

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

R.YUVAREKHA (43739034)



**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

www.sathyabama.ac.in

DEPARTMENT OF BIOINFORMATICS
AND DATA SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of
R.YUVAREKHA (43739034) who carried out the project entitled
“**Integrated Analysis: Gene Expression, Survival, Biomarkers, and
Drug Targets**” under my / our supervision from January 2024 to April
2024.

Internal Guide

External Guide

Head of the Department

Submitted for Viva voce Examination held on _____

DECLARATION

I **R.YUVAREKHA** (43739034) hereby declare that the Project Report entitled **“Integrated Analysis: Gene Expression, Survival, Biomarkers, and Drug Targets”** done by me under the guidance of Dr. /Prof./ Mr./Ms._____ and _____(External) at _____ is submitted in partial fulfilment of the requirements for the award of Masters of Science degree in Bioinformatics and Data Science.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

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 ----- and Dr.-----, Head of the Department, Dept. of Bioinformatics and DataScience 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./Mr./Ms-----for their 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 and DataScience 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	10
6	REVIEW ARTICLE	15
7	MATLAB COURCES	20

ABSTRACT

The approach to undo the molecular complexities underlying lung cancer pathogenesis and therapeutic interventions. Firstly, we conducted a comprehensive differential expression analysis using GEO2R, elucidating key genes dysregulated in lung cancer compared to normal tissue. Subsequent functional analysis using Cytoscape and ClueGO provided valuable insights into the biological processes and pathways implicated in disease progression.

Building upon these findings, we performed survival analysis utilizing Kaplan-Meier plots to assess the prognostic significance of identified genes. This analysis allowed us to stratify patients based on gene expression profiles, uncovering potential prognostic biomarkers essential for patient risk stratification and personalized treatment strategies. Then we used TheMarker to conduct biomarker screening specifically tailored for lung cancer, identifying potential diagnostic and prognostic markers with high specificity and sensitivity. These markers offer promise for early detection and prognosis assessment in clinical settings.

Additionally, we conducted drug target screening using SwissTargetPrediction to identify potential targets for therapeutic intervention in lung cancer. Subsequent molecular docking analysis with the drug doxorubicin assessed the binding affinity of identified targets, offering insights into potential treatment efficacy. Overall, our integrated approach provides a comprehensive understanding of lung cancer biology, from dysregulated gene expression to prognostic biomarkers and potential therapeutic targets.

AIM

To comprehensively investigate the molecular landscape of lung cancer, elucidating dysregulated genes, molecular pathways, and potential therapeutic targets, while also assessing the prognostic significance of treatment modalities, including the drug doxorubicin.

OBJECTIVE

1. Conduct a differential expression analysis using GEO2R to identify genes dysregulated in lung cancer compared to normal lung tissue.
2. Perform functional analysis utilizing Cytoscape and ClueGO to elucidate the biological processes and pathways associated with the dysregulated genes.
3. Utilize the KM plotter to analyze survival plots
4. Explore potential biomarkers and therapeutic targets for lung cancer through integrated analysis of gene expression data and survival outcomes
5. Assess the binding affinity of doxorubicin with potential therapeutic targets identified in lung cancer using molecular docking methods, providing insights into its mechanism of action and therapeutic potential.

MATERIALS:

Geo2R:

GEO2R is an interactive web tool that allows users to compare two or more groups of Samples in a GEO Series in order to identify genes that are differentially expressed across experimental conditions.

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, miRNA, 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:

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.

SwissTargetPrediction:

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

Molecular Docking Software (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 smiles 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 23793 genes were found to be differentially expressed between lung cancer samples and normal lung tissue.
- Of these, 35 genes were upregulated, and 23758 genes were downregulated in lung cancer compared to normal tissue.

