Essential Math for Data Science

SKILLSOFT ASPIRE JOURNEY

skillsoft | percipio

Aspire Journeys

Essential Math for Data Science

Mathematics form the foundation for Machine Learning algorithms and Data Science, necessary for working and research in the Data Science field. Many Data Science elements depend on mathematical concepts such as probability, statistics, calculus, linear algebra, and so on. Hence, it is important for data scientists, to understand the principles of these concepts and how these principles might affect their models and day-to-day tasks.

In the Essential Math for Data Science journey, you will explore important concepts of mathematics that form the foundation for Machine Learning algorithms, Data Science and Artificial Intelligence.

View Less

33 courses | 45h 36m 14s

skillsoft

Earn a Badge

Tracks

Track 1: Introduction to Math

In this track of the Essential Math for Data Science Skillsoft Aspire journey, you will focus on the fundamentals of linear algebra and calculus. This includes discrete math concepts and their ...

View More

Explore 10 courses | 12h 44m 33s

Track 2: Statistics and Probability

In this track of the Essential Math for Data Science Skillsoft Aspire journey, you will acquire a deeper understanding of probability and statistical concepts including probability distributions, various types ...

View More

Explore 12 courses | 17h 23m 14s

Track 3: Math Behind ML Algorithms

In this track of the Essential Math for Data Science Skillsoft Aspire journey, the focus will be on math applied in various machine learning algorithms. You will understand the intuition behind these ...

View More

Explore 9 courses | 12h 49m 36s

Track 4: Advanced Math

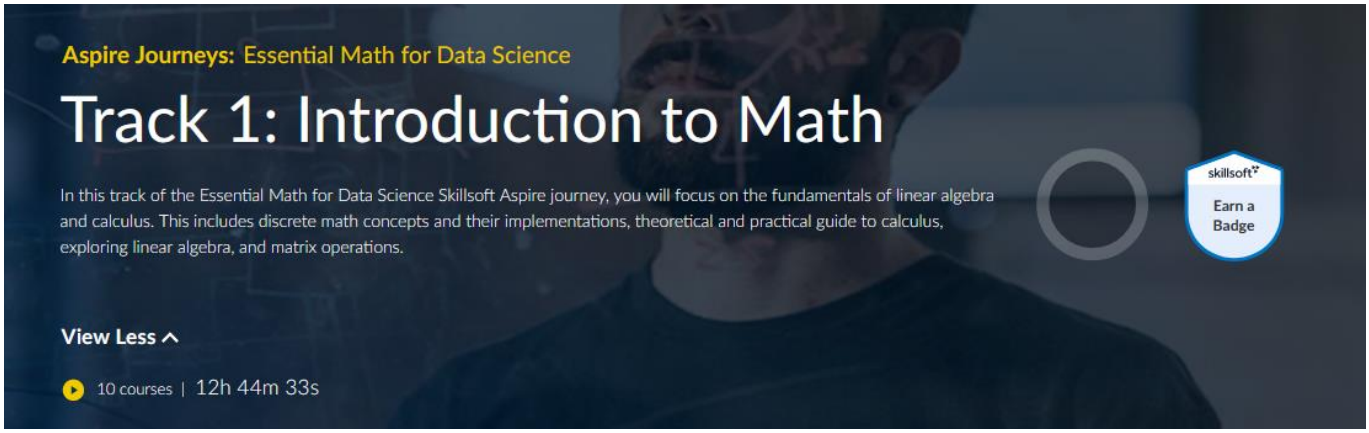
In this track of the Essential Math for Data Science Skillsoft Aspire journey, the focus will be on math behind advanced concepts such as principal component analysis, recommendation systems, and ...

