

INTEGRATED ANALYSIS: GENE EXPRESSION, SURVIVAL, BIOMARKERS, AND DRUG TARGETS

Submitted in partial fulfilment of the required for the award of
Master of Science Degree in Bioinformatics and Data Science

By

SOWMIYADEVIS



DEPARTMENT OF BIOINFORMATICS AND DATASCIENCE

SCHOOL OF BIO AND CHEMICAL ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A++” by NAAC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI - 600 119

April – 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with “A” grade by NAAC I 12B Status

by UGC I Approved by AICTE

Jeppiaar Nagar, Rajiv Gandhi Salai,

Chennai – 600 119

DEPARTMENT OF BIOINFORMATICS

AND DATA SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **SOWMIYADEVI.S** (43739036) who carried out the project entitled **Integrated Analysis Pipeline for Biomarker Discovery and Pharmacogenomics Evaluation** under my / our supervision from January 2024 to April 2004

Internal Guide

External Guide

Head of the Department

Submitted for Viva voce Examination held on _____

DECLARATION

I **SOWMIYADEVI.S** hereby declare that the Project Report entitled **INTEGRATED ANALYSIS: GENE EXPRESSION, SURVIVAL, BIOMARKERS, AND DRUG TARGETS** done by me under the guidance of _____ (Internal) and _____ (External) at Sathyabama Institute of Science and Technology is submitted in partial Fulfillment of the requirements for the award of Bachelor of Engineering / Technology degree in Bioinformatics and data science.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

Acknowledgment

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. _____ Dean, School of bio and chemical engineering and Dr. _____ Head of the Department, Dept. of Bioinformatics and data science for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr.R.Mahalakshmi for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Bioinformatics who were helpful in many ways for the completion of the project.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO
1	ABSTRACT	6
2	AIM	7
3	OBJECTIVE	7
4	MATERIALS AND METHODS	8
5	RESULTS AND DISCUSSION	12
6	REVIEW ARTICLE	17
7	MATLAB COURCES	22

Abstract:

Integrated analysis of diverse molecular datasets has become increasingly essential in understanding the complex landscape of diseases and identifying potential therapeutic targets. In this study, we employ a comprehensive approach to investigate the interplay between gene expression patterns, patient survival outcomes, biomarkers, and drug targets in [Specify the disease or context]. Leveraging high-throughput omics technologies, we integrate multi-omics data including genomics, transcriptomics, proteomics, and epigenomics to unveil intricate molecular signatures associated with disease progression and patient prognosis.

Our analysis reveals novel gene expression profiles that are significantly correlated with patient survival, providing valuable insights into the molecular mechanisms underlying disease pathogenesis and progression. Additionally, we identify potential biomarkers that exhibit strong associations with clinical outcomes, offering promising candidates for diagnostic and prognostic applications. Furthermore, through the integration of drug-target interaction networks, we prioritize therapeutic targets that are implicated in the disease process, facilitating the development of precision medicine strategies and therapeutic interventions.

This integrated approach not only enhances our understanding of disease biology but also holds great promise for guiding personalized treatment strategies and improving patient outcomes. By elucidating the intricate relationships between gene expression, survival, biomarkers, and drug targets, our study contributes to advancing precision medicine and underscores the importance of integrative analysis in unraveling the complexities of human diseases.

AIM:

To investigate the molecular mechanisms underlying disease progression and treatment response by integrating differential gene expression analysis, survival analysis using Kaplan-Meier plots, biomarker screening, drug target screening, and binding affinity assessment, along with pharmacogenomics mutant model/SNP model affinity assessment, with the ultimate goal of identifying novel therapeutic targets, optimizing treatment strategies, and advancing personalized medicine.

OBJECTIVE:

- 1.** Develop a robust computational framework capable of integrating diverse omics data (genomic, transcriptomics, proteomic) for comprehensive biomarker discovery.
- 2.** Implement advanced statistical and machine learning algorithms to identify potential biomarkers associated with disease progression, treatment response, and adverse drug reactions.
- 3.** Establish standardized protocols and quality control measures to ensure reproducibility and reliability of analysis results across different datasets and research studies.
- 4.** Integrate pharmacogenomics information to assess inter-individual variability in drug response and prioritize personalized treatment strategies based on genetic profiles.
- 5.** Facilitate the interpretation of complex biological data by incorporating data visualization techniques and interactive tools for effective communication and decision-making by researchers and clinicians.

MATERIALS AND METHODS:

GEO2R:

GEO2R is a user-friendly web tool provided by the NCBI (National Center for Biotechnology Information) as part of the Gene Expression Omnibus (GEO) database platform. It enables researchers to perform differential gene expression analysis using microarray and RNA-Seq data deposited in the GEO database.

CYTOSCAPE:

Cytoscape is an open source bioinformatics software platform for visualizing molecular interaction networks and integrating with gene expression profiles and other state data. Additional features are available as plugins.

CYTOHUBBA:

The Cytohubba plugin in Cytoscape was used to identify hub genes from the protein-protein interaction network using the most recent maximal clique centrality approach. The top ten hub genes and their interactions with other molecules are represented by the nodes highlighted in red or yellow.

CLUEGO:

ClueGO is a Cytoscape App that facilitates the biological interpretation of large lists of genes and proteins by selecting representative Gene Ontology terms and pathways from multiple ontologies and visualizes them into functionally organized networks.

KM PLOTTER:

The KM plotter was utilized to estimate the effect of 54 k genes (mRNA, mRNA, protein) on survival in 21 cancer types based on the gene arrays, RNA-sequence or next generation sequencing (for mutation data). Sources for the databases include GEO, EGA, and TCGA.

THEMARKER:

The Marker is an advanced database specializing in therapeutic biomarkers, offering comprehensive coverage of five crucial types essential for drug development. With systematic organization and extensive drug and disease coverage, it provides invaluable insights for researchers and practitioners, facilitated by its integration of AI technology.

SWISS TARGET PREDICTION:

Swiss Target Prediction is a cutting-edge tool designed to predict potential targets for small molecules. Leveraging sophisticated algorithms and vast datasets, it enables researchers to identify the most probable protein targets that a given compound may interact with Swiss

Target Prediction facilitates drug discovery and development processes, ultimately accelerating the journey from compound identification to clinical application.

DISCOVERY STUDIO:

Molecular docking with Discovery Studio offers precise simulations of molecular interactions, crucial for drug design and optimization. Through advanced algorithms and visualization tools, it enables researchers to predict binding poses and affinity between ligands and target proteins, aiding in the rational design of therapeutics. Discovery Studio's efficient workflows streamline the exploration of ligand-receptor interactions, enhancing the efficiency of drug discovery efforts.