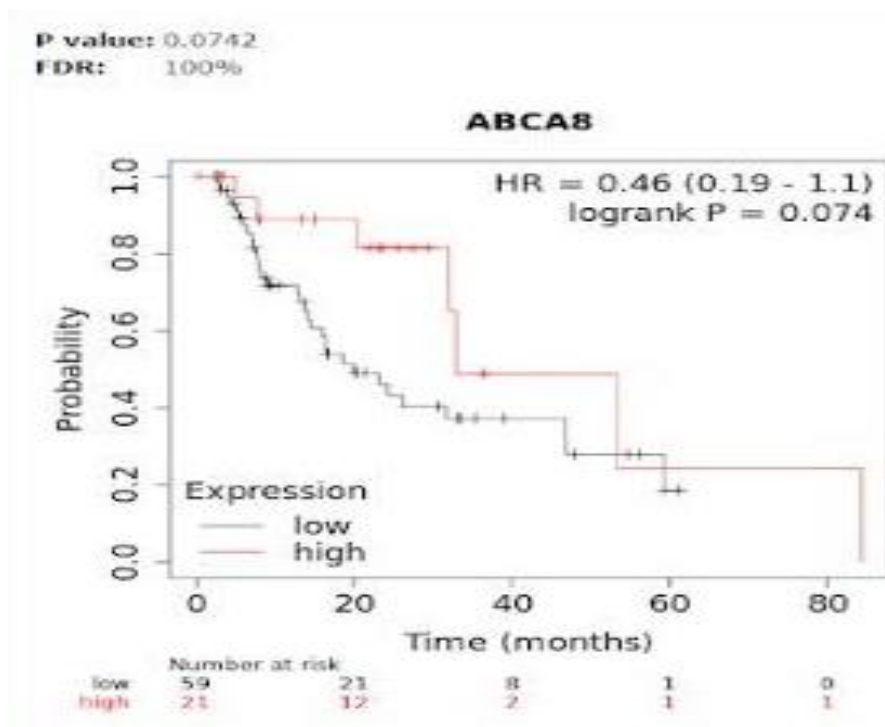


Fig 3 : Survival Plot

- The Biomarker selected is ABCA8 mRNA(mRNA)
- Drug name : Azathioprine

Search for Drug(s)

You are searching by: "Rheumatoid arthritis"

View by Card View by Table Filter Search Results by Keyword:

Drug ID: DFX24Q			
	Drug Name	Azathioprine	Drug Info
	Drug Type	Small molecule	
	Drug Status	Approved	
	Representative Biomarker	Nucleotide triphosphate diphosphatase NUDT15 (Protein)	Biomarker Info
	Representative Disease	Rheumatoid arthritis [ICD-11: FA20]	Disease Info

Fig: 4 Drug Details

- 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.