View More

Explore 2 courses | 2h 38m 50s

Prerequisites

Knowledge of Data Science | Knowledge of Machine Learning | Knowledge of Artificial Intelligence



ML & Dimensionality Reduction: Performing Principal Component Analysis

Objectives:

- recall the use of matrix operations to represent linear transformations
- recall the intuition behind principal component analysis
- define principal components and their uses
- define eigenvalues and eigenvectors
- mathematically compute principal components
- compute eigenvalues and eigenvectors
- perform principal component analysis
- build a baseline model using logistic regression
- build a logistic regression model using principal components

Recommender Systems: Under the Hood of Recommendation Systems

Objectives:

- summarize the use cases of recommendation systems and the different techniques applied to build such models, with emphasis on the content-based filtering approach
- describe the intuition behind collaborative filtering, its main advantages, and how ratings matrices, the nearest neighbour approach, and latent factor analysis are involved
- decompose a ratings matrix into its latent factors
- apply gradient descent to compute the factors of a ratings matrix
- compute a penalty for a large number of latent factors when computing the factors of a ratings matrix
- use NumPy and Pandas to define a ratings matrix that can be fed into a recommendation system
- implement the gradient descent algorithm to decompose a ratings matrix
- compute the predicted ratings given by users for various items by using matrix decomposition

Aspire Journeys: Essential Math for Data Science

Track 2: Statistics and Probability

In this track of the Essential Math for Data Science Skillsoft Aspire journey, you will acquire a deeper understanding of probability and statistical concepts including probability distributions, various types of statistical tests, and hypothesis testing. You will deep dive into understanding conditional probability concepts that forms the crux of naïve Bayes classification algorithms.

skillsoft

Earn a Badge

View Less

12 courses | 17h 23m 14s

Core Statistical Concepts: An Overview of Statistics & Sampling

Objectives:

- describe what statistics, populations, and samples are
- recognize how metrics such as mean, median and mode describe data
- recall what information is conveyed by measures such as standard deviation and variance
- summarize the workings a number of probability sampling techniques
- outline how to create balanced samples from an imbalanced dataset

Core Statistical Concepts: Statistics & Sampling with Python

Objectives:

- install the latest versions of pandas and visualization modules used to analyze data
- load data from a CSV file into a pandas DataFrame and perform some initial analysis
- calculate the mean and median of a distribution using your own function and compare it with the built-in pandas function
- use Seaborn and Matplotlib to visualize a distribution and where the mean, median, and mode fit in
- compute and visualize the standard deviation and variance of a distribution
- implement simple random and stratified sampling on a data frame
- use pandas to generate a sample using cluster and systematic sampling
- create a balanced sample using random undersampling and oversampling
- generate synthetic data in order to create a balanced sample using the Synthetic Minority Over-sampling Technique (SMOTE)

Probability Theory: Getting Started with Probability

Objectives:

- enumerate differences between random and non-random variables and describe the role of probability
- define terms such as event, outcome, and experiment
- import python libraries needed to work with probabilities
- simulate the flipping of a coin in Python
- simulate the rolling of a die in Python
- simulate the picking of marbles from a bag in Python

Probability Theory: Understanding Joint, Marginal, & Conditional Probability

Objectives:

- discover the key concepts covered in this course
- define joint, marginal, and conditional probability
- link the definitions of marginal and conditional probability
- outline the chain rule of probability
- calculate joint probabilities associated with the rolling of a die
- compute marginal probabilities
- compute conditional probabilities
- simulate the rolling of two die to test joint probability
- calculate the joint probability of dependent variables
- calculate marginal and conditional probability on dependent variables
- define the formula of the expected value of a random variable
- compute the expected value of the rolling of a die

Probability Theory: Creating Bayesian Models

Objectives:

- define and understand the Bayes theorem
- enumerate the architecture of Bayesian networks
- use the chain rule with Bayesian networks
- create probability tables for a Bayesian network
- explore the probability tables of nodes in a Bayesian network
- query Bayesian networks to measure probabilities
- define a Bayesian model in Python
- predict values with Bayesian models
- explore probabilities associated with a Bayesian model
- create naive Bayes models in Python
- predict values with naive Bayes models

Probability Distributions: Getting Started with Probability Distributions

Objectives:

- use SciPy to generate uniformly distributed samples
- analyze a uniform distribution by using cumulative distribution and probability density functions
- generate discrete uniform data using NumPy and SciPy and evaluate the distributions
- describe binomial distributions and generate one using SciPy
- simulate trials with binomial distributions in SciPy
- explore cumulative distribution, probability mass, and survival functions with binomial data
- describe the Poisson distribution and its applications
- invoke functions available in SciPy to work with Poisson distributions
- apply Poisson distributions to make estimates in real-life situations

Probability Distributions: Uniform, Binomial, & Poisson Distributions

Objectives:

- describe normal distributions and their characteristics
- use the cumulative distribution function (CDF) of a normal distribution and recognize how the mean and standard deviation (SD) influence it
- visualize the cumulative distribution function (CDF) for different standard deviations
- recall the symmetrical features of normal distributions
- explain the law of large numbers programmatically
- recall the central limit theorem and recognize its applications

Probability Distributions: Understanding Normal Distributions

Objectives:

- describe normal distributions and their characteristics
- use the cumulative distribution function (CDF) of a normal distribution and recognize how the mean and standard deviation (SD) influence it
- visualize the cumulative distribution function (CDF) for different standard deviations
- recall the symmetrical features of normal distributions
- explain the law of large numbers programmatically
- recall the central limit theorem and recognize its applications

Statistical & Hypothesis Tests: Getting Started with Hypothesis Testing

Objectives:

- outline how descriptive and inferential statistics work
- describe the fundamentals of hypothesis testing
- set up null and alternative hypotheses for statistical tests
- interpret p-values using alpha levels
- explore the one-sample, two-sample, and paired-sample T-tests
- compare and contrast type I and type II errors in hypothesis testing
- apply the ANOVA test for multiple groups

Statistical & Hypothesis Tests: Using the One-sample T-test

Objectives:

- install various modules in python
- create a function to manually perform a T-test
- compare a manual one-sample T-test to a built-in test
- explore Laplace and Wald distributions with T-tests
- test data to see if it is normally distributed using Shapiro-Wilk and Anderson-Darling tests
- perform T-tests on real-world data
- explore one-sided and two-sided T-tests
- perform the Wilcoxon signed-rank test to compare medians
- test medians using the Wilcoxon signed-rank test

Statistical & Hypothesis Tests: Performing Two-sample T-tests & Paired T-tests

Objectives:

- recall the assumptions of the two-sample T-test
- use Levene's test to check for equal variances
- use the two-sample T-test to compare means
- recognize when the Welch's T-test should be used
- use the Welch's T-test to compare means
- describe type I and type II errors
- outline the relationship between type I errors and alpha levels
- outline the relationship between type II errors and alpha levels
- set up and visualize data for the paired difference T-test
- perform the paired T-test on paired samples
- use the paired T-test to compare before and after values
- perform paired T-tests with varying outcomes for the null hypothesis

Statistical & Hypothesis Tests: Using Non-parametric Tests & ANOVA Analysis

Objectives:

- recognize the use of the Mann-Whitney U-test
- use the Mann-Whitney U-test
- set up data for the paired Wilcoxon signed-rank test
- compare the paired T-test and the paired Wilcoxon signed-rank test
- identify the pairwise T-test for multiple categories
- use the pairwise T-test to test for different means
- outline the use of one-way ANOVA analysis
- outline one-way ANOVA and linear regression
- use Tukey's HSD to know which categories differ significantly
- describe how ANOVA requires residuals to be normally distributed
- use the non-parametric Kruskal-Wallis test
- outline the use of the two-way ANOVA analysis
- use two-way ANOVA with interaction between the independent variables

Aspire Journeys: Essential Math for Data Science

Track 3: Math Behind ML Algorithms

In this track of the Essential Math for Data Science Skillsoft Aspire journey, the focus will be on math applied in various machine learning algorithms. You will understand the intuition behind these algorithms along with math used in their optimization/loss/cost functions. You will understand the math behind regression algorithms, decision trees, distance-based models, kernel methods and SVM and neural networks.

View Less

9 courses | 12h 49m 36s

skillsoft

Earn a Badge

Regression Math: Getting Started with Linear Regression

Objectives:

- define linear regression and outline how regression is used in prediction
- outline how residuals are used in regression
- describe what's meant by the least square error
- compute the best fit using partial derivatives
- calculate R-squared of a regression model
- summarize what comprises the normal equation
- visualize correlations of features
- split train and test data and create computations
- manually define a regression line
- perform regression and view the predicted values
- view the R-squared and residuals in regression
- implement regression models using libraries

Regression Math: Using Gradient Descent & Logistic Regression

Objectives:

- outline how gradient descent works
- describe what gradients are used for
- work through a calculation of an epoch
- standardize and shape data for gradient descent
- implement a single epoch
- perform gradient descent
- recall how logistic regression can be used for classification
- calculate an S-curve in logistic regression
- identify correlations for performing logistic regression
- set up training and testing data for logistic regression
- explore and perform stochastic gradient descent

The Math Behind Decision Trees: An Exploration of Decision Trees

Objectives:

- define what's meant by classification, describing classification rules and rule-based classifier properties and limitations
- contrast rule-based and ML-based classifiers
- outline the structure of a decision tree, the process it uses to "decide," its advantages, and some core considerations when building one
- work through the creation of a decision tree and list some decision tree algorithms
- define what's meant by entropy and outline how it's used in relation to decision trees, referencing the ID3 algorithm and information gain
- summarize how information gain and entropy are used in tandem
- define GINI impurity and calculate it for a dataset
- split decision trees based on GINI impurity
- import modules and set up data
- decide splits for a rule-based decision tree
- define a rule-based decision tree
- illustrate the use of decision trees for continuous values
- visualize a decision tree
- create a rule-based decision tree
- train an ML-based decision tree
- use a trained ML-based decision tree to make decisions

Distance-based Models: Overview of Distance-based Metrics & Algorithms

Objectives:

- recall how distance-based models work at a high level and identify the use cases of such models
- describe the Hamming and Cosine distance metrics
- recount how the KNN and K-means algorithms use distance metrics to perform ML operations
- define and visualize two points in a two-dimensional space using Python
- calculate the Euclidean and Manhattan distance between two points using SciPy as well as your own function
- implement a Minkowski and Hamming distance calculator and use the built-in ones available in SciPy
- compute the cosine distance between vectors

Distance-based Models: Implementing Distance-based Algorithms

Objectives:

- analyze the data used to implement a classification model using K Nearest Neighbors
- implement a function that classifies a point using the K Nearest Neighbors algorithm
- classify test data points using your own KNN classifier and evaluate the model using a variety of metrics
- implement a function that uses KNN in order to perform regression
- obtain predictions on test data for your own implementation of a KNN regressor
- code the individual steps involved in performing a clustering operation using the K-means algorithm
- define a function that clusters the points in a dataset using the K-means algorithm and then test it

Support Vector Machine (SVM) Math: A Conceptual Look at Support Vector Machines

Objectives:

- recognize the place of support vector machines (SVMs) in the machine learning landscape
- outline how SVMs can be used to classify data, how hyperplanes are defined, and the qualities of an optimum hyperplane
- recall the qualities of an optimum hyperplane, outline how scaling works with SVM, distinguish soft and hard margins, and recognize when and how to use either margin
- recall the techniques that can be applied to classify data that are not linearly separable
- formulate the optimization problem for support vector machines
- apply the gradient descent algorithm to solve for the optimum hyperplane

Support Vector Machine (SVM) Math: Building & Applying SVM Models in Python

Objectives:

- use scikit-learn to generate blob data that is linearly separable
- separate a dataset into training and test sets
- code the steps to apply gradient descent to find the optimum hyperplane
- load a dataset from a CSV file into a pandas DataFrame and analyze it in preparation for binary classification
- generate a heatmap to visualize the correlations between features in a dataset
- build and evaluate an SVM classifier and recognize the importance of scaling the inputs to such a model
- use boxplots, a pair plot, and a heatmap to analyze a dataset in preparation for training a regression model
- build and evaluate an SVM regressor from the LIBSVM library

Neural Network Mathematics: Understanding the Mathematics of a Neuron

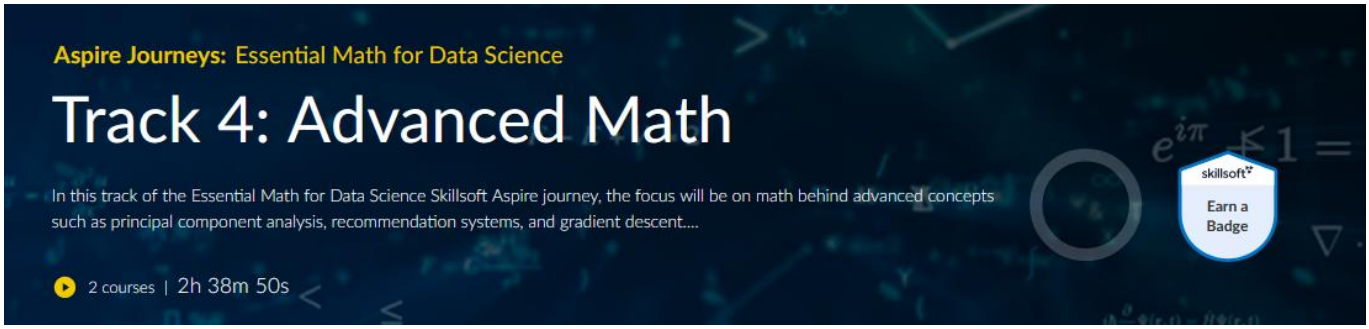
Objectives:

- recall the architecture and components that make up neural networks
- summarize the mathematical operation of a neuron
- install Python modules
- compute the weighted sum of inputs with bias
- process data in batches and with multiple layers

Neural Network Mathematics: Exploring the Math behind Gradient Descent

Objectives:

- outline how gradient descent works
- summarize how to compute the gradient vector of partial derivatives
- recall the characteristics of activation functions
- illustrate step, sigmoid, and tangent activation functions
- illustrate ReLU, Leaky ReLU, and ELU activation functions
- describe how unstable gradients can be mitigated using variants of the ReLU activation function
- create a simple neural network with one neuron for regression
- illustrate the impact of learning rate and number of epochs of training
- illustrate the classification dataset
- write Python code from scratch to represent and train a single neuron



Objectives

- recall the use of matrix operations to represent linear transformations
- recall the intuition behind principal component analysis
- define principal components and their uses
- define eigenvalues and eigenvectors
- mathematically compute principal components
- compute eigenvalues and eigenvectors
- perform principal component analysis
- build a baseline model using logistic regression
- build a logistic regression model using principal components

Objectives

- summarize the use cases of recommendation systems and the different techniques applied to build such models, with emphasis on the content-based filtering approach
- describe the intuition behind collaborative filtering, its main advantages, and how ratings matrices, the nearest neighbor approach, and latent factor analysis are involved
- decompose a ratings matrix into its latent factors
- apply gradient descent to compute the factors of a ratings matrix
- compute a penalty for a large number of latent factors when computing the factors of a ratings matrix
- use NumPy and Pandas to define a ratings matrix that can be fed into a recommendation system
- implement the gradient descent algorithm to decompose a ratings matrix
- compute the predicted ratings given by users for various items by using matrix decomposition