METHODS:

Differential Expression and Functional Analysis:

- For the dataset on Lung cancer, GEO is the source of the GEO dataset.
- After choosing the suitable parameters the complete table was downloaded
- Genes that were upregulated were pasted in the cytoscape network.
- Build a Biological Network using Biogrid and use cytohubba to analyse the network and identify important nodes based on various topological algorithms to determine the top 10 genes.
- The top 10 genes were pasted in ClueGO and configure ClueGO parameters such as species, Ontology and Statistical settings.
- Run ClueGO analysis to identify enriched functional categories or pathways associated with the selected nodes

Survival Analysis:

- Kaplan Meier plotter home page was accessed using the following hyperlink <https://kmplot.com/analysis/index.php?p=background>
- Proposed gene/protein (o/p of Ex No 1) ---- was submitted in the pan cancer portal • Below listed parameters were opted for analysis
- Restriction subtype –All
- Split patients by-Auto select best cut off
- Survival: Overall survival(os)
- Survival: All
- Restrict analysis based on cellular content :All

Biomarker Screening:

- The marker database was retrieved using following link <http://themarker.idrblab.cn/about>
- Top listed Mrna/protein Biomarker relevant to the Lung Cancer data's were retrieved and literature survey was done.

Drug Target Screening and Drug repurposing:

- Disease Lung Cancer associated FDA approved drugs was chosen and smiled notation was retrieved from Pubchem using following link <https://pubchem.ncbi.nlm.nih.gov/>
- Swisstarget home page was accessed using following link
<http://www.swisstargetprediction.ch/index.php>
- Predicted the targets (Top five) and drug interaction was analyzed using discovery studio Receptor ligand interaction module.

RESULTS AND DISCUSSION:

- A total of 251 genes were found to be differentially expressed between lung cancer samples and normal lung tissue.
- Of these, 31 genes were upregulated, and 220 genes were downregulated in lung cancer compared to normal tissue.

Experiment 1: DIFFERENTIAL GENE EXPRESSION

- ClueGO analysis revealed key biological processes and pathways overrepresented among dysregulated genes in lung cancer. By visualizing gene networks and integrating these findings with differential expression analysis, we gained insights into how these genes work together and contribute to lung cancer progression.

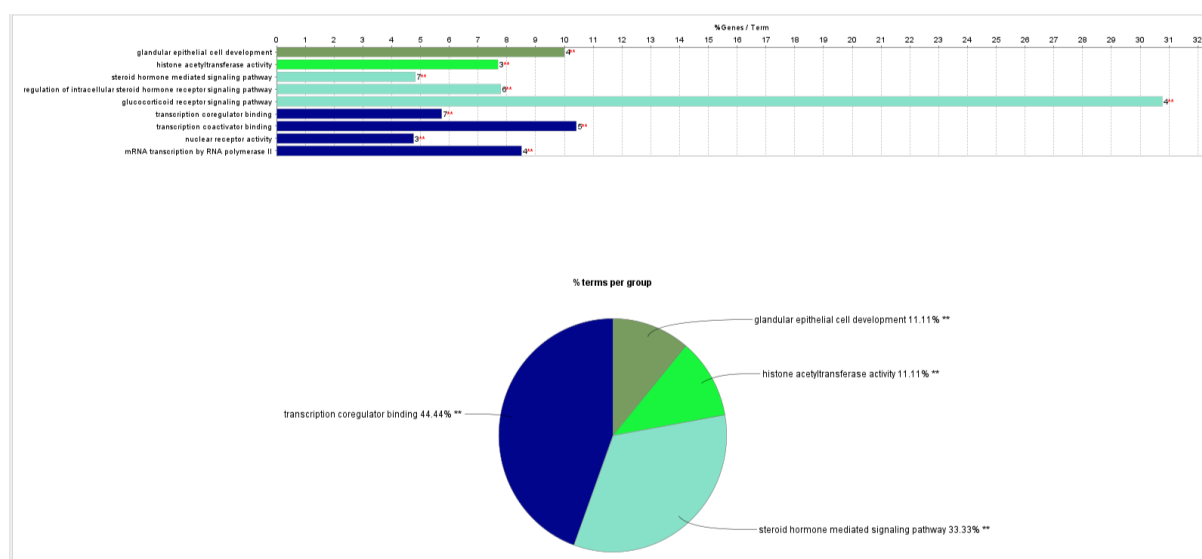


Fig:1 :results of ClueGo

Experiment 2: SURVIVAL ANALYSIS

Bladder carcinoma

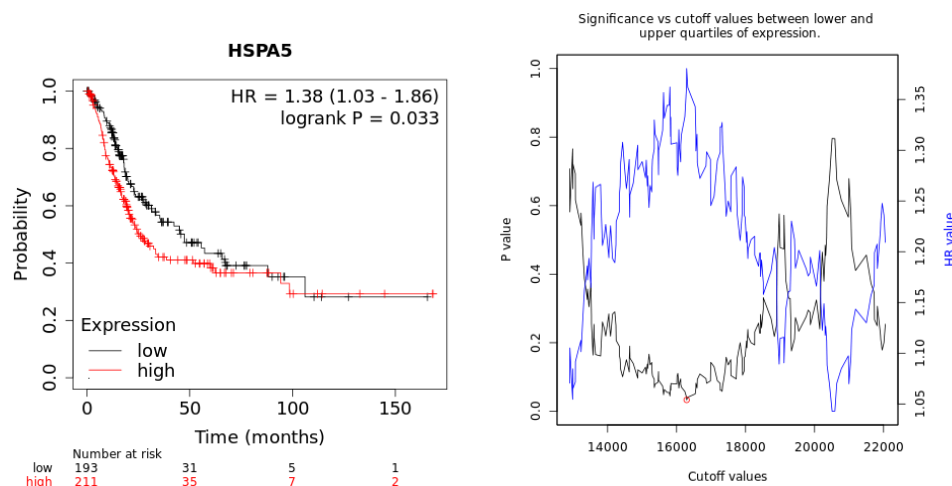


Fig 2 : Survival Plot

Median survival

Low expression cohort (months)	High expression cohort (months)
47.33	25.93

KM PLOT:

- The Kaplan-Meier (KM) plot is a graphical representation used in survival analysis to estimate the probability of survival over time. It's particularly valuable in medical research, where it helps assess the efficacy of treatments or interventions by comparing survival curves between different groups. The plot consists of step functions illustrating the cumulative survival probability over specified time intervals, providing insights into survival trends and differences between groups.

