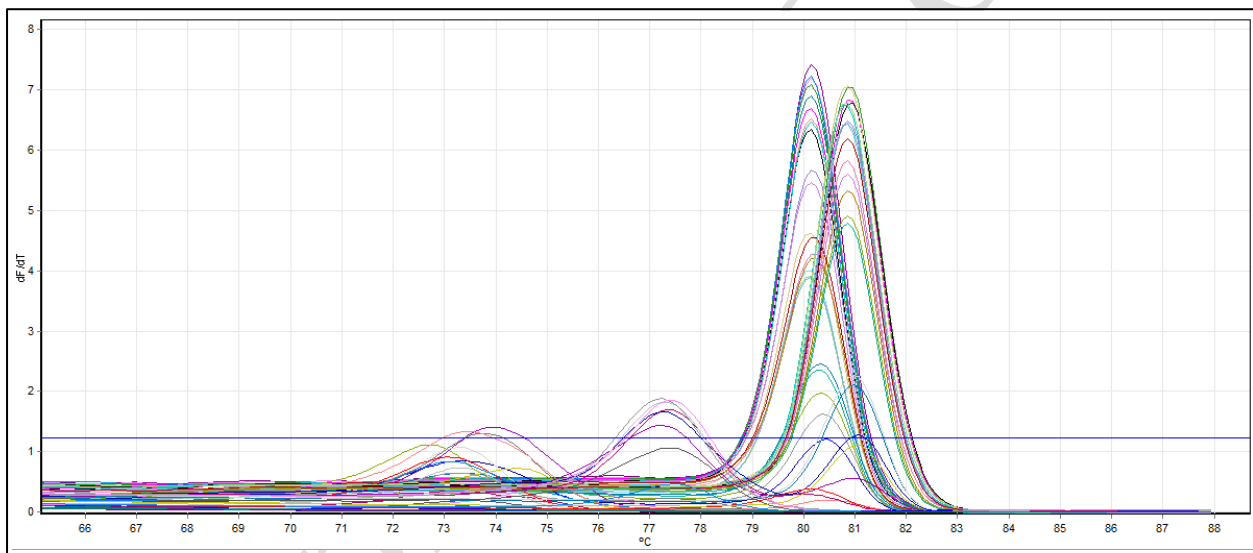


Figure 4: Negative derivative plot of the Melting curve

Source: QIAGEN's Q-Rex Software

In looking at Fig. 4, the characteristics of the Melt curves are observed and studied by concerning various features like:

- Peaks
- Shape of the curve
- Height
- Range
- Area under the curve

## 1.5.4 AMPLIFICATION CURVE ANALYSIS

The Amplification Curves are also known as ‘growth curves’ that display the graph of Cycle number vs Fluorescence (fig. 5), these data from real-time PCR are used to detect the presence of the PCR product (i.e., target DNA) and a threshold ( $C_t$ ) line is to be set in between the exponential and linear phase, to identify which PCR product is amplified earlier in the PCR reaction cycles [7], [20, p. 212].

The expression levels of genes can be measured by either absolute or relative quantification. In absolute quantification, a calibration curve is used to relate the PCR signal to the input copy number, while relative quantification measures the relative change in mRNA expression levels. The accuracy of an absolute real-time rtPCR assay depends on the identical amplification efficiencies of both the native target and the calibration curve in the RT reaction and kinetic PCR. Relative quantification is a simpler method compared to absolute quantification since it does not require a calibration curve. It involves comparing the expression levels of a target gene to a reference gene and is sufficient for most investigations into changes in gene expression. The units used for relative quantification are unimportant and can be compared across multiple real-time RT-PCR experiments [14, Sec. 3.2.6].

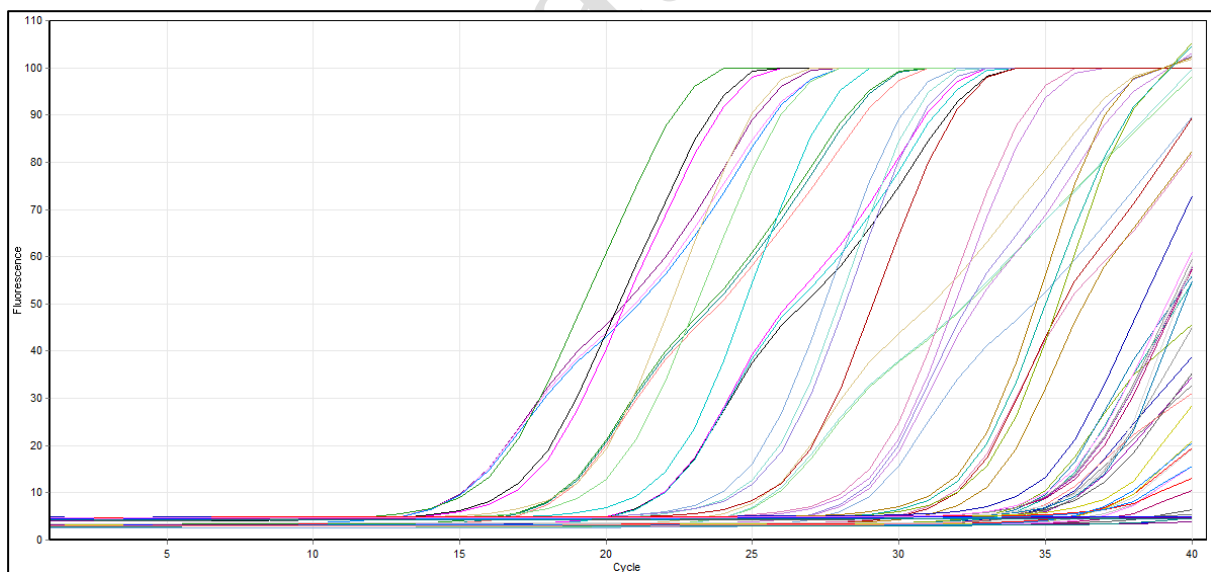


Figure 5: PCR Amplification Curve  
Source: QIAGEN's Q-Rex Software

In [15, Fig. 1, A] depicts the three phases of PCR:

- Exponential phase
- Linear phase
- Plateau phase

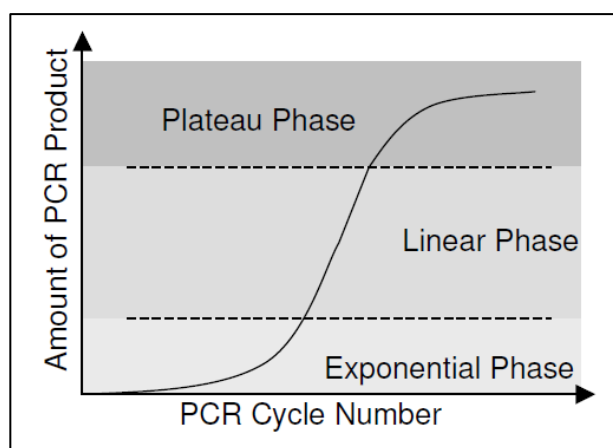


Figure 6: Theoretical plot of PCR, three phases (adapted from Yuan *et al.* [15])

The PCR will eventually reach the plateau phase during later cycles and the amount of product will not change because some reagents become depleted. The exponential phase is the earliest segment of the PCR, in which the product increases exponentially because the reagents are not limited. the linear phase is characterized by a linear increase in the product as PCR reagents become limited. the PCR will eventually reach the plateau phase during later cycles [15, pp. 1 – 2].

### 1.5.5 HIGH-RESOLUTION DNA MELT ANALYSIS

A novel DNA analysis technique called High-Resolution Melting (HRM/HRMA) is a significant method that was developed in 2002 through a collaborative effort between the University of Utah, USA, and Idaho Technology Inc., USA [16, p. 219]. for analyzing genetic variations such as SNPs (single nucleotide polymorphisms), mutations, and methylations in PCR amplicons. It is a homogeneous, close-tube, post-PCR technique that allows researchers to study the thermal denaturation of double-stranded DNA in greater detail than *traditional melting curve analysis*, resulting in higher information yield [16, 17], with the advanced hardware.

By analyzing the disassociation (melting) behaviour of nucleic acid samples, HRMA can differentiate between samples based on their sequence, length, guanine-cytosine (GC) content, or strand complementarity and it can even detect single base changes like SNPs [18]. It is a powerful tool that enables the detection of unknown variations in PCR amplicons, making it a valuable alternative to sequencing and the range of applications including [19]:

- Mutation discovery
- Screening for loss of heterozygosity
- DNA fingerprinting
- SNP genotyping
- DNA methylation analysis

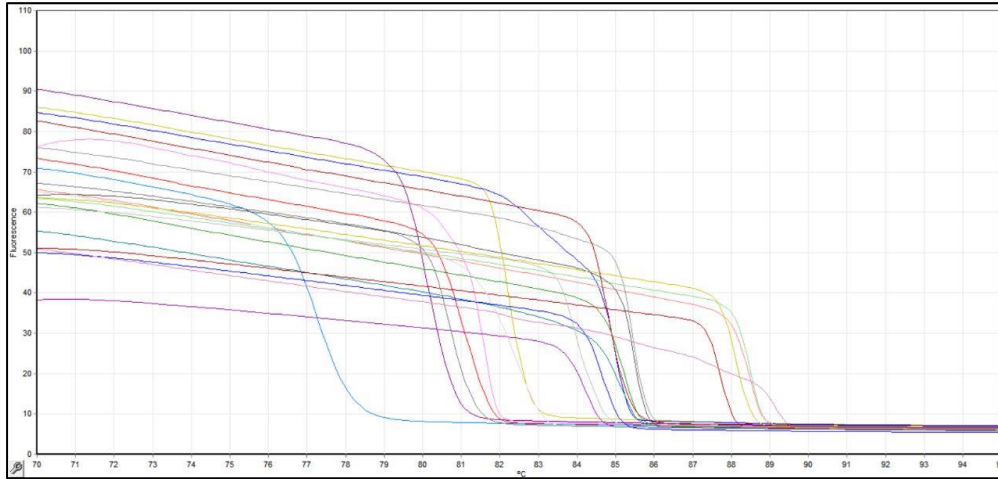


Figure 7: High-Resolution Melt Curve  
Source: QIAGEN's Q-Rex Software

Duplicate

## 1.5.6 OVERVIEW OF DNA MELT SIGNAL INTERPRETATION

DNA melt signals are the important output of PCR experiments, as they provide information about the characteristics of the amplified DNA. As the temperature increases, the double-stranded DNA begins to denature into single-strand, and the DNA-binding dye will dissociate from the DNA, causing a decrease in fluorescence. The temperature at which half of the double-stranded DNA is denatured is called the melting temperature ( $T_m$ ).

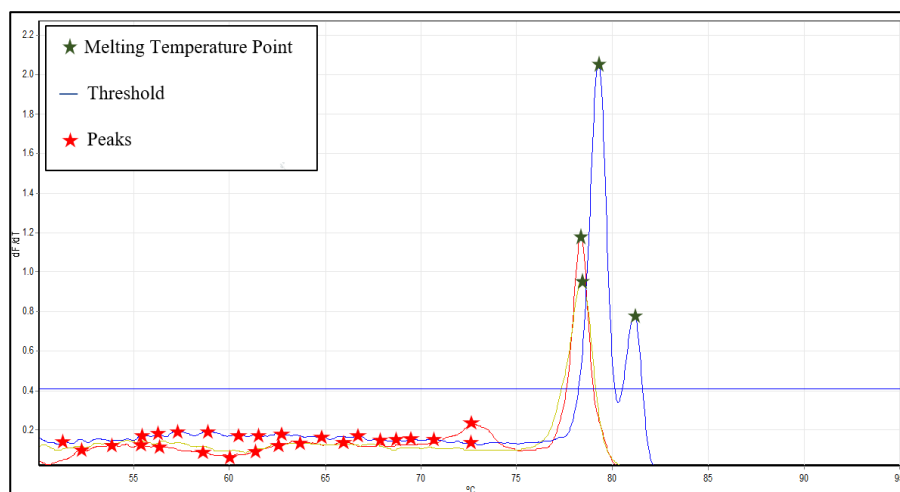


Figure 8: Reference DNA melting signal – Meningitis Panel

Each pathogen will be having different DNA melting temperatures and their respective signals will also possess different shapes and sizes. Usually, DNA melting signals are in bell-shaped curve with peaks, which denotes the melting point or the melting temperature of the DNA. Interpretation will be made through visual inspection, performed on such signal's shape, peaks, and size. Fig. 8 shows the DNA melting signals of **Haemophilus influenzae (HI)**, **Streptococcus pneumoniae (SP)**, and **Neisseria meningitidis (NM)**.

No.	Colour	Pathogen Name	Temperature of melt-Observed $T_m$	Temperature of melt-Expected $T_m$	Cycle of Threshold	Cycle of Threshold	Threshold
1	Red	Haemophilus influenzae	78.35 °C	77±1 °C	27.25	26	0.4
2	Yellow	Streptococcus pneumoniae	78.40 °C	78±1 °C	24.88	25	0.4
3	Blue	Neisseria meningitidis	79.27 °C /81.13 °C	79/81±1 °C	19.52	2	0.4

Table 1: DNA melting temperature standards for Meningitis Panel of the positive sample [fig. 8]



## A. Melting Temperature

Pathogen	Target $T_m$
<i>Acinetobacter baumannii</i>	81°C
<i>Bacteriodes fragillis</i>	80 °C
CONS-coagulase-negative <i>Staphylococci</i> .	80 °C
<i>Enterobacter</i> spp.	84 °C
<i>Enterococcus faecalis</i>	82 °C
<i>Enterococcus</i> spp.	84 °C
Group A <i>Streptococcus</i> -GAS	82 °C
<i>Serratia macesens</i>	85 °C
<i>Staphylococcus</i> spp. (Gram-positive)	77 °C
<i>Streptococcus agalactiae</i> -GBSC	77 °C
<i>Streptococcus pneumoniae</i> (Gram-positive)	78 °C

Table 2: Sample DNA melting temperature standards for Sepsis Panel

There is a set of pre-defined DNA melting temperature standards recorded by clinicians and microbiologists for identifying and distinguishing melt signals during the interpretation process. Peaks found from such melt signals (Fig. 8) for a proposed pathogen, that seemed to be satisfying ( $\pm 1$ ) the standard, will be treated as ‘Positive’, and if not, will be treated as ‘Negative’ and vice-versa.

## B. Thresholding

Thresholding, on the other hand, plays a crucial role in this process, which is being set manually during the analysis, for eliminating noisy signals and unwanted peaks. It is a simple numerical figure on the *y-axis* (i.e., derivative of fluorescence over temperature), where only those peaks will be considered on or above such numerical figure (Fig. 8 and Table 1).

Apart from Melting temperature and threshold, there are several other parameters that must also be put into consideration for interpreting signals, and they are,

- Temperature point at which the signal start rising.
- Temperature point at which the signal falls/saturated.
- Prominence of the signal.
- Area under the curve.

Such attributes of DNA melt signals will be considered as features, and relevant feature engineering techniques must be employed to bring the best out of them. The techniques and methodologies are briefly elaborated in the upcoming sections.

## 1.6 ROLE OF DATA IN REAL-TIME PCR

The post-PCR methods such as Melt Curve Analysis, Amplification Curve Analysis and HRMA are undergone using rtPCR machines, followed by the PCR experiment using thermal-cycler machines. In common, most post-PCR data are collectively known as DNA melting curves, amplification curves, and HRMA [1.5.3 – 1.5.5].

- **Amplification curves:** PCR generates amplification curves that show the increase in fluorescence signal over time, indicating the amount of amplified DNA. This data can be used to determine the starting amount of target DNA and to assess the efficiency and sensitivity of the PCR reaction.
- **Melting curves:** PCR melting curves show the dissociation of double-stranded DNA into single strands as the temperature is increased. These curves can be used to determine the melting temperature ( $T_m$ ) of the amplified DNA, which can help to identify specific DNA sequences and to detect mutations.
- **HRMA data:** HRMA provides information on the melting behaviour of PCR products, which can be used to identify and distinguish different PCR amplicons based on their melting temperature ( $T_m$ ). HRMA data can be used to detect mutations, SNPs, and other sequence variations in DNA samples. It can also be used to evaluate PCR performance, including specificity and sensitivity, and to optimize PCR conditions.

From the discussion of Vaerman *et al.*, the interpretation of rtPCR results data including various numerical data (other than melting curves, amplification curves and HRMA) that grant the assessment of various analytical parameters such as linearity, accuracy, precision, specificity, and so on [20] to determine the rtPCR instrument's efficiency, specificity and other results. From the post-PCR data, the HRMA data (i.e., melting curves) is promising to do analyses and interpretations about the specificity and identity of the amplified product (i.e., target DNA) by employing the pioneering ML techniques. Overall, HRMA data plays an important role in real-time PCR by providing valuable information on PCR products and helping to improve the accuracy and reliability of PCR-based assays.

## 1.7 PREDICTIVE ANALYSIS IN DIAGNOSIS

Diagnostic is itself an analysis of “What happened?”, “Why happened?” and “Where happened?”. There are a lot of advancements are been introduced day by day in healthcare and some of them already exist. In such a way, predictive analysis of diagnostic data is not a new approach. There are a lot of provisions and support were already been introduced to aid many researchers and organizations in boosting their routine work.

Some examples are,

- Predicting heart disease using electronic data, medical data, and patient information.
- Predicting various health complexities using medical images like X-rays, CT scans, and MRIs.
- Predicting Cancer with data on tumours (benign or malignant).

In the field of molecular diagnosis, predicting targets may involve considering several factors. Melt curve analysis is one of the steps in diagnosing with PCR and it gives information more on the melting nature of a dsDNA. Clinicians/Microbiologists will study the melt curves and observe the distinct variations in the graphs thus determine the presence of a specific target DNA. It is important to note that only melting analysis would not suffice to confirm any presence of target DNA (pathogen) in a patient sample, and additional analysis may require such as

- Sequencing
- Phylogenetic analysis
- Multiplex PCR
- Specific primer/probe design
- Culture and isolation
- Serology

As a result, this project will cover predictive analysis on PCR diagnosis data, specifically HRM data, that gives Melt curves and peaks. With relevant feature extraction and feature engineering, finally, predictions will be made on classifying pathogen classes using various predictive analysis techniques.

## 1.7.1 MACHINE LEARNING IN DIAGNOSIS

Machine Learning is a great choice of predictive modelling, and it is a robust technology, which has been globally applied in various applications ranging from classifying spam mail to predicting diseases. Due to its high compatibility and sound algorithmic resource, many real-world problems can be solved using Machine Learning and diagnosis is not an exception for applying it.

### **Machine Learning over Statistical Modelling**

Both tend to use similar predictive approaches like regression, classification, and clustering, but they differ in many ways.

Statistical modelling is a subset of mathematical modelling where it hugely involves assumptions, relationships between random and non-random variables, and estimating population with sample data. Choosing statistical modelling as a predictive analysis technique will require more understanding of the variables involved in the data and the respective relationships between them. Once these perceptions are satisfied, a sensible assumption has to be made, to explain the relationships between variables and the resulting prediction. This is why statistical modelling is highly preferred when proper interpretation and explanations are demanded.

On the other hand, Machine Learning is also another predictive analysis technique, which is a branch of computer science and artificial intelligence, that mainly works on the principle of pattern analysis and often introduces challenges in interpreting their learning pattern. Compared to statistical modeling ML models can work with large data sets, and it rejects the chances of making assumptions on the given data, as it learns from the pattern through a weight-based approach. As a result, predictions by these models are powerful and more accurate.

In the context of performing predictive analysis with HRM data, typically it's biological data which is complicated due to its complexity and high variability in nature because every experiment is done and influenced under the clinical environment. So, the Machine Learning approach is preferable over statistical modeling, owing to its pattern learning process which gives highly accurate results while statistical modeling is best if the characteristics of the data should not vary for formulating the hypothesis. Concurrently, several statistical techniques

were also used to find insights during the development of this project. Besides both techniques have their advantages and limitations [21, 22].

## **1.8 OVERALL RESEARCH AIM AND OBJECTIVES**

The overall aim of the project is to create an AI-based framework for analyzing and interpreting the HRM data without involving any human assistance. The scope of the project starts from the necessary data extraction/acquisition to the end report presentation.

The objectives can be enumerated as:

- To setup data acquisition pipelines for extracting and acquiring all the data, necessary for the analysis and interpretation.
- To develop data pre-processing modules for cleaning and transforming acquired data.
- To conduct research on the domain and the given problem to formulate the best and most suitable solution.
- To conduct relative research on previously undergone research and solutions made for problems in the same regard.
- To formulate an effective approach to applying Machine Learning algorithms to the problem.
- To evaluate and validate results with domain expertise and look for further improvisation.
- To setup modules and components for effective data storing and accessing.
- To develop supporting software components to aid the workflow.
- To augment all the components into a single apex system.

## CHAPTER 2

### EXISTING SYSTEM

The High-Resolution Melting Analysis are being done with the help of specific software and existing hardware components. In practice, without these components, HRM cannot be done. However, doing manually is more complex and prone to error. Since the term “**High-Resolution**” itself depicts the technology of capturing fluorescence in “**High-Resolution**” which is thereby demanding sophisticated and engineered technical components. There are already many commercially available instruments and plugin tools used for running PCR tests, and most of them are engineered with cutting-edge technology.

#### 2.1. DATA ANALYSIS SOFTWARE

##### 2.1.1 QIAGEN’s ROTOR-GENE

Rotor-Gene Q series are commercial thermal cycle instruments used in many laboratories for running PCR tests and analyzing the results using their respective versions of plugin software [24].

Rotor-Gene offers several individual components for analysis like,

- *Melt Analysis*
- *HRM Analysis*
- *ScreenClust Analysis*

The components come with various names and versions, and among them, QIAGEN’s ‘**Rotor-Gene Q-Rex**’ is a default software tool that comes with every Rotor-Gene instrument for analyzing the run files of completed PCR tests. Q-Rex offers both *Melt* and *HRM* analysis in a single package.

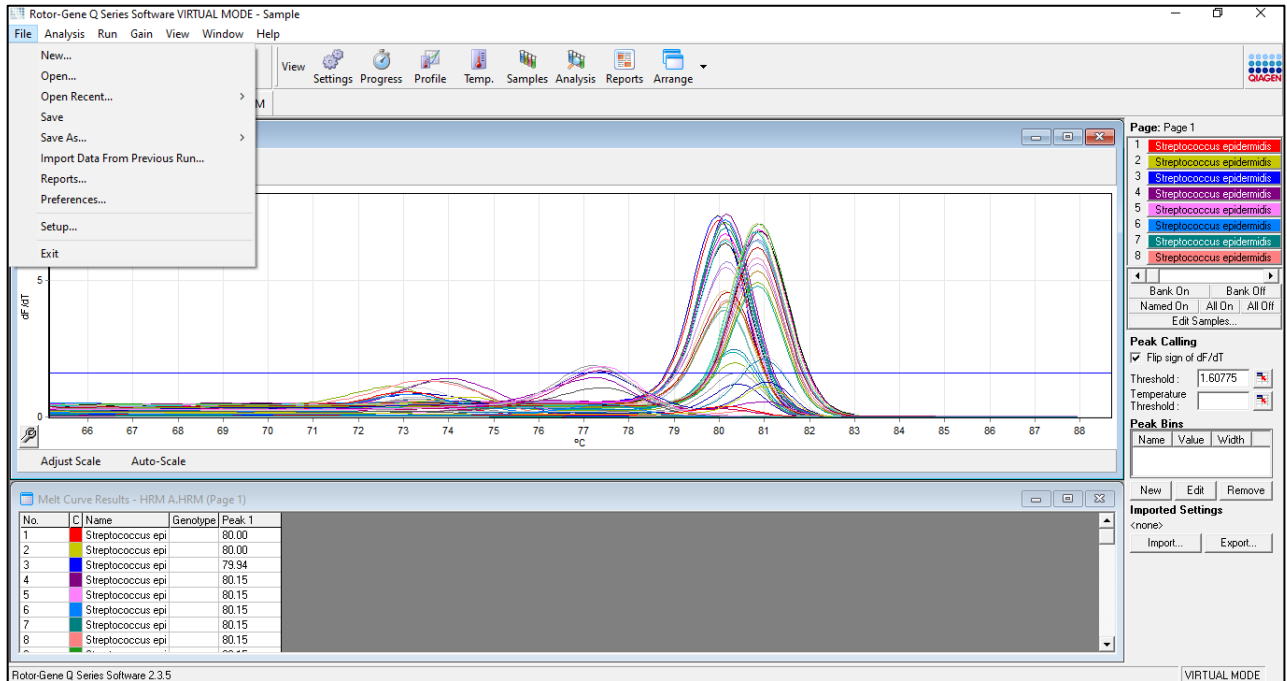
##### 2.1.1 (a) Rotor-Gene Q-Rex Software

###### Melt Analysis

The "Melt Curve Analysis" function of the Rotor-Gene Q software can be used for checking the specificity of a reaction, genotyping, and measuring protein stability with differential scanning fluorimetry. The function analyses the first derivative ( $dF/dT$ ) of the raw

melting data and identifies peaks in the selected temperature and fluorescence range. The data can be used for genotyping by defining "Peak Bins" based on peak characteristics of known genotypes [24].

**A**



**B**

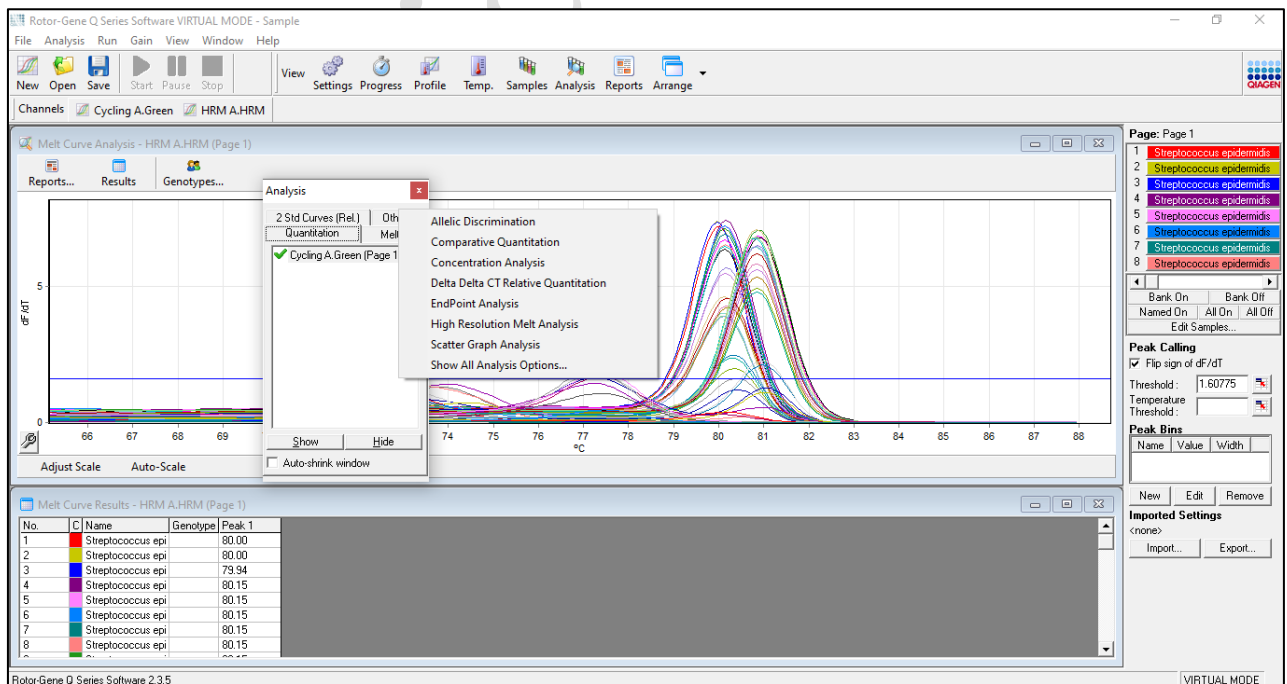


Fig 9, A and B: Rotor-Gene Q-Rex Interface  
Source: QIAGEN's Q-Rex Software

## HRM Analysis

High-resolution melting (HRM) analysis using Rotor-Gene Q software involves selecting a data set and defining normalization regions to compensate for variations between samples. Genotype names are assigned, and control samples are selected. Results are displayed in the “HRM Results” table with automatic identification results for each genotype. Confidence levels are assigned to each sample, and a threshold value for the "Confidence Percentage" can be defined. The “HRM Normalized Graph” plot displays different curves relative to a selected genotype in a “Difference Graph” to emphasize differences between samples. Once the process is complete, tables and graphs can be exported [24].

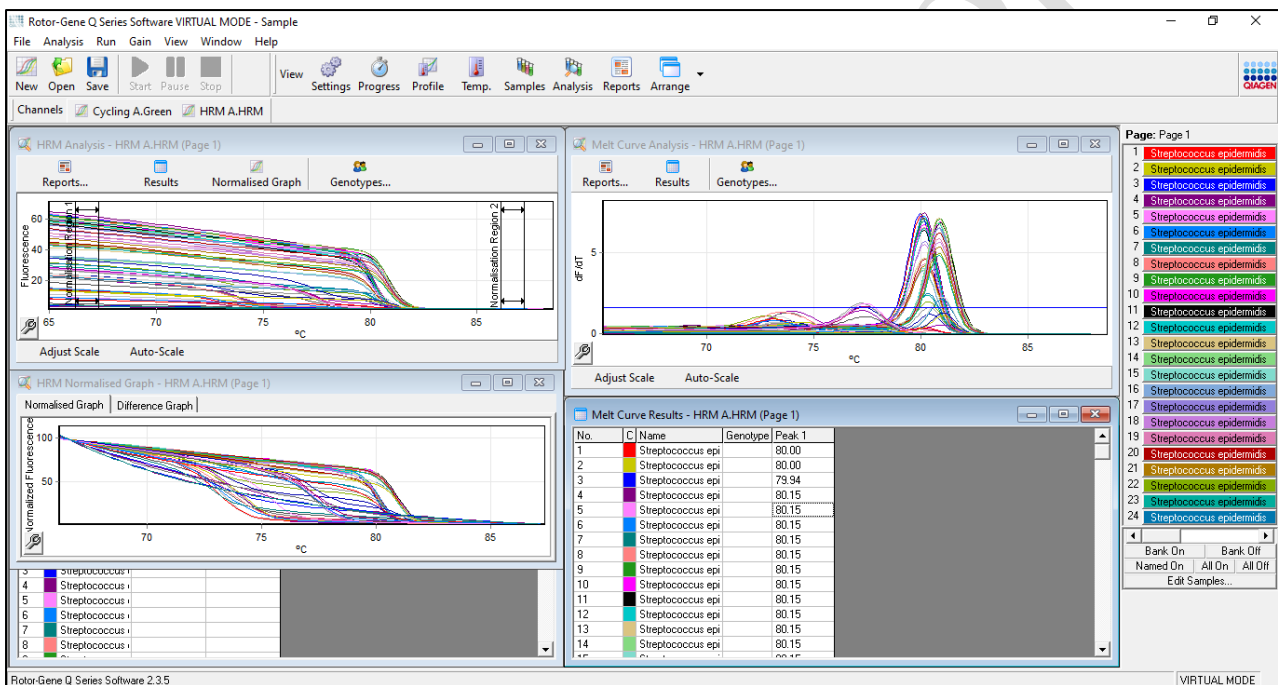


Fig 10: Rotor-Gene Q Rex Interface  
Source: QIAGEN's Q-Rex Software

### 2.1.1 (b) Rotor-Gene ScreenClust HRM Software

Rotor-Gene ScreenClust HRM Software is a powerful tool for the analysis of high-resolution melting (HRM) data from the Rotor-Gene Q or Rotor-Gene 6000 cyclers. By grouping samples into clusters based on their dissociation (melting) curve characteristics, Rotor-Gene ScreenClust HRM Software enables applications such as genotyping and mutation scanning. The number of clusters can either be defined by the user, if they have known controls



for each genotype (supervised mode) or the software can aid the user in determining the number of clusters in a sample set (unsupervised mode).