- **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

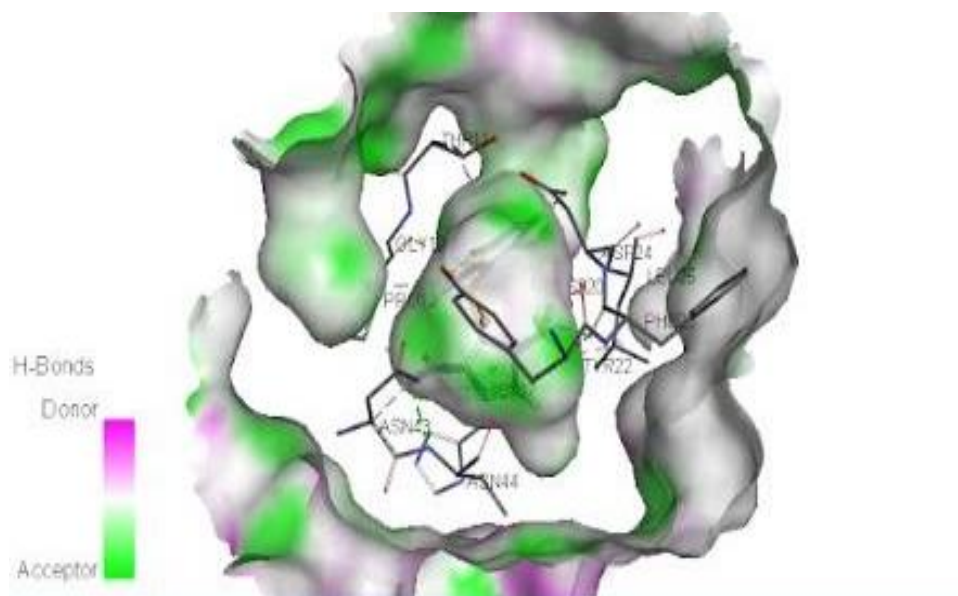


Fig 5 : 3D Structure

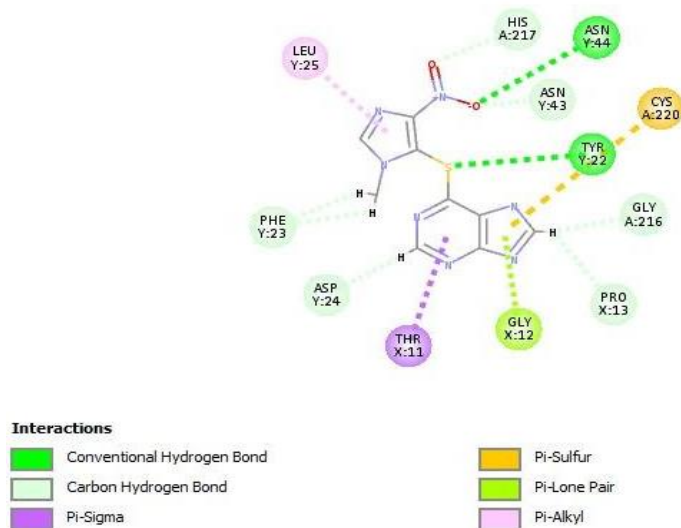
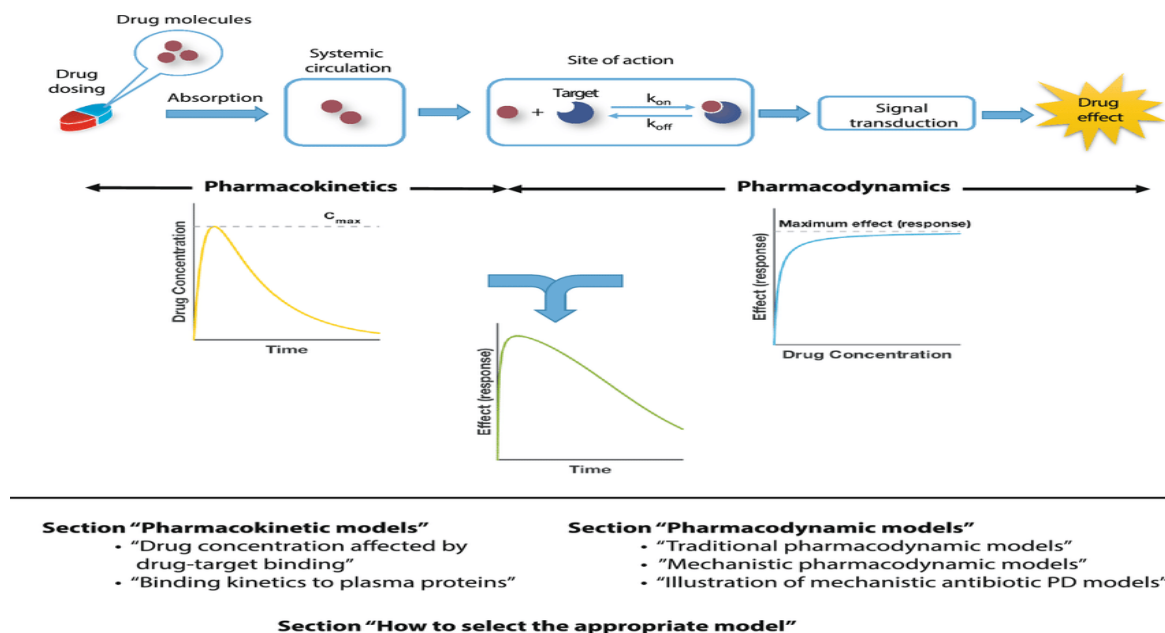