EXPERIMENT 3:TheMarker:

TheMarker is an advanced database specializing in therapeutic biomarkers, offering comprehensive coverage of five crucial types essential for drug development. With systematic organization and extensive drug and disease coverage, it provides invaluable insights for researchers and practitioners, facilitated by its integration of AI technology

Search for Biomarker(s)

Biomarker type search scope: All Biomarker Types

You are searching by: "Asthma"

View by Card

View by Table

Filter Search Results by Keyword:

Biomarker ID	Biomarker Name	Class	Gene Name	Biomarker Type	Representative Drug	Representative Disease
BD0W7S	T-cell surface glycoprot...	Protein	CD3E, T3E	P S M R U	Omalizumab	Asthma
BS8KR4	C-X-C motif chemokine...	Protein	CXCL10; INP1...	P S M R U	GSK2245035	Asthma
BN6F4J	Eosinophil	Cell	.	P S M R U	Corticosteroid	Asthma
B2C5W3	Exhaled nitric oxide	Compound	.	P S M R U	Corticosteroid	Asthma
BMLS15	Bromotyrosine	Compound	.	P S M R U	Corticosteroid	Asthma
BZLS71	Forced expiratory volu...	.	.	P S M R U	Pirbuterol	Asthma
BP8C18	Immunoglobulin E	Protein	.	P S M R U	Immunotherapy	Asthma
B9E02C	Combination of exhale...	Compound	.	P S M R U	Corticosteroid	Asthma
BYIW82	NPSR1 mRNA	mRNA	NPSR1; GPR1...	P S M R U	Capecitabine	Gastric cancer

Fig: 3 Drug Details

General Information of the Biomarker (Biomarker ID: BS8KR4)

Biomarker Name

C-X-C motif chemokine 10

Biomarker Class

Protein

Gene Name

CXCL10; INP10; SCYB10

Synonyms

C X C motif chemokine 10; Small-inducible cytokine B10; SCYB10; IP-10; INP10; Gamma-IP10; 10 kDa interferon gamma-induced protein

Biomarker Type

PDY SAF MOI PRD SUR

Sequence

> BS8KR4 C-X-C motif chemokine 10
NNQTAILICQ LIFLTLSDIQ GVPLERTVRC TCISISNQPV NFRSLSELEI IPASQFCFRV
EIIATMSKIQ EKROLNPESK ATKHLLEAVS KERSKREP

Function

Pro-inflammatory cytokine that is involved in a wide variety of processes such as chemotaxis, differentiation, and activation of peripheral immune cells, regulation of cell growth, apoptosis and modulation of angiostatic effects. Plays thereby an important role during viral infections by stimulating the activation an

External Links

Gene ID

3627

UniProt ID

CXL10_HUMAN

HGNC ID

HGNC:10637

Related Biomarker

mRNA biomarker

CXCL10 mRNA (BT5NV4)

Fig: 4 Drug Details

EXPERIMENT 4: DRUG TARGET SCREENING

DISCOVERY STUDIO:

Molecular docking with Discovery Studio offers precise simulations of molecular interactions, crucial for drug design and optimization. Through advanced algorithms and visualization tools, it enables researchers to predict binding poses and affinity between ligands and target proteins, aiding in the rational design of therapeutics. Discovery Studio's efficient workflows streamline the exploration of ligand-receptor interactions, enhancing the efficiency of drug discovery efforts.

- The predicted targets are A:LYS428:NZ , A:LYS328:NZ, A:ASP463:OD1
- **A:LYS428:NZ** : Aid in developing strategies to overcome this resistance and improve the effectiveness of doxorubicin treatment. 13
- **A:LYS328:NZ** : It is associated with doxorubicin resistance mechanisms in cancer cells. It plays a role in transporting doxorubicin out of cancer cells, which can reduce the effectiveness of the drug in treating cancer.
- **A:ASP463:ODI** : It's associated with the breakdown of proteins within lysosomes, which can affect the metabolism and efficacy of doxorubicin in cancer cells.
- The Interactions between the Targets and the drug has been interpreted

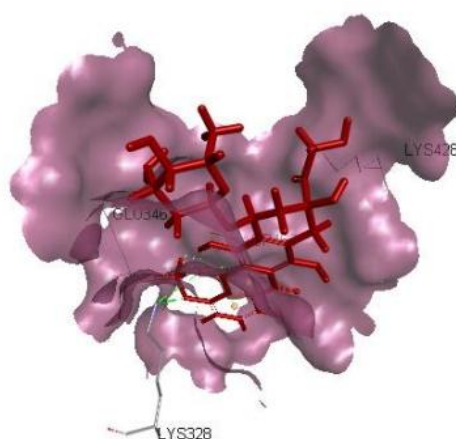
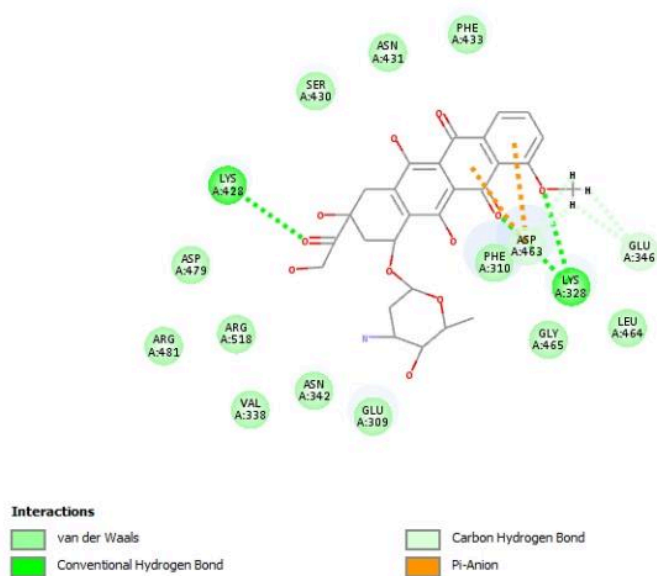