Rotor-Gene ScreenClust HRM Software provides:

- Innovative mathematical approach to HRM analysis.
- Highly accurate identification of genotypes in supervised mode.
- Automatic detection of new mutations in unsupervised mode.
- Robust statistics for classifying and interpreting HRM data.
- Minimal effort and standardized processes for data interpretation.

HRM analysis on a Rotor-Gene cycler produces raw data that can be further analyzed using Rotor-Gene ScreenClust HRM Software. Rotor-Gene ScreenClust HRM Software analyses HRM data in 4 steps:

1. Normalization
2. Generation of a residual plot
3. Principal component analysis
4. Clustering

HRM curves can have different starting points, therefore the scale of each melt is different [25, Fig. 1-A). Rotor-Gene ScreenClust HRM Software only compares samples that are on the same scale, which is achieved by normalization. Raw data are normalized by applying curve scaling to a line of best fit so that the highest fluorescence value is equal to 100 and the lowest is equal to zero [25, Fig. 1-B]. Next, the curves are differentiated and a composite median curve is constructed using the median fluorescence of all samples. The melt traces for each sample are subtracted from this composite median curve to draw a residual plot [25, Fig. 1-C). Consecutively, the individual sample characteristics are extracted by principal component analysis from the residual plot. The principal component analysis is a well-established method of data analysis.

However, Rotor-Gene ScreenClust HRM Software is the first software application to apply principal component analysis to HRM data. The principal component analysis highlights similarities and differences in the data and is used to create a cluster plot [25, Fig.1-D].

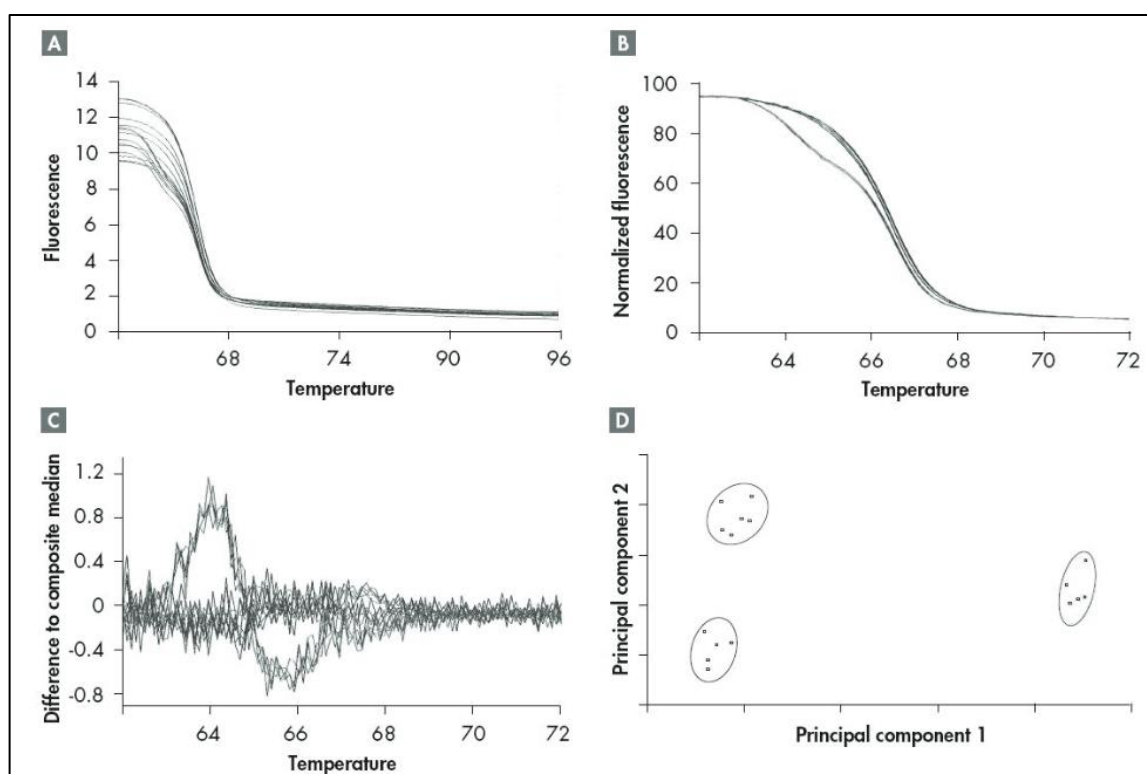


Figure 11: Data analysis performed in ScreenClust HRM Software (adapted from ScreenClust Software user manual [25, Fig. 1])

Rotor-Gene ScreenClust HRM Software performs clustering (grouping) of data according to allele in either supervised or unsupervised mode. Supervised mode is often used for SNP genotyping, where the genotypes are known. In supervised mode, the user assigns one or more control samples for each cluster and the software classifies (auto calls) all unknown samples to clusters according to their characteristics. The unsupervised mode is used when there is no or only partial prior knowledge of the genotypes present in the samples. In unsupervised mode, the software calculates the optimal number of clusters by itself. This feature is an excellent tool for the discovery of new polymorphisms. In addition to the easy-to-interpret cluster plot, Rotor-Gene ScreenClust HRM Software provides statistical probabilities and typicalities in a results table to allow easy comparison of results from different experiments.

## 2.1.2 BIO-RAD'S CFX SERIES

### 2.1.2 (a) CFX Manager

The software plots the relative fluorescence unit (RFU) data collected during a melt curve as a function of temperature. To analyze melt peak data, the software assigns a beginning and ending temperature to each peak by moving the threshold bar. The floor of the peak area is specified by the position of the melt threshold bar. A valid peak must have a minimum height relative to the distance between the threshold bar and the height of the highest peak.

- Melt Curve: Viewing the real-time data for each fluorophore as RFUs per temperature for each well.
- Melt Peak: Viewing the negative regression of the RFU data per temperature for each well.
- Well selector: Wells to show or hide the data.
- Peak spreadsheet: Viewing as a spreadsheet of the data collected in the selected well.

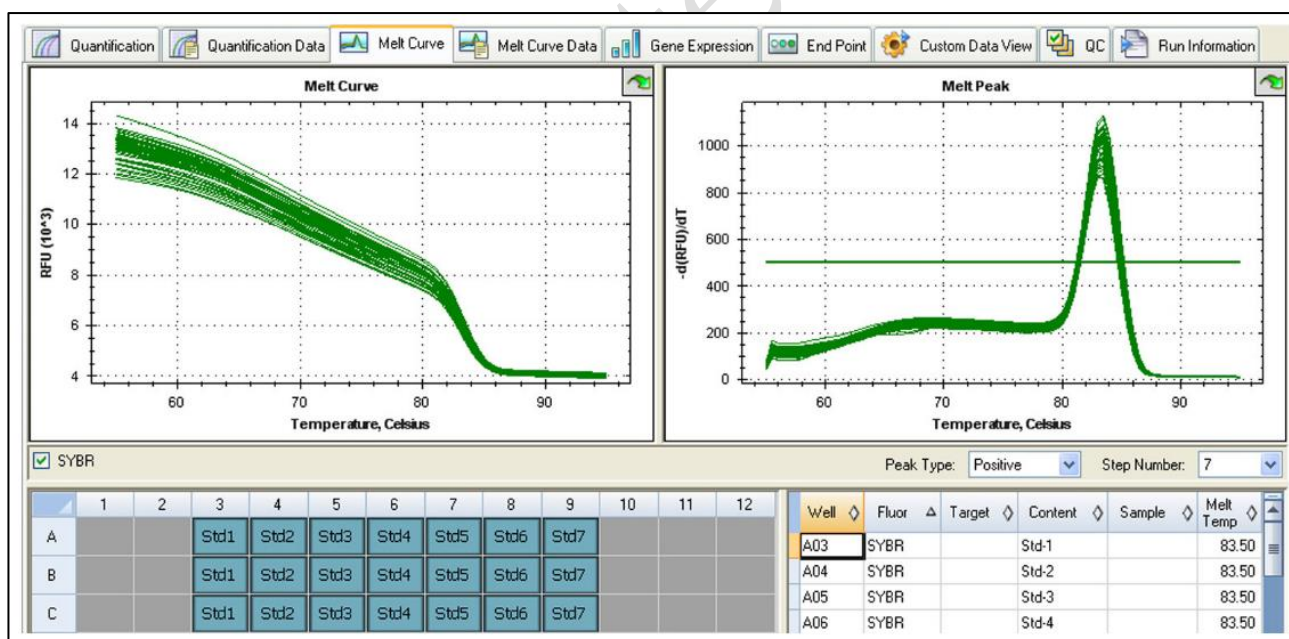


Figure 12: Interface of BIO-RAD CFX Manager

The Melt Curve Data shows the data from the Melt Curve in multiple spreadsheets, including all the melt peaks for each trace. Select one of these four options to show the melt curve data in different spreadsheets:

- Melt Peaks: Listing all the data, including all the melt peaks, for each trace
- Plate: Listing a view of the data and contents of each well in the plate
- RFU: Listing the RFU quantities at each temperature for each well
- $-d(\text{RFU})/dT$ : Listing the negative rate of change in RFU as the temperature (T) changes. This is the first regression plot for each well in the plate

Well	Fluor	Content	Target	Sample	Melt Temperature	Peak Height	Begin Temperature	End Temperature
A01	SYBR	Std-1			86.00	1502.14	82.00	88.00
A02	SYBR	Std-2			86.00	1496.90	81.50	88.00
A03	SYBR	Std-3			86.00	1496.51	82.00	88.00
A04	SYBR	Std-4			86.00	1523.68	81.50	88.00
A05	SYBR	Std-5			86.00	1369.55	82.00	88.00
A06	SYBR	Std-6			86.00	1379.17	82.00	88.00
A07	SYBR	Std-7			86.00	1282.97	82.00	88.00

Figure 13: Melt Peak Spreadsheet – CFX Manager

## 2.1.3 BIO MOLECULAR SYSTEMS – MIC

### 2.1.3 (a) micPCR Software

The micPCR software offers a Melt Analysis option that enables the determination of the peak dissociation temperature ( $T_m$ ) of a sample from the melt data. This feature is useful in detecting non-specific amplicons like primer dimers, thereby serving as a measure of analytical specificity for an assay. Melt Analysis can also be applied for genotyping using chemistries such as dual hybridization probes. The software displays a graph of the first derivative curve plotted as  $dF/dT$  (y-axis) against temperature ( $^{\circ}C$ , x-axis) for the first target selected in the Assays list. Users can set the melt curve threshold to any value and adjust other melting parameters available for genotyping.

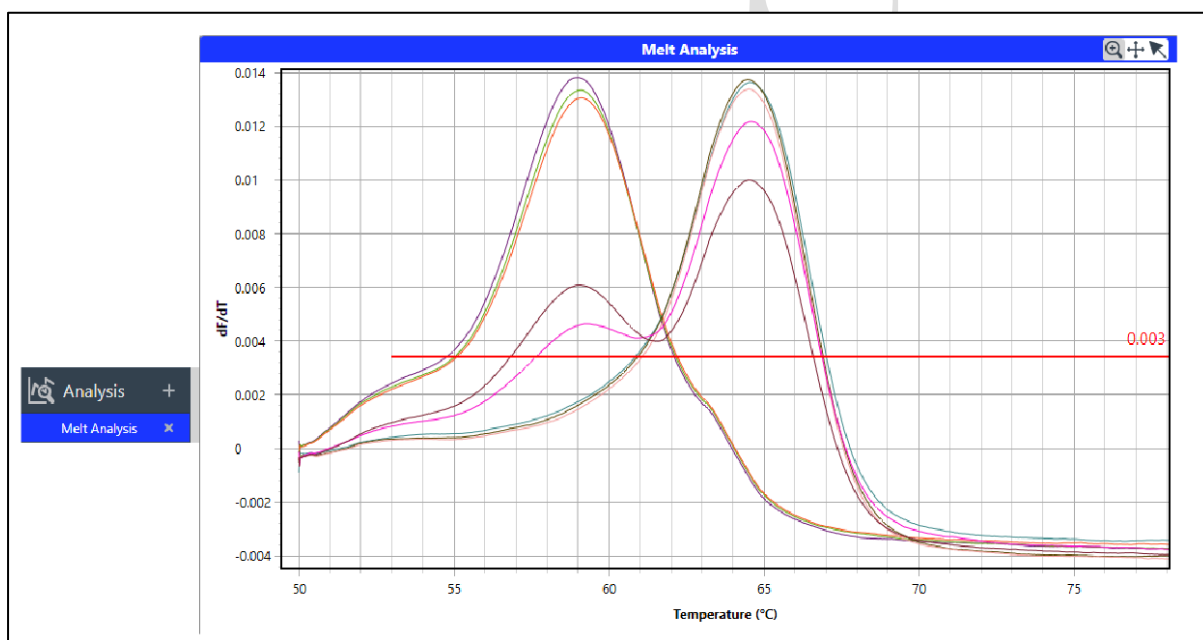


Figure 14: Interface of melt curve graph in Bio Molecular System micPCR software

## 2.1.4 THERMO FISHER – QUANTSTUDIO

### 2.1.4 (a) QuantStudio Design & Analysis Software

The Melt Curve experiment is used in Thermo Fisher PCR reactions with SYBR Green dye to determine the melting temperature ( $T_m$ ) of the amplification products.  $T_m$  is the temperature at which 50% of the DNA is double-stranded and 50% is dissociated into single-stranded DNA. Melt Curve analysis is included in the default run method for any experiment type that uses SYBR Green reagents. Multiple peaks in a melt curve indicate additional amplification products, often due to non-specific amplification or primer-dimer formation.

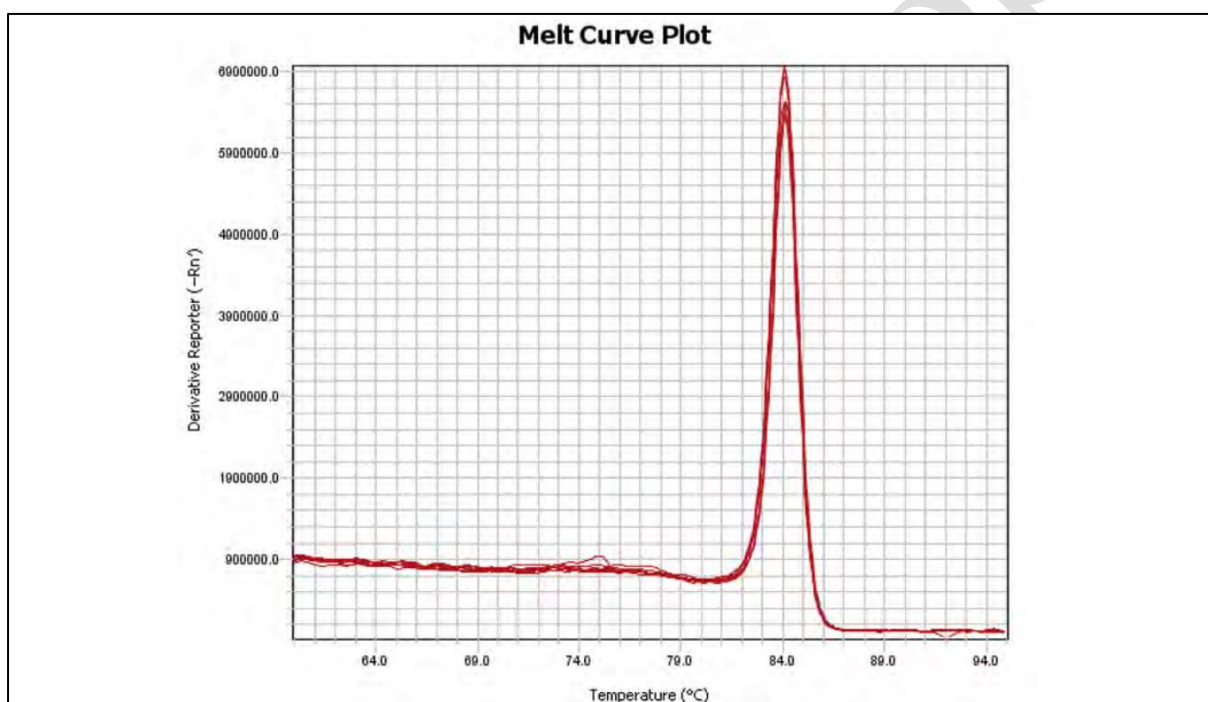


Figure 15: Interface of Melt curve graph in ThermoFisher QuantStudio software

## 2.1.5 ROCHE – LIGHTCYCLER SERIES

### 2.1.5 (a) LightCycler Software

The LightCycler uses fluorescence measurements to perform melting temperature analysis, which determines the melting temperature ( $T_m$ ) of each sample. The analysis produces a Melting Curves chart that shows the downward curve in fluorescence as samples melt and a Melting Peaks chart that plots the first negative derivative of sample fluorescent curves to display the melting temperature of each sample as a peak. This allows for easier comparison between samples.

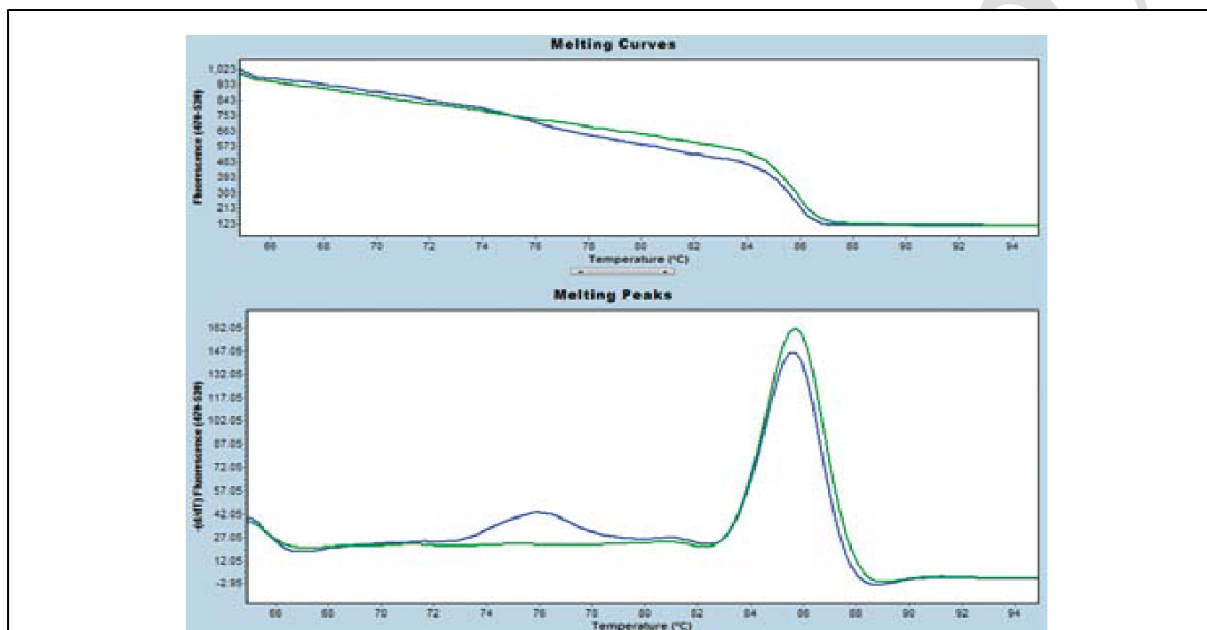


Figure 16: Interface of Roche's LightCycler

## 2.2 DEMOCRATIZED SOFTWARE

### 2.2.1 DNA-UTAH

#### 2.2.1 (a) uANALYZE

uAnalyze is a web-based tool that analyses high-resolution melting data of PCR products. It uses recursive nearest-neighbour thermodynamic calculations to predict a melting curve. The tool accepts unprocessed melting data from *LightScanner-96*, *LS32*, or *HR-1* data files or via a generic format for other instruments. A fluorescence discriminator identifies low-intensity samples, and the background is removed either as an exponential or by linear baseline extrapolation. The precision and accuracy of experimental melting curves are quantified, and a temperature overlay is provided to focus on the curve shape.



Figure 17: uANALYZE Interface





## CHAPTER 3

### LITERATURE REVIEW

Untergasser *et al.* 2021, made the analyses of amplification and melting curves to provide valuable information on the quality of individual reactions in quantitative PCR (qPCR) experiments and result in more reliable and reproducible quantitative results. The new web-based LinRegPCR web application provides visualization and analysis of a single qPCR run, displaying the analysis results on the amplification curve and melting curve analysis in tables and graphs. It also provides a stand-alone back-end RDML (Real-time PCR Data Markup Language) Python library and several companion applications for data visualization, analysis, and interactive access. The use of the RDML data standard enables machine-independent storage and exchange of qPCR data, and the RDML tools assist with importing the data from the files exported by the qPCR instrument.

Moniri *et al.* (2020) demonstrates that the large volume of raw data obtained from real-time PCR instruments can be exploited to perform data-driven multiplexing in a single channel using machine learning methods. This approach, referred to as Amplification Curve Analysis (ACA), was used to multiplex 3 carbapenem-resistant genes in the presence of single targets, resulting in an accuracy of 99.1% (N = 16188). To support the analysis, a formula was derived to estimate co-amplification occurrence in PCR based on multi-variate Poisson statistics. Combining this method with probe-based assays will increase multiplexing capabilities.

Wisittipanit *et al.* 2020 used a modified high-resolution DNA melting curve analysis (m-HRMa) to classify *Salmonella* spp. into clusters and a machine learning (dynamic time warping) algorithm (DTW) to create a phylogeny tree of *Salmonella* strains (n = 40) collected from homes, farms, and slaughterhouses in northern Thailand. DTW and ms-HRMa clustering analyses were able to generate molecular signatures of the *Salmonella* isolates, resulting in 25 ms-HRM and 28 DTW clusters compared to 14 clusters from a standard HRM analysis. The new *Salmonella* sub-typing protocol identified five *S. Weltevreden* subtypes with *S. Weltevreden* subtype DTW4-M1 being predominant. This suggests that transmission of salmonellosis in northern Thailand is likely to be farm-to-farm through contaminated chicken stool.

Athamanolap *et al.* 2014 made an automated HRM curve classification based on machine learning methods and learned tolerance for reaction condition deviations enables reliable, scalable, and automated HRM genotyping analysis with broad potential clinical and epidemiological applications.

Roediger *et al.* 2013. implemented the MBmca package with R, for DNA Melting Curve Analysis on microbead surfaces. Particularly, for the use of the second derivative melting peaks as an additional parameter to characterize the melting behaviour of DNA duplexes.

Dwight *et al.* 2011 created a web-based tool called “uMeltSM” for predicting DNA melting curves and denaturation proles of PCR products. It uses an accelerated partition function algorithm to calculate and visualize the mean helicity and dissociation probability at each sequence position at different temperatures. Results from fluorescent melting experiments match the number of predicted domains and their relative temperatures, but current libraries do not account for the rapid melting rates and helix-stabilizing dyes used in experiments.

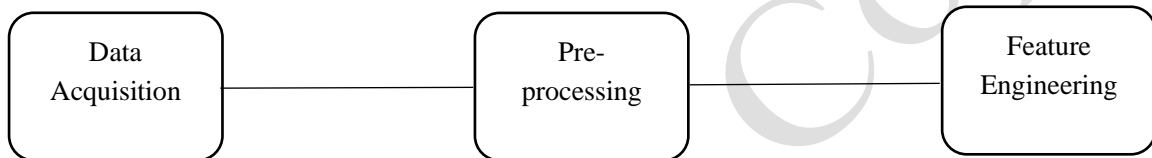
Smith *et al.* 2009 defines Methylation of DNA as a common mechanism for silencing genes and is increasingly being implicated in many diseases. They describe and validate a rapid, in-tube method to quantitate DNA methylation using the melt data obtained following the amplification of bisulphite-modified DNA in a real-time thermocycler. The parameters derived provide an objective description and quantitation of the methylation in a specimen and can be used for statistical comparisons of methylation between specimens.

## CHAPTER 4

### PROPOSED METHODOLOGIES

Developing an AI-based framework for interpreting and reporting PCR tests, require necessary research and primary attention towards understanding key result analyzing techniques such as melt curve analysis and cycle threshold analysis. Therefore, conducting proper research is crucial for formulating any robust solution for the problem given.

#### 4.1 KEY ASPECTS



The research and methodologies in this project have three main key aspects, namely Data Acquisition, Pre-processing, and Feature Engineering. Approaches were made, concerning all the above aspects as stated.

- How data acquisition can be performed?
- What type of data is required?
- How pre-processing should be done?
- Is pre-processing necessary?
- How feature engineering can be done effectively?

The research and methodology of this project predominantly focuses on the above stated key aspects, and satisfying all of them, will constitute a larger contribution. Most of the work in this project follows, how data is being collected and how it is leveraged, such that expected outcomes are brought in. Following that, several methodologies was found and applied on various scenarios, which gave deterministic and non-deterministic results.

Since the project aims in introducing Machine Learning based approach for interpreting the results, proper attention has to be made for data acquisition and feature engineering.

## 4.2 PROPOSED METHODOLOGIES

The project comprises of various methodologies, proposed and implemented in various sections, where each of them provide certain observable results, that are notable and few of them demand for future implementation.

- Approach on images of DNA melting signal graphs.
- Approach on co-ordinates of raw fluorescence signal.
- Approach on co-ordinates of DNA melting signal graphs.
- Combination of approach on images and the co-ordinates of DNA melting signal.

The above methodologies are proposed and implemented in this project as part of feature engineering process. Moreover, not all the methodologies stated above are truly applicable and successful, as some of them are preliminary experiments and trails. Each methodology relied on data and respective data acquisition methods are also covered in the following sections.

Duplicate

# CHAPTER 5

## APPROACH ON IMAGES OF DNA MELT SIGNALS

As discussed earlier, melt curves are interpreted by visual inspection of their respective graphs that is the result of the rtPCR assay. Experts then examine every feature in the graph and draft their report/interpretation accordingly.

In the onset of this project the type of data that is more suitable for developing the AI framework was unknown. Hence, initially the available images of the melt curve graph for the different patient sample were used as the source data for developing the algorithm.

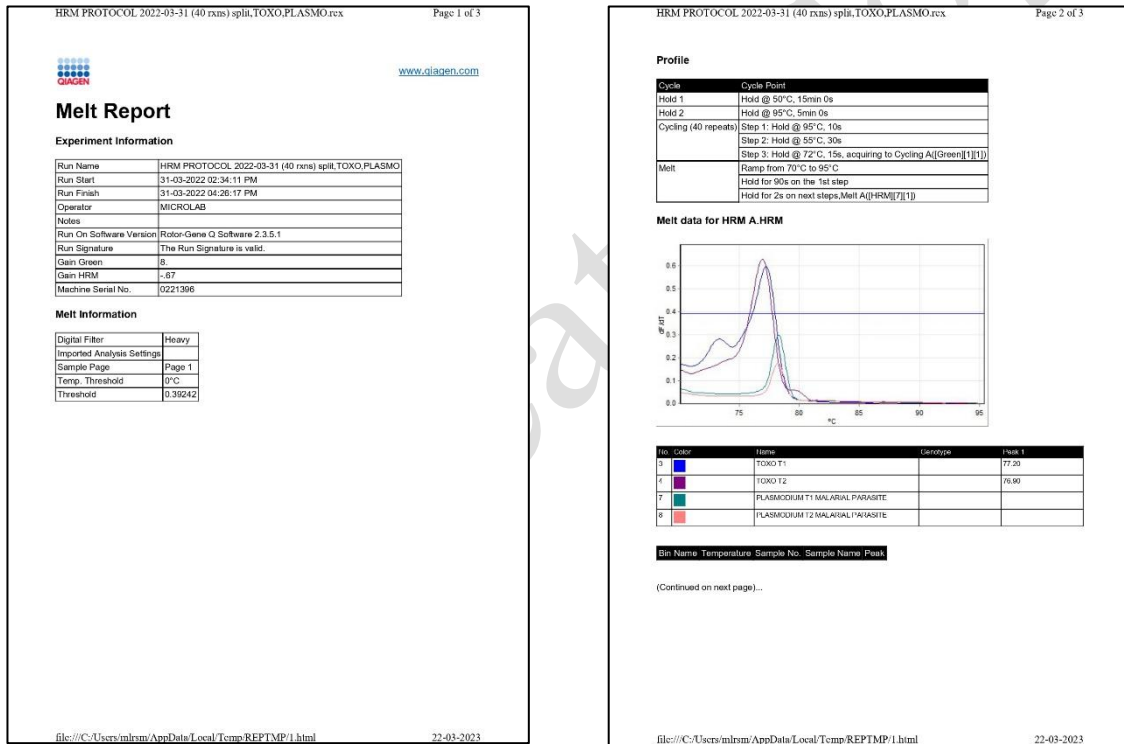


Figure 19: DNA Melt data report generated by thermal cycler machine

Reports are the only initial source of data, where information on melt data are available. These reports are created by commercially available thermal cycler machines, after performing the interpretation by clinicians/microbiologists. Since the data is available in the form of images, image processing techniques are performed as an initial step, to explore the available image source.

## 5.1 IMAGE PROCESSING

Image Processing is a standard technique, widely used on various applications like computer vision, video processing, remote sensing etc., to extract and restore valuable information in the image by the use of algorithms. Image processing has various applications like image sharpening, image restoration, image reconstruction, image reformatting, image generation and so on, for enhancing the existing image resource.

In this context, as the data available from the physical paper reports, it is likely to have poor quality and there are more chances of having noises. To suppress such constraints, image processing is a promising solution. There are many libraries and packages, that are readily available for image-processing tasks, and among those, OpenCV is a familiar image-processing library, predominantly used on many image-based tasks and projects. Likewise, OpenCV has been used in this methodology with Python as a code language tool.

Pre-processing with OpenCV, provides many image processing methods, to manipulate images in a desired manner. Such a way to separate or distinguish the melt signals from the given plot, image masking can be performed.

Image masking is a simple technique, where only the desired pixels are taken for consideration, ignoring the background or unwanted pixels by creating appropriate masks.