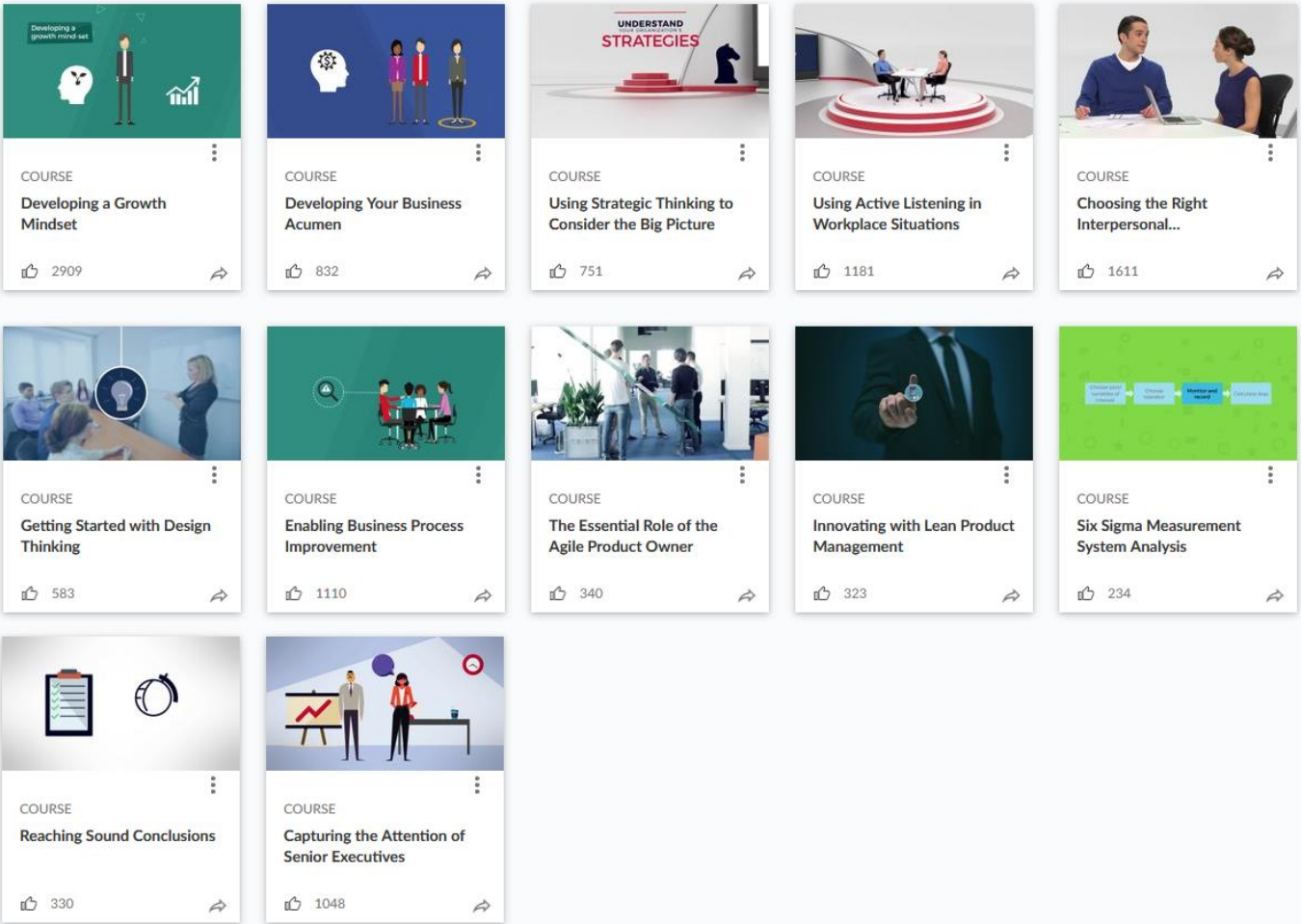
Objectives

- apply gradient descent to compute the factors of a ratings matrix
- build a baseline model using logistic regression
- build a logistic regression model using principal components
- compute a penalty for a large number of latent factors when computing the factors of a ratings matrix
- compute eigenvalues and eigenvectors
- compute the predicted ratings given by users for various items by using matrix decomposition
- decompose a ratings matrix into its latent factors
- define eigenvalues and eigenvectors
- define eigenvectors and eigenvalues

- define principal components and their uses
- describe the intuition behind collaborative filtering, its main advantages, and how ratings matrices, the nearest neighbor approach, and latent factor analysis are involved
- describe the use cases of recommendation systems and the different techniques applied to build such models, with emphasis on the content-based filtering approach
- implement the gradient descent algorithm to decompose a ratings matrix
- mathematically compute principal components
- perform principal component analysis
- recall the intuition behind principal component analysis
- recall the use of matrix operations to represent linear transformations
- summarize the intuition behind collaborative filtering, its main advantages, and how ratings matrices, the nearest neighbor approach, and latent factor analysis are involved
- summarize the use cases of recommendation systems and the different techniques applied to build such models, with emphasis on the content-based filtering approach
- use NumPy and Pandas to define a ratings matrix that can be fed into a recommendation system

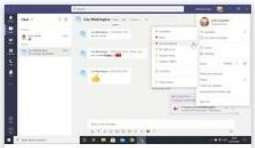
Business & Leadership for Essential Math for Data Science

Optional



Productivity Tools for Essential Math for Data Science

Optional



COURSE

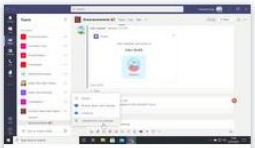
Exploring and setting up Microsoft Teams