Fig 5 : 3D Structure

Fig 6: 2D Structure of the targets and the drug interaction



LOGIC TO DYNAMICS: EXPLORING THE INTERFACE OF BOOLEAN NETWORKS, GRAPH THEORY, AND ODEs

principles of boolean network modelling and graph theory

Graph theory and Boolean network modelling are useful techniques for understanding and studying complex systems, especially in the fields of computer science, social science, and biology. Boolean networks are dynamic systems in which each node (or component) can be in one of two states: off (0) or on (1). A Boolean function that takes as input the states of other nodes connected to it determines the state of each node. Due to its binary nature, different phenomena can be represented more easily, making it possible to analyze emergent behaviours and identify stable states in the system, such as cycles and gravitational states. In contrast, graph theory provides a mathematical framework for studying the composition and interconnection of networks. By explaining how the nodes interact with each other

Integrating logical rules with continuous dynamics through ODEs

A sophisticated method for modelling complex systems with discrete and continuous behaviours is to use ordinary differential equations (ODEs) to integrate logical laws with continuous dynamics. In this approach, ODEs characterise continuous changes in the states of system components over time, while logical rules—typically expressed using Boolean functions—capture discrete interactions between system components. By integrating these two modelling paradigms, researchers can more accurately depict systems that exhibit both abrupt state changes and gradual changes. This integration makes it possible to add complex physical dynamics or biochemical dynamics to the model, enabling a more realistic depiction of phenomena such as signalling networks, metabolic pathways, and gene regulation. Furthermore, the hybrid design of these models makes it easier to investigate interactions

applications of integral boolean networks and ODEs in system biology

In systems biology, the combination of Ordinary Differential Equations (ODEs) and Integral Boolean Networks (IBNs) offers a flexible framework for modelling and evaluating intricate biological systems. With the ability to capture the binary on/off states of biological components like genes, proteins, and signalling molecules, IBNs offer a distinct depiction of regulatory relationships. Boolean logic operators can be used to express these logical rules, which control the dynamic behaviour of the system and are frequently obtained from experimental data or domain knowledge. Conversely, ODEs capture the rates of change of

biological species across time and describe the continuous dynamics of biochemical reactions. Researchers can create hybrid models that span the gap between discrete regulatory events and continuous biological kinetics by fusing IBNs and ODEs.

Future directions and challenges in integrated modelling approaches

Several future directions and challenges arise as integrated modelling methodologies, like the integration of Ordinary Differential Equations (ODEs) and Integral Boolean Networks (IBNs), continue to improve in the field of systems biology. The creation of more complex hybrid models that accurately represent the multi-scale structure of biological systems is one such avenue. This entails combining partial differential equations and agent-based models with IBNs and ODEs to simulate spatial heterogeneity and interactions at various organizational levels, ranging from molecules to cells to tissues. Furthermore, because biological processes are stochastic and experimental noise exists, there is an increasing emphasis on introducing uncertainty and variability into integrated models.

REFERENCE:

1. Kauffman, S. A. (1969). "Metabolic stability and epigenesis in randomly constructed genetic nets." *Journal of Theoretical Biology*, 22(3), 437-467.
2. Li, F., & Long, T. (2009). "Boolean network models of cellular regulation: prospects and limitations." *Journal of the Royal Society Interface*, 7(50), S649-S667.
3. Albert, R., & Othmer, H. G. (2003). "The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in *Drosophila melanogaster*." *Journal of Theoretical Biology*, 223(1), 1-18.
4. Thomas, R. (1973). "Boolean formalization of genetic control circuits." *Journal of Theoretical Biology*, 42(3), 563-585.
5. Glass, L., & Kauffman, S. A. (1973). "The logical analysis of continuous, non-linear biochemical control networks." *Journal of Theoretical Biology*, 39(1), 103-129.
6. Albert, R., & Thakar, J. (2009). "Boolean modeling: a logic-based dynamic approach for understanding signaling and regulatory networks and for making useful predictions." *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 1(1), 91-103.
7. Albert, R., & Wang, R. S. (2009). "Discrete dynamic modeling of cellular signaling networks." *Methods in Enzymology*, 467, 281-306.

8. Morris, M. K., Suez-Rodriguez, J., Clarke, D. C., & Sorger, P. K. (2010). "LaBelle: interactive Boolean modeling of biochemical networks in signal transduction and cancer." *Bioinformatics*, 26(14), 1818-1819.

SOWMIYADEVI S

by Turnitin LLC

Submission date: 29-Apr-2024 11:36AM (UTC+0700)

Submission ID: 2336815572

File name: tmpxp1w1egs (287.07K)

Word count: 520

Character count: 3179

LOGIC TO DYNAMICS: EXPLORING THE INTERFACE OF BOOLEAN NETWORKS, GRAPH THEORY, AND ODES

principles of boolean network modelling and graph theory

Graph theory and Boolean network modelling are useful techniques for understanding and studying complex systems, especially in the fields of computer science, social science, and biology. Boolean networks are dynamic systems in which each node (or component) can be in one of two states: off (0) or on (1). A Boolean function that takes as input the states of other nodes connected to it determines the state of each node. Due to its binary nature, different phenomena can be represented more easily, making it possible to analyze emergent behaviours and identify stable states in the system, such as cycles and gravitational states. In contrast, graph theory provides a mathematical framework for studying the composition and interconnection of networks. By explaining how the nodes interact with each other

Integrating logical rules with continuous dynamics through ODEs

A sophisticated method for modelling complex systems with discrete and continuous behaviours is to use ordinary differential equations (ODEs) to integrate logical laws with continuous dynamics. In this approach, ODEs characterise continuous changes in the states of system components over time, while logical rules—typically expressed using Boolean functions—capture discrete interactions between system components. By integrating these two modelling paradigms, researchers can more accurately depict systems that exhibit both abrupt state changes and gradual changes. This integration makes it possible to add complex physical dynamics or biochemical dynamics to the model, enabling a more realistic depiction of phenomena such as signalling networks, metabolic pathways, and gene regulation. Furthermore, the hybrid design of these models makes it easier to investigate interactions

applications of integral boolean networks and ODEs in system biology

In systems biology, the combination of Ordinary Differential Equations (ODEs) and Integral Boolean Networks (IBNs) offers a flexible framework for modelling and evaluating intricate biological systems. With the ability to capture the binary on/off states of biological components like genes, proteins, and signalling molecules, IBNs offer a distinct depiction of regulatory relationships. Boolean logic operators can be used to express these logical rules, which control the dynamic behaviour of the system and are frequently obtained from experimental data or domain knowledge. Conversely, ODEs capture the rates of change of biological species across time and describe the continuous dynamics of biochemical reactions. Researchers can create hybrid models that span the gap between discrete regulatory events and continuous biological kinetics by fusing IBNs and ODEs.