Figure 20: Colour mask to track yellow colour in the input image

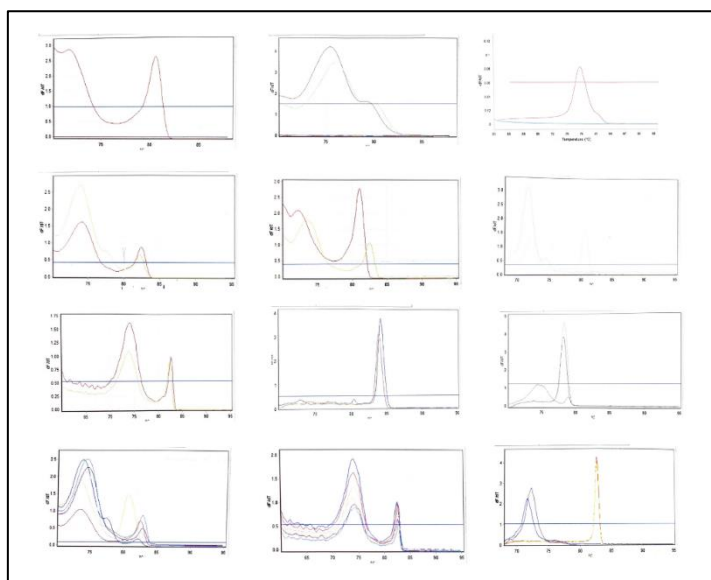


Figure 21: Melt signal images scanned and cropped from PCR reports

After collecting several report files, images of melt signals are scanned and segmented individually. The signal plots in the reports are partially clear and some of the signals are pale in colour. Most of the images, have white background that possess the RGB value of (255,255,255). Masking has to be done in such a way to remove those white coloured pixels from the images, so that only the coloured pixels are remained for consideration.

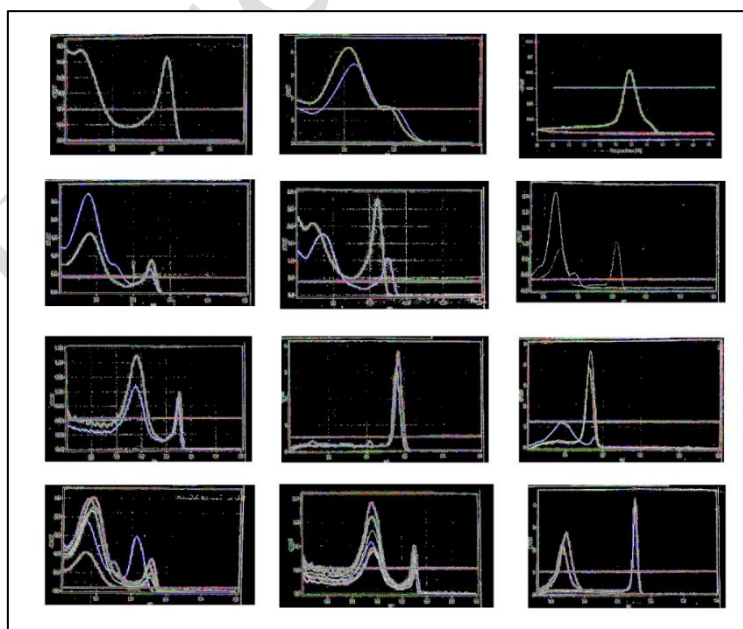


Figure 22: Image masking performed on melt signal images



Performing image masking on the DNA melt images, has removed all the white background pixels, keeping only those pixels of melt signals. However, few sections in the picture such as x-axis and y-axis legends, labels, threshold line etc., (fig. 21)) were not removed. Hence, there is a requirement of additional techniques to be applied for removing such unwanted information.

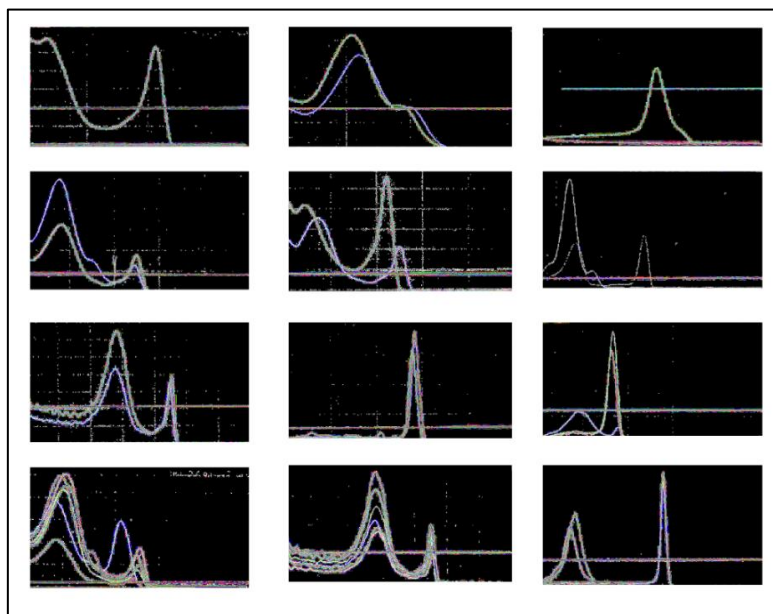


Figure 23: Cropping the images to retain only melt signals

Cropping the images helped to retain the target area of the input images. As a result, all the unwanted sections like labels, x-axis and y-axis are removed. However, the melt signals in the images are still noisy and due to the poor pixel quality of the images and the inaccuracy of the edges. Additionally, images in the reports were compressed, which is entirely not as good as image formats like SVG and PNG.

On the other hand, information on the numerical values seems to be missing (say  $T_m$ ). Unlike problems like object detection, which has the aim of identifying the desired object at any position, at any size, at any colour irrespective of numerical information (say the exact size of the object in inches or centimetres), image processing is unable to assess such attributes (numerical).

## RESULT AND DISCUSSION

The experiment on the images of melt signals has introduced image processing methods, with the aim of extracting important information from the signals. The experiment focuses on creating mask for such melt signals, which as a result, background is separated. Despite this happens, the process has not seemed to be satisfying several requirements, such as numerical attributes of the signal, sound pixel quality etc.,

The image-based approach, as a result, could capture and partially enhance the melt signals from the image, but could not assess any other import features, required for analysis. Though great-quality images are acquired, extracting/assessing the numerical attributes is challenging and that may introduce methods like *plot digitizing*.

Explorations with respect to the above constraints have to be done, and a proper substitute or applicable alternate method needs to be introduced. Instead of acquiring physical reports, exploration shall be made on the software components of the thermal cycler machines, so that, any new data sources or any alternative in this regard can be found.

## CONCLUSION

Approach on images of DNA melt signals, has provided critical results, that demand further implantation and exploration. The prevailing approach lacks areas like feature extraction and data acquisition, where the source of data is not intact and with that source, formulating a robust foundation for the problem is difficult.

## CHAPTER 6

### APPROACH ON CO-ORDINATES OF RAW FLUORESCENCE SIGNAL

In the previous approach, physical paper reports were the initial source of data, where image processing techniques are applied and respective results are observed at the end. It was found that, there were no reliable data source for extracting information on melt signals, and applying image-based techniques are only relevant with several limitations. The approach is not feasible, because of the following reasons,

- Unreliable Data
- Time consuming (Involves much pre-processing)
- Hard to differentiate signals within themselves
- Involves plot digitizing
- Could not cover numerical features

Concerning all these drawbacks, further exploration has been performed in this successive approach to address the above stated constraints.

The science behind DNA melting signal, is a light reaction, called 'fluorescence'. Fluorescence is a process that belongs to the ubiquitous luminescence family in which sensitive molecules produce light from electronically excited states created by a physical (for e.g., light absorption), mechanical (friction), or chemical mechanism. The DNA melt signals are actually this fluorescence emission, emitted by fluorescent dyes during the PCR run. Such emissions are captured and recorded by thermal cycle machines in real time.

The captured fluorescence signal is then plotted against the temperature, which as a result, signals are produced. Like all other signals (say sin waves, spike waves etc.,) melting signals also have co-ordinates (x, y co-ordinates), that can be plotted in a cartesian plane. Exploring the thermal cycler machines and its corresponding software plugin, it is discovered that, such co-ordinates can be extracted and exported into various file formats. As a result, any melting signals can be plotted manually with any data visualization tool, without depending on the software plugin of thermal cycler machines.

Text	X	Y	Text	X	Y	Text	X	Y	Text	X	Y
1: 0400437684	70	56.25	5: 0400437685	70	77.52	9: 1000675295	70	48.9	13: 1000675298	70	78.69
1: 0400437684	70.2	55.43	5: 0400437685	70.2	76.74	9: 1000675295	70.2	48.2	13: 1000675298	70.2	77.87
1: 0400437684	70.4	54.61	5: 0400437685	70.4	75.95	9: 1000675295	70.4	47.5	13: 1000675298	70.4	77.06
1: 0400437684	70.6	53.84	5: 0400437685	70.6	75.30	9: 1000675295	70.6	46.9	13: 1000675298	70.6	76.38
1: 0400437684	70.8	53.07	5: 0400437685	70.8	74.69	9: 1000675295	70.8	46.1	13: 1000675298	70.8	75.71
1: 0400437684	71	52.31	5: 0400437685	71	74.07	9: 1000675295	71	45.4	13: 1000675298	71	75.14
1: 0400437684	71.2	51.47	5: 0400437685	71.2	73.45	9: 1000675295	71.2	44.6	13: 1000675298	71.2	74.48
1: 0400437684	71.6	49.75	5: 0400437685	71.6	72.19	9: 1000675295	71.6	43.0	13: 1000675298	71.6	73.15
1: 0400437684	72.2	46.57	5: 0400437685	72.2	69.88	9: 1000675295	72.2	40.0	13: 1000675298	72.2	71.03

Table 3: Sample data - co-ordinates of raw fluorescence

The Table 3 shows the tabular representation of fluorescence intensity, captured in real time. The data specifies the 'X' and 'Y' co-ordinates of raw fluorescence signals, where 'X' denotes the temperature and 'Y' denotes the fluorescence intensity. The 'Text' column represents the name/reference id of the sample, which is being loaded in the wells. It is also clearly observable that, the 'X' column is common for all the 'Y' (fluorescence intensity) columns, which is the temperature value, that range from 70°C to 90 °C.

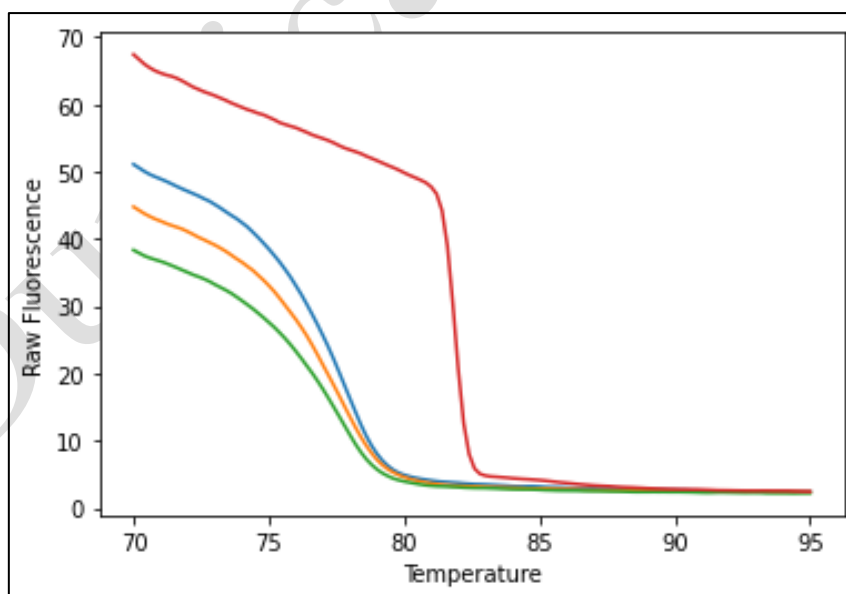


Figure 24: Raw fluorescence signal plotted using matplotlib

The shape of raw fluorescence signal is different from the DNA melting signal. In fact, the DNA melting signals are produced after applying mathematical function on the raw fluorescence signals. And the mathematical function is the first negative derivative of raw fluorescence intensity.

$$-\frac{\text{change in } y}{\text{change in } x} = -\frac{\Delta y}{\Delta x} = -\frac{dy}{dx} \quad (1)$$

There raw fluorescence signal has several properties and also provide certain information on the melting characteristic of a DNA sample. According to the method described by Smith *et al.* the team stretched out the mathematical method to find the start and end of melting temperature from a fluorescence signal. The raw fluorescence signal has an inverse proportion with the temperature, that, as temperature increases, fluorescence decrease. Such decreasing nature of fluorescence has some properties and it can be distinguished in three different phases.

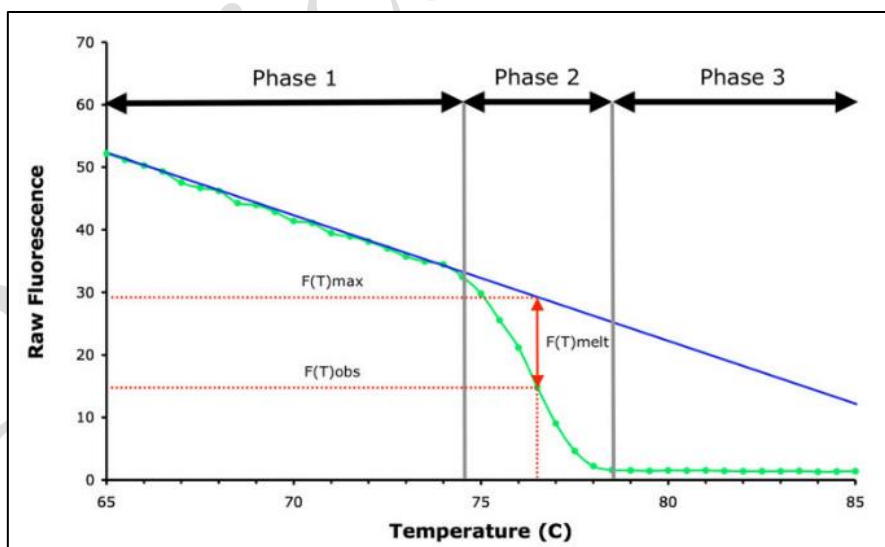


Figure 25: Properties of raw fluorescence signal

In the fig. 25, a standard fluorescence signal has been divided into three different phases, where each illustrate appropriate properties of raw fluorescence signal. According to the study, the first property of a fluorescence signal is that, it follows linearity in the beginning, that is, every raw fluorescence signal follows a linearity in its initial stage, where the decrease against the increasing temperature will be linear for specific time interval.

Usually, fluorescence-based PCR test uses fluorescent dyes, that bound to the double strand DNA. These fluorescent dyes follow a mechanism that, they do not fluoresce when they bound to double strand DNA. When the DNA sample is heated with a gradual increase in temperature, the double strand DNA will be separated into a single strand DNA. As a result, the fluorescent dye bound to the double strand DNA will fluoresce. Such state, is often called as melting state, where half of the DNA will be separated into single strand DNA. Until this process happens (melting), there will be linear decrease in fluorescence intensity, which cannot be avoided, as heating happens in a continuous increasing manner. Hence, there will be some amount of fluorescence release is observed, which is termed as Phase 1.

The Phase 2 describes the melting stage, where the fluorescence release is aggressive, as the DNA sample is being converted into single strand DNA. The magnitude of fluorescence will be high, that cause the sudden massive decrease in the fluorescence intensity. The Phase 3 is the final stage, where the fluorescent dye gets saturated and no longer fluoresce will happen as the dye gets exhausted.

With the above information, raw fluorescence signals can be processed in more significant manner along with the methods described by Smith *et al.*, to find the start and end of melting temperature from a fluorescence signal. There are several mathematical algorithms are followed and applied in this approach, to extract the earlier stated features like, temperature at which the melting started and the temperature at which the melting ends or saturated. Distinct mathematical and statistical algorithms like Linear regression and line fitting methods are also introduced and applied in this methodology.

The algorithm to apply line fitting method in the raw fluorescence signal is followed as,

**ALGORITHM 1: ALGORITHM TO APPLY LINE FITTING METHOD IN THE RAW FLUORESCENCE SIGNAL USING LINEAR REGRESSION**

---

*Input:* Data frame and the index of the required column  
*Output:* Straight line extrapolated and fitted on the features of the data

- 1 **Importing necessary libraries:** sklearn, numpy, matplotlib, pandas
- 2 **Initialization of objects:** create an empty list object 'List1'.
- 3 **FUNCTION predict** (data frame, index of column)
  - 4  $X \leftarrow$  create an array of shape (10,1), taking the first 10 elements of temperature column from the data frame.
  - 5  $Y \leftarrow$  create an array of shape (10,1), taking the first 10 elements of given column index from the data frame.
  - 6 Fit a linear equation on the arrays X, Y using sklearn.linear\_model.LinearRegression()
  - 7 **Prediction**  $\leftarrow$  extrapolate the temperature column in the data frame using the above fitted linear equation.
  - 8 **RETURN Prediction**
- 9 **END FUNCTION**
- 10 **FOR**  $j = 1$  TO len (data frame.columns) **DO**
- 11  $List1 \leftarrow$  Call the predict function for each column and append the results to the list
- 12 **END FOR**
- 13 **FOR**  $k = 1$  TO len (data frame.columns) **DO**
- 14 Plot the predictions against the actual values for each column using matplotlib.pyplot
- 15 Create a new plot for each column
- 16 Plot the actual values for the given column
- 17 Plot the predicted values for the given column
- 18 Show the plot
- 19 **END FOR**

The above algorithm illustrates how to extrapolate a straight line to the temperature values in the data set using Linear Regression. Following that, the algorithm starts by capturing the initial linear phase of the signal by choosing the first n (10 (a random choice, must be a least figure)) numbers.

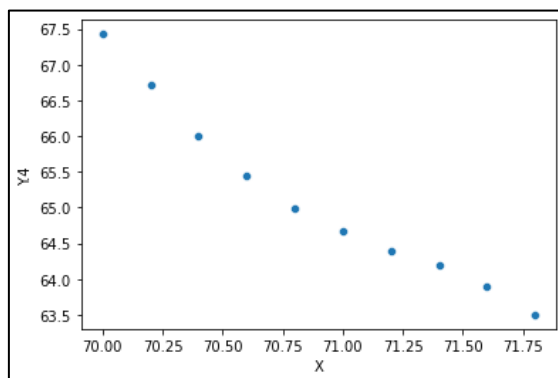


Figure 26: First 10 linear points of the fluorescence signal

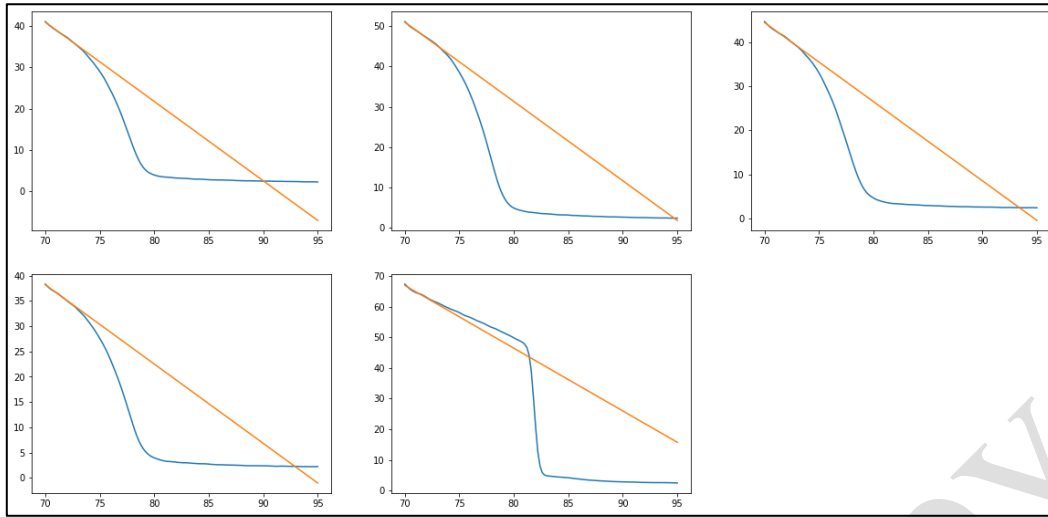


Figure 27: Fitting a straight line along the fluorescence signal's linear phase

The fitted straight line is an assumption that, the fluorescence intensity is decreased linearly throughout the temperature increase and it is denoted as “ $F(T)_{max}$ ”. This  $F(T)_{max}$  is the function of temperature, and the same has been described in the above algorithm as ‘FUNCTION predict’. The fluorescence signals are denoted as  $F(T)_{obs}$ , which is also treated as a function of temperature, but this denotation is not appropriate, as the fluorescence signals are not computed by a mathematical function, and it is wholly depends on various parameters, combining that, it is observed in real-time. For the discussion, it is denoted as such, and it is simply the observed fluorescence values.

To compute the melting phase of the raw fluorescence signal, the following formula can be applied,

$$F(T)_{melt} = F(T)_{max} - F(T)_{obs} \quad (2)$$

Again  $F(T)_{melt}$  is considered as a function of temperature, but here it is expressed as the difference between the assumed and observed values of fluorescence signals. Fitting an extrapolation line helps to differentiate fluorescence release caused by DNA melting, and the release that is caused by heating.



On clearly observing the values of fluorescence intensity, it is obvious that, the values are ranging from 0 to 100, which can also be furtherly processed with pre-processing techniques like normalization. Using appropriate mathematical functions, normalization can be achieved.

---

**ALGORITHM 2: ALGORITHM TO NORMALIZE THE RAW FLUORESCENCE SIGNAL**

---

**Input:** Data frame and the List1 from the Algorithm 1 (extrapolated values)  
**Output:** Normalized values of fluorescence signals

- 1 **Importing necessary libraries:** numpy, pandas
- 2 **Initialization of objects:** create an empty list object 'List2', 'List3'.
- 3 **FOR**  $i = 1$  TO  $len(data\ frame.columns)$  **DO**
- 4     **FOR**  $x, y$  IN  $zip(List1[i-1], data\ frame.iloc[:,i])$  **DO**
- 5         |  $List2 \leftarrow$  perform element wise division ( $y/x$ ) and append to the list
- 6     **END FOR**
- 7      $List3 \leftarrow$  append all the lists after computing the element wise division for all the columns.
- 8 **END FOR**

The algorithm takes data frame (fluorescence signals co-ordinates) and the extrapolated values computed in Algorithm 1 as inputs, to perform normalization. The mathematical notion implemented in the algorithm is the ratio of observed values to the extrapolated values.

$$Normalization = \frac{F(T)_{obs}}{F(T)_{max}} \quad (3)$$

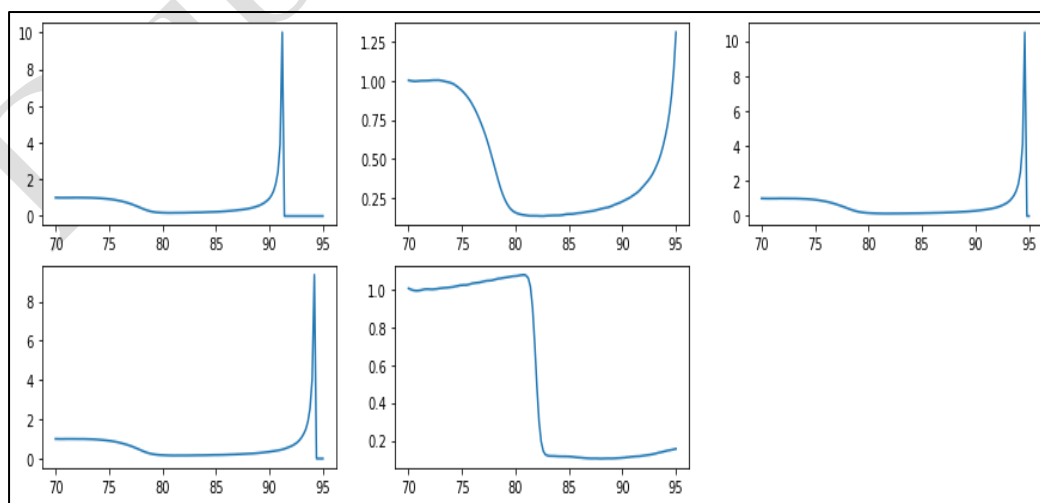


Figure 28: Fluorescence signals after performing normalization

The fig. 28, shows the normalized raw fluorescence signals, but the results are not as expected. On clearly observing the plots, there are some noises, being introduced in the end, i.e., between 85°C to 90°C temperature range. This is an error, and it is not a properly normalized fluorescence signal. In fact, the algorithm, on a specific stage has introduced an outlier value, which has to be backtracked to fix such erroneous.

On backtracking the error, it was observed that, the extrapolated values, being fitted along the linear phase of fluorescence signal, is intersecting the observed signal on a specific point (fig. 29).

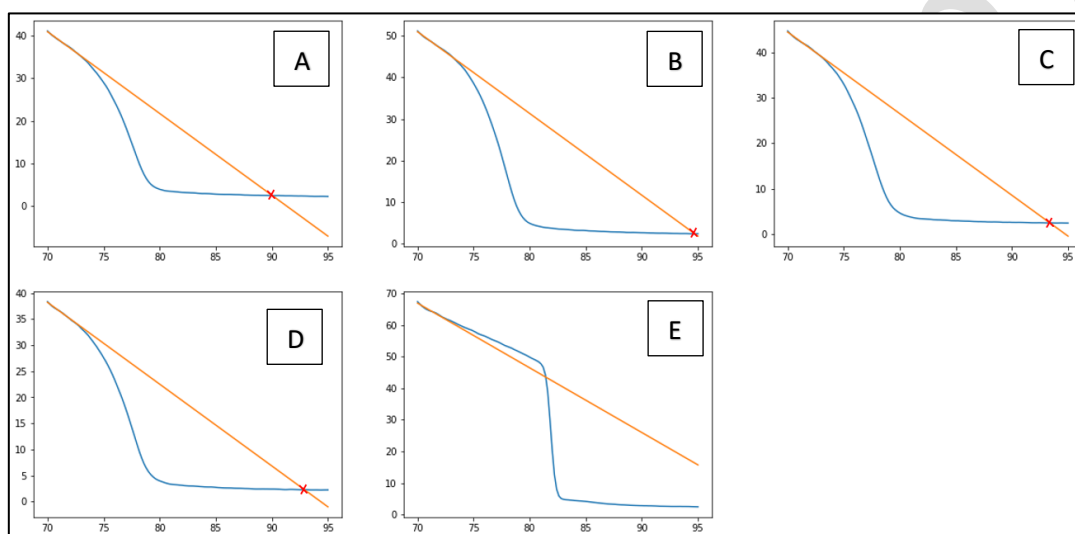


Figure 29: Erroneous intersection spotted in the signals

The extrapolated values fitted along the linear phase, should not intersect the observed values. This is because, normalization makes the fluorescence intensity as percentage or the ratio of observed values to the extrapolated values. On performing the element wise division, as described in Algorithm 2, observed and extrapolated value on the intersected point will be same, and the resultant ratio will be a whole number or a fraction that is near to a whole number. As a result, the intersection produces outlier and will be reflected in the resultant signal (Figure 3.12).

Moreover, on further analysis, it is also observed that, the linear phase in the observed fluorescence signal is narrow for fig 29 A, B, C, D as compared to fig 29 E and extrapolating a linear line on such phase will definitely intersect the fluorescence signal. This may be due to the reason that, the signals are actually missing the linear phase, as the has the temperature is starting from 70°C. Capturing the linear phase is crucial for this approach, and lacking on such

attribute, may lead to erroneous results. These issues can be fixed by acquiring data from earlier temperature values (say 50 °C), so that, linear phase of the fluorescence signals can be captured.

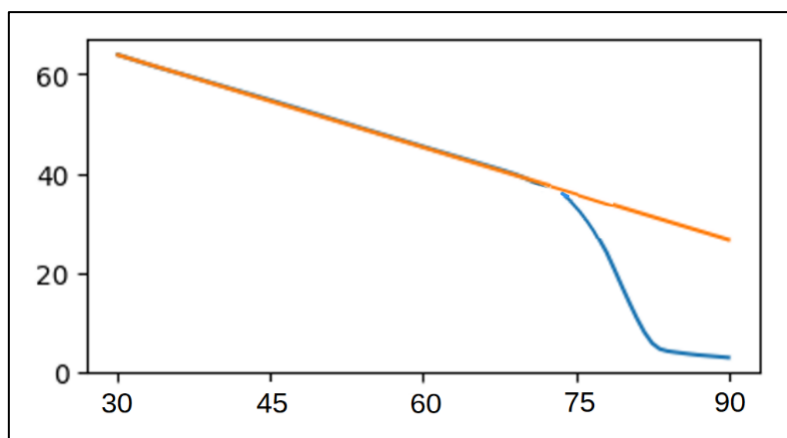


Figure 30: Raw fluorescence signal – Temperature range (30°C to 90°C)

Once the signals are captured with linear phase, Algorithm 2 can be applied hence, such that normalization can be performed properly.

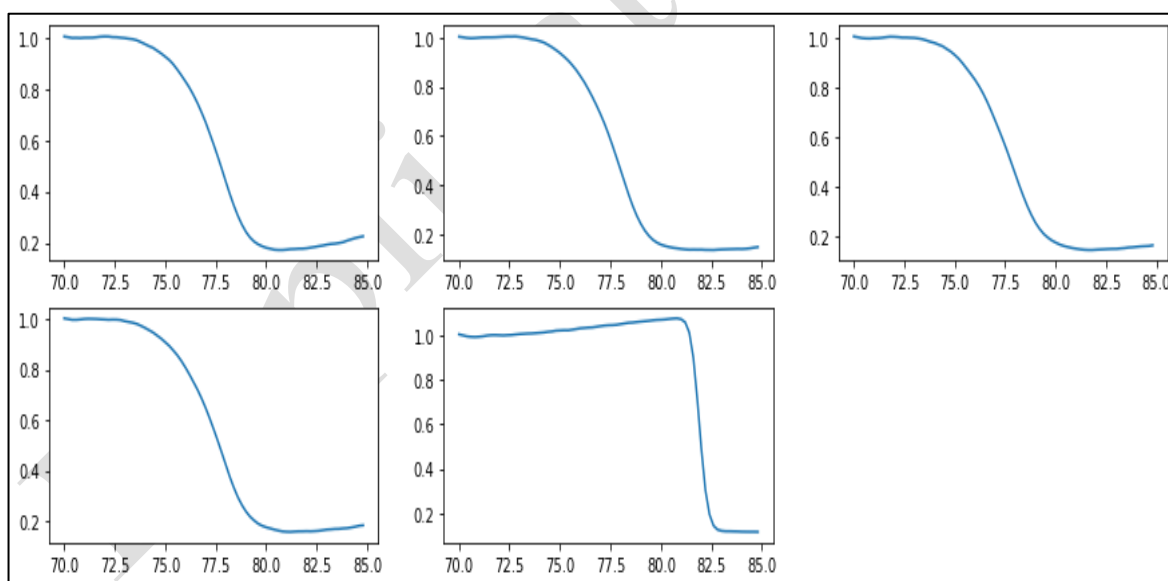


Figure 31: Normalized fluorescence signals

In (fig 31), windowing method has been applied, such that, only the required the range of temperature values and its corresponding fluorescence intensities will be taken. Following that, only the temperature range 70°C to 90°C has been fixed as a window size.

Raw fluorescence signals are taken into consideration, mainly to extract information on the take-off temperature value and the touch-down temperature value. This information is phenomenal, as it provides additional information like take-off and touch-off values, which can be considered as a valuable characteristic or a feature of the DNA melt signal. As far as, all the methods applied in this approach are stated and derived by E. Smith *et al.*, [1], as a part of fluorescence signal processing (FSP) to extract desired information for the analysis. The methods were described by E. Smith *et al.*, were performed on different context, with different samples, and following the same, has introduced several practical exceptions.