Fig 6: 2D Structure of the targets and the drug interaction

INTRODUCTION :

Mechanism-based pharmacokinetic and pharmacodynamics (PKPD) and disease system (DS) models have been introduced in drug discovery and development research, to predict in a quantitative manner the effect of drug treatment in vivo in health and disease. This requires consideration of several fundamental properties of biological systems behavior including: hysteresis, non-linearity, variability, interdependency, convergence, resilience, and multi-stationarity. Classical physiology-based PKPD models consider linear transduction pathways, connecting processes on the causal path between drug administration and effect, as the basis of drug action. Depending on the drug and its biological target, such models may contain expressions to characterize i) the disposition and the target site distribution kinetics of the drug under investigation, ii) the kinetics of target binding and activation and iii) the kinetics of transduction. When connected to physiology-based DS models, PKPD models can characterize the effect on disease progression in a mechanistic manner. These models have been found useful to characterize hysteresis and non-linearity, yet they fail to explain the effects of the other fundamental properties of biological systems behavior.



Recently systems pharmacology has been introduced as novel approach to predict in vivo drug effects, in which biological networks rather than single transduction pathways are considered as the basis of drug action and disease progression. These models contain expressions to characterize the functional interactions within a biological network. Such interactions are relevant when drugs act at multiple targets in the network or when homeostatic feedback mechanisms are operative. As a result systems

pharmacology models are particularly useful to describe complex patterns of drug action (i.e. synergy, oscillatory behavior) and disease progression (i.e. episodic disorders). In this contribution it is shown how physiology-based PKPD and disease models can be extended to account for internal systems interactions. It is demonstrated how SP models can be used to predict the effects of multi-target interactions and of homeostatic feedback on the pharmacological response. In addition it is shown how DS models may be used to distinguish symptomatic from disease modifying effects and to predict the long term effects on disease progression, from short term biomarker responses. It is concluded that incorporation of expressions to describe the interactions in biological network analysis opens new avenues to the understanding of the effects of drug treatment on the fundamental aspects of biological systems behavior.

NETWORK ANALYSIS IN SYSTEMS PHARMACOLOGY:

Systems pharmacology is an emerging area of pharmacology which utilizes network analysis of drug action as one of its approaches. By considering drug actions and side effects in the context of the regulatory networks within which the drug targets and disease gene products function, network analysis promises to greatly increase our knowledge of the mechanisms underlying the multiple actions of drugs. Systems pharmacology can provide new approaches for drug discovery for complex diseases. The integrated approach used in systems pharmacology can allow for drug action to be considered in the context of the whole genome. Network-based studies are becoming an increasingly important tool in understanding the relationships between drug action and disease susceptibility genes. This review discusses how analysis of biological networks has contributed to the genesis of systems pharmacology and how these studies have improved global understanding of drug targets, suggested new targets and approaches for therapeutics, and provided a deeper understanding of the effects of drugs. Taken together, these types of analyses can lead to new therapeutic options while improving the safety and efficacy of existing medications

PHARMACOLOGICAL INTERVENTION IN SYSTEM BIOLOGY MODELS

The field of medical systems biology aims to advance understanding of molecular mechanisms that drive disease progression and to translate this knowledge into therapies to effectively treat diseases. A challenging task is the investigation of long-term effects of a (pharmacological) treatment, to establish its applicability and to identify potential side effects. We present a new modeling approach, called Analysis of Dynamic Adaptations in Parameter Trajectories (ADAPT), to analyze the long-term effects of a pharmacological intervention. A concept of time-dependent evolution of model parameters is introduced to study the dynamics of molecular adaptations. The progression of these adaptations is predicted by identifying necessary dynamic changes in the model parameters to describe the transition between experimental data obtained during different stages of the treatment. The trajectories provide insight in the affected underlying biological systems and identify the molecular events that should be studied in more detail to unravel the mechanistic basis of treatment outcome. Modulating effects caused by interactions with the proteome and transcriptome levels, which are often less well understood, can be captured by the time-dependent descriptions of the parameters. ADAPT was employed to identify metabolic adaptations induced upon pharmacological activation of the liver X receptor (LXR), a potential drug target to treat or prevent atherosclerosis. The trajectories were investigated to study the cascade of adaptations. This provided a counter-intuitive insight concerning the function of scavenger receptor class B1 (SR-B1), a receptor that facilitates the hepatic uptake of cholesterol. Although activation of LXR promotes cholesterol efflux and -excretion, our computational analysis showed that