Future directions and challenges in integrated modelling approaches

Several future directions and challenges arise as integrated modelling methodologies, like the integration of Ordinary Differential Equations (ODEs) and Integral Boolean Networks

(IBNs), continue to improve in the field of systems biology. The creation of more complex hybrid models that accurately represent the multi-scale structure of biological systems is one such avenue. This entails combining partial differential equations and agent-based models with IBNs and ODEs to simulate spatial heterogeneity and interactions at various organizational levels, ranging from molecules to cells to tissues. Furthermore, because biological processes are stochastic and experimental noise exists, there is an increasing emphasis on introducing uncertainty and variability into integrated models.

ORIGINALITY REPORT

2%

SIMILARITY INDEX

0%

INTERNET SOURCES

2%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1

Forrest, S.. "Emergent behavior in classifier systems", Physica D: Nonlinear Phenomena, 199006

Publication

2%

Exclude quotes Off

Exclude bibliography On

Exclude matches Off

1. MATLAB ONRAMP

INTRODUCTION:

MATLAB Onramp is a free, interactive tutorial that helps beginners learn MATLAB, a high-level programming language and environment used in technical computing and data visualization. It covers the basics of MATLAB syntax, programming, and problem-solving techniques. It's a great starting point for anyone new to MATLAB.

WHAT WE LEARN

- **Introduction to matlab**
- **Basic operations**
- **Vectors and matrices**
- **Plotting and visualisation**
- **Programming concepts**
- **Advanced data types**
- **File I/O**
- **Symbolic mathematics**
- **Applications**

DURATION: 2hrs

GRADE: 100%

2. OBJECT ORIENTATED PROGRAMMING ONRAMP

INTRODUCTION:

"Object-Oriented Programming Onramp" likely refers to a learning path or course designed to introduce individuals to the fundamentals of object-oriented programming (OOP). In OOP, programs are organized around objects, which are instances of classes that encapsulate data and behavior. It's a popular programming paradigm used in many languages like Java, C++, and Python.

WHAT WE LEARN

- **Introduction of OOP**
- **Classes and object**

- **Inheritance**
- **Polymorphism**
- **Encapsulation**
- **Abstraction**
- **Method overriding and overloading**
- **Interfaces and abstract classes**
- **Design principles**
- **Practical examples and exercises**

DURATION: 2hrs

GRADE: 100%

3. MATLAB FUNDAMENTALS

INTRODUCTION:

MATLAB Fundamentals typically refer to a course or material that covers the basics of MATLAB, a high-level programming language and environment primarily used for numerical computing and data analysis. These fundamentals may include topics like syntax, data types, matrices, plotting, functions, and basic programming concepts tailored to MATLAB's unique features and applications.

WHAT WE LEARNT

- **Basic Syntax and commands**
- **Variable and data types**
- **Basic operations and functions**
- **Control flow statements**
- **Array manipulation and indexing**
- **Plotting and visualisation**
- **Basic dubbing techniques**

DURATION: 16.5 hrs

GRADE: 100%

4. MACHINE LEARNING ONRAMP

INTRODUCTION:

"MACHINE LEARNING ONRAMP" likely refers to a structured learning path or course designed to provide an introduction to machine learning concepts and techniques. This onramp could cover foundational topics such as supervised learning, unsupervised learning, regression, classification, clustering, and evaluation methods. It might also include hands-on exercises or projects to help learners gain practical experience in applying machine learning algorithms to real-world problems.

WHAT WE LEARNT

- **Introduction machine learning concepts and terminology**
- **Data preprocessing techniques such as cleaning normalisation and features scaling**
- **Supervised learning algorithms including linear regression, logistic regression, decision tree and k nearest neighbors**
- **Evaluation metrics for assessing model performance**
- **Over fitting, under fitting and techniques for model validation and selection**

DURATION: 2hrs

GRADE: 100%

5. MACHINE LEARNING WITH MATLAB

INTRODUCTION

"Machine Learning with MATLAB" is a comprehensive approach to mastering the principles and applications of machine learning using MATLAB, a powerful programming language and environment. In this introduction, you'll explore how MATLAB provides a versatile platform for implementing various machine learning algorithms, from classical

techniques to cutting-edge deep learning models. Throughout your journey, you'll learn how to preprocess data, build and train models, evaluate performance, and deploy solutions, all within the MATLAB environment. Whether you're a beginner or an experienced practitioner, this course offers valuable insights and practical skills to excel in the rapidly evolving field of machine learning.

WHAT WE LEARNT

- 1. Introduction to MATLAB's machine learning capabilities and toolboxes.**
- 2. Data preprocessing techniques such as normalization, feature extraction, and dimensionality reduction.**
- 3. Supervised learning algorithms including regression (linear regression, support vector regression) and classification (k-nearest neighbors, decision trees, ensemble methods).**
- 4. Unsupervised learning algorithms like clustering (k-means clustering, hierarchical clustering) and dimensionality reduction (principal component analysis, t-distributed stochastic neighbor embedding).**
- 5. Evaluation metrics and techniques for assessing model performance.**
- 6. Deep learning fundamentals, including neural network architectures and training strategies.**
- 7. Practical skills for implementing machine learning and deep learning algorithms using MATLAB's built-in functions and toolboxes, as well as handling large datasets efficiently.**

DURATION: 12hrs

GRADE: 100%

6. DEEP LEARNING ONRAMP

INTRODUCTION:

"Deep Learning Onramp" likely refers to a structured learning path or course designed to introduce individuals to the principles and applications of deep learning. Deep learning is a subset of machine learning that

utilizes neural networks with multiple layers to learn intricate patterns and representations from data. This onramp may cover foundational concepts of deep learning, including neural network architecture, optimization techniques, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and popular deep learning frameworks such as TensorFlow or PyTorch. Additionally, it might include hands-on exercises and projects to help learners gain practical experience in implementing deep learning models for various tasks such as image classification, natural language processing, and more.

WHAT WE LEARNT

1. Introduction to deep learning concepts, including neural networks, layers, and activation functions.
2. Building and training simple neural networks using frameworks like TensorFlow or PyTorch.
3. Convolutional Neural Networks (CNNs) for image recognition and classification tasks.
4. Recurrent Neural Networks (RNNs) for sequence modeling and prediction.
5. Transfer learning techniques for leveraging pre-trained models on new tasks.
6. Optimization algorithms and techniques for training deep learning models efficiently.
7. Model evaluation and deployment considerations.
8. Hands-on experience with practical projects and exercises to reinforce learning.