The next section of this approach will be an extended approach of E. Smith *et al.*, whom described the imaginary/extrapolated line fitting on the fluorescence signals. To capture such take-off and touch down temperature values, normalized fluorescence signals are used in the further experiments where the same line fitting method will be performed in a different manner. This time the existing Algorithm 1 can be used with some additional steps. In Algorithm 1 extrapolation has been done on the raw fluorescence signals, especially on the linear phase, by taking first n least observations. As a change, the same steps can be performed in a reciprocal way, to extrapolate the same straight line for the least n elements from the end of the signal (saturated part). This process as a result, will provide two lines extrapolated on both the ends (top and down) of the normalized fluorescence signals (fig 31).

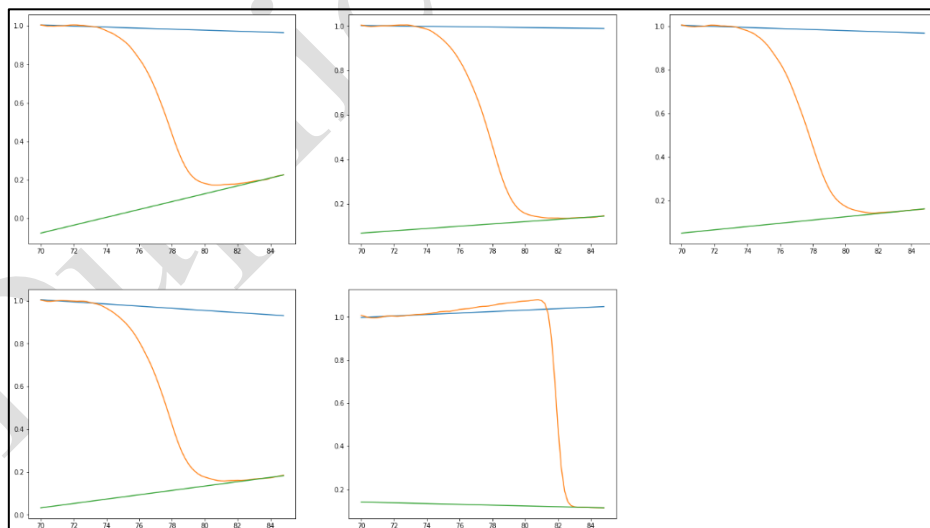


Figure 32: Extrapolating imaginary lines on both the ends

---

**ALGORITHM 3: ALGORITHM TO RECORD TAKE-OFF AND TOUCH-DOWN TEMPERATURE VALUES IN NORMALIZED RAW FLUORESCENCE SIGNAL**

---

*Input:* Normalized fluorescence values computed in Algorithm 2

*Output:* Take-off and Touch-down temperature values.

```
1 Importing necessary libraries: numpy, pandas
2 FUNCTION predict (normalized fluorescence data, index of column)
3     X1 ← create an array of shape (10,1), taking the first 10 elements of temperature values
        from the normalized fluorescence data
4     X2 ← create an array of shape (10,1), taking the last 10 elements of temperature values
        from the normalized fluorescence data
5     Y1 ← create an array of shape (10,1), taking the first 10 elements of given column index
        from the normalized fluorescence data.
6     Y2 ← create an array of shape (10,1), taking the last 10 elements of given column index
        from the normalized fluorescence data.
7     Fit a linear equation on the arrays X1, Y1 and X2, Y2 using
        sklearn.linear_model.LinearRegression()
8     Prediction1 ← extrapolate the temperature column in the data frame using the fitted
        linear equation for X1 and Y1 arrays.
9     Prediction2 ← extrapolate the temperature column in the data frame using the fitted
        linear equation for X2 and Y2 arrays.
10    RETURN Prediction1, Prediction2
11 END FUNCTION
13 FOR i = 1 TO len (Normalized fluorescence data frame.columns) DO
14     Extrapolated_values ← predict(Normalized fluorescence dataframe, i)
15     FOR x, y IN zip (Extrapolated_values [0][i-1], data frame.iloc[:,i]) DO
16         TakeOff ← compare the extrapolated values and observed fluorescence values to
            detect the point of deviation.
17     END FOR
18     FOR x, y IN zip (Extrapolated_values [1][i-1], data frame.iloc[:,i]) DO
19         TouchDown ← compare the extrapolated values and observed fluorescence
            values to detect the point of convergence.
20     END FOR
21 END FOR
```

The above algorithm describes a function to compute the extrapolated values for both the ends of a normalized fluorescence signal, and iterates through each column (signals) of normalized fluorescence data frame, such that, it could compare the actual signal and the extrapolated values, to detect the point of deviation and the point of convergence (fig 33). The resultant touch-down and take-off values are considered as the starting temperature point and the ending temperature point, and such points shall be cross validated with the corresponding DNA melting signal.

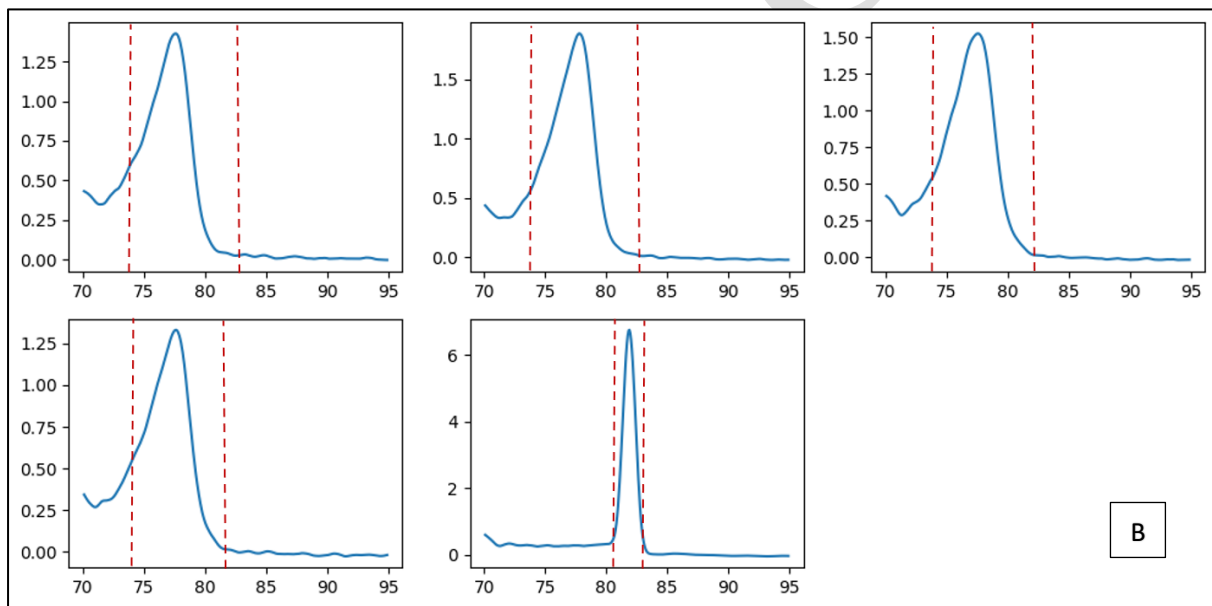
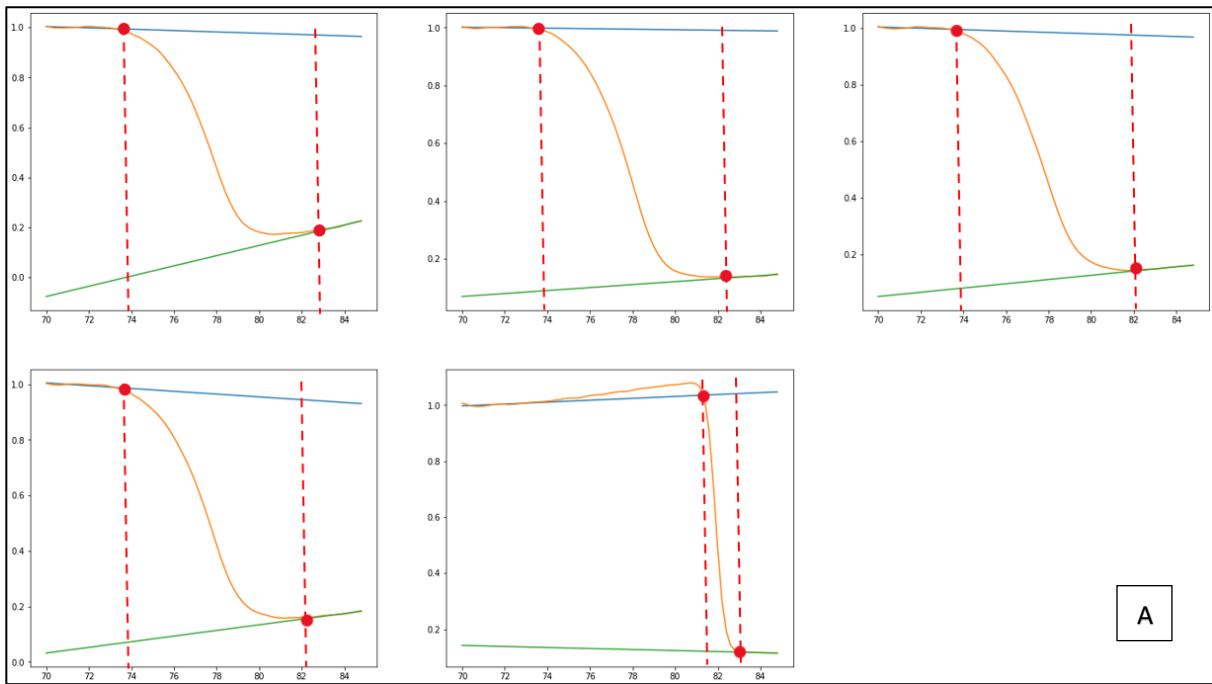


Figure 33: Mapping the take-off & touch-down points with Normalized fluorescence signal in DNA Melt signals.

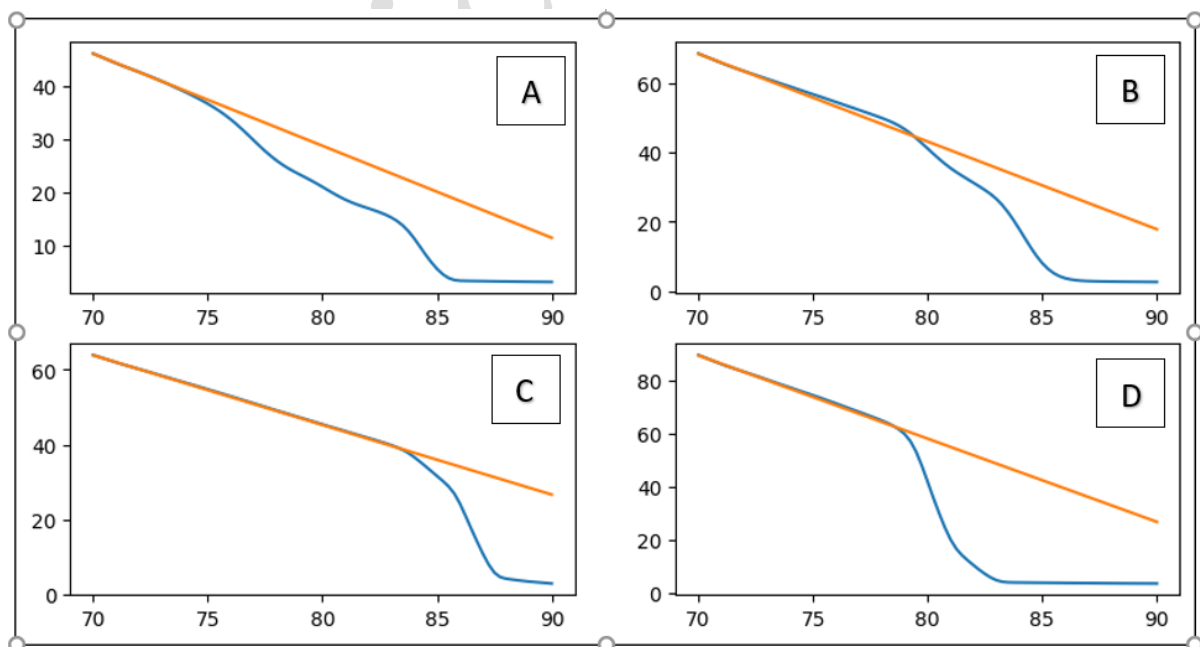
The results of the algorithm looks relevant, as the take-off and touch-down points are cross validated with the corresponding DNA melting signals. These points can be taken as features for analysis, and can be applied on successive samples, especially on the raw fluorescence signals. By the way, this algorithm works as a part of feature engineering process.

For calculating the melting temperature, same raw fluorescence signal co-ordinates can be processed further, to produce DNA melting signal. As stated earlier, DNA melting signals

are actually produced after applying a differential equation on the co-ordinates of raw fluorescence signals. The differential equation (1), captures the rate of change of fluorescence intensity for every change in temperature values. The negative derivative has been taken, concerning the order of observations (as the temperature values are in ascending order). As a result, the melting phase in the raw fluorescence will be reflected as a peak (fig 33 B), which is more appropriate for interpretation.

The exceptions of the algorithm is quite observable that, where the algorithm finds it difficult to capture further more additional informations. When it come to those signals, which has double peaks and double attributes (depends on primers) the algorithm could not pick the additional information on successive peak, which is prominent for several type of pathogen. Not all the pathogen would be expected with a single peaked DNA melting signals, where different pathogens markers will be captured with different set of primers designed dedicatedly. Such variations in primers could result in differernt DNA melting signals, and it is important that, capturing all the information is prominent.

Despite the algorithm works fine, there are also several scenarios where the algorithm fails to provide expected results. As the HRM data is prone to high variability, applying the same algorithm on different samples, does not work, and it follows certain exceptions.



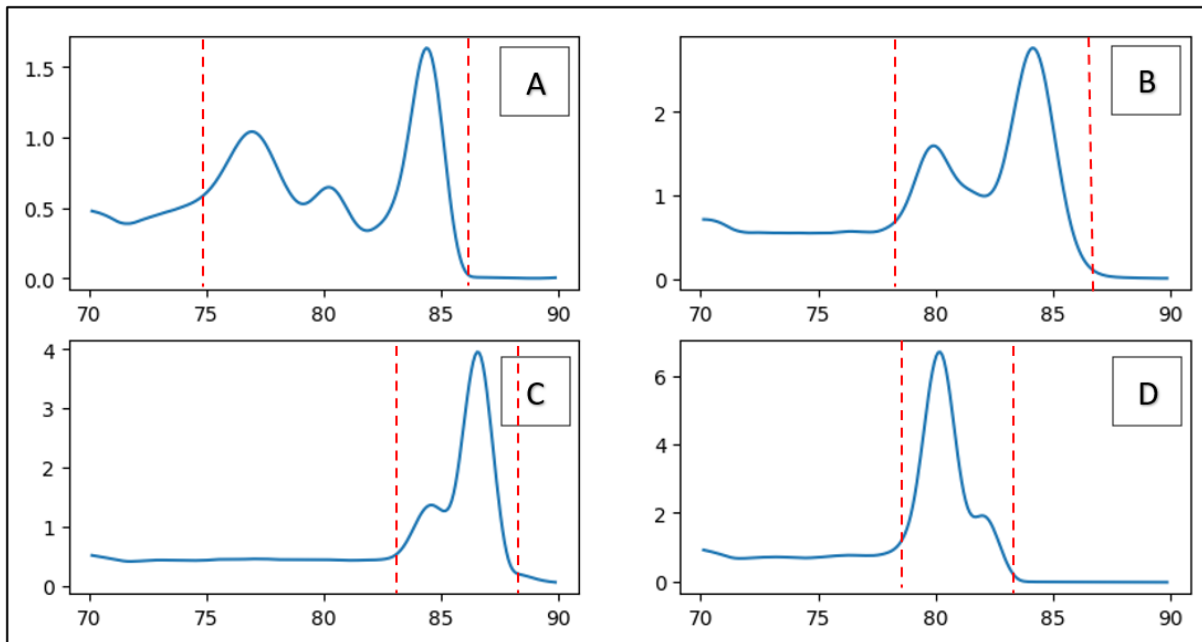


Figure 34: DNA melting signals with double peaks and its corresponding raw fluorescence.

With the algorithm using raw fluorescence curve, capturing the melting phase can be achieved in an overall sense, but could not provide any further detailing on additional peaks, which is being introduced along with the major peak. This will lead to, lack of information extraction, as the additional peaks also provides additional relevant information, which is a valuable feature, that has to be considered during interpretation. Since the algorithm could not do this, further implementation on capturing those additional information has to be performed.



## **RESULT AND DISCUSSION**

Data of Raw fluorescence signal has been acquired in this methodology as a primary data, and processed respectively using several line fitting techniques like linear regression and extrapolation. As a result, linear phase of fluorescence intensity has been captured and compared with actual observed signal, to differentiate melting phase, from the raw fluorescence signal. Furtherly data pre-processing techniques like normalization has also introduced to normalize the fluorescence intensity in a range of 0 to 1, and performed further line fitting methods on both the end of the resultant normalized curve to detect take off and touch down points.

## **CONCLUSION**

The experiment on raw fluorescence signals, has provided some significant results on single peaked DNA melting signal, but failed to work with double peaked signal, as it could not capture any additional information on additional peaks, which is being introduced along the previous peak. It is necessary to employ some alternate processing technique to capture all relevant information of peaks as much as possible. Efficient and gold standard pre-processing techniques like signal processing methods, has to be trialed, and corresponding results must be captured and evaluated, which is being covered in the upcoming chapters.

## CHAPTER 7

### APPROACH ON CO-ORDINATES OF DNA MELTING SIGNAL

In the previous chapter, processing raw fluorescence signals has several short comes, where it could only extract partial information from DNA melting signals. In this chapter, alternative methods to processing raw fluorescence signals will be introduced, and its corresponding results will be observed and evaluated duly. As stated earlier, DNA melting signals can be produced after applying a differential function over a raw fluorescence signal, such that peaks will be formed, denoting the melting phase. Such a way, raw fluorescence signals will be converted into melting signals and further processing steps will be covered in this chapter, to extract features from melting signals.

#### 7.1 MELT CONVERSION

Converting a raw fluorescence signal into a melting signal involves a gradient function, which takes the first negative derivative of fluorescence over the temperature values. The gradient function simply captures the rate of change of fluorescence intensity over each change happens in a temperature value. This as a result will produce signals with peaks, which is so called Melt curve or Melting signal.

---

#### ALGORITHM 4: ALGORITHM TO CONVERT RAW FLUORESCENCE SIGNAL TO MELTING SIGNAL

---

*Input:* Raw fluorescence signal values

*Output:* Melting signal values

- 1 **Importing necessary libraries:** *numpy, pandas*
- 2 **Initialization of objects:** *create an empty dataframe object to store melting signal values.*
- 3 **FOR** *i = 1 TO len (data frame.columns)* **DO**
- 4     **gradient** ← *Calculate the rate of change for fluorescence intensity and temperature values using np.gradient() and multiply with np.negative*
- 5     **Melting Signal** ← *append each signal columns of raw fluorescence to its corresponding columns of Melting signal data frame.*
- 6 **END FOR**

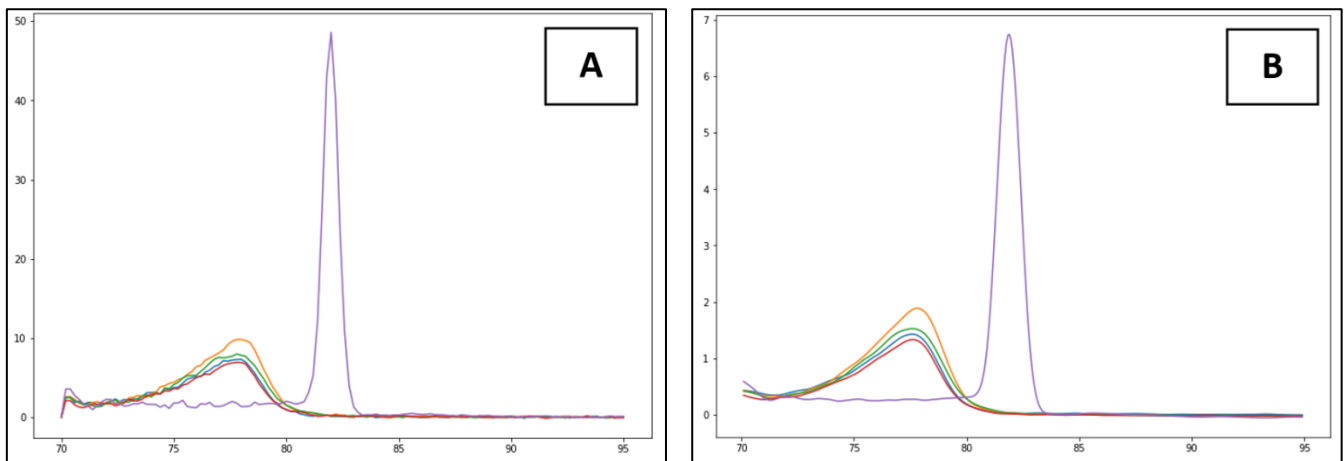


Figure 35: Comparison for manually converted melting signal(A) to machine converted melting signal(B).

Converting raw fluorescence signals into melting signals, provide similar results, as shown in the (fig 35). Though there are still some differences are observed, with respect to the smoothness of the signal. The machine converted melting signals (fig 35 B) are smooth and did not provide any noise at either the ends, whereas, manually converted signals (fig 35 A) introduce noises, which makes a slighter difference from the original machine converted signals. The machine may apply some noise reducing filters, that reduce the excess noise in the melting signals.

Harnessing raw fluorescence signals furtherly, throughout the approach, paves way for common and generalized solution, since raw fluorescence data can be extracted from any thermal cycler machines of any manufactures. Following the same will provide a universal solution, and hence can be applied anywhere, with any machines.

Commercially available thermal cycler machines, uses several signal smoothening algorithms like Savitzky-Golay algorithm, moving average algorithm, and ensemble average etc., to smooth the DNA melting signals. Despite signal smoothening, the machine also performs data interpolation and some of the familiar algorithms are spline, radial basis function, nearest interpolation etc.

Interpolation and smoothing are two related concepts in signal processing that are used to enhance or modify signals. Interpolation refers to the process of estimating values of a signal at points where it has not been sampled, based on the values of the signal at nearby sampled points. This is typically done using mathematical algorithms, such as *linear interpolation* or

*spline interpolation*. The purpose of interpolation is to increase the sampling rate of a signal, or to fill in missing or corrupted data points.

Smoothing, on the other hand, refers to the process of reducing the noise or irregularities in a signal by averaging adjacent data points or using a filtering algorithm. Smoothing can be used to remove high-frequency noise or artifacts in a signal, or to remove short-term fluctuations that are not relevant to the underlying trend of the signal.

Interpolation and smoothing are related in that they both involve modifying a signal to improve its quality or make it more useful for analysis or processing. In some cases, smoothing may be applied before interpolation to remove noise or irregularities in the signal, while in other cases interpolation may be applied before smoothing to fill in missing data points. Both techniques are widely used in signal processing and are essential for many applications, such as image processing, audio processing, and data analysis.

Interpolating method depends on the type of data being processed and the specific requirements of the application. Some common interpolating methods used in signal processing include:

1. **Linear interpolation:** This method involves connecting two adjacent data points with a straight line and estimating the value of the signal at a new point based on the position of that point on the line.
2. **Cubic spline interpolation:** This method involves fitting a cubic polynomial curve to the data points in a local region, which provides a smoother estimate of the signal at the new point.
3. **Fourier interpolation:** This method involves using the Fourier transform to estimate the signal at a new point based on its frequency content and the values of the signal at neighbouring points.
4. **Akima interpolation:** This method uses a modified cubic spline method that is specifically designed to handle noisy or irregular data.
5. **B-spline interpolation:** This method involves fitting a spline curve to the data points using a set of basis functions known as B-splines.

## **7.2 SPLINE AND SAVGOL FILTER**

Spline and Savitzky-Golay (Savitzky-Golay) filter are two popular methods used for smoothing data.

Spline is a mathematical technique that is used to interpolate or smooth data points by fitting a piecewise polynomial curve through the data points. The curve is designed to have a smooth appearance by minimizing the curvature of the function. The resulting curve can be used to interpolate between data points or to smooth out noise or irregularities in the data.

On the other hand, the Savitzky-Golay filter is a technique used for digital signal processing. The filter is designed to smooth out noisy data by fitting a series of polynomial functions to the data points. The polynomial functions are then used to calculate the smoothed values of the data points. The filter is designed to minimize the impact of the noise on the resulting smoothed values by fitting a polynomial curve of a certain order to the data.

The key difference between the two methods is that spline is designed to interpolate between data points and smooth out noise, whereas the Savitzky-Golay filter is designed to smooth out noise while preserving the shape of the data. In other words, spline can introduce new data points that were not present in the original data, whereas SavGol filter only uses the existing data points. In general, if the goal is to preserve the shape of the data, SavGol filter may be a better choice. On the other hand, if the goal is to interpolate between data points and smooth out noise, spline may be a better choice.

### **7.3 B SPLINE**

B-spline representation of a 1-D curve is a mathematical technique that approximates a smooth curve using a set of control points and a set of piecewise polynomial functions known as basis functions. The degree of the basis functions determines the degree of the B-spline curve, and the knot vector determines the locations of the knots that connect the basis functions.

To evaluate a point on the B-spline curve, the basis functions are computed at that point and weighted sums of the control points are computed using those basis functions. B-splines have the advantage of being able to approximate complex shapes using a small number of control points, and they can be used to interpolate or approximate a given set of data points.

B-splines are widely used in computer graphics, computer-aided design (CAD), and other fields that require precise and efficient curve and surface modeling. They offer several advantages over other curve representations, including their flexibility, efficiency, and ability to approximate complex shapes with a relatively small number of control points. To smoothen the converted melting signal from the raw fluorescence signal, spline functions has been used.

**ALGORITHM 5: ALGORITHM TO SMOOTH THE CONVERTED MELTING SIGNAL FROM RAW FLUORESCENCE SIGNAL**

---

**Input:** Converted Melting signal values

**Output:** Smoothened Melting signal values

- 1 **Importing necessary libraries:** *numpy, pandas, scipy.interpolate*
- 2 **Initialization of objects:** *create an empty dataframe object to store smoothened melting signal values.*
- Interpolated temperature**  $\leftarrow$  *Interpolate the temperature values using `np.linspace` with a constant multiplying to the actual number of the temperatures values.*
- 3 **FOR**  $i = 1$  TO  $\text{len}(\text{data frame.columns})$  **DO**
- 4     **Interpolated Signals**  $\leftarrow$  *Interpolate the signal values with the corresponding interpolated temperature values using `scipy.interpolate.splrep()` with a 's' value.*
- 5     **Smoothened Melting Signal**  $\leftarrow$  *append each signal columns of raw fluorescence to its corresponding columns of Melting signal data frame.*
- 6 **END FOR**

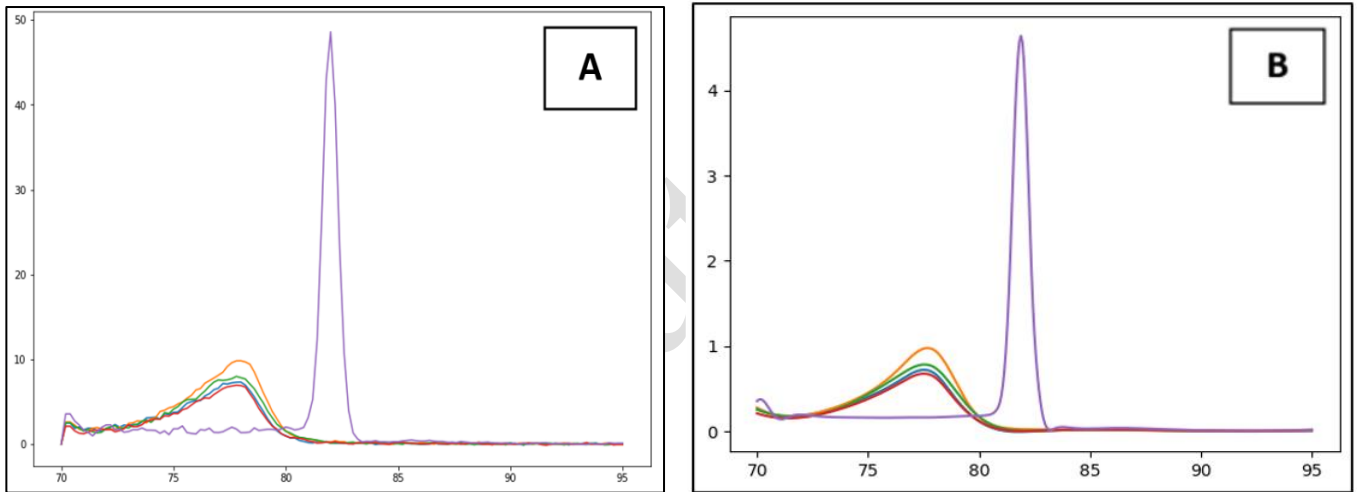


Figure 36: Before (A) and After (B) applying smoothing filter.

After applying smoothing filter in the signal, it looks fine and there are no noises observed as compared to the previous one (A). The spline function smoothens the signal with the use of 'S' parameter, which is a smoothing condition. The amount of smoothness is determined by satisfying the conditions:

$$\sum (w * (y - g)) ** 2 \leq s$$

where  $g(x)$  is the smoothed interpolation of  $(x,y)$ . It can be controlled to make a trade-off between closeness and smoothness of fit. Larger  $s$  means more smoothing while smaller values of  $s$  indicate less smoothing. Recommended values of  $s$  depend on the weights,  $w$ . If the weights

represent the inverse of the standard-deviation of  $y$ , then a good  $s$  value should be found in the range

$$(m - \sqrt{(2 * m)}, m + \sqrt{(2 * m)})$$

where  $m$  is the number of datapoints in  $x$ ,  $y$ , and  $w$ .

Default:

$$s = m - \sqrt{(2 * m)}$$

if weights are supplied.  $s = 0.0$  (interpolating) if no weights are supplied.

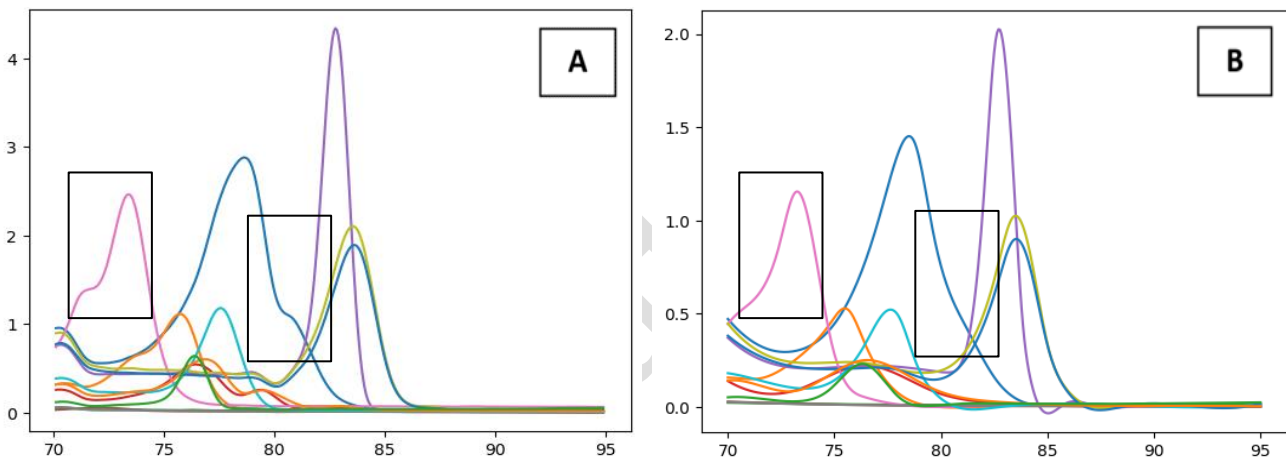


Figure 37: Machine converted melting signal (A) and manually converted melting signal (B) by applying smoothing filter.