the hepatic capacity to clear cholesterol was reduced upon prolonged treatment. This prediction was confirmed experimentally by immunoblotting measurements of SR-B1 in hepatic membranes. Next to the identification of potential unwanted side effects, we demonstrate how ADAPT can be used to design new target interventions to prevent these.

UNRAVELING COMPLEX INTERACTION IN POLYGENIC DISORDER

Polygenic diseases are caused by the joint contribution of a number of independently acting or interacting polymorphic genes; the individual contribution of each gene may be small or even unnoticeable. The carriage of certain combinations of genes can determine the occurrence of clinically heterogeneous forms of the disease and treatment efficacy. This review describes the approaches used in a polygenic analysis of data in medical genomics, in particular, pharmacogenomics, aimed at identifying the cumulative effect of genes. This effect may result from the summation of gains of different genes or be caused by the epistatic interaction between the genes. Both cases are undoubtedly of great interest in investigating the nature of polygenic diseases. The means that allow one to discriminate between these two possibilities are discussed. The methods for searching for combinations of alleles of different genes associated with the polygenic phenotypic traits of the disease, as well as the methods for presenting and validating the results, are described and compared. An attempt is made to evaluate the applicability of the existing methods to an epistasis analysis. The results obtained by the authors using the APSampler software are described and summarized. Individuals with complex disorders typically have a heritable burden of common variation that can be expressed as a polygenic risk score (PRS). While PRS has some predictive utility, it lacks the molecular specificity to be directly informative for clinical interventions. We therefore sought to develop a framework to quantify an individual's common variant enrichment in clinically actionable systems responsive to existing drugs. This was achieved with a metric designated the pharmagenic enrichment score (PES), which we demonstrate for individual SNP profiles in a cohort of cases with schizophrenia. A large proportion of these had elevated PES in one or more of eight clinically actionable gene-sets enriched with schizophrenia associated common variation. Notable candidates targeting these pathways included vitamins, antioxidants, insulin modulating agents, and cholinergic drugs. Interestingly, elevated PES was also observed in individuals with otherwise low common variant burden. The biological saliency of PES profiles were observed directly through their impact on gene expression in a subset of the cohort with matched transcriptomic data, supporting our assertion that this gene-set orientated approach could integrate an individual's common variant risk to inform personalised interventions, including drug repositioning, for complex disorders such as schizophrenia.

References:

Admiraal et al., 2014

R. Admiraal, C. van Kesteren, J.J. Boelens, R.G. Bredius, D. Tibboel, C.A. Knibbe
Towards evidence-based dosing regimens in children on the basis of population pharmacokinetic
pharmacodynamic modelling
Arch. Dis. Child., 99 (2014), pp. 267-272
[View at publisher CrossRef](#)[View in Scopus](#)[Google Scholar](#)

Ballinger et al., 2015
T.J. Ballinger, M.E. Sanders, V.G. Abramson
Current HER2 testing recommendations and clinical relevance as a predictor of response to
targeted therapy
Clin. Breast Cancer, 15 (2015), pp. 171-180
[View in Scopus](#)[Google Scholar](#)

Bauer and Fung, 1994
J.A. Bauer, H.L. Fung
Pharmacodynamic models of nitroglycerin-induced hemodynamic tolerance in experimental
heart failure
Pharm. Res., 11 (1994), pp. 816-823
[View in Scopus](#)[Google Scholar](#)