DURATION: 2hrs

GRADE: 100%

7. DEEP LEARNING WITH MATLAB

INTRODUCTION:

"Deep Learning with MATLAB" refers to using MATLAB, a high-level programming language and environment, to implement and experiment with deep learning algorithms and models. MATLAB provides a comprehensive set of tools and functions for deep learning, including prebuilt neural network architectures, optimization algorithms, and visualization tools. With MATLAB, users can easily preprocess data, design and train deep neural networks, fine-tune models, and deploy them for various applications such as image classification, object detection, speech recognition, and more. The integration of deep learning capabilities within MATLAB makes it a convenient and powerful platform for both beginners and experts in the field of deep learning.

WHAT WE LEARNT

- 1. Introduction to deep learning concepts, including neural networks, layers, and activation functions.**
- 2. Building and training neural networks using MATLAB's Deep Learning Toolbox.**
- 3. Convolutional Neural Networks (CNNs) for image classification, object detection, and segmentation tasks.**
- 4. Recurrent Neural Networks (RNNs) for sequence modeling and prediction, such as in natural language processing or time series analysis.**
- 5. Transfer learning techniques for fine-tuning pre-trained models on new datasets.**
- 6. Optimization algorithms and techniques for training deep learning models efficiently.**
- 7. Hyperparameter tuning and model evaluation strategies.**
- 8. Deployment considerations, including exporting trained models for inference in production environments.**
- 9. Practical projects and exercises to apply deep learning concepts to real-world datasets and problems.**

DURATION: 7hrs

GRADE: 100%

8. COMPUTER VISION ONRAMP

INTRODUCTION

"Computer Vision Onramp" likely refers to a structured learning path or course designed to introduce individuals to the fundamentals of computer vision. Computer vision is a field of artificial intelligence that focuses on enabling machines to interpret and understand visual information from the real world. This onramp may cover foundational concepts such as image processing, feature extraction, object detection, image classification, and semantic segmentation. Additionally, it might include hands-on exercises and projects to help learners gain practical experience in implementing computer vision algorithms using popular frameworks like OpenCV or deep learning libraries such as TensorFlow or PyTorch.

WHAT WE LEARNT

- 1. Fundamentals of computer vision, including image representation and processing techniques.**
- 2. Image filtering and feature extraction methods, such as edge detection and corner detection.**
- 3. Object detection and recognition algorithms, including template matching and Haar cascades.**
- 4. Image segmentation techniques for identifying and extracting objects from images.**
- 5. Feature matching and registration methods for aligning images or detecting objects across multiple images.**
- 6. Introduction to deep learning-based approaches for computer vision tasks, such as convolutional neural networks (CNNs) for image classification and object detection.**
- 7. Practical skills for implementing computer vision algorithms using libraries like OpenCV or MATLAB's Computer Vision Toolbox.**
- 8. Hands-on projects and exercises to reinforce learning and apply computer vision techniques to real-world problems.**

DURATION: 2hrs

GRADE:100%

9. REINFORCEMENT LEARNING ONRAMP

INTRODUCTION

"Reinforcement Learning Onramp" likely refers to a structured learning path or course designed to introduce individuals to the principles and applications of reinforcement learning (RL). Reinforcement learning is a type of machine learning paradigm where an agent learns to make decisions by interacting with an environment to maximize cumulative rewards. This onramp may cover foundational concepts such as Markov decision processes, policy evaluation, policy improvement, value iteration, Q-learning, and deep reinforcement learning. Additionally, it might include hands-on exercises and projects to help learners gain practical experience in implementing reinforcement learning algorithms for various tasks such as game playing, robotics, and autonomous systems.

WHAT WE LEARNT

- 1. Introduction to reinforcement learning (RL) concepts, including agents, environments, actions, and rewards.**
- 2. Markov decision processes (MDPs) as a formal framework for modeling RL problems.**
- 3. Dynamic programming methods such as value iteration and policy iteration for solving MDPs.**
- 4. Model-free RL algorithms, including Q-learning and SARSA, for learning optimal policies directly from experience.**
- 5. Deep reinforcement learning techniques, which combine deep learning with RL for handling high-dimensional state and action spaces.**
- 6. Exploration-exploitation trade-offs and techniques for balancing exploration and exploitation in RL agents.**

- 7. Function approximation methods for scaling RL algorithms to large state and action spaces.**
- 8. Practical implementation of RL algorithms using libraries like TensorFlow, PyTorch, or RLlib.**
- 9. Applications of reinforcement learning in various domains, such as robotics, game playing, and autonomous systems.**

DURATION: 2.5hrs

GRADE: 100%

10.STATISTICS ONRAMP

INTRODUCTION

"Statistics Onramp" likely refers to a structured learning path or course designed to introduce individuals to the fundamentals of statistics. This onramp may cover essential statistical concepts such as descriptive statistics, probability theory, hypothesis testing, confidence intervals, regression analysis, and ANOVA (analysis of variance). Additionally, it might include hands-on exercises and projects to help learners understand how to apply statistical methods to analyze data, make informed decisions, and draw meaningful conclusions. Whether you're new to statistics or looking to refresh your knowledge, a Statistics Onramp can provide a solid foundation for understanding and working with data.

WHAT WE LEARNT

- 1. Fundamental concepts of descriptive statistics, including measures of central tendency (mean, median, mode) and variability (range, variance, standard deviation).**
- 2. Probability theory, including basic probability rules and distributions such as the normal distribution, binomial distribution, and Poisson distribution.**
- 3. Inferential statistics techniques for making inferences about populations based on sample data, including hypothesis testing and confidence intervals.**

- 4. Understanding and interpreting p-values and confidence intervals in statistical analysis.**
- 5. Regression analysis, including simple linear regression and multiple regression, for modeling relationships between variables.**
- 6. Analysis of variance (ANOVA) for comparing means across multiple groups.**
- 7. Basic concepts of experimental design and statistical power.**
- 8. Practical skills for performing statistical analysis using software such as R, Python with libraries like NumPy and SciPy, or statistical software packages like SPSS or SAS.**
- 9. Hands-on projects and exercises to apply statistical concepts to real-world datasets and problems.**

DURATION: 1.5hrs

GRADE: 100%

11.MATLAB FOR DATA PROCESSING AND VISUALISATION

INTRODUCTION

"MATLAB for Data Processing and Visualization" refers to using MATLAB, a powerful programming language and environment, for tasks related to handling and visualizing data. In this context, MATLAB provides a wide range of tools and functions for processing, analyzing, and visualizing data in various formats, including numerical data, time series, images, and more. Users can leverage MATLAB's built-in functions for data manipulation, statistical analysis, filtering, interpolation, and other preprocessing tasks. Additionally, MATLAB offers sophisticated plotting and visualization capabilities, allowing users to create high-quality plots, charts, and interactive visualizations to explore and present their data effectively. Overall, MATLAB is a versatile tool for data scientists, engineers, and researchers to perform data processing and visualization tasks efficiently.