Selecting an appropriate value for 'S' looks prominent, and the impact of such parameter is variable as data changes. Using a constant 'S' value to the spline function will not give a generalized solution, and smoothness applied to the signal must be proportional and dynamic. This also becomes a hindrance for feature extraction, as over smoothness cause, over smoothing the important curves and details, which as a result brings lack of performance. Finding a suitable 'S' parameter for each set of data is challenging, and making a dynamic 'S' value requires more attention. Despite proper hyper parameter has chosen, performing a manual conversion involves certain risk of losing the authenticity of data, since the machine-

made DNA signals will undergo certain stages, which follows some technical process, that could only handle efficiently by dedicated PCR machines.

A PCR machine converts raw fluorescence to melt data by involving the following steps:

1. **Raw fluorescence data acquisition:** The instrument measures the fluorescence signal during the PCR reaction and generates a raw fluorescence data file.
2. **Background correction:** The machine subtracts the baseline fluorescence level from the raw fluorescence data to correct for any background signal.
3. **Normalization:** The fluorescence signal is normalized to a reference dye signal to account for any variations in fluorescence intensity caused by differences in sample concentration or dye binding.
4. **Melting curve generation:** The normalized fluorescence data is plotted against the temperature of the reaction to generate a melting curve. The software uses an algorithm to smooth the curve and calculate the first derivative of the curve.
5. **Baseline subtraction:** The instrument calculates the baseline of the melting curve by fitting a straight line to the lowest fluorescence signal values.
6. **T<sub>m</sub> determination:** The machine identifies the temperature at which 50% of the double-stranded DNA has dissociated by calculating the maximum of the first derivative of the melting curve.
7. **Melt curve analysis:** The machine can also perform a derivative analysis on the melting curve to identify the number of melting domains or different DNA sequences present in the sample. This information can be used to identify the presence of mutations or genetic variations in the sample.
8. **Data output:** The machine finally, outputs the melt curve data in various formats such as a graph, a table of T<sub>m</sub> values, and a report containing detailed analysis results.

#### 7.4 BASELINE SUBTRACTION

Baseline subtraction is a critical step in converting raw fluorescence data to melt data. The purpose of baseline subtraction is to correct for any background signal and to ensure that the fluorescence signal represents the DNA melting signal.

Performing baseline subtraction can be achieved using the following method:



1. **Baseline region selection:** The PCR machine selects a region of the melting curve that represents the baseline fluorescence level. This region is usually chosen as the region before the start of the DNA melting transition, where the fluorescence signal is relatively stable.
2. **Straight line fitting:** Then it fits a straight line to the baseline region of the melting curve using linear regression analysis.
3. **Baseline subtraction:** The machine subtracts the values of the fitted straight line from the entire melting curve to obtain the baseline-subtracted curve. By fitting a straight line to the baseline region of the melting curve and subtracting it from the entire curve, the baseline-subtracted curve is obtained. This ensures that the fluorescence signal represents only the DNA melting signal and not any background signal. The baseline subtraction method eliminates any background signal that may interfere with the accurate detection of DNA melting. The result is a more accurate and reliable measurement of the melting temperature and any genetic variations in the sample.

## 7.5 BACKGROUND CORRECTION

1. **Baseline fluorescence measurement:** The instrument measures the fluorescence signal from a region of the reaction that does not contain any DNA, such as the negative control or a blank sample.
2. **Baseline fluorescence subtraction:** Then it subtracts the baseline fluorescence level from the raw fluorescence data for each sample to correct for any background signal.
3. **Correction factor calculation:** Calculates a correction factor for each sample by dividing the fluorescence signal of the sample by the fluorescence signal of the reference dye. This correction factor accounts for any variations in fluorescence intensity caused by differences in sample concentration or dye binding.
  - Reference dye measurement: The instrument measures the fluorescence signal of a reference dye that is added to each reaction. The reference dye has a constant fluorescence intensity and serves as a standard for the fluorescence measurement.
  - Raw fluorescence correction: The machine divides the raw fluorescence signal of each sample by the raw fluorescence signal of the reference dye to correct for any variations in fluorescence intensity caused by differences in sample concentration or dye binding.

4. **Normalization:** Then it normalizes the fluorescence data for each sample by dividing the corrected fluorescence signal by the average of the correction factors for all samples.

However, there are some limitations to performing these steps manually. For example, the real-time PCR instrument used for fluorescence measurement would need to be capable of precise temperature control and fluorescence detection, and the analysis would require specialized software for data processing and analysis. Additionally, manual analysis may be more time-consuming and prone to errors than using dedicated software. While it is possible to perform some of the steps involved in converting raw fluorescence data to melt data manually, it would require specialized equipment and expertise and may not be as efficient or accurate as using PCR machines.

Concerning the study and technical aspects behind converting raw fluorescence signals into melt signals, the process of manual conversion can be put in hold, and can be kept as a future research work of this project where those technical steps mentioned above must be implemented further with domain experts. To sum up everything, acquiring the machine converted DNA melting signal data is feasible as far as concerning the scope of the project and the relevant feature extraction process can be applied on such data, effectively.

## 7.6 SIGNAL PROCESSING ON DNA MELTING SIGNAL

To process the DNA melting signal, several techniques from signal processing can be used. For example, noise reduction techniques, such as filtering and averaging, can be used to remove random fluctuations in the signal and improve its accuracy. Signal processing techniques can also be used to extract features from the melting curve, such as the melting temperature, the shape of the curve, and the width of the transition region. These features can then be used to compare different DNA samples or to detect mutations and structural variations in the DNA molecule. Another important aspect of DNA melting signal processing is the use of mathematical models to describe the melting process. Signal processing techniques play a critical role in analysing and interpreting DNA melting signals, and can provide valuable insights into the properties of the DNA molecule and its interactions with other molecules.

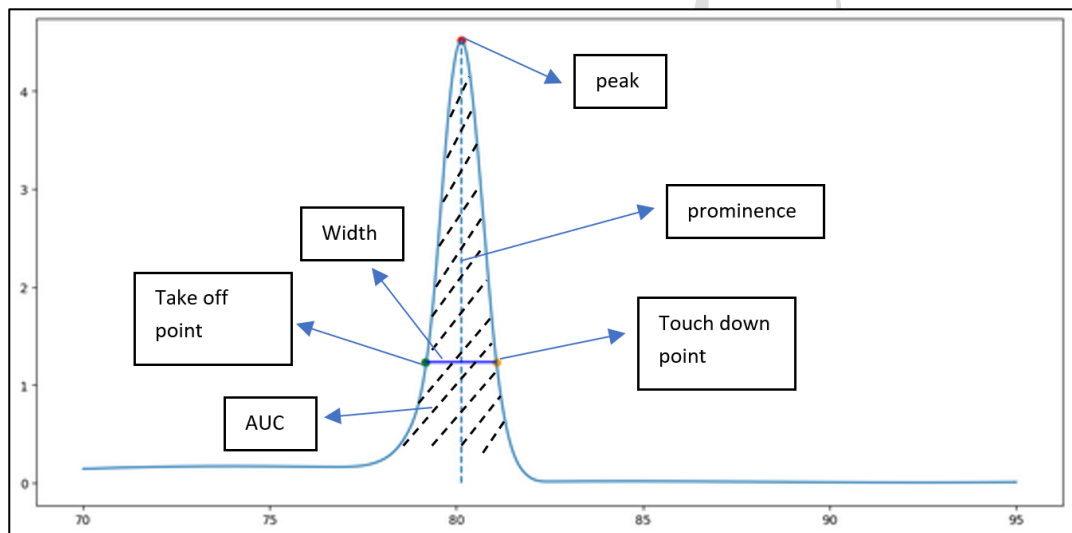


Figure 38: Features can be extracted from a melt curve using signal processing

### 7.6.1 PEAK FINDING

Signal peak finding is a common problem in signal processing, where the goal is to identify and extract significant features, such as peaks or valleys, from a given signal. A peak is a local maximum or high point in the signal, while a valley is a local minimum or low point in the signal. Peak finding algorithms are used in a variety of fields, including image processing, speech recognition, and data analysis. In many cases, peak finding is a critical step in the data analysis pipeline as it can reveal important information about the underlying process

generating the signal. There are many different methods for peak finding, ranging from simple threshold-based approaches to more sophisticated techniques like wavelet transform and machine learning-based approaches. These methods differ in terms of their accuracy, computational complexity, and robustness to noise and other types of signal artifacts. One commonly used approach for peak finding is the so-called "local maxima" method, where the signal is scanned for local maxima using a sliding window. Another popular approach is the "derivative-based" method, where the derivative of the signal is computed and peaks are identified as the points where the derivative changes sign.

The peak finding algorithm function takes a 1-D array and finds all local maxima by simple comparison of neighbouring values.

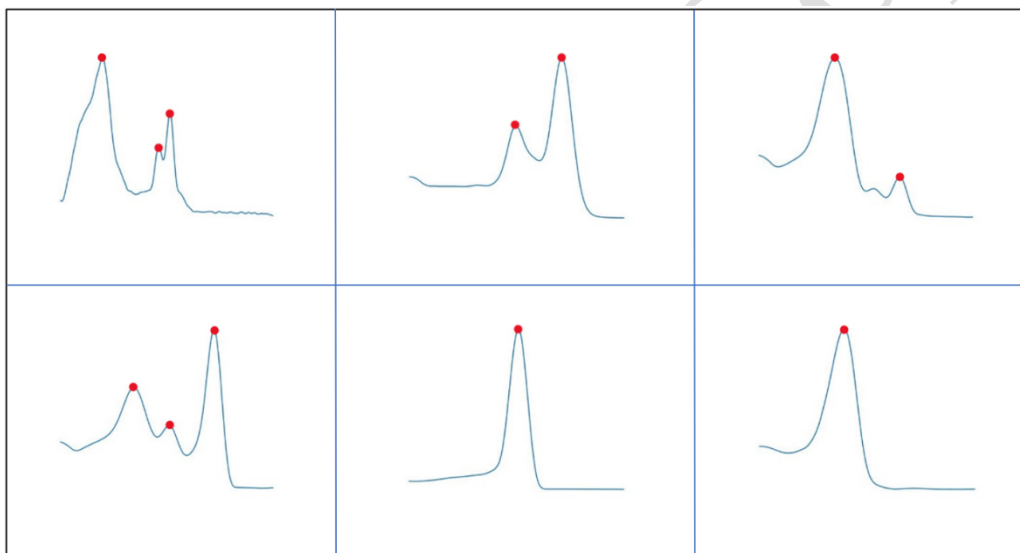


Figure 39: Detecting peaks in the DNA melting signal

### 7.6.2 PEAK PROMINENCE

Signal peak prominence is a measure of the relative height of a peak in a signal compared to the surrounding peaks. It is a useful feature for signal processing and analysis because it can provide information about the importance or significance of a peak in the overall signal. The prominence of a peak is defined as the vertical distance between the peak and the lowest point on the curve that connects all neighbouring peaks that are higher. In other words, it measures the minimum height required to descend from the peak to a higher neighbouring

peak or the baseline of the signal. Signal peak prominence is commonly used in peak detection algorithms to filter out noise and identify only the most significant peaks in a signal. Peaks with high prominence are typically more important and relevant to the underlying process generating the signal than those with low prominence.

Prominence can also be used to compare and quantify the differences between peaks in different signals or datasets. For example, it can be used to compare the amplitude of peaks in EEG signals recorded from different brain regions or to compare the intensity of peaks in spectra obtained from different chemical compounds.

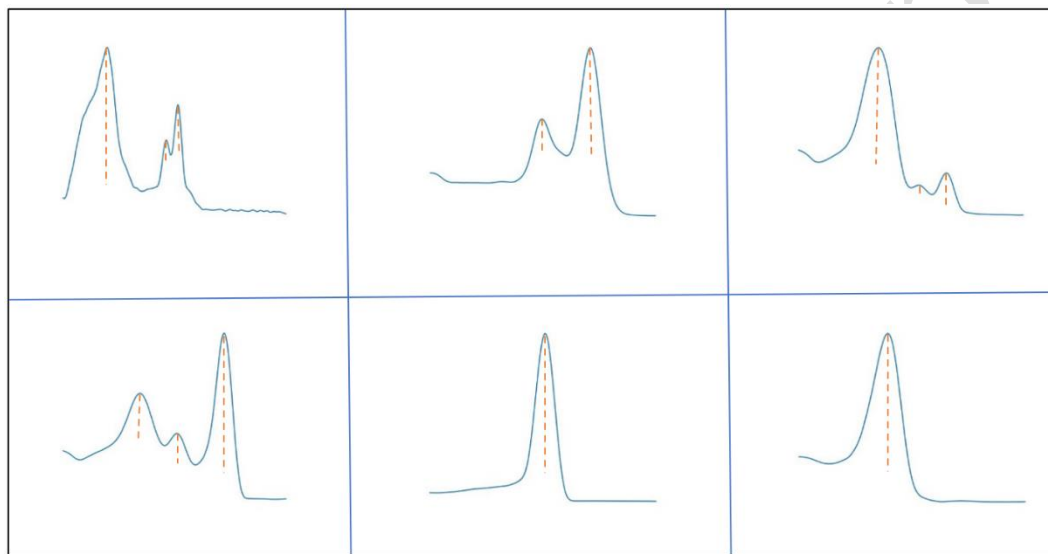


Figure 40: Calculating the peak prominence of the DNA melting signal

### 7.6.3 PEAK WIDTH

Signal peak width is a measure of the extent of a peak in a signal along the x-axis (usually time or frequency). It is an important feature in signal processing and analysis because it can provide information about the duration or spread of a particular phenomenon represented by the peak. Peak width is commonly used in peak detection algorithms to distinguish between closely spaced or overlapping peaks. In such cases, the width of a peak can help to differentiate between distinct peaks and noise or artifacts in the signal. Peak width can be quantified in a number of ways, depending on the nature of the signal and the application. One common measure is the full width at half maximum (FWHM), which is the width of the peak at half its

maximum amplitude. Another common measure is the peak width at the baseline, which is the width of the peak at a certain fraction of its maximum amplitude.

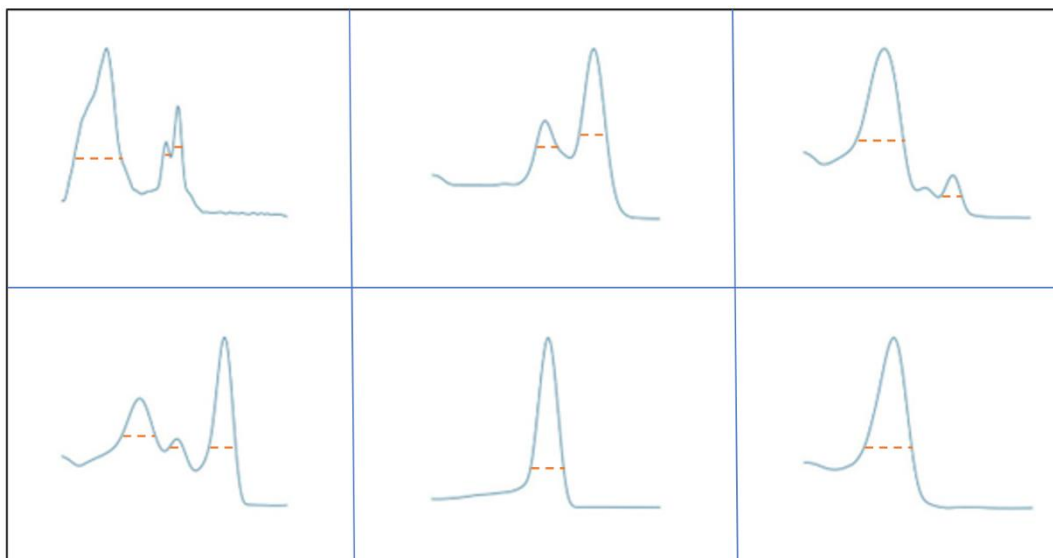


Figure 41: Calculating the peak width of the DNA melting signal

Using the width of the peak, the take off and touch down points can be calculated using the start and end points of the width. Moreover, the width of the peak is being calculated using the relative height of the peak which has been set to 75%.

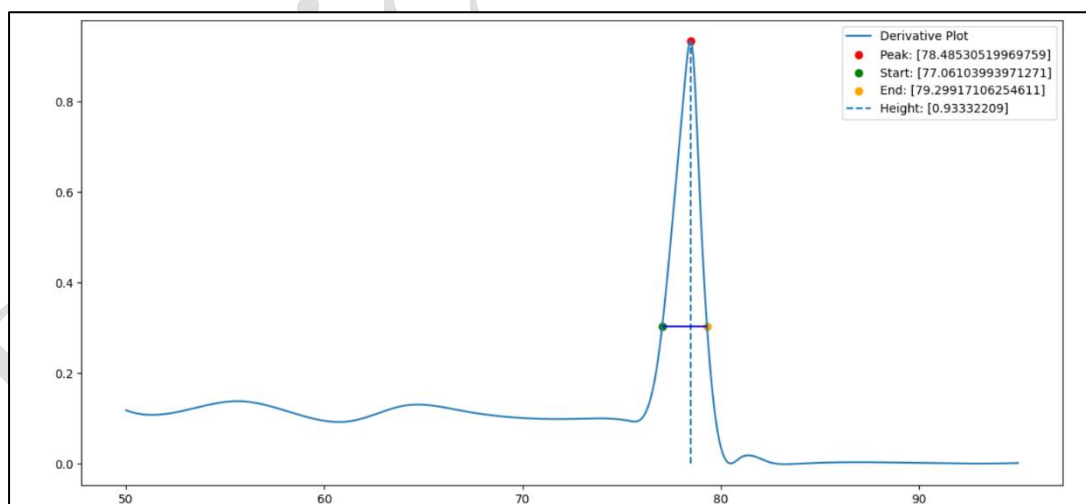


Figure 42: Detecting all the features from a melting signal.

#### 7.6.4 AREA UNDER THE CURVE (AUC)

The area under the curve of a signal is a fundamental concept in signal processing and analysis. It represents the total energy or power of the signal over a given time period. This concept is widely used in many fields, including physics, engineering, biology, and finance. The calculation of the area under the curve can provide important information about the signal, such as its mean, variance, and distribution. It is also a useful tool for detecting anomalies and patterns in the signal, which can be used for various applications such as signal classification, signal denoising, and signal compression. There are various methods and techniques used for its computation, including numerical integration, trapezoidal rule, Simpson's rule, and Monte Carlo integration.

##### Simpson's Rule

Simpson's rule is a numerical integration technique used to approximate the area under a curve. It is based on the idea of approximating the curve with a parabolic function and then computing the area of the resulting parabola.

$$\int [a, b] f(x)dx \approx (b - a)/6 * [f(a) + \frac{4f(a + b)}{2} + f(b)]$$

where  $f(x)$  is the function to be integrated,  $[a,b]$  is the interval over which the integration is to be performed, and  $(a+b)/2$  is the midpoint of the interval. The formula can be understood as follows: the area under the curve is approximated by dividing the interval  $[a,b]$  into subintervals of equal width, and then approximating the curve over each subinterval with a parabolic function. The area under each parabola is then computed and added together to obtain an approximation of the total area under the curve.

Simpson's rule is known to be more accurate than other numerical integration techniques, such as the trapezoidal rule, for functions that are smooth and have a relatively simple shape. However, it may not be as effective for functions that have sharp changes or irregularities in their shape.

As far as, the peak detection and feature extraction using signal processing methods, provided significant results, and it detects all possible features of a DNA melting signal

including peak prominence, width, and area under the curve. However, it is also notable that, the peak detection process is performed without any thresholding, i.e., whatever the peaks introduced in a signal, are being captured, which in turn provides all the peaks, disregarding the desired peak necessitate for analysis. This is the scenario, where thresholding plays a crucial role in removing unwanted signals.

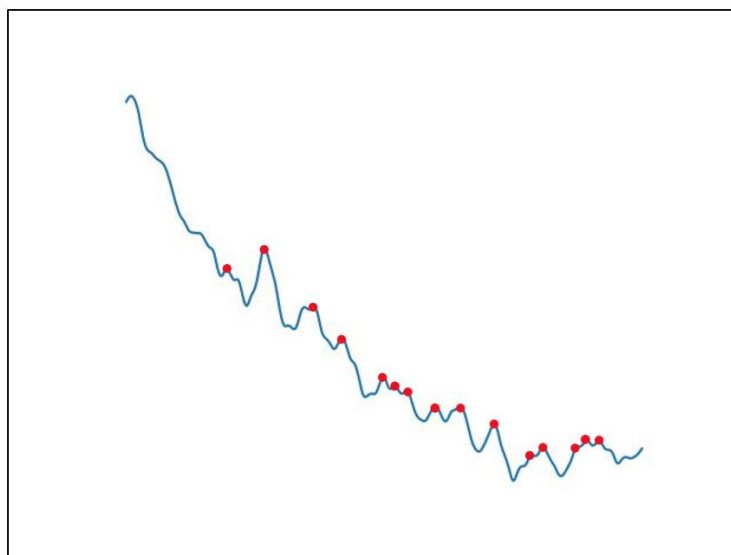


Figure 43: Negative(noise) signal with peaks detected

The negative signal as shown in the fig 43 should not be taken into consideration, as there is no any DNA melting is observed, but performing signal processing methods, has captured the minute noises as peaks, which is inappropriate. Proper thresholding techniques must be applied on such signals, such that noisy and unwanted peaks can be removed earlier, before bringing them into analysis.

Thresholding has been performed in the prevailing context, with a visual inspection, which involve manual configurations on the prominences of DNA melting signals. A decent prominence level will be fixed, so that, peaks above such points are only considered, where remaining will be left as noisy/ unwanted signals.

Since it is an AI based framed, which should perform tasks on its own, without any human intervenes, proper logic must be applied to set self-thresholding levels, so that, only necessary peaks will be taken for analysis.



## 7.7 THRESHOLDING LOGIC

As stated earlier, logic for thresholding must be very appropriate and should be in a way to detect only genuine signals. In most of the melting signals, peaks with sound prominences will be chosen for analysis. In fact, even small peaks with smaller prominences also taken into consideration at some times, concerning all other parameters like melting temperature etc.,

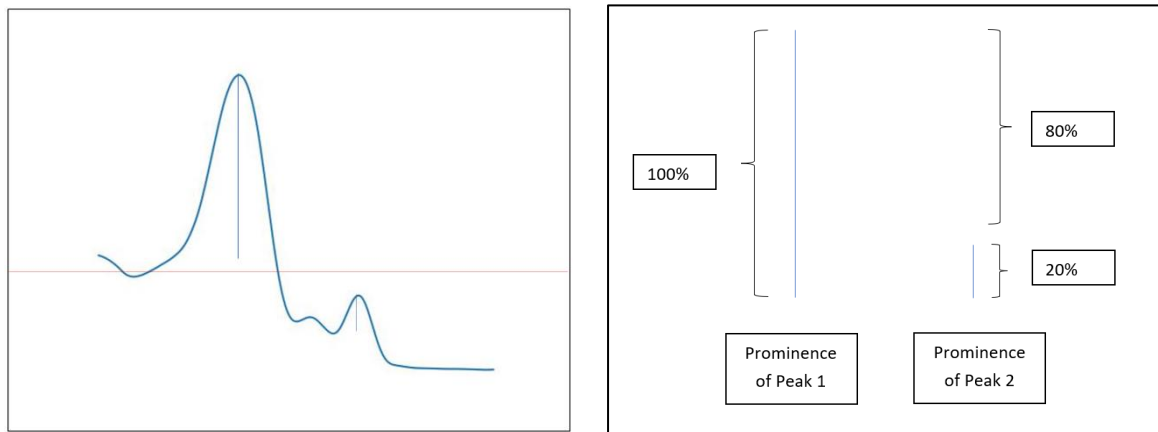


Figure 44: Measuring the prominences

Threshold is being set (red coloured line) for the peaks shown in the Figure 3.23, which rejects the second peak over the first peak, that has higher prominence as compared to the second one. On the other hand, while measuring the prominences of the peaks, it is observed that, the peaks introduced later are with the ~20% prominence of the first peak. Hence, the logic has to applied in a manner to detect the highest peak in the signal, and to compare the prominences of its other neighbouring peaks, such that, the peaks which has more than ~20% prominence of the highest peak (first peak) will be considered.

## **RESULT AND DISCUSSION**

In this approach, raw fluorescence signals were initially converted into a melting signal using a gradient function and applied interpolation algorithm like Spline, to reduce the noise introduced in the signals. As a result, signals are smoothed, but does not provide authenticity, since the parameter of the spline function is highly variable. Therefore, a machine processed DNA melting signals are adopted and processed further with signal processing methods like peak finding, peak prominences, peak width and area under the curve, to extract necessary information of such signals. Later, Thresholding logic has been approached to keep only genuine and desired peak, but on applying such logic, does not provide significant result, as the ~20% value is not significant, and the value varies data to data.

## **CONCLUSION**

Apart from feature extraction, developing a sensible logic on thresholding is mandatory. Applying a mathematical equation for thresholding provides significant result for certain data, but has rate of variability, and the result which comes out of such equation becomes insignificant at some scenarios. Proper, alternate solution must be employed, so that, unwanted peaks and noisy signals are removed from the analysis.

## CHAPTER 8

# COMBINATION OF APPROACH ON IMAGES AND THE COORDINATES OF DNA MELTING SIGNAL

In the previous section, thresholding becomes a challenging process, where applying logic, based on peak prominences is not significant, and highly variable. To tackle such hindrance, an image-based approach is being performed in this chapter, to introduce an image-based thresholding, as like how conventional thresholding is being done through visual inspection. This can be achieved through the traditional computer vision algorithm like Convolution Neural Network, that can be used to classify genuine and non-genuine peaks, by training them on multiple images of DNA melting signals, generated randomly from a pool of samples.

### 8.1 CONVOLUTION NEURAL NETWORK

A Convolutional Neural Network (CNN) is a type of neural network that is commonly used in image and video recognition tasks. It is a deep learning algorithm that learns to automatically extract features from input images, through a process called convolution. The key feature of a CNN is its ability to learn and identify spatial patterns in an image. This is done through the use of convolutional layers, which apply filters to an input image to extract specific features. These features are then passed on to other layers in the network, such as pooling layers and fully connected layers, to further refine the information and classify the input image.

CNNs have achieved state-of-the-art performance in various computer vision tasks such as object detection, face recognition, and image segmentation. They have also been used in other domains, such as natural language processing and speech recognition.

Classifying genuine and non-genuine DNA melting signals is an important task in DNA analysis and sequencing. Convolutional Neural Networks (CNNs) have been used to address this problem by learning to automatically extract features from the melting signals and classify them as genuine or non-genuine.

During training, the CNN is fed with a large dataset of labelled DNA melting signals, consisting of both genuine and non-genuine examples. The network learns to differentiate between these two classes by adjusting the weights of its filters through a process called backpropagation. Once trained, the CNN can be used to classify new DNA melting signals as genuine or non-genuine. By the way using CNNs to classify genuine and non-genuine DNA melting signals will be a promising approach.

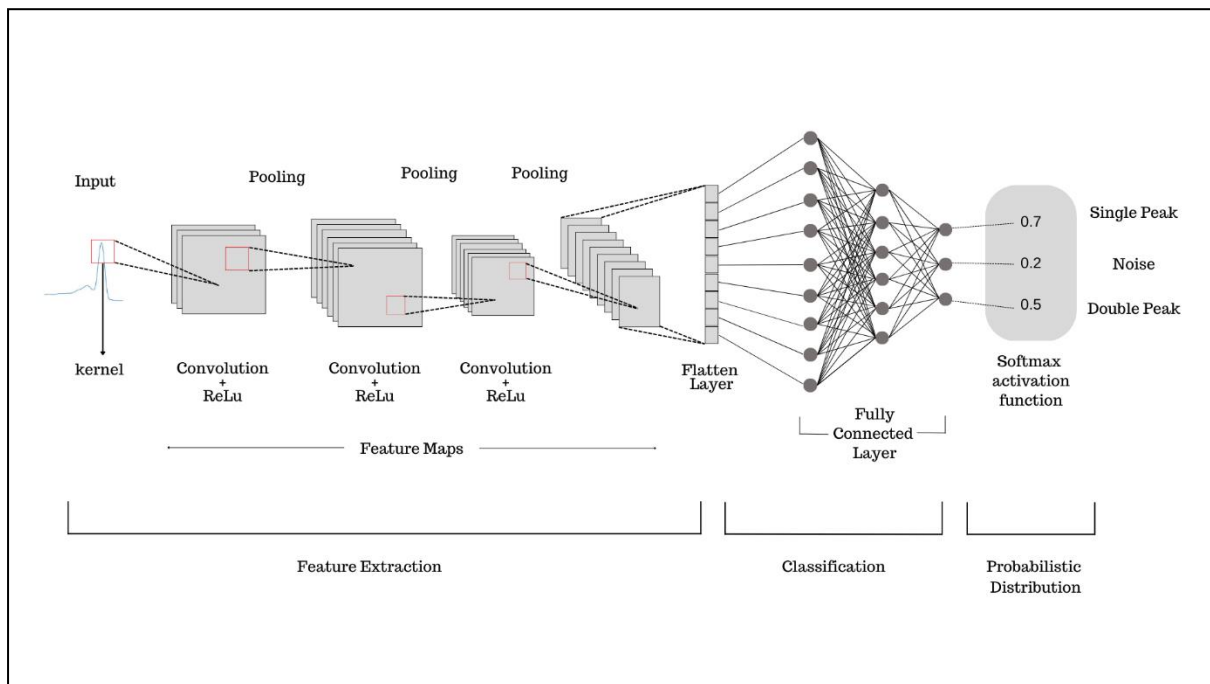


Figure 45: Concept of Convolution Neural Network for classifying DNA melting signal

To train a CNN model, to classify DNA melting signals of “*Single Peaked*”, “*Double Peaked*” and “*Noise*”, necessary training images has to be generated and must be labelled accordingly. ConvNets, or Convolutional Neural Networks, have shown promising results in DNA analysis, particularly in the classification of DNA melting signals. They are able to automatically extract relevant features from the data and learn to classify the signals based on these features. In the context of DNA signal thresholding, ConvNets can be used to generate an image of the signal based on provided coordinates. The image is then processed through the trained neural network, which has learned to identify and extract specific features from the signal. The network's feature maps, which are created during training, enable it to provide a probability distribution indicating the likelihood that the signal belongs to a particular probability density function

(PDF). For example, the network might output a probability distribution indicating that the signal is more likely to belong to the PDF of "Single Peaked" than to the PDF of "Double Peaked" or "Noise". This information can be used to threshold the signal and separate genuine signals from noise or other artifacts.