Beard et al., 2002
D.A. Beard, S.D. Liang, H. Qian
Energy balance for analysis of complex metabolic networks
Biophys. J., 83 (2002), pp. 79-86
[View at publisher](#)
This article is free to access.
[View in Scopus](#)[Google Scholar](#)

Bliss, 1939
C.I. Bliss
The toxicity of poisons applied jointly
Ann. Appl. Biol., 26 (1939), pp. 585-615
[View at publisher CrossRef](#)[View in Scopus](#)[Google Scholar](#)
Cox et al., 1998a

YUVA REKHA.R

ORIGINALITY REPORT

3%

SIMILARITY INDEX

2%

INTERNET SOURCES

2%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

1

[fastercapital.com](https://www.fastercapital.com)

Internet Source

1%

2

Chenglong Wang, Xiaohui Lv, Wen Liu, Song Liu, Zongxi Sun. "Uncovering the pharmacological mechanism of motherwort (*Leonurus japonicus* Houtt.) for treating menstrual disorders: A systems pharmacology approach", Computational Biology and Chemistry, 2020

Publication

1%

3

www.ncbi.nlm.nih.gov

Internet Source

1%

4

cid.oxfordjournals.org

Internet Source

<1%

5

Aditya K. Padhi, Shweeta Maurya. "Uncovering the secrets of resistance: An introduction to computational methods in infectious disease research", Elsevier BV, 2024

Publication

<1%

Exclude quotes Off
Exclude bibliography On

Exclude matches Off

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 visualization
- Programming concepts
- Advanced data types
- File I/O
- Symbolic mathematics
- Applications

DURATION: 2hrs

GRADE: 100%

OBJECT ORIENTED 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%

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 debugging techniques

DURATION: 16.5 hrs

GRADE: 100%

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%

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%

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 LEARN

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%

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%

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%

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%

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%

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%

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%

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%

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%

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%

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

Yuvarekha R

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

Signal Processing Onramp

A handwritten signature in black ink, reading "Ray L. Santos".

DIRECTOR, TRAINING SERVICES

6 April 2024



Course Completion Certificate

Yuvarekha R

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

Solving Nonlinear Equations with MATLAB



DIRECTOR, TRAINING SERVICES

31 March 2024



Course Completion Certificate

Yuvarekha R

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

MATLAB Programming Techniques



DIRECTOR, TRAINING SERVICES

29 March 2024



Course Completion Certificate

Yuvarekha R

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

Reinforcement Learning Onramp

A handwritten signature in black ink, reading "Ray L Santos".

DIRECTOR, TRAINING SERVICES

29 March 2024



Course Completion Certificate

Yuvarekha R

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

MATLAB for Data Processing and Visualization

A handwritten signature in black ink, reading "Ray L Santos".

DIRECTOR, TRAINING SERVICES

21 March 2024



Course Completion Certificate

Yuvarekha R

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

Object-Oriented Programming Onramp



Craig Santos

DIRECTOR, TRAINING SERVICES

21 March 2024



Course Completion Certificate

Yuvarekha R

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

Computer Vision Onramp



Craig Santos

DIRECTOR, TRAINING SERVICES

13 March 2024



Course Completion Certificate

Yuvarekha R

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

Deep Learning with MATLAB



DIRECTOR, TRAINING SERVICES

10 March 2024



Course Completion Certificate

Yuvarekha R

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

Statistics Onramp



DIRECTOR, TRAINING SERVICES

4 March 2024



Course Completion Certificate

Yuvarekha R

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

Deep Learning Onramp



DIRECTOR, TRAINING SERVICES

29 February 2024



Course Completion Certificate

Yuvarekha R

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

Machine Learning Onramp



DIRECTOR, TRAINING SERVICES

29 February 2024



Course Completion Certificate

Yuvarekha R

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

MATLAB Fundamentals

A handwritten signature in black ink, reading "Craig L. Santos".

DIRECTOR, TRAINING SERVICES

22 November 2023



Course Completion Certificate

Yuvarekha R

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

MATLAB Onramp

A handwritten signature in black ink, reading "Craig L. Santos".

DIRECTOR, TRAINING SERVICES

17 November 2023