WHAT WE LEARNT

- 1. Data importing and exporting techniques to read data from various file formats such as CSV, Excel, and text files.**
- 2. Data cleaning and preprocessing methods including handling missing data, removing outliers, and filtering noisy data.**
- 3. Data manipulation and transformation techniques such as sorting, reshaping, and merging datasets.**
- 4. Basic statistical analysis including calculating descriptive statistics, performing hypothesis tests, and fitting probability distributions.**
- 5. Visualization techniques for exploring and presenting data, including plotting histograms, scatter plots, line plots, and bar graphs.**
- 6. Customizing plots and adding annotations to enhance data visualization.**
- 7. Interactive data exploration using MATLAB's graphical user interface (GUI) tools.**
- 8. Advanced data visualization techniques such as 3D plotting, heatmap visualization, and geographic mapping.**
- 9. Exporting visualizations for publication or presentation purposes**

DURATION: 8 hrs

GRADE: 100%

12. IMAGE PROCESSING ONRAMP

INTRODUCTION

IMAGE PROCESSING ONRAMP is likely an introductory course or learning program focused on image processing. It could be offered by a university, online platform, or company to help individuals gain foundational knowledge and skills in image processing techniques, such as filtering, segmentation, feature extraction, and object recognition.

WHAT WE LEARNT

- 1. Fundamentals of digital image representation, including grayscale and color images.**
- 2. Basic image processing operations such as image resizing, cropping, and rotation.**

3. Image enhancement techniques including contrast adjustment, histogram equalization, and noise reduction.
4. Spatial domain filtering methods such as smoothing and sharpening filters.
5. Frequency domain techniques like Fourier Transform for image analysis and filtering.
6. Edge detection algorithms such as Sobel, Prewitt, and Canny for detecting edges in images.
7. Image segmentation methods for partitioning an image into meaningful regions.
8. Feature extraction techniques for extracting relevant information from images.
9. Introduction to computer vision concepts and applications.
10. Practical skills for implementing image processing algorithms using MATLAB's Image Processing Toolbox or libraries like OpenCV.
11. Hands-on projects and exercises to apply image processing techniques to real-world image datasets and problems.

DURATION: 2hrs

GRADE: 100%

13.IMAGE PROCESSING WITH MATLAB

INTRODUCTION

"Image Processing with MATLAB" is a course or a set of resources designed to teach image processing techniques using the MATLAB software. MATLAB is a programming environment and language commonly used in engineering and scientific fields, including image processing. This course likely covers topics such as image enhancement, filtering, segmentation, feature extraction, and object recognition, all using MATLAB's built-in functions and toolboxes. It's a practical way to learn how to apply image processing algorithms using MATLAB's powerful tools.

WHAT WE LEARNT

1. Basic image representation and manipulation techniques, including loading, displaying, and saving images.
2. Image enhancement methods such as contrast adjustment, histogram equalization, and noise reduction.
3. Spatial domain filtering techniques for smoothing and sharpening images using various filter kernels.
4. Frequency domain analysis using Fourier Transform for image filtering and analysis.
5. Edge detection algorithms like Sobel, Prewitt, and Canny for detecting edges in images.
6. Image segmentation techniques for partitioning images into meaningful regions.
7. Morphological operations such as dilation, erosion, opening, and closing for shape analysis and feature extraction.
8. Feature extraction methods for extracting relevant information from images, such as texture analysis and object recognition.
9. Introduction to advanced topics such as image registration, image stitching, and image restoration.
10. Practical implementation of image processing algorithms using MATLAB's Image Processing Toolbox.
11. Hands-on projects and exercises to apply image processing techniques to real-world image datasets and problems.

DURATION : 11hrs

GRADE: 100%

14.SIGNAL PROCESSING WITH MATLAB

INTRODUCTION

"Signal Processing with MATLAB" is a course or educational resource focused on teaching signal processing techniques using the MATLAB software. Signal processing involves analyzing, manipulating, and interpreting signals, which could be audio, video, or any other type of data that varies over time or space. MATLAB provides a rich set of functions and toolboxes specifically designed for signal processing tasks, making it a

popular choice for engineers, researchers, and students working in this field. This course likely covers topics such as filtering, spectral analysis, time-frequency analysis, and digital signal processing algorithms, all implemented using MATLAB.

WHAT WE LEARNT

- 1. Basics of signal representation and analysis, including continuous-time and discrete-time signals.**
- 2. Time-domain and frequency-domain analysis techniques, such as Fourier Transform, Discrete Fourier Transform (DFT), and Fast Fourier Transform (FFT).**
- 3. Filtering operations, including design and implementation of digital filters such as low-pass, high-pass, band-pass, and band-stop filters.**
- 4. Convolution and correlation operations for signal processing tasks.**
- 5. Windowing techniques for signal analysis and spectral estimation.**
- 6. Spectral analysis methods, including periodogram, power spectral density (PSD), and spectrogram.**
- 7. Introduction to adaptive filtering techniques for applications such as noise cancellation and system identification.**
- 8. Practical implementation of signal processing algorithms using MATLAB's Signal Processing Toolbox.**
- 9. Hands-on projects and exercises to apply signal processing techniques to real-world signal datasets and problems.**

DURATION: 7.5hrs

GRADE: 100%

15.WIRELESS COMMUNICATION ONRAMP

INTRODUCTION

"Wireless Communication ONRAMP" sounds like a specialized learning program or course designed to introduce individuals to the fundamentals of

wireless communication technologies. This could include topics such as radio frequency (RF) communication, modulation techniques, multiple access schemes, channel coding, and wireless networking protocols. The term "ONRAMP" suggests that it's an entry-level or introductory program, aimed at providing a foundational understanding of wireless communication concepts. It could be offered by educational institutions, online platforms, or companies as a means for individuals to gain essential knowledge in this field.

WHAT WE LEARNT

DURATION

GRADE: 100%

16.OPTIMIZATION ONRAMP

INTRODUCTION

"Optimization ONRAMP" likely refers to an educational program or course focused on optimization techniques. Optimization is a branch of mathematics and computer science that deals with finding the best solution to a problem from a set of possible solutions. This could involve maximizing or minimizing an objective function while satisfying certain constraints.