## 8.2 GENERATING IMAGE DATA SET

Images of DNA melting signal, will be generated, from the acquired co-ordinates data from PCR machines. The image generation includes all type of signals like 'Single peaked', 'Double peaked' and 'Noisy' signals.



Figure 46: Training images of DNA melting signals of various class 'Single', 'Double' and 'Noise'



### 8.3 MODEL ARCHTECTURE

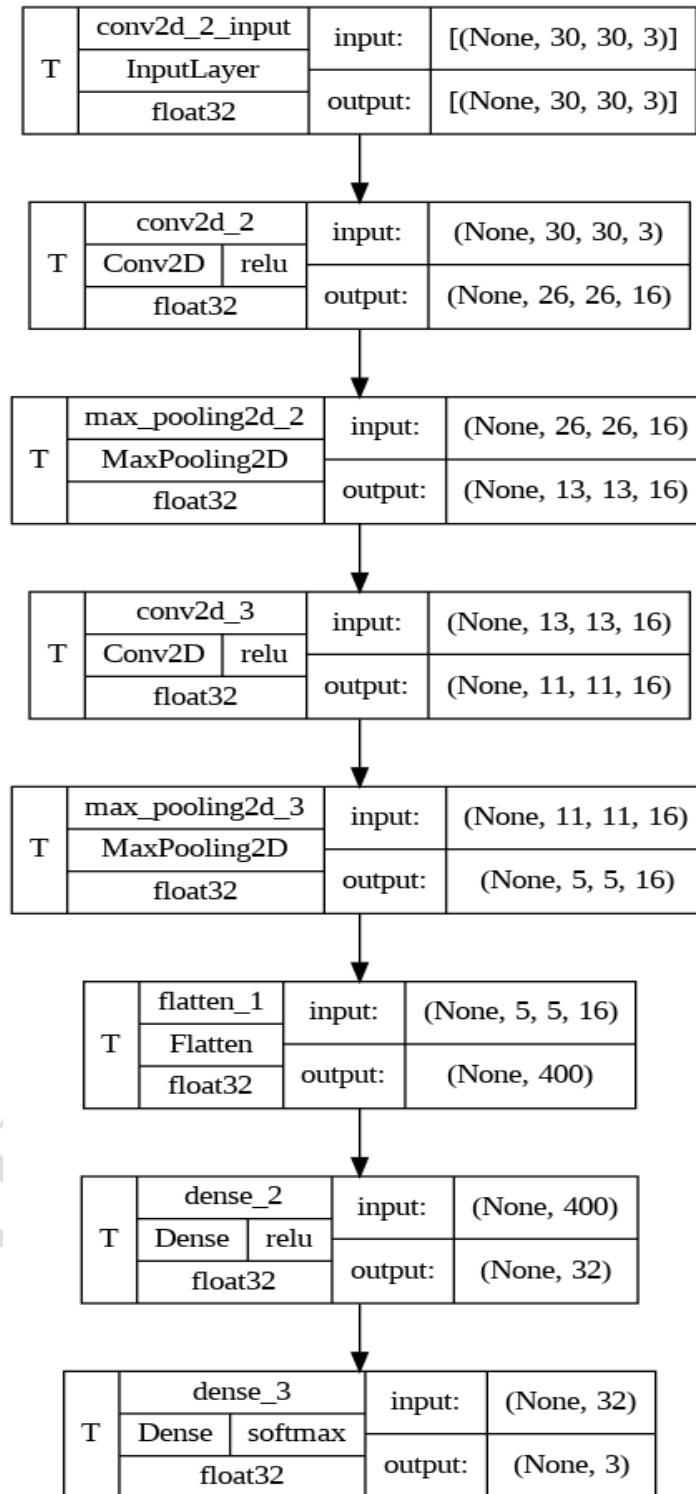


Figure 47: Model architecture with layers

There are totally 569 images, which has 206 “Noisy Signals” labelled as 0, 197 “Single peaked signals” labelled as 1, and finally 166 “Double peaked signals” labelled as 2.

The architecture of the model has 2 convolution layers with 2 pooling layers and 2 Dense layers, which as a whole, has 16,467 total trainable parameters. The input shape of the image is reshaped into (30,30) with 3 channels. The first convolution layer has a kernel size of (5,5), with pooling size of (2,2) and the second convolution layer has a kernel size of (3,3) with a pooling size of (2,2). The activation function used in these layers is *ReLU*. There are 2 Dense layers in the architecture, which has 12832 trainable parameters, with *ReLU* and *Softmax* as activation functions. The model has been trained with 20 epochs with a batch size of 8.

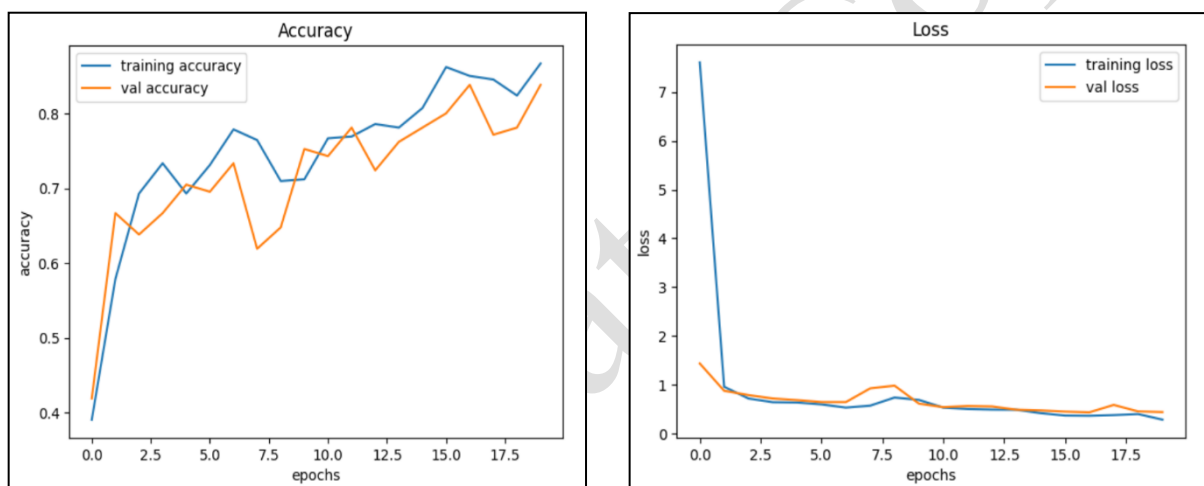


Figure 48: Model performance with Accuracy and Loss

The Model has a validation accuracy of **83.81%** and the training accuracy of **86.67%**, which looks like, the model doesn't overfit to the data. Since it is a multiclass classification problem, looking on the accuracy is not sufficient. The true accuracy of the model will be assessed by looking on metrics like precision and recall. Furtherly, on combing both the metrics, f1 score can be taken into consideration, as it is harmonic mean of both precision and recall, will produce a significant and reliable result if the model truly performs good.



	0	1	2	accuracy	macro avg	weighted avg
precision	0.8	0.857143	1	0.871795	0.885714	0.885714
recall	0.923077	0.923077	0.769231	0.871795	0.871795	0.871795
f1-score	0.857143	0.888889	0.869565	0.871795	0.871866	0.871866
support	13	13	13	0.871795	39	39

Table 4: Model performance metrics

The f1 score provides 85+% of accuracy, which denotes both the precision and recall are high. This result has been evaluated on testing the model on different test images, which the model hasn't yet seen before.

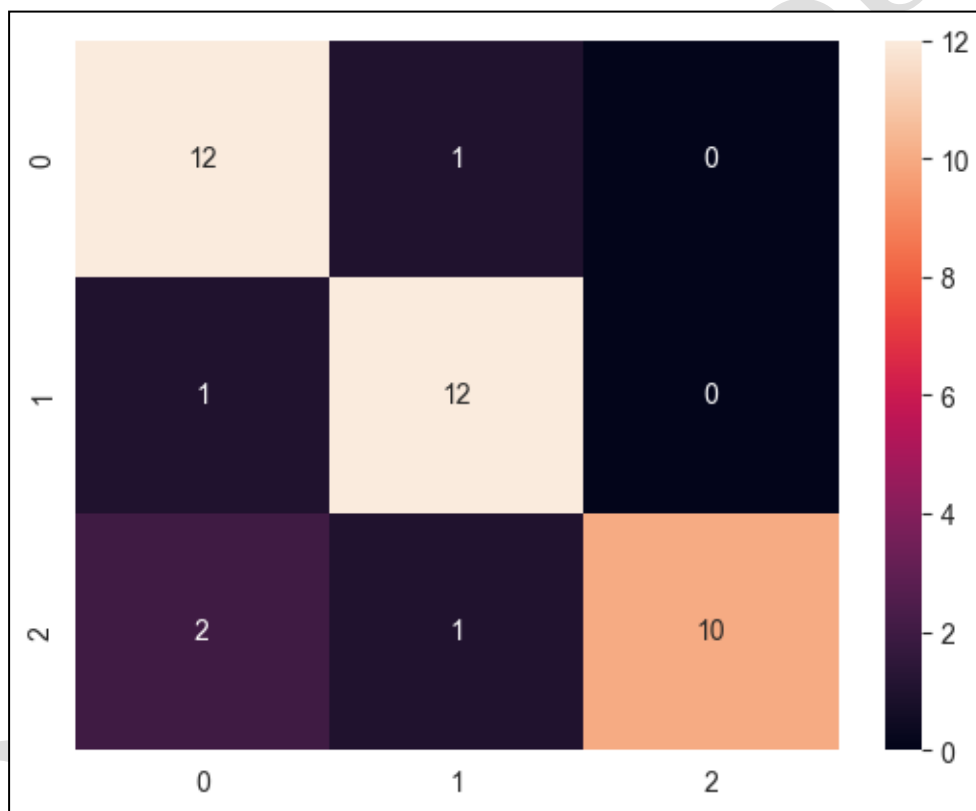


Figure 49: Confusion matrix for the results of CNN model.

Once the model has been validated and tested, it can be applied in the process of extracting information of DNA melting signals with image- based thresholding along with the signal processing method. The model will provide a probabilistic result (0 or 1 or 2), based on which, number of peaks will be considered. If model says '1', feature extraction will be done on, the first highest peak (peaks will be sorted in descending order of prominence), leaving the remaining peaks as out of concern.

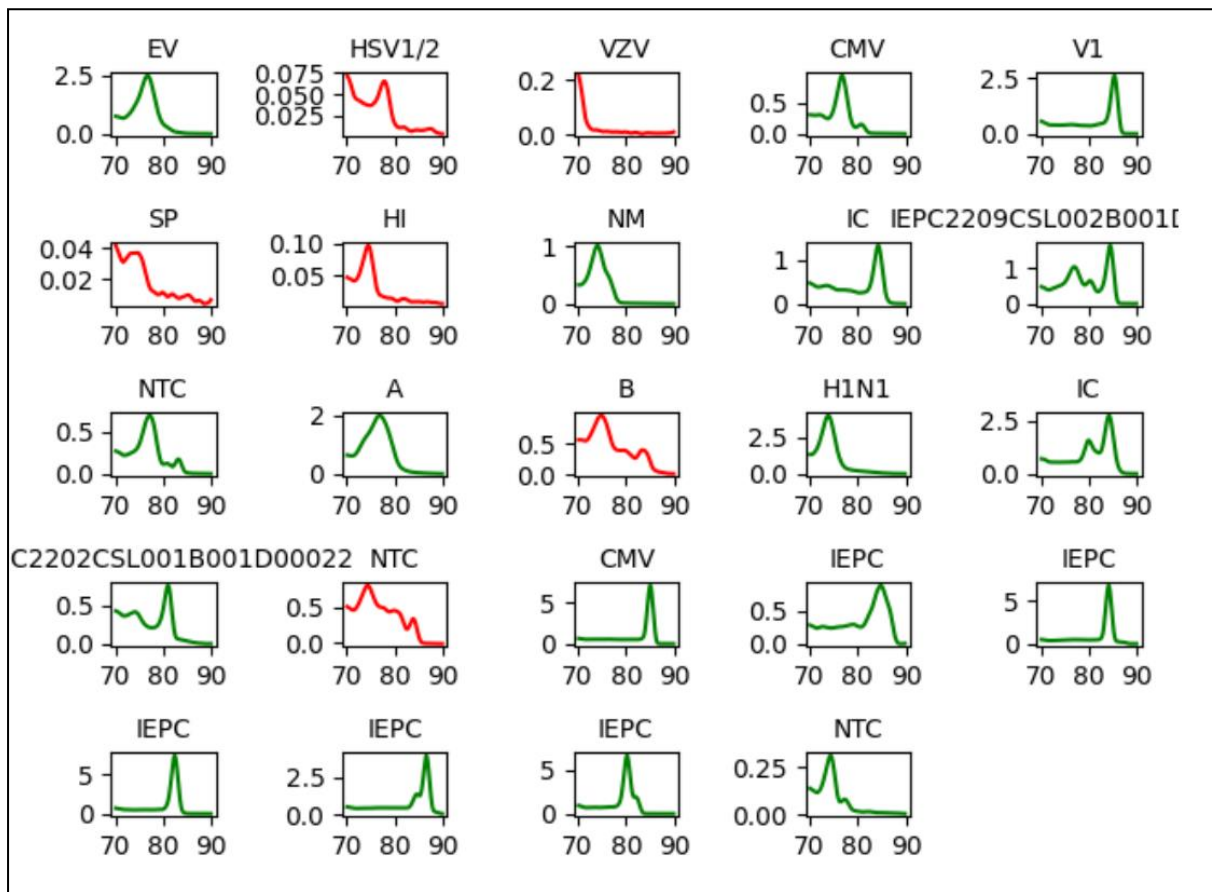


Figure 50: Classification of CNN model between genuine and non-genuine peaks.

In the Figure 3.28 'Green' coloured signals are identified as 'genuine' and 'Red' coloured signals are identified as 'non-Genuine'. For each signal, number of peaks, that has to be considered will be given (0,1,2) based on that, feature extraction will be made.

Tm1	Tstart1	Tend1	Prom1	Width1	AUC1	Tm2	Tstart2	Tend2	Prom2	Width2	AUC2	Target
76.71	73.88	78.62	1.12	57.83	8.75	0.0	0.0	0.0	0.0	0.0	0.0	EV
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	HSV1/2
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	VZV
76.79	75.12	78.12	0.42	36.16	2.15	0.0	0.0	0.0	0.0	0.0	0.0	CMV
85.29	84.04	86.21	0.9	26.43	4.0	76.29	75.04	77.29	0.38	26.87	0.86	V1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	SP
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	HI
74.04	72.04	76.21	0.5	50.28	3.18	0.0	0.0	0.0	0.0	0.0	0.0	NM
84.29	82.88	85.46	0.52	30.84	2.51	73.54	72.46	74.38	0.39	22.59	0.74	IC

## **RESULT AND DISCUSSION**

The CNN based thresholding has provided a significant result, as the model was trained with images of genuine and non-genuine peaks, such that it could learn patterns of such signal, which as a result can identify and classify signals. Based on the result of the model, feature extraction process has been performed and relevant features of the classified signals are captured and stored in the feature store for model building.

Duplicate Copy

# CHAPTER 9

## SYSTEM DESIGN & DEVELOPMENTS

### 9.1 COMPONENTS

For developing the AI-based framework for automated analysis, interpretation and data management for the HRM data, which is generalized and optimized, three major components were developed by the team are

- EXTRACTOR (data extraction tool)
- PyHRM (Feature engineering)
- Meltcurve Interpreter (prediction)

The team has developed three major components for developing an AI-based framework for automated analysis, interpretation, and data management of High-Resolution Melting (HRM) data. These components include the data extraction tool called "EXTRACTOR," the feature engineering tool called "PyHRM," and the prediction tool called "Meltcurve Interpreter."

The EXTRACTOR tool is used to extract data from the raw HRM files, while the PyHRM tool is used for feature engineering, which involves extracting relevant features from the HRM data. Finally, the Meltcurve Interpreter tool uses predictive analytics and deep learning models to interpret the extracted features and predict the presence of the intended molecular target in a clinical sample tested.

These three components work together to develop an AI-based framework that can automate the analysis and interpretation of HRM data, allowing for faster and more accurate diagnosis of infectious diseases. With this framework, clinicians can make more informed decisions and plan the course of treatment for their patients.

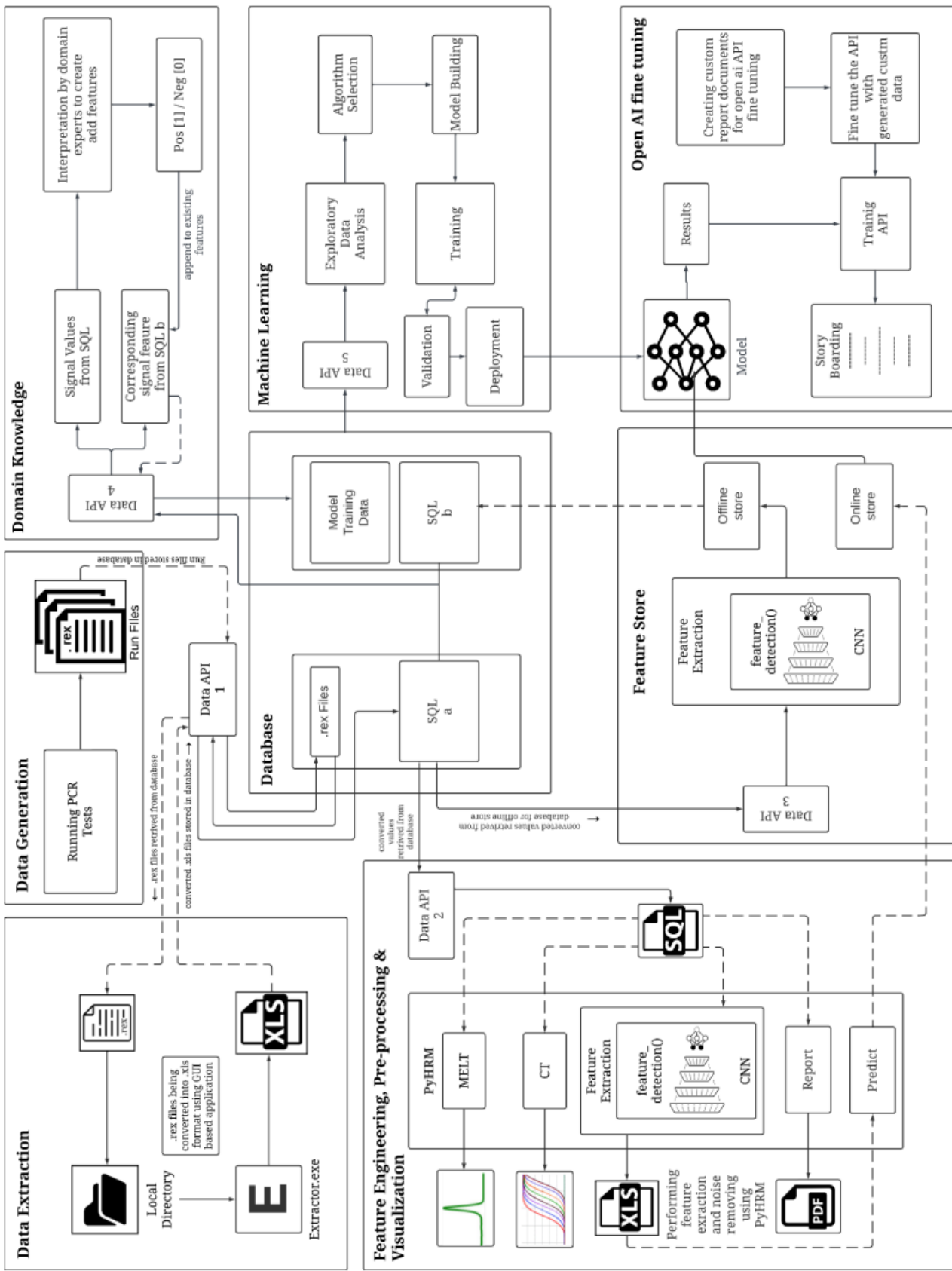


Figure 51: AI-framework

## 9.2 EXTRACTOR

Extractor is a lightweight simple GUI-based application (figure.) that extracts '.rex' files from the **Qiagen's Rotor-Gene Q Software** to the necessary '.xls' file. It's built for the users such as laboratory technicians and clinicians who handle and run PCR experiments especially in **Qiagen's Rotor-Gene Q** thermal cycler machine.

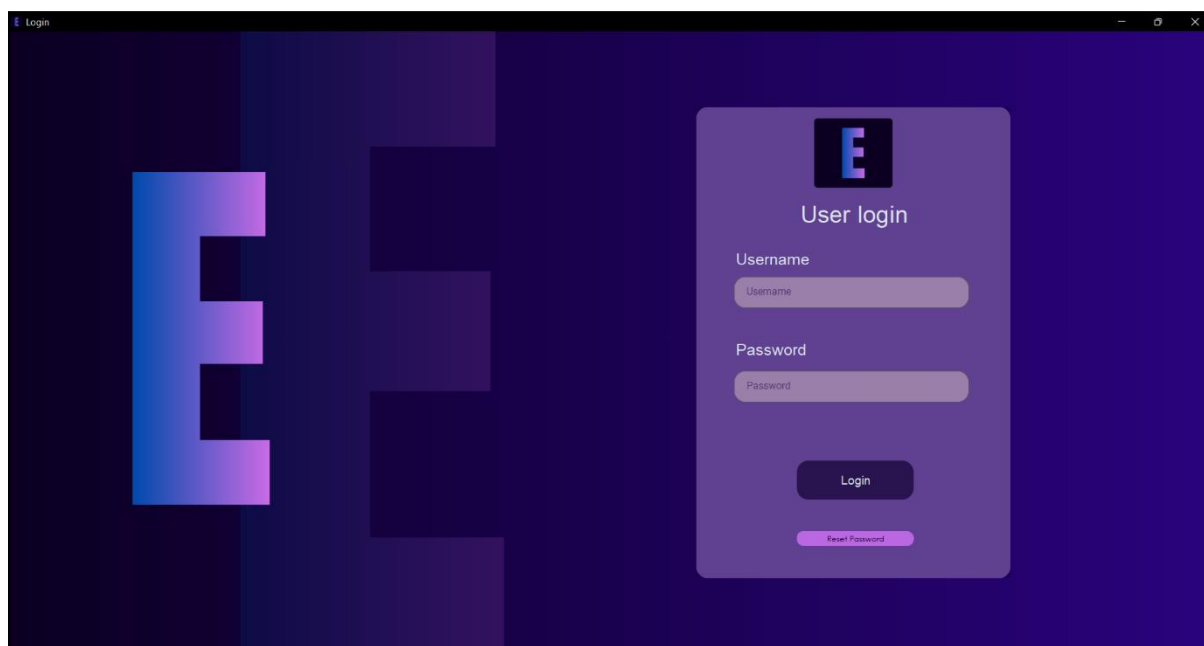


Figure 52: User interface of Extractor

After the successful experiment ran in *Rotor-Gene Q* cyclers, it produces the raw data and the users which can be only opened and analyzed via **Qiagen's Q-Rex Software**. If a specific run file (raw data) has to be exported into desired formats such as *text(.txt)*, *HTML Table(.html)*, *XML(.xml)*, *excel(.xls)* given by the **Qiagen Rotor-Gene Q-Rex Software**. Here we automated the user role by our **EXTRACTOR** Software, by which you simply put the raw data file directory and desired directory to which the excel files are stored in your system, which saves time and not to be burned out from this repetitive task.

## 9.2.1 FRAMEWORK

The framework for the extractor application is below

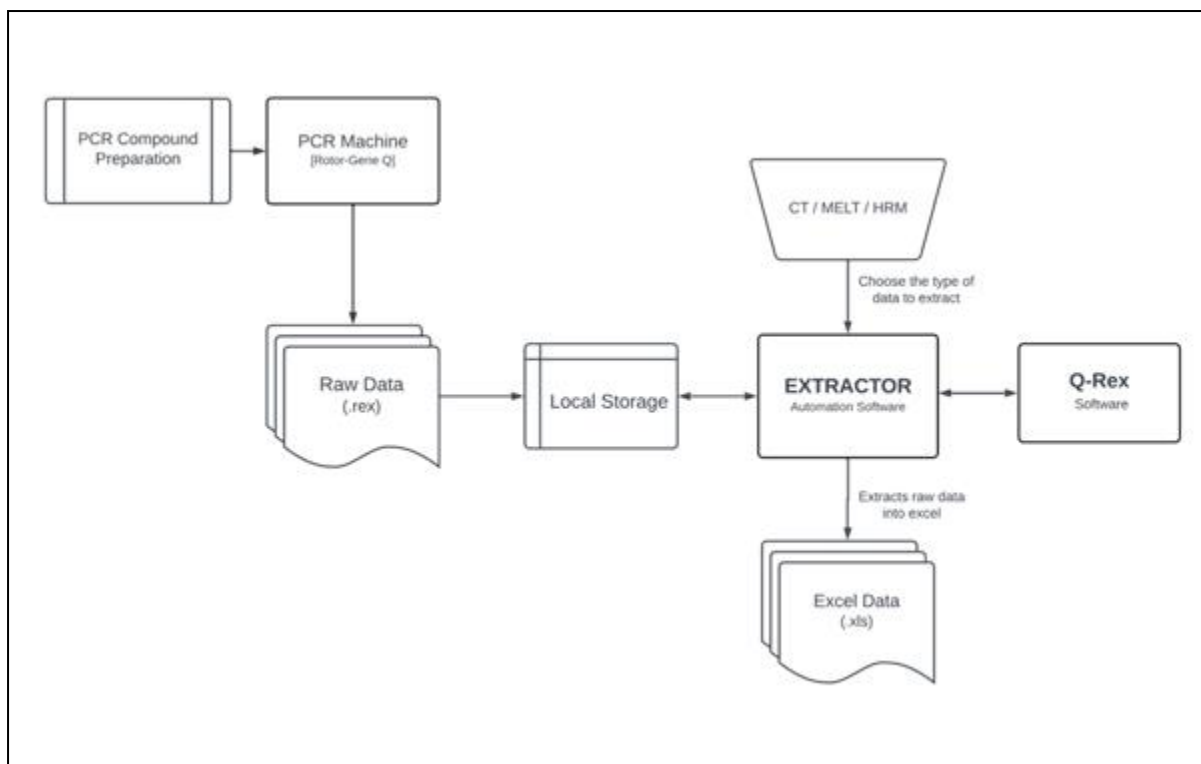


Figure 53: Framework of extractor

### Manual Conversion

Time consuming and often introduce frustration and inconsistency, while converting bulks of raw data by manual process.

### Automatic Conversion

Runs in a constant time and it handle countless of raw data automatically without any human intervence.

### Features

- Able to set user credentials.
- Selection of which type of data to extract from the raw data, i.e.,
  - CT (Amplification Curve)
  - MELT (derivative) &
  - HRM (Normalized fluorescence)
- Supports desired output format as excel file.

## Types of data to extracts

Select the type of data from the drop-down menu and enter the respected fields below and finally enter submit (figure).

- CT
- MELT
- HRM

## System Requirements

The following are the essential requirements for this software to run:

Q-Rex Software : Rotor-Gene Q Software v2.3.5 (Build 1)

Platforms : Windows 10 / 11

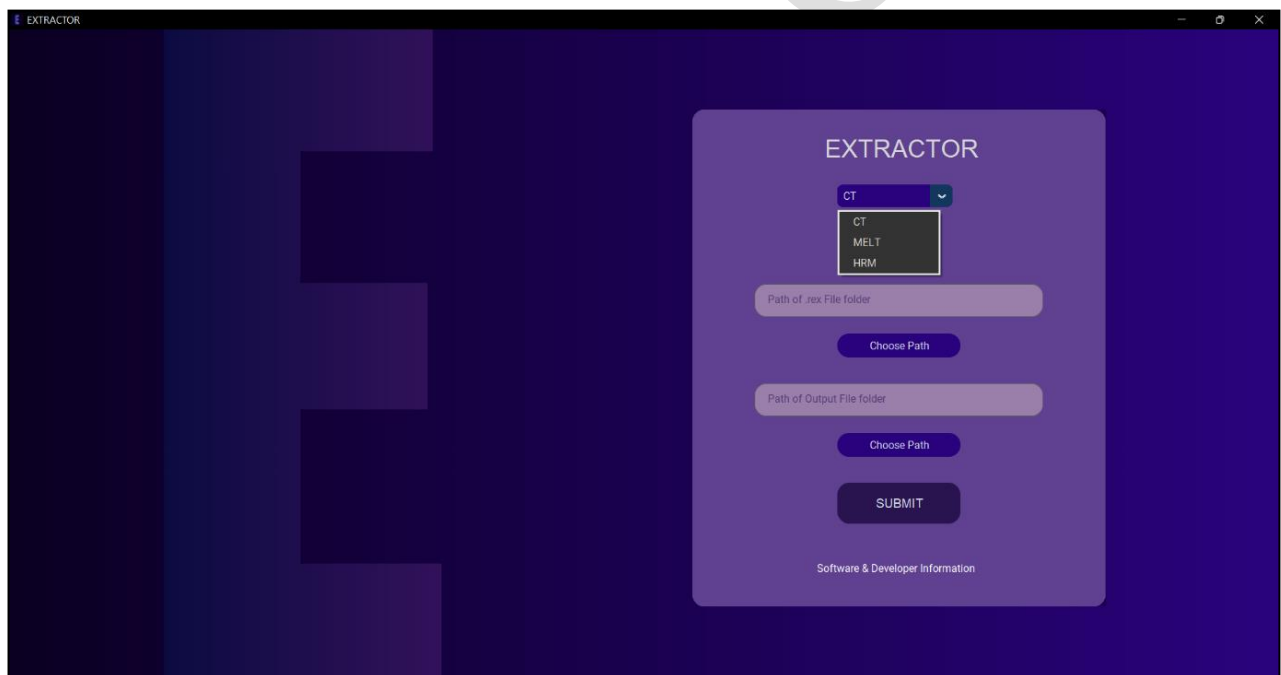


Figure 54: Type of data to extract



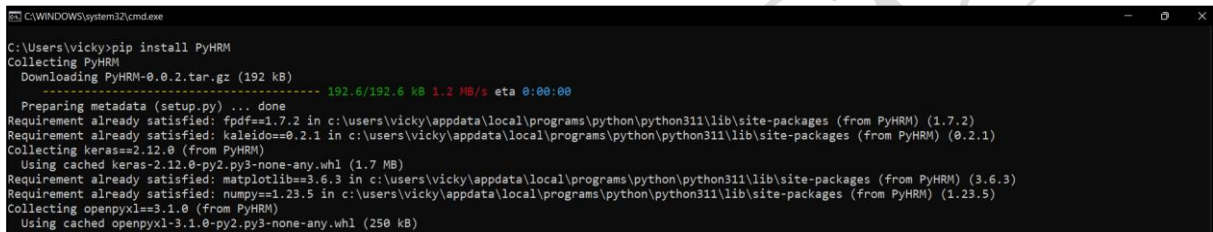
### 9.3 PyHRM

We developed a python-based library called PyHRM for processing High-Resolution Melting (HRM) data, especially, DNA melting signals to extract features like 'Melting Temperatures', 'Take-off and Touch-down points of melting signal (Temperature at which peak start rising and temperature at which peak falls down)', 'Peak prominences', and 'Area Under the curve'. Additionally, the library offers interactive visualization for DNA melting signal and vision based filtering, to eliminate noisy signals from the data and provides only genuine peaks with all the above mentioned features.