138

COURSE

Creating and managing teams & channels

42



COURSE

Formatting, illustrating & reacting to messages

29

COURSE

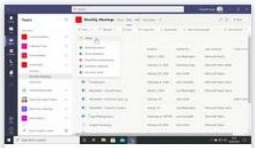
Using private messaging & call tools

37

COURSE

Creating, joining, and managing meetings

50



COURSE

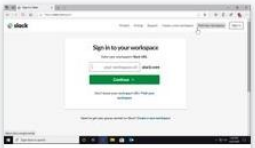
Creating, finding & organizing files

32

COURSE

Working with Tabs & Apps

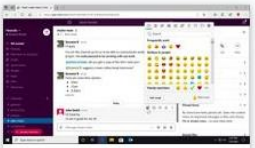
36



COURSE

Signing in & Setting Up Slack

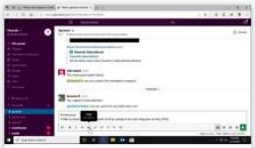
33



COURSE

Using Channels in Slack

18



COURSE

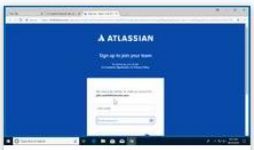
Using Private Messaging & Communication Tools in...

19

COURSE

Reporting in Jira Software

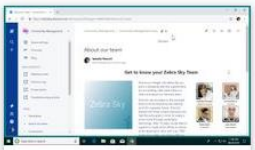
114



COURSE

Signing in & Navigating within Spaces

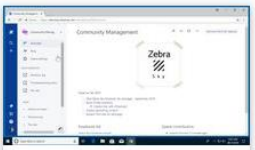
55



COURSE

Setting Up & Managing Spaces

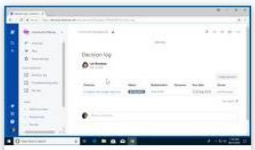
41



COURSE

Working with Space

34



COURSE

Working with Team Members

91

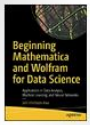
COURSE

Configuring Spaces

22

Bookshelf

Optional



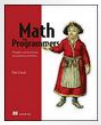
BOOK

Beginning Mathematica and Wolfram for Data Science:...

BOOK

Mathematical Theory of Advanced Computing

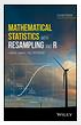
1



BOOK

Math for Programmers: 3D Graphics, Machine Learning...

3

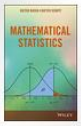


BOOK

Mathematical Statistics with Resampling and R, Second...

BOOK

Mathematical Statistics: An Introduction to Likelihood...



BOOK

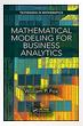
Mathematical Statistics

1

BOOK

Mathematical Methods in Science and Engineering,...

5



BOOK

Mathematical Modeling for Business Analytics

4

BOOK

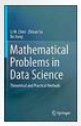
An Introduction to Mathematical Cognition

2

BOOK

Advanced Thermodynamics: Fundamentals - Mathemati...

2



BOOK

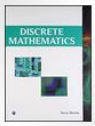
Mathematical Problems in Data Science: Theoretical...

14

BOOK

Advanced Engineering Mathematics, 10th Edition

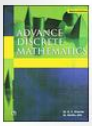
1



BOOK

Discrete Mathematics

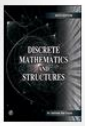
2



BOOK

Advance Discrete Mathematics, 2nd Edition

5



BOOK

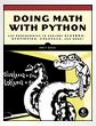
Discrete Mathematics and Structures, Sixth Edition

2

BOOK

Fundamentals of Discrete Math for Computer Science...

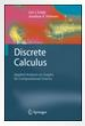
4



BOOK

Doing Math with Python: Use Programming to Explor...

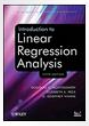
5



BOOK

Discrete Calculus: Applied Analysis on Graphs for...

1



BOOK

Introduction to Linear Regression Analysis, Fifth...

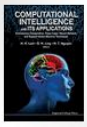
7

BOOK

Solutions Manual to Accompany Introduction to...

BOOK

Solutions Manual to Accompany Introduction to...



BOOK

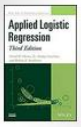
Computational Intelligence and its Applications:...

2

BOOK

MIT Sloan Management Review Article on The...

1



BOOK

Applied Logistic Regression, Third Edition

2

BOOK

Testing Statistical Assumptions in Research

2

FOLLOW US ON:

www.skilltech.pl
email: biuro@skilltech.pl
tel. +48 22 44 88 827

SkillTech
Technology hired for excellence