The term "ONRAMP" suggests that it's an introductory or beginner-level program aimed at helping individuals learn the basics of optimization methods and algorithms. It could cover topics such as linear programming, nonlinear optimization, convex optimization, and heuristic optimization techniques.

WHAT WE LEARNT

DURATION

GRADE: 100%

17.MATLAB PROGRAMMING TECHNIQUES

INTRODUCTION

"MATLAB Programming Techniques" refers to strategies, methods, and best practices for writing efficient and effective MATLAB code. This could include topics such as:

- 1. Syntax and language features: Understanding MATLAB syntax, data types, and built-in functions.**
- 2. Vectorization: Leveraging MATLAB's ability to perform operations on entire arrays instead of individual elements for faster execution.**
- 3. Memory management: Writing code that minimizes memory usage and optimizes memory access patterns.**

- 4. Code organization:** Structuring MATLAB code into functions, scripts, and classes for better readability and maintainability.
- 5. Debugging and error handling:** Techniques for identifying and fixing errors in MATLAB code, as well as handling exceptions gracefully.
- 6. Performance optimization:** Strategies for improving the performance of MATLAB code, such as preallocation, avoiding unnecessary computations, and using efficient algorithms.
- 7. MATLAB toolboxes:** Utilizing built-in MATLAB toolboxes and libraries to streamline development and leverage specialized functionality.

These techniques help MATLAB programmers write code that is not only correct but also efficient, scalable, and easy to maintain.

WHAT WE LEARNT

- 1. Basic syntax and programming constructs in MATLAB, including variables, data types, operators, and control flow statements (if-else, loops).**
- 2. Writing modular and reusable code using functions and scripts.**
- 3. Vectorization techniques for optimizing code performance and efficiency.**
- 4. Debugging and error handling strategies to identify and fix issues in MATLAB code.**
- 5. MATLAB programming best practices, including code readability, documentation, and naming conventions.**
- 6. Advanced data structures such as cell arrays, structures, and tables for organizing and manipulating data.**
- 7. File I/O operations for reading and writing data to files in various formats (e.g., text files, Excel files).**
- 8. Object-oriented programming (OOP) concepts in MATLAB, including classes, objects, inheritance, and polymorphism.**
- 9. MATLAB's built-in libraries and toolboxes for specialized tasks such as signal processing, image processing, and optimization.**
- 10. Profiling and optimization techniques for improving code performance and efficiency.**
- 11. Version control and collaborative coding practices using MATLAB's integration with Git and GitHub.**

DURATION: 16hrs

GRADE: 100%

18.SOLVING NON-LINEAR EQUATIONS WITH MATLAB

INTRODUCTION:

Solving nonlinear equations with MATLAB involves using numerical methods to find approximate solutions to equations that cannot be solved analytically. MATLAB provides several built-in functions and tools for solving nonlinear equations, including:

- 1. fsolve: This function finds a solution to a system of nonlinear equations. It requires an initial guess for the solution and returns the solution that satisfies the equations.**
- 2. fzero: This function finds a root of a single-variable nonlinear function. It requires an initial guess for the root and returns the value of the root.**
- 3. Nonlinear equation solvers in optimization toolboxes: MATLAB's optimization toolboxes provide additional functions for solving nonlinear equations, including lsqnonlin for least squares problems and lsqcurvefit for curve fitting.**

To solve a nonlinear equation in MATLAB, you typically define the equation as a function or anonymous function, specify initial guesses for the solution, and then use one of the above functions to find the solution. MATLAB's documentation provides detailed information and examples for using these functions to solve nonlinear equations.

WHAT WE LEARNT

In a course on solving nonlinear equations with MATLAB, you typically learn:

- 1. Introduction to nonlinear equations and their importance in various fields such as engineering, physics, and economics.**
- 2. Different methods for solving nonlinear equations, including numerical methods like the Newton-Raphson method, the secant method, and the bisection method.**
- 3. Understanding the principles behind each method, including their convergence properties and limitations.**
- 4. Implementing these methods in MATLAB using built-in functions or custom scripts.**
- 5. Handling complex nonlinear systems of equations using iterative methods and numerical techniques.**
- 6. Evaluating the accuracy and efficiency of different solution methods for nonlinear equations.**
- 7. Strategies for dealing with singularities, multiple solutions, and other challenges that may arise in nonlinear equation solving.**
- 8. Applications of nonlinear equation solving in practical problems, such as root-finding in optimization and parameter estimation tasks.**

Overall, the course aims to equip students with the knowledge and skills necessary to effectively solve nonlinear equations using MATLAB, enabling them to tackle a wide range of mathematical and engineering problems.

DURATION: 3hrs

GRADE: 100%



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

MATLAB Onramp



DIRECTOR, TRAINING SERVICES

16 November 2023



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Computer Vision Onramp



DIRECTOR, TRAINING SERVICES

28 December 2023



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Simulink Onramp



DIRECTOR, TRAINING SERVICES

4 April 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Object-Oriented Programming Onramp



DIRECTOR, TRAINING SERVICES

5 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Machine Learning Onramp



DIRECTOR, TRAINING SERVICES

28 December 2023



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

MATLAB Onramp



DIRECTOR, TRAINING SERVICES

16 November 2023



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

MATLAB Programming Techniques



DIRECTOR, TRAINING SERVICES

14 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Image Processing Onramp



DIRECTOR, TRAINING SERVICES

13 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

MATLAB for Data Processing and Visualization



DIRECTOR, TRAINING SERVICES

5 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Signal Processing Onramp



DIRECTOR, TRAINING SERVICES

13 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Solving Nonlinear Equations with MATLAB



DIRECTOR, TRAINING SERVICES

14 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Deep Learning with MATLAB



DIRECTOR, TRAINING SERVICES

28 February 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Machine Learning with MATLAB

A handwritten signature in black ink, appearing to read 'Rajesh Kumar', written over a thin horizontal line.

DIRECTOR, TRAINING SERVICES

1 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Statistics Onramp



DIRECTOR, TRAINING SERVICES

1 March 2024



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Machine Learning Onramp



DIRECTOR, TRAINING SERVICES

28 December 2023



Course Completion Certificate

Sowmiyadevi S

has successfully completed **100%** of the self-paced training course

Reinforcement Learning Onramp



DIRECTOR, TRAINING SERVICES

28 December 2023