#### Installing with PyPi

The PyHRM library can be installed with using pip command,

`python -m pip install PyHRM` or `pip3 install PyHRM`



```
C:\WINDOWS\system32\cmd.exe
C:\Users\vicky>pip install PyHRM
Collecting PyHRM
  Downloading PyHRM-0.0.2.tar.gz (192 kB)
-----192.6/192.6 kB 1.2 MB/s eta 0:00:00
Preparing metadata (setup.py) ... done
Requirement already satisfied: fpdf==1.7.2 in c:\users\vicky\appdata\local\programs\python\python311\lib\site-packages (from PyHRM) (1.7.2)
Requirement already satisfied: kaleido==0.2.1 in c:\users\vicky\appdata\local\programs\python\python311\lib\site-packages (from PyHRM) (0.2.1)
Collecting keras==2.12.0 (from PyHRM)
  Using cached keras-2.12.0-py2.py3-none-any.whl (1.7 MB)
Requirement already satisfied: matplotlib==3.6.3 in c:\users\vicky\appdata\local\programs\python\python311\lib\site-packages (from PyHRM) (3.6.3)
Requirement already satisfied: numpy==1.23.5 in c:\users\vicky\appdata\local\programs\python\python311\lib\site-packages (from PyHRM) (1.23.5)
Collecting openpyxl==3.1.0 (from PyHRM)
  Using cached openpyxl-3.1.0-py2.py3-none-any.whl (250 kB)
```

Figure 55: PyHRM installation

#### File stack of the library

The file stack of the PyHRM library consists of various files including .py and .h5 files which are basement for feature detection of HRM data.

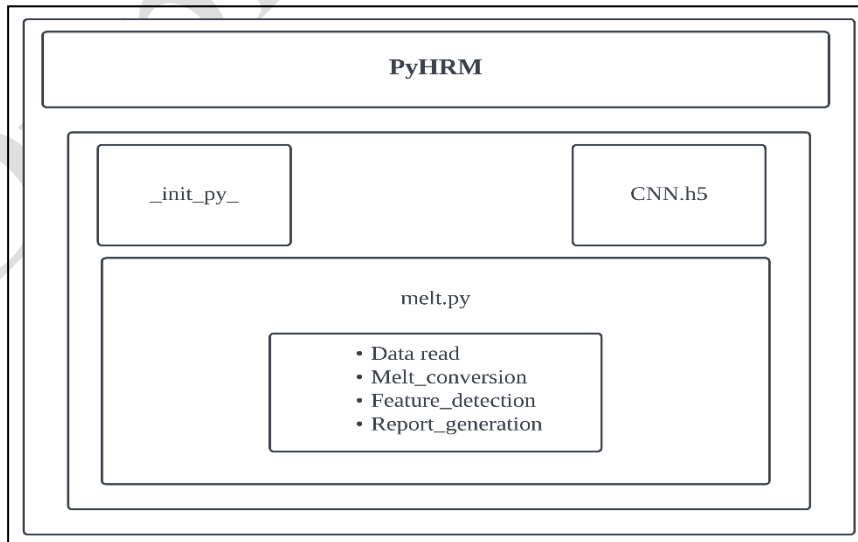


Figure 56: File stack of PyHRM

The Deep Learning model has been saved using the method in Keras module as CNN.h5 for meltcurve peak classification. The melt.py has major functions like data read, melt\_conversion, feature\_detection, report\_generation for processing HRM data.

### Dependencies of PyHRM

The following are the necessary dependencies for the PyHRM.

- fpdf==1.7.2
- kaleido==0.2.1
- keras==2.12.0
- matplotlib==3.6.3
- numpy==1.23.5
- openpyxl==3.1.0
- packaging==23.1
- pandas==1.5.3
- Pillow
- plotly==5.13.0
- Requests==2.28.2
- scipy==1.10.1
- tensorflow==2.12.0
- tqdm==4.64.1
- xlrd==2.0.1

### Features

The PyHRM library has the following features

- Rapid preprocessing.
- Feature Extraction
  - Tm (Melting Temperature (Max 2))
  - Tstart (Starting temperature point)
  - Tend (Ending Temperature)
  - Prominence
  - Area Under the curve
- Interactive Visualization.
- Computer Vision based thresholding for eliminating noisy signals.
- Report Generation.

### Input data format

The PyHRM has only supports .xls and .xlsx formats for seamless analysis.

	Text	X	Y	Text.1	X.1	Y.1	Text.2	X.2	Y.2	Text.3	...
0	1:	70.050000	0.236726	2:	70.050000	0.017526	3:	70.050000	0.019620	4:	...
1	1:	70.083333	0.238449	2:	70.083333	0.017746	3:	70.083333	0.019589	4:	...
2	1:	70.116667	0.240134	2:	70.116667	0.017974	3:	70.116667	0.019551	4:	...

Figure 57: Input data format for PyHRM

## Working of library

The library has been imported in the any notebook IDEs, by with the following commands in figure. Next to create a class instance for the meltcurve interpreter module present in the PyRM

```
[1]: from PyHRM.melt import MeltcurveInterpreter as mlt
[2]: obj=mlt()
[3]: data=obj.data_read(path="C:\\Users\\wicky\\Desktop\\Melt output\\Melt Extracted MEP 2019-05-26 (12).xls")
*** No CODEPAGE record, no encoding_override: will use 'iso-8859-1'
[4]: fig=obj.plot(data=data,save=True)
[5]: fig.show()
***
```

Figure 58: Import PyHRM

To process the HRM data with the desired input format using data\_read() with the necessary parameters and to visualize the HRM data with using the plot() figure

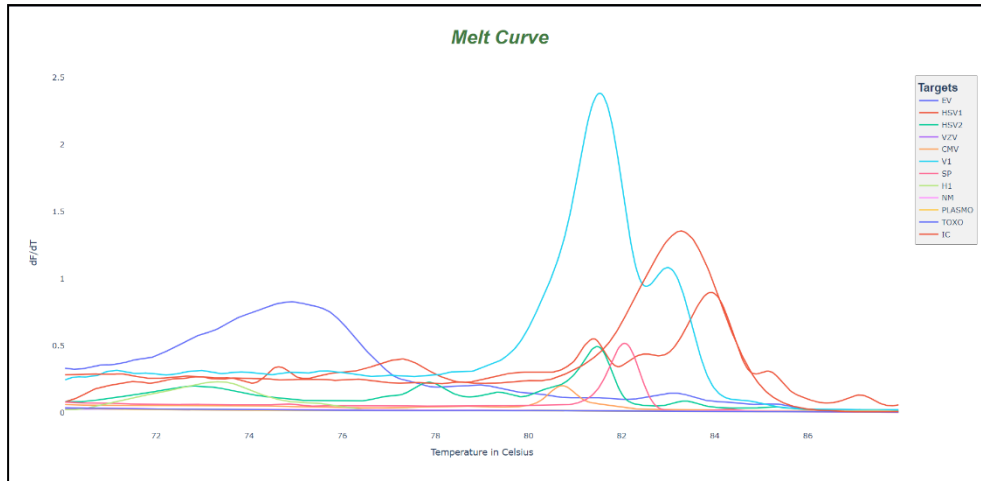


Figure 59: Output of plot()

The major function in this library which detects the features of HRM signals using customized Deep Learning model using `feature_detection()` which returns the outputs in dataframe format. To generate and download the report of features using `report()` in pdf format figure.

```
[6]: feature_detectobj.feature_detection(return_values = True, download = False)
***
[7]: feature_detect
[7]:
```

	Tm1	Tstart1	Tend1	Prom1	Width1	AUC1	Tm2	Tstart2	Tend2	Prom2	Width2	AUC2	Target
1	74.916667	72.183333	76.55	0.450997	131.295163	2.950736	0.0	0.0	0.0	0.0	0.0	0.0	EV
2	83.916667	79.45	85.283333	0.286413	174.865372	2.660655	81.383333	80.983333	81.75	0.395556	22.290177	0.358412	HSV1
3	81.483333	80.45	81.95	0.185348	44.459363	0.483385	72.783333	71.183333	74.483333	0.115892	99.897611	0.529285	HSV2
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	VZV
5	80.716667	80.183333	81.316667	0.077816	33.851843	0.15933	0.0	0.0	0.0	0.0	0.0	0.0	CMV
6	81.516667	80.216667	83.416667	0.779951	96.286576	4.532545	0.0	0.0	0.0	0.0	0.0	0.0	V1
7	82.05	81.45	82.55	0.166694	32.166637	0.39067	0.0	0.0	0.0	0.0	0.0	0.0	SP
8	73.35	71.05	75.316667	0.074913	127.406003	0.643621	0.0	0.0	0.0	0.0	0.0	0.0	H1
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NM
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	PLASMO
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	TOXO
12	83.283333	81.683333	84.483333	0.500495	84.577407	2.750904	0.0	0.0	0.0	0.0	0.0	0.0	IC

```
[8]: Report = obj.report()
Enter the path to save: C:\Users\vicky\Desktop\report.pdf
```

Figure 60: Features of HRM data using `feature_detection()`

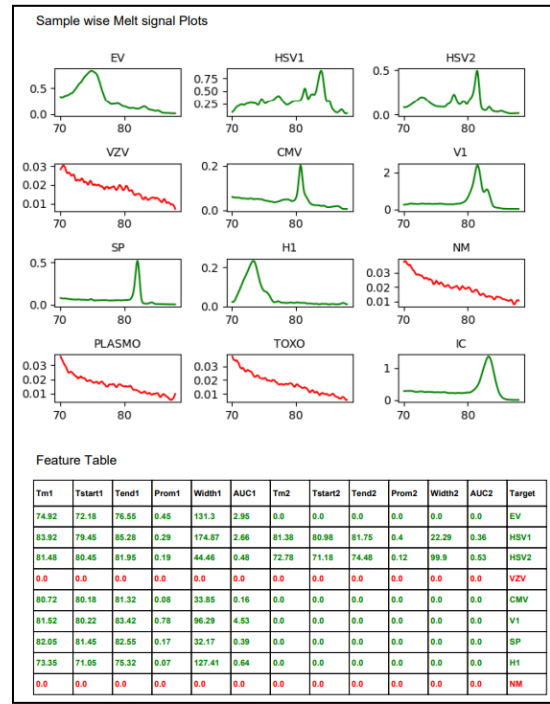
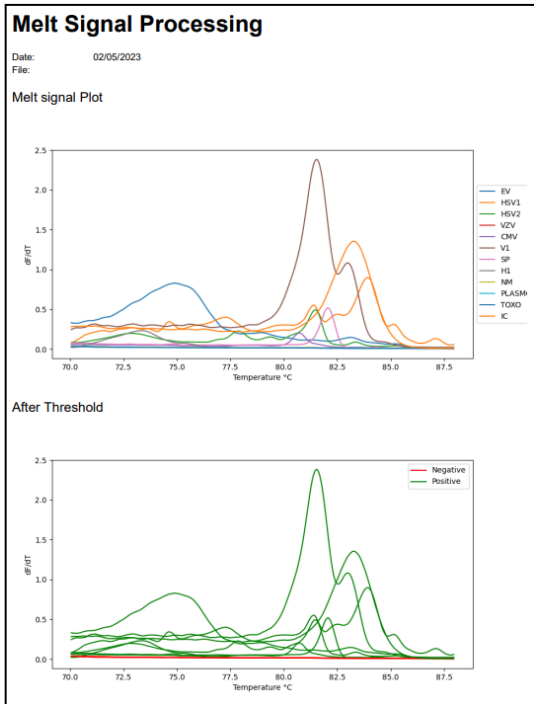


Figure 61: Reports of features detection

## 9.4 MELTCURVE INTERPRETER

The MeltcurveInterpreter is a web-based application for analysing and interpreting the final results from the consequences of extractor and PyHRM library. This module consists of various files and folders such as .py, .html, .css, .h5 for the final classification and interpretation of Meningitidis panel figure .

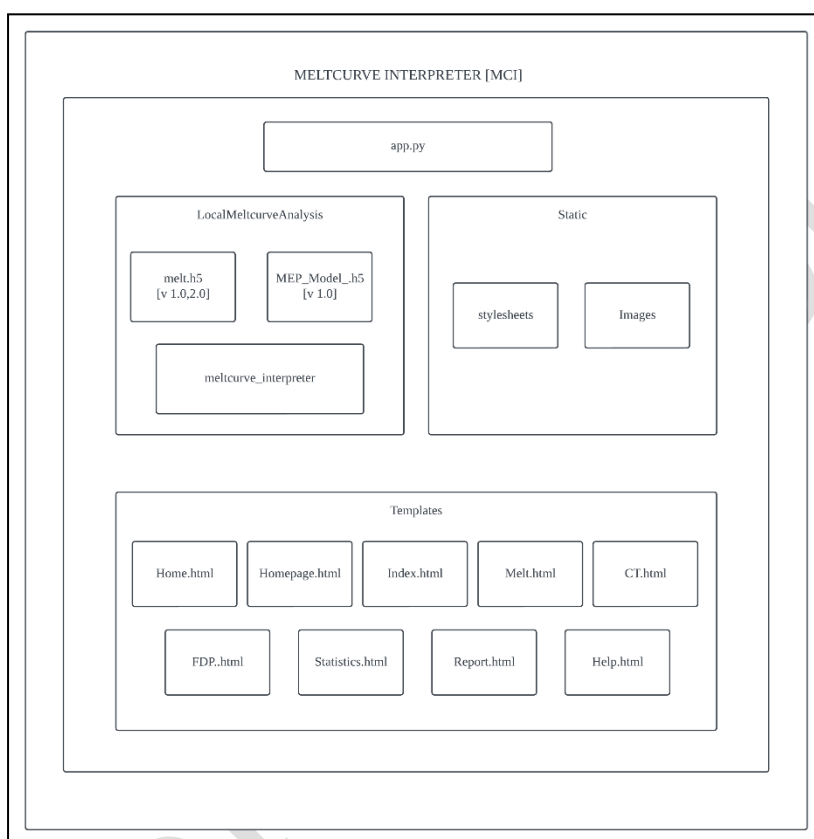


Figure 62: File stack of meltcurve interpreter

Here Meltcurve interpreter [MCI], there are two main Deep Learning models melt.h5, MEP\_Model.h5 for the meltcurve peak classification and the molecular target classification respectively.

The front-end of the MCI has the following files in the directory:

### Templates

- Home.html
- Homepage.html
- Index.html
- Melt.html
- CT.html
- FDP.html
- Statistics.html
- Report.html
- Help.html

- Static
- Stylesheets
- Images

The back-end of the MCI has the following files in the directory:

LocalMeltcurveAnalysis

- melt.h5
- MEP\_Model.h5
- meltcurve\_interpreter

The meltcurve interpreter was deployed using a simple web development framework in python called flask.

MCI is designed as user-friendly with multiple features such as file upload, visualization of interactive graphs, feature detection, EDA, final report.

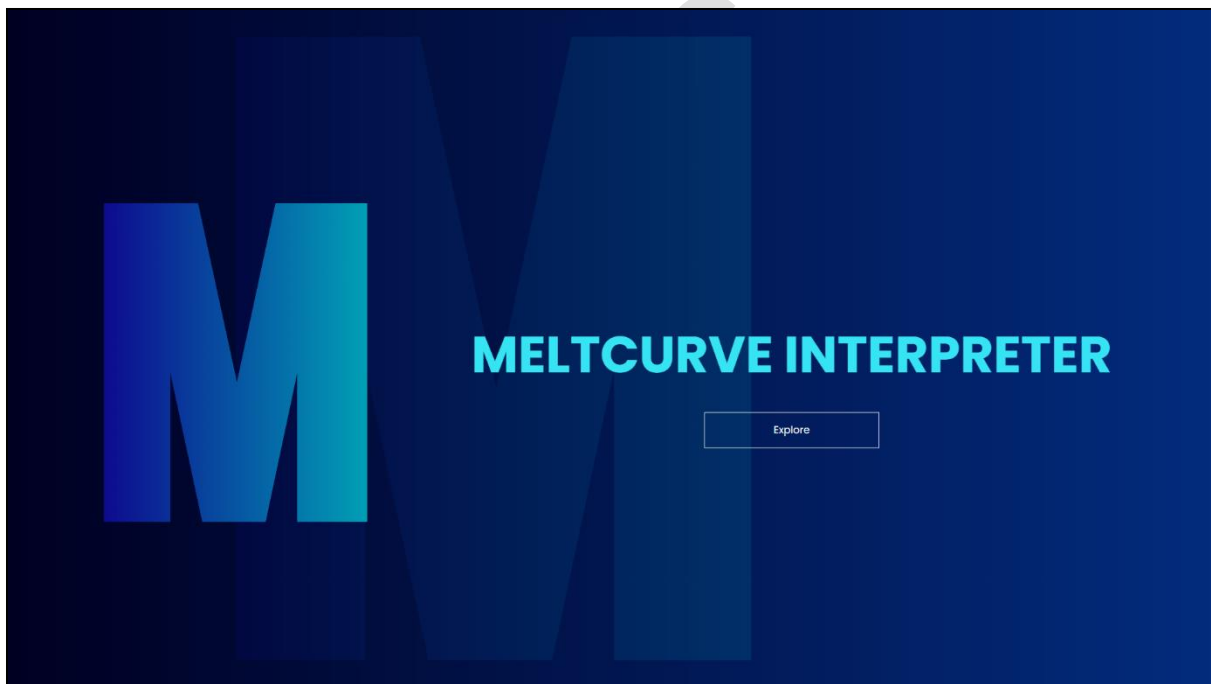


Figure 63: MCI Interface

The home page of MCI describes the organization profiles and working modules as shown in figure 64

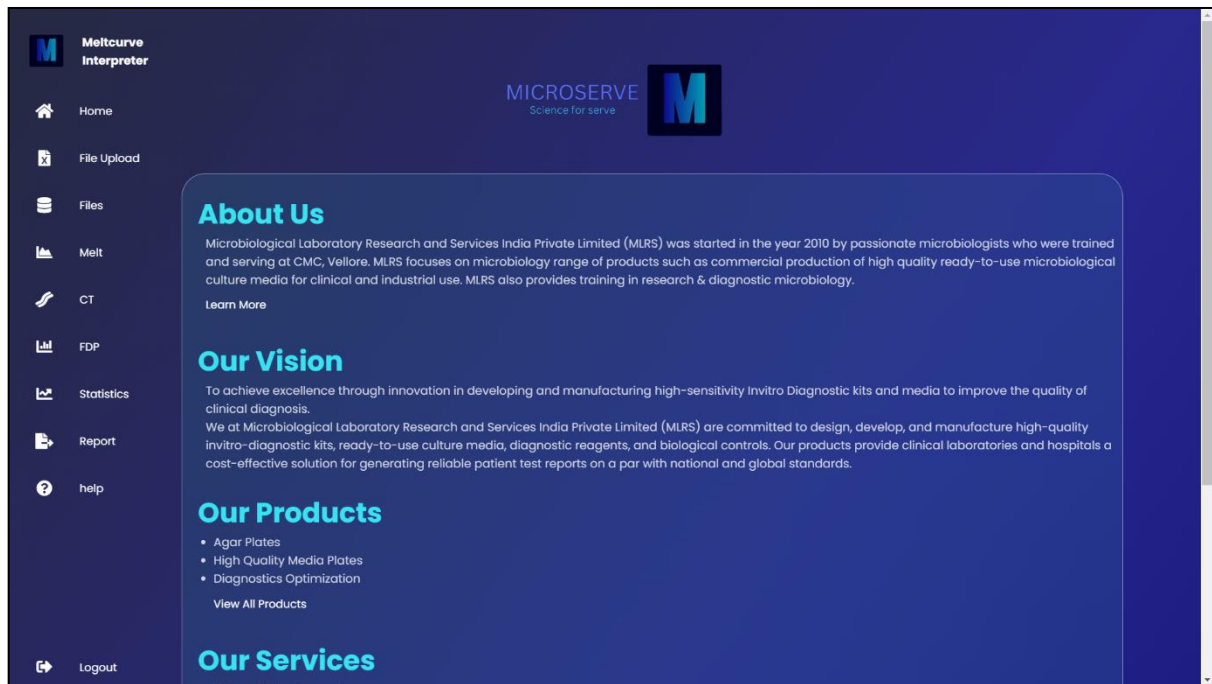


Figure 64: MCI Home page

The file upload page consists of Melt and Ct files in excel format which is mandatory, and it gives a token which is used to retrieved the data at anytime. The uploaded files are storing and retrieval in the centralized database using PostgreSQL figure.

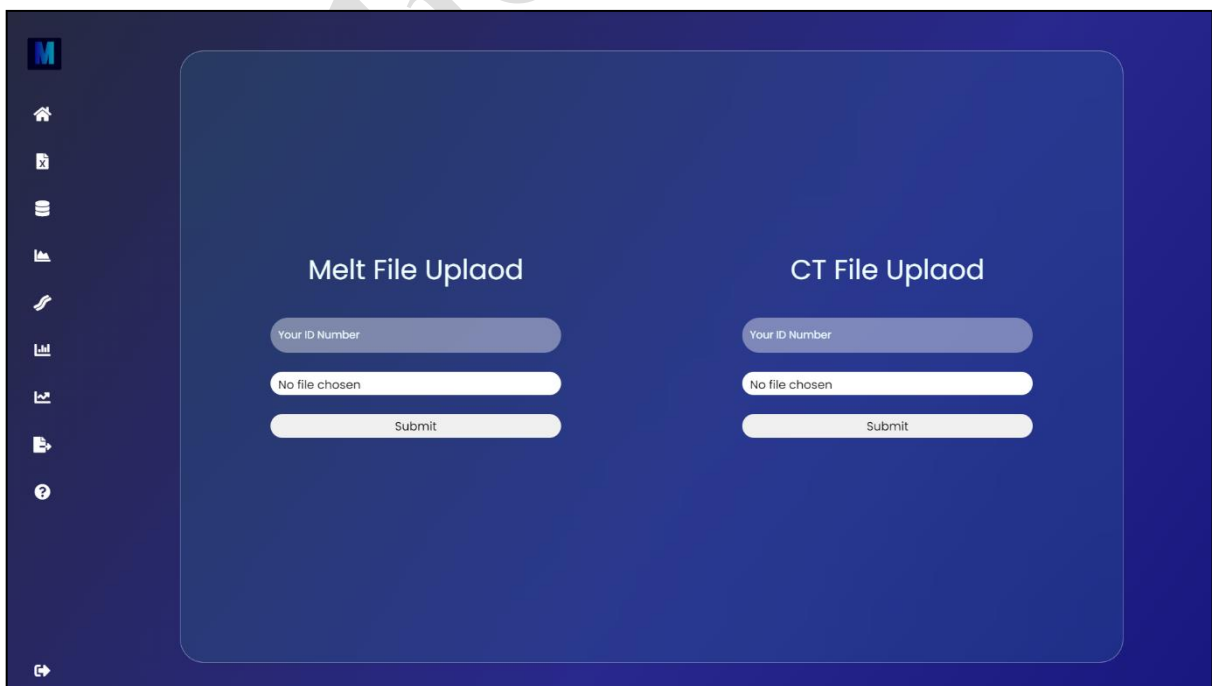


Figure 65: MCI file upload



Once the file has been uploaded ,it can be retrieved with the token and username for further analysis and for the visualisation of Melt and Amplification curves figure .

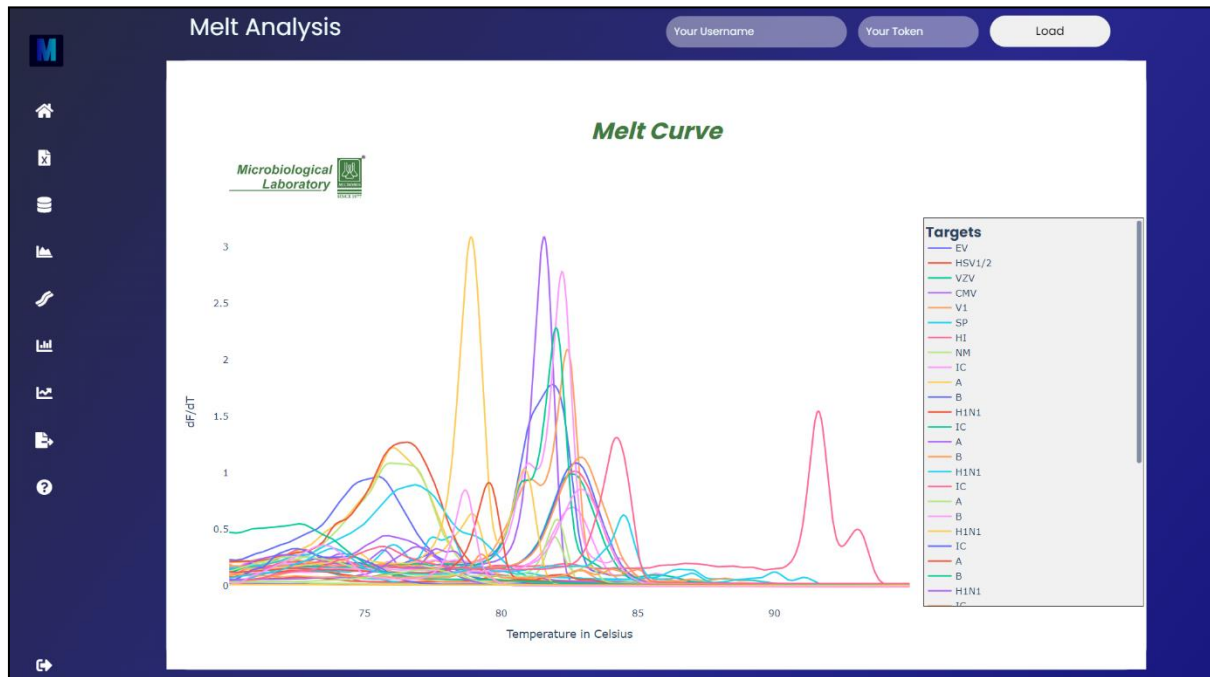


Figure 66: MCI Melt curve visualisation

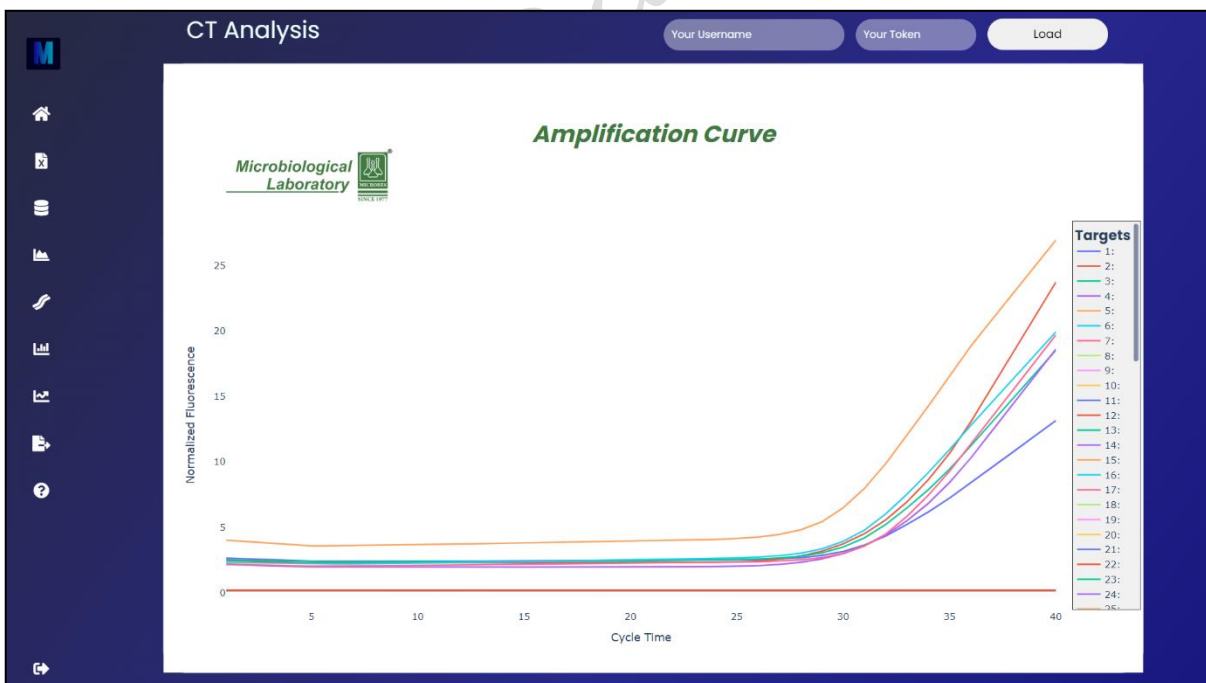


Figure 67: MCI amplification curvet visualisation

The classification of Melt peaks for thresholding and features of melt signals are detected by the feature detection panel its gave a peak features like temperature start, end, AUC, prominence, width and target of the samples tested figure.

	Tm1	Tstart1	Tend1	Prom1	Width1	AUC1	Tm2	Tstart2	Tend2	Prom2	Width2	AUC2	Target
1	75.483333	72.416667	77.05	0.410498	139.007766	3.257496	0.0	0.0	0.0	0.0	0.0	0.0	EV
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	HSV1/2
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	VZV
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	CMV
5	74.983333	71.316667	76.783333	0.063821	164.105161	0.755914	82.883333	81.016667	83.55	0.052557	75.785382	0.228626	VI
6	72.583333	70.883333	74.583333	0.068906	110.63459	0.491146	0.0	0.0	0.0	0.0	0.0	0.0	SP
7	72.716667	70.65	74.05	0.113542	101.400699	0.778337	0.0	0.0	0.0	0.0	0.0	0.0	HI
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NM
9	82.616667	81.25	83.65	0.236339	71.430159	1.168199	0.0	0.0	0.0	0.0	0.0	0.0	IC
10	76.05	73.45	77.95	0.456655	134.51186	3.769492	0.0	0.0	0.0	0.0	0.0	0.0	A
11	81.883333	80.216667	82.816667	0.503152	78.084307	3.408177	0.0	0.0	0.0	0.0	0.0	0.0	B
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	H1N1
13	82.616667	81.25	83.816667	0.351185	77.055999	1.849754	0.0	0.0	0.0	0.0	0.0	0.0	IC
14	81.583333	80.85	82.083333	0.878243	37.888495	2.601717	0.0	0.0	0.0	0.0	0.0	0.0	A
15	82.416667	80.383333	82.983333	0.63553	77.4605	3.0538	0.0	0.0	0.0	0.0	0.0	0.0	B
16	78.15	75.25	78.616667	0.197177	101.689018	1.1052	76.05	75.483333	76.583333	0.253694	32.805753	0.344145	H1N1
17	82.716667	81.35	83.95	0.363649	78.131312	1.920552	0.0	0.0	0.0	0.0	0.0	0.0	IC
18	75.95	73.616667	77.983333	0.402866	131.148814	3.5107	82.05	81.35	82.516667	0.208962	34.593107	0.481868	A

Figure 68: MCI feature detection panel

The statistical measures of the HRM data to gain insights with stipulated statistical analysis figure

Dataset statistics		Variable types	
Number of variables	13	Numeric	12
Number of observations	40	Categorical	1
Missing cells	0		
Missing cells (%)	0.0%		
Duplicate rows	2		
Duplicate rows (%)	5.0%		
Total size in memory	5.4 KIB		
Average record size in memory	138.8 B		

Figure 69: MCI Statistical measures

Finally, the report has been generated in pdf format with the melt curve graphs of classified peaks with features table as shown in fig.71

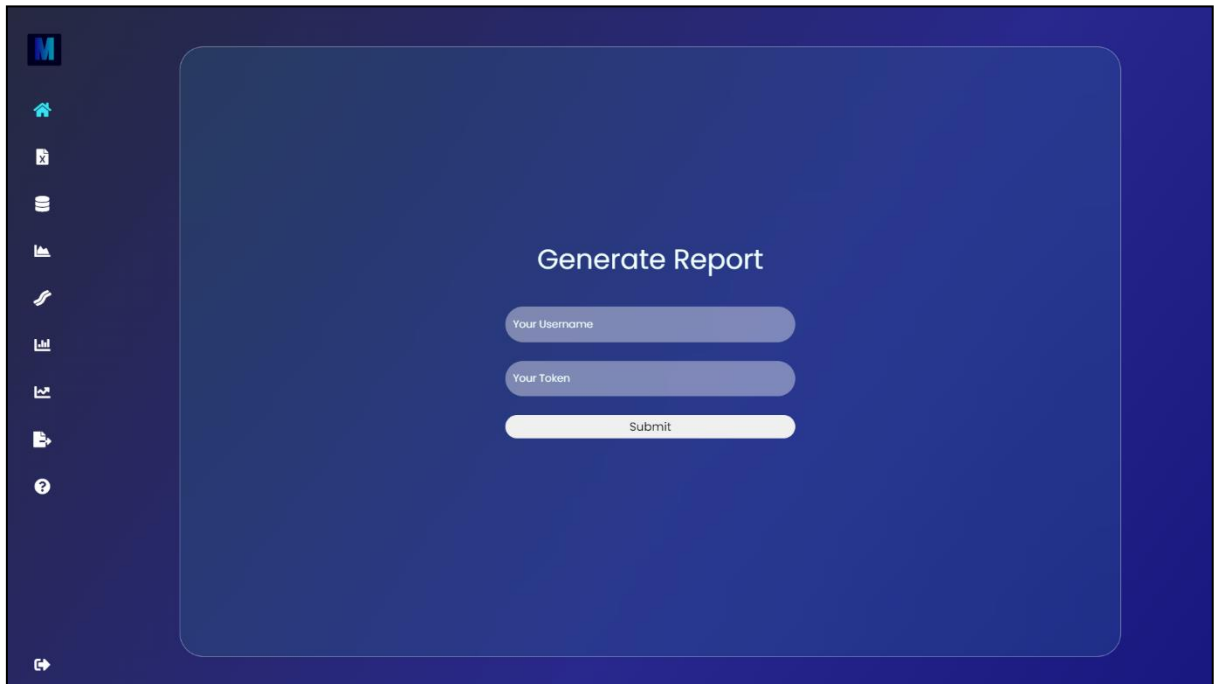


Figure 70: MCI Report Generation

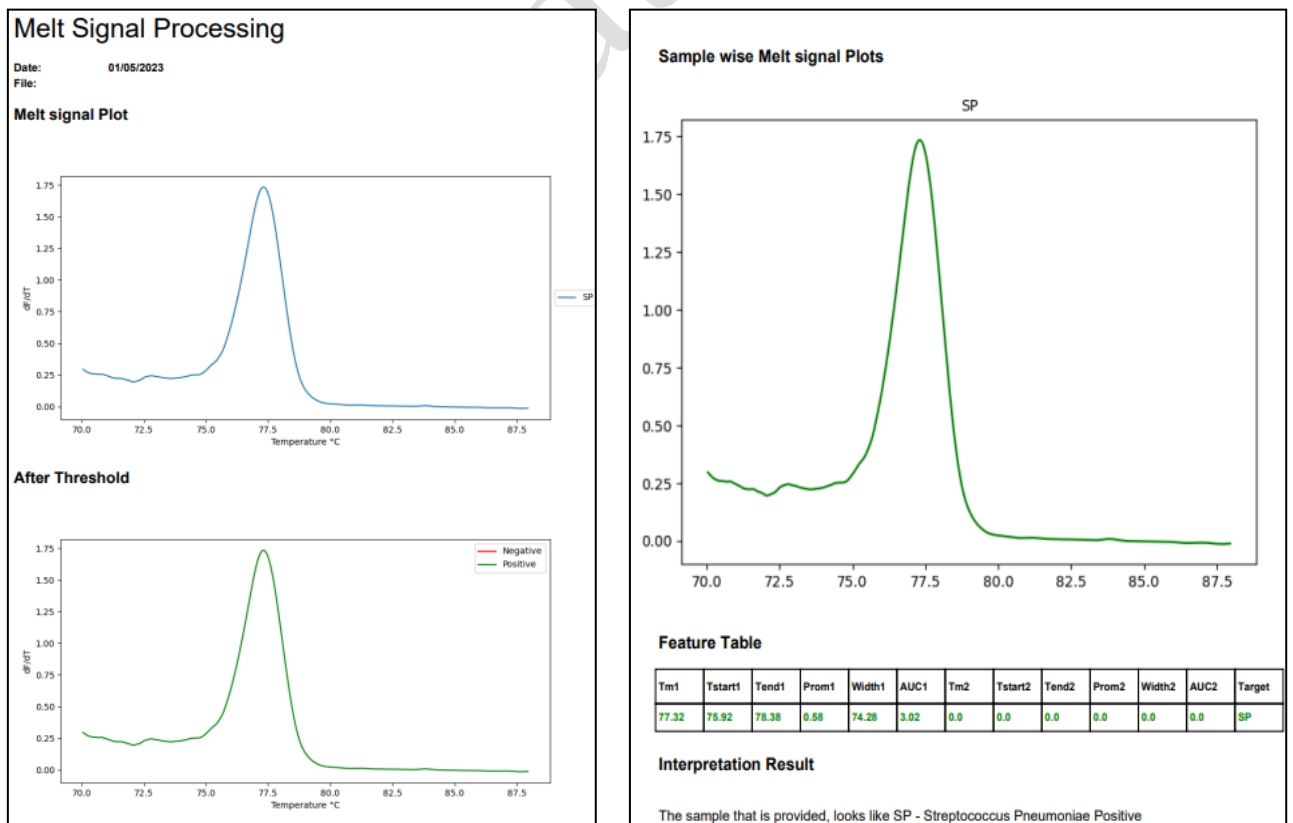


Figure 71: MCI Final Report

## 9.5 ER DIAGRAM

The Entity-Relationship Diagram for the back-end database component of MCI as shown below fig.

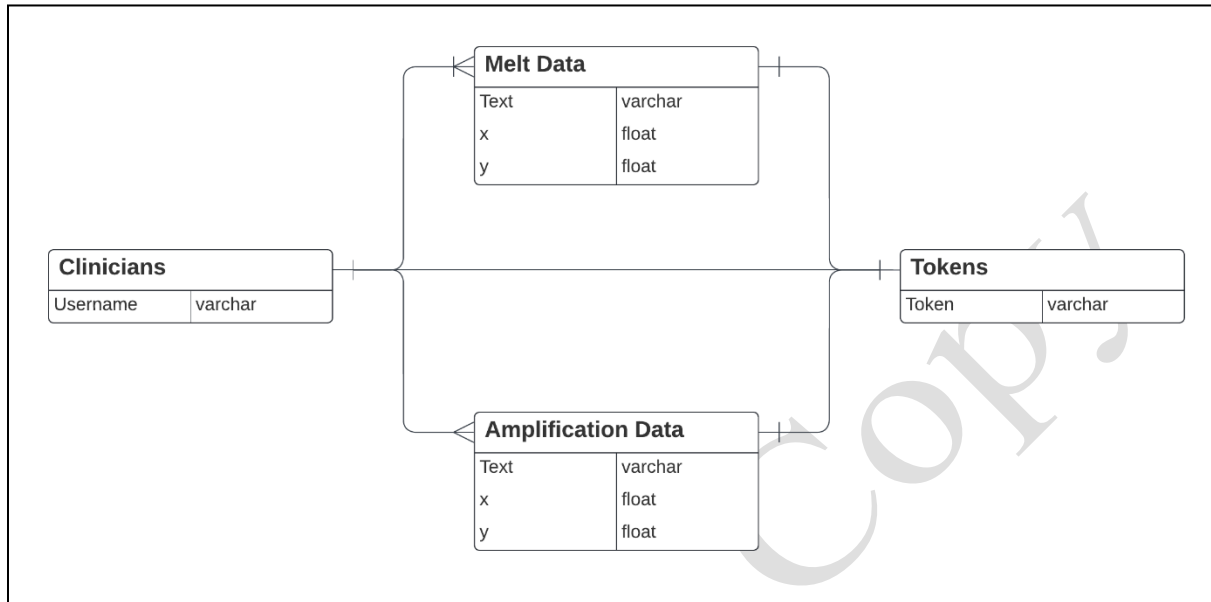


Figure 72: ER diagram of MCI database component

# CHAPTER 10

## TESTING AND RESULTS

### 10.1 TEST DATA

The following unknown patient sample data (fig.73 ) has been used to test the MCIs Deep Learning Model of pathogen classification (fig. ).

Text	X	Y	Text	X	Y	Text	X	Y
1: 930000637 HARSHITHAA EV	70.05	0.028589929	2: 930000637 HARSHITHAA HSV1/2	70.05	0.022799896	3: 930000637 HARSHITHAA VZV	70.05	0.010236624
1: 930000637 HARSHITHAA EV	70.08333333	0.028041001	2: 930000637 HARSHITHAA HSV1/2	70.08333333	0.022326115	3: 930000637 HARSHITHAA VZV	70.08333333	0.010253346
1: 930000637 HARSHITHAA EV	70.11666667	0.027579476	2: 930000637 HARSHITHAA HSV1/2	70.11666667	0.021880992	3: 930000637 HARSHITHAA VZV	70.11666667	0.010263267
1: 930000637 HARSHITHAA EV	70.15	0.027292755	2: 930000637 HARSHITHAA HSV1/2	70.15	0.021493181	3: 930000637 HARSHITHAA VZV	70.15	0.010259582
1: 930000637 HARSHITHAA EV	70.18333333	0.0272489	2: 930000637 HARSHITHAA HSV1/2	70.18333333	0.021185295	3: 930000637 HARSHITHAA VZV	70.18333333	0.010237308
1: 930000637 HARSHITHAA EV	70.21666667	0.027438601	2: 930000637 HARSHITHAA HSV1/2	70.21666667	0.020955773	3: 930000637 HARSHITHAA VZV	70.21666667	0.01019873
1: 930000637 HARSHITHAA EV	70.25	0.027833205	2: 930000637 HARSHITHAA HSV1/2	70.25	0.020797008	3: 930000637 HARSHITHAA VZV	70.25	0.010147948
1: 930000637 HARSHITHAA EV	70.28333333	0.02841337	2: 930000637 HARSHITHAA HSV1/2	70.28333333	0.020702955	3: 930000637 HARSHITHAA VZV	70.28333333	0.010086781
1: 930000637 HARSHITHAA EV	70.31666667	0.029196993	2: 930000637 HARSHITHAA HSV1/2	70.31666667	0.020673823	3: 930000637 HARSHITHAA VZV	70.31666667	0.010007908
1: 930000637 HARSHITHAA EV	70.35	0.030211279	2: 930000637 HARSHITHAA HSV1/2	70.35	0.020711377	3: 930000637 HARSHITHAA VZV	70.35	0.009901722
1: 930000637 HARSHITHAA EV	70.38333333	0.031465539	2: 930000637 HARSHITHAA HSV1/2	70.38333333	0.020817182	3: 930000637 HARSHITHAA VZV	70.38333333	0.009765939
1: 930000637 HARSHITHAA EV	70.41666667	0.032897496	2: 930000637 HARSHITHAA HSV1/2	70.41666667	0.020991976	3: 930000637 HARSHITHAA VZV	70.41666667	0.009627559
1: 930000637 HARSHITHAA EV	70.45	0.034426977	2: 930000637 HARSHITHAA HSV1/2	70.45	0.02123629	3: 930000637 HARSHITHAA VZV	70.45	0.009520903
1: 930000637 HARSHITHAA EV	70.48333333	0.035984487	2: 930000637 HARSHITHAA HSV1/2	70.48333333	0.021542994	3: 930000637 HARSHITHAA VZV	70.48333333	0.009467845
1: 930000637 HARSHITHAA EV	70.51666667	0.037543248	2: 930000637 HARSHITHAA HSV1/2	70.51666667	0.021874317	3: 930000637 HARSHITHAA VZV	70.51666667	0.009440466
1: 930000637 HARSHITHAA EV	70.55	0.039087158	2: 930000637 HARSHITHAA HSV1/2	70.55	0.022184823	3: 930000637 HARSHITHAA VZV	70.55	0.009398398
1: 930000637 HARSHITHAA EV	70.58333333	0.040605443	2: 930000637 HARSHITHAA HSV1/2	70.58333333	0.022444341	3: 930000637 HARSHITHAA VZV	70.58333333	0.009312452
1: 930000637 HARSHITHAA EV	70.61666667	0.042108624	2: 930000637 HARSHITHAA HSV1/2	70.61666667	0.022683753	3: 930000637 HARSHITHAA VZV	70.61666667	0.009198158
1: 930000637 HARSHITHAA EV	70.65	0.04361255	2: 930000637 HARSHITHAA HSV1/2	70.65	0.022949204	3: 930000637 HARSHITHAA VZV	70.65	0.00908222

Figure 73: Melt curve test data

The Model has a test accuracy of 85% and the training accuracy of 86.67%, which looks like, the model doesn't overfit to the data. Since it is a multiclass classification problem, looking on the accuracy is not sufficient.

Tm1	Tstart1	Tend1	Prom1	Width1	AUC1	Tm2	Tstart2	Tend2	Prom2	Width2	AUC2	Target
79.20673	78.53113	79.91713	1.170876	32.23155	5.67836	0	0	0	0	0	0	0
77.82929	76.65565	78.73931	0.727953	34.23747	4.054774	0	0	0	0	0	0	1
76.63901	76.15175	78.04208	0.968974	86.68756	4.119516	0	0	0	0	0	0	2
79.99024	79.01599	80.99954	1.957374	38.33559	11.35592	0	0	0	0	0	0	0
78.94352	77.7458	79.97168	0.357917	59.90459	1.865729	0	0	0	0	0	0	1
77.0693	75.64377	78.63368	1.427196	16.0056	4.268992	0	0	0	0	0	0	2
77.80822	76.57445	78.81159	0.528863	86.52769	2.813339	0	0	0	0	0	0	1
77.50538	76.11217	78.49173	0.367561	75.85445	2.003881	0	0	0	0	0	0	1
79.75506	78.87188	80.38561	1.325147	37.87491	4.653992	0	0	0	0	0	0	0
79.78578	79.16101	80.41159	1.212435	33.88572	4.850704	0	0	0	0	0	0	0
78.91551	77.36373	79.88107	0.478192	85.50107	2.568193	0	0	0	0	0	0	1
78.97358	77.44945	79.92444	1.027488	38.56002	4.743832	0	0	0	0	0	0	1
78.35532	76.76537	78.94676	1.071179	57.32071	4.519926	0	0	0	0	0	0	2
77.78803	76.53383	78.9107	0.739046	61.63718	3.458974	0	0	0	0	0	0	1
79.88083	78.79163	80.64687	2.18468	29.86326	9.572137	0	0	0	0	0	0	0
78.25328	76.79547	79.18695	1.032894	37.4954	4.638876	0	0	0	0	0	0	1
78.072	76.94997	79.00368	0.960863	72.93558	4.479077	0	0	0	0	0	0	1
79.73457	78.92594	80.40963	2.045144	43.18679	9.222937	0	0	0	0	0	0	0
78.46522	77.17707	79.46159	0.291389	52.64943	1.457615	0	0	0	0	0	0	1
80.34924	79.38222	81.06112	1.647746	41.69772	8.321952	0	0	0	0	0	0	0
80.12126	79.16552	80.88382	2.371694	52.29316	11.03641	0	0	0	0	0	0	0
79.92697	79.00546	80.8197	2.055965	26.13481	9.048377	0	0	0	0	0	0	0
77.49257	76.27082	78.46962	0.090408	69.66065	0.255822	0	0	0	0	0	0	1
79.60977	78.62805	80.64108	0.168169	31.80285	0.266516	0	0	0	0	0	0	0
76.80093	75.86558	78.00504	1.964497	56.11184	7.679188	0	0	0	0	0	0	2
78.06685	77.22797	79.10114	0.68536	91.96432	3.806107	0	0	0	0	0	0	1
76.71037	75.85796	77.75554	0.961182	82.54195	3.174506	0	0	0	0	0	0	2

Figure 74: Features of melt curve test data

The true accuracy (fig 75)of the model will be assessed by looking on metrics like precision and recall. Furtherly, on combing both the metrics, f1 score can be taken into consideration, as it is harmonic mean of both precision and recall, will produce a significant and reliable result if the model truly performs good.

```
{'0': {'precision': 1.0, 'recall': 1.0, 'f1-score': 1.0, 'support': 19},
'1': {'precision': 0.8076923076923077,
'recall': 0.9130434782608695,
'f1-score': 0.8571428571428572,
'support': 23},
'2': {'precision': 0.9090909090909091,
'recall': 0.8,
'f1-score': 0.8510638297872342,
'support': 25},
'accuracy': 0.8955223880597015,
'macro avg': {'precision': 0.9055944055944055,
'recall': 0.9043478260869566,
'f1-score': 0.9027355623100304,
'support': 67},
'weighted avg': {'precision': 0.900062623943221,
'recall': 0.8955223880597015,
'f1-score': 0.8953862904323369,
'support': 67}}
```

Figure 75: Accuracy and loss metrics for the MEP\_Model

The confusion matrix for the pathogen classification model (fig.76 ) which classifies the pathogens present in the Meningitidis panel. Here, the model classifies the SP, HI and NM pathogens with less false-positive rates.

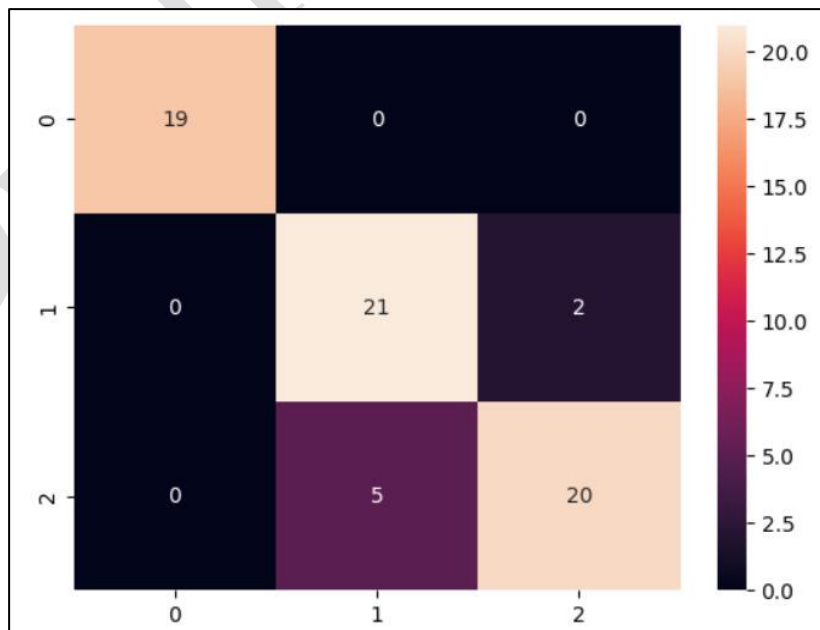


Figure 76: Confusion Matrix for MEP\_model

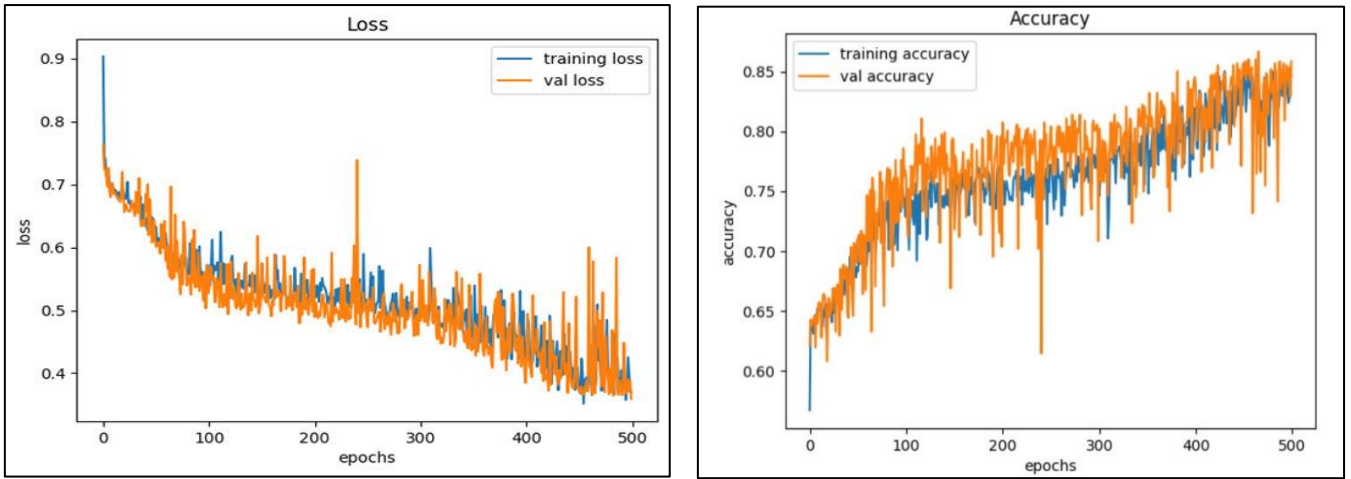


Figure 77: Accuracy and loss metrics for the MEP\_Model

Duplicate Copy

## CONCLUSION

In the current scenario, the diagnosis of infectious diseases is rapidly moving towards molecular assays, with several major biotechnology companies developing ready-to-use molecular kits. However, the reporting of these molecular assays is largely dependent on visual interpretation and analysis by technicians, which has significantly affected the acceptability of these assays in commercial diagnostic setups such as clinical laboratories and hospitals.

This project aims to lay the foundation for developing a framework for automated analysis of molecular assays, which is first-of-its-kind. We have successfully shown that by using predictive analytics and deep learning models on High-Resolution Melting data, several distinct features can be extracted that can be used to develop an algorithm to indicate the presence of the intended molecular target in a clinical sample tested.

This project can be further developed into a full-fledged software that can aid clinicians in diagnosing several diseases and planning the course of treatment. This software has the potential to revolutionize the molecular diagnosis field and improve the digital compatibility of molecular assay interpretation with the existing laboratory information management system.



## REFERENCES

### Bibliography

M. T. Dorak, Ed., *Real-time PCR*. New York: Taylor & Francis Group, 2007. Accessed: Mar. 15, 2023. pp. 1-83. [Online]. Available: <https://www.gene-quantification.de/dorak-book-real-time-pcr-2006.pdf>

S. F. Dobrowolski and C. T. Wittwer, "High-Resolution Melt Profiling," in *Molecular Analysis and Genome Discovery*, R. Rapley, S. Harbron, Eds., 2nd ed. West Sussex, UK: Wiley-Blackwell, 2012. Accessed: Mar. 29, 2023. pp. 81-113. [Online]. doi: 10.1002/9781119977438.ch5.

### Primary Literature

[1] Thermo Fisher Scientific, Inc., Laboratory Information Management Systems [Online]. Available: <https://www.thermofisher.com/in/en/home/digital-solutions/lab-informatics/lab-information-management-systems-lims.html>. [Accessed Apr. 06, 2023].

[2] Thermo Fisher Scientific, Inc., "Lab Software Integration Tools," Available: <https://www.thermofisher.com/in/en/home/digital-solutions/lab-informatics/integration.html>. [Accessed Apr. 06, 2023].

[3] M. L. Bayot, J. E. Lopes, and P. Naidoo, "Clinical Laboratory," *StatPearls - NCBI Bookshelf*, Dec. 19, 2022. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK535358/>. [Accessed Apr. 06, 2023].

[4] B. H. Shirts *et al.*, "Clinical laboratory analytics: Challenges and promise for an emerging discipline," *Journal of Pathology Informatics*, vol. 6, no. 1, p. 9, Feb. 2015, doi: 10.4103/2153-3539.151919.

[5] Agaram, "LIMS for speciality Diagnostic labs," *Agaram Tech*, Jan. 18, 2022. [Online]. Available: <https://www.agaramtech.com/lims-for-specialty-diagnostic-labs/>

[6] G. P. Patrinos, P. B. Danielson, and W. J. Ansorge, "Molecular Diagnostics," in *Elsevier eBooks*, Elsevier BV, 2017, pp. 1–11. doi: 10.1016/b978-0-12-802971-8.00001-8.

[7] R. G. L. Pryor and C. T. Wittwer, "Real-Time Polymerase Chain Reaction and Melting Curve Analysis," in *Humana Press eBooks*, vol. 336, pp. 19–32, Jan. 2006, doi: 10.1385/1-59745-074-x:19.

[8] D. V. Rebrikov and D. Y. Trofimov, "Real-time PCR: A review of approaches to data analysis," *Applied Biochemistry and Microbiology*, vol. 42, no. 5, pp. 455–463, Sep. 2006, doi: 10.1134/s0003683806050024.

[9] L. Garibyan and N. Avashia, "Polymerase Chain Reaction," *Journal of Investigative Dermatology*, vol. 133, no. 3, pp. 1–4, Mar. 2013, doi: 10.1038/jid.2013.1.

- [10] C. Wang *et al.*, “Veterinary PCR Diagnostics,” *BENTHAM SCIENCE PUBLISHERS eBooks*, Mar. 2012, doi: 10.2174/97816080534831120101.
- [11] G. L. Shipley, “Real-Time Quantitative PCR: Theory and Practice,” *Encyclopedia of Molecular Cell Biology and Molecular Medicine*, vol. 11, Sep. 2006, doi: 10.1002/3527600906.mcb.200500012.
- [12] A. Tahamtan and A. Ardebili, “Real-time RT-PCR in COVID-19 detection: issues affecting the results,” *Expert Review of Molecular Diagnostics*, vol. 20, no. 5, pp. 453–454, Apr. 2020, doi: 10.1080/14737159.2020.1757437.
- [13] I. M. Artika, Y. P. Dewi, I. M. Nainggolan, J. E. Siregar, and U. Antonjaya, “Real-Time Polymerase Chain Reaction: Current Techniques, Applications, and Role in COVID-19 Diagnosis,” *Genes*, vol. 13, no. 12, p. 2387, Dec. 2022, doi: 10.3390/genes13122387.
- [14] M. W. Pfaffl, “Quantification strategies in real-time PCR,” in *A-Z of quantitative PCR*, S. A. Bustin, Ed., California, USA: International University Line (IUL), 2004. Accessed: Apr. 15, 2023. pp. 87 – 112. [Online]. Available: <https://www.gene-quantification.de/chapter-3-pfaffl.pdf>
- [15] J. S. Yuan, A. M. Reed, F. Chen, and C. N. Stewart, “Statistical analysis of real-time PCR data,” *BMC Bioinformatics*, vol. 7, no. 1, Feb. 2006, doi: 10.1186/1471-2105-7-85.
- [16] J. L. Montgomery, L. N. Sanford, and C. T. Wittwer, “High-resolution DNA melting analysis in clinical research and diagnostics,” *Expert Review of Molecular Diagnostics*, vol. 10, no. 2, pp. 219–240, Mar. 2010, doi: 10.1586/erm.09.84.
- [17] G. H. Reed, J. Kent, and C. T. Wittwer, “High-resolution DNA melting analysis for simple and efficient molecular diagnostics,” *Pharmacogenomics*, vol. 8, no. 6, pp. 597–608, Jun. 2007, doi: 10.2217/14622416.8.6.597.
- [18] J. S. Farrar and C. T. Wittwer, “High-Resolution Melting Curve Analysis for Molecular Diagnostics,” *Elsevier eBooks*, pp. 79–102, Jan. 2017, doi: 10.1016/b978-0-12-802971-8.00006-7.
- [19] R. H. a. M. Vossen, E. Aten, A. Roos, and J. T. D. Dunnen, “High-Resolution Melting Analysis (HRMA)-More than just sequence variant screening,” in *Human Mutation*, vol. 30, no. 6, pp. 860–866, Jun. 2009, doi: 10.1002/humu.21019.
- [20] J. L. Vaerman, P. Saussoy, and I. Ingargiola, “Evaluation of real-time PCR data.,” *Journal of Biological Regulators and Homeostatic Agents*, vol. 18, no. 2, pp. 212–4, Apr. 2004.
- [21] L. M. Sullivan, J. Weinberg, and J. F. Keaney, “Common Statistical Pitfalls in Basic Science Research,” *Journal of the American Heart Association*, vol. 5, no. 10, Oct. 2016, doi: 10.1161/jaha.116.004142.

- [22] S. Prakash, “Statistical approaches to make sense of data in biology and medicine,” in *Indian Journal of Medical Sciences*, vol. 74, pp. 103–105, Aug. 2022, doi: 10.25259/ijms\_197\_2021.
- [23] M. W. Pfaffl, J. Vandesompele, and M. Kubista, “Data Analysis Software,” in *Real-time PCR: Current Technology and Applications*, J. Logan, J. M. J. Logan, K. J. Edwards, and N. A. Saunders, Eds., Caister Academic Press, 2009. pp. 65 – 83. [Online]. Available: <https://www.gene-quantification.de/Pfaffl-Kubista-Vandesompele-real-time-PCR-chapter-5.pdf>
- [24] QIAGEN GmbH, QIAGEN Strasse 1, D-40724 Hilden. *Rotor-Gene Q User Manual, Version 2*. (2012). Accessed: Apr. 10, 2023. [Online]. Available: <https://www.qiagen.com/us/resources/resourcedetail?id=d29cab50-f102-4faa-b453-4a57463610fa&lang=en>
- [25] QIAGEN GmbH, QIAGEN Strasse 1, D-40724 Hilden. *Rotor-Gene ScreenClust HRM Software User Guide*. Accessed: Apr. 10, 2023. [Online]. Available: <https://www.qiagen.com/cn/resources/download.aspx?id=af33be05-14c6-4ac3-ace2-d85aa7ad0434&lang=en>
- [26] Bio-Rad Laboratories, Inc., Hercules, California, USA. *CFX96™ and CFX384™ Real-Time PCR Detection Systems Instruction Manual*. Accessed: Apr. 10, 2023. [Online]. Available: <https://www.bio-rad.com/sites/default/files/webroot/web/pdf/lsr/literature/10010424.pdf>
- [27] BIO MOLECULAR SYSTEMS (BMS), Upper Coomera QLD 4209, Australia. *MIC, User Manual, Version 1.2*. Accessed: Apr. 10, 2023. [Online]. Available: <https://biomolecularsystems.com/mic-qpcr/software/>
- [28] Thermo Fisher Scientific, Inc., Waltham, Massachusetts, USA. *QuantStudio™ 5 Real-Time PCR Instrument*,