Read first it just overview:- {https://drive.google.com/drive/folders/1t9JDEgcQ6PFPI8BaMx19pc3oLhK1tGjD}

Distinguish between Dictionary and Tuple.

|  |  |
| --- | --- |
| **Tuple** | **Dictionary** |
| A tuple is a non-homogeneous data structure that can hold a single row as well as several rows and columns. | Dictionary is a non-homogeneous data structure that contains key-value pairs. |
| Tuples are represented by brackets (). | Dictionaries are represented by curly brackets {}. |
| Tuples are immutable i.e, we can not make changes. | Dictionaries are mutable and keys do not allow duplicates. |
| A tuple is ordered. | Dictionary is ordered (python 3.7 and above). |
| Tuple can be created using tuple() function. | Dictionary can be created using the dict() function. |
| Creating an empty Tuple: () | Creating an empty dictionary: {} |
| As tuples are immutable, the reverse() method is not defined in them. | Because the dictionary's entries are in the form of key-value pairs, the elements cannot be reversed. |
| Example: ('Tutorialspoint', 'simple’, ‘easy learning’) | Example : {'companyname': 'Tutorialspoint', 'tagline': 'simplyeasylearning'} |

Distinguish between Tuple and Lists.

| SR.NO. | LIST | TUPLE |
| --- | --- | --- |
| 1 | Lists are mutable | Tuples are immutable |
| 2 | The implication of iterations is Time-consuming | The implication of iterations is comparatively Faster |
| 3 | The list is better for performing operations, such as insertion and deletion. | Tuple data type is appropriate for accessing the elements |
| 4 | Lists consume more memory | Tuple consumes less memory as compared to the list |
| 5 | Lists have several built-in methods | Tuple does not have many built-in methods. |
| 6 | The unexpected changes and errors are more likely to occur | In tuple, it is hard to take place. |

Write a Pandas program to create a time series using three months frequency.

import pandas as pd

time\_series = pd.date\_range('1/1/2021', periods = 36, freq='3M')

print("Time series using three months frequency:")

print(time\_series)

Write a Pandas program to create a time-series with two index labels and random

values. Also print the type

import pandas as pd

import numpy as np

import datetime

from datetime import datetime, date

dates = [datetime(2011, 9, 1), datetime(2011, 9, 2)]

print("Time-series with two index labels:")

time\_series = pd.Series(np.random.randn(2), dates)

print(time\_series)

print("\nType of the index:")

print(type(time\_series.index))

Compare and contrast IDLE and Spyder?

From ppts

Defifine Reinforcement machine learning technique?

Reinforcement learning focuses on regimented learning processes, where a machine learning algorithm is provided with a set of actions, parameters and end values. By defining the rules, the machine learning algorithm then tries to explore different options and possibilities, monitoring and evaluating each result to determine which one is optimal. Reinforcement learning teaches the machine trial and error. It learns from past experiences and begins to adapt its approach in response to the situation to achieve the best possible result.

Defifine Supervised machine learning technique?

In supervised learning, the machine is taught by example. The operator provides the machine learning algorithm with a known dataset that includes desired inputs and outputs, and the algorithm must find a method to determine how to arrive at those inputs and outputs. While the operator knows the correct answers to the problem, the algorithm identifies patterns in data, learns from observations and makes predictions. The algorithm makes predictions and is corrected by the operator – and this process continues until the algorithm achieves a high level of accuracy/performance.

Under the umbrella of supervised learning fall: Classification, Regression and Forecasting.

**Classification**: In classification tasks, the machine learning program must draw a conclusion from observed values and determine towhat category new observations belong. For example, when filtering emails as ‘spam’ or ‘not spam’, the program must look at existing observational data and filter the emails accordingly.

**Regression**: In regression tasks, the machine learning program must estimate – and understand – the relationships among variables. Regression analysis focuses on one dependent variable and a series of other changing variables – making it particularly useful for prediction and forecasting.

**Forecasting**: Forecasting is the process of making predictions about the future based on the past and present data, and is commonly used to analyse trends.

Defifine Unsupervised machine learning technique?

Here, the machine learning algorithm studies data to identify patterns. There is no answer key or human operator to provide instruction. Instead, the machine determines the correlations and relationships by analysing available data. In an unsupervised learning process, the machine learning algorithm is left to interpret large data sets and address that data accordingly. The algorithm tries to organise that data in some way to describe its structure. This might mean grouping the data into clusters or arranging it in a way that looks more organised.

As it assesses more data, its ability to make decisions on that data gradually improves and becomes more refined.

Under the umbrella of unsupervised learning, fall:

**Clustering**: Clustering involves grouping sets of similar data (based on defined criteria). It’s useful for segmenting data into several groups and performing analysis on each data set to find patterns.

**Dimension reduction**: Dimension reduction reduces the number of variables being considered to find the exact information required.

Demonstrate how to get the items that are not common to both series X and series Y using pandas python?

**import** pandas as pd

**import** numpy as np

# create the series

ser1 **=** pd.Series([1, 2, 3, 4, 5])

ser2 **=** pd.Series([3, 4, 5, 6, 7])

# union of the series

union **=** pd.Series(np.union1d(ser1, ser2))

# intersection of the series

intersect **=** pd.Series(np.intersect1d(ser1, ser2))

# uncommon elements in both the series

notcommonseries **=** union[~union.isin(intersect)]

# displaying the result

print(notcommonseries)

Demonstrate how will you add a column to a pandas DataFrame?

# Import pandas package

**import** pandas as pd

# Define a dictionary containing Students data

data **=** {'Name': ['Jai', 'Princi', 'Gaurav', 'Anuj'],

        'Height': [5.1, 6.2, 5.1, 5.2],

        'Qualification': ['Msc', 'MA', 'Msc', 'Msc']}

# Convert the dictionary into DataFrame

df **=** pd.DataFrame(data)

# Using DataFrame.insert() to add a column

df.insert(2, "Age", [21, 23, 24, 21], True)

# Observe the result

print(df)

Describe Overfifitting, and How Can You Avoid It?

* Overfitting & underfitting are the two main errors/problems in the machine learning model, which cause poor performance in Machine Learning.
* Overfitting occurs when the model fits more data than required, and it tries to capture each and every datapoint fed to it. Hence it starts capturing noise and inaccurate data from the dataset, which degrades the performance of the model.
* An overfitted model doesn't perform accurately with the test/unseen dataset and can’t generalize well.
* An overfitted model is said to have low bias and high variance.

## Ways to prevent the Overfitting

Although overfitting is an error in Machine learning which reduces the performance of the model, however, we can prevent it in several ways. With the use of the linear model, we can avoid overfitting; however, many real-world problems are non-linear ones. It is important to prevent overfitting from the models. Below are several ways that can be used to prevent overfitting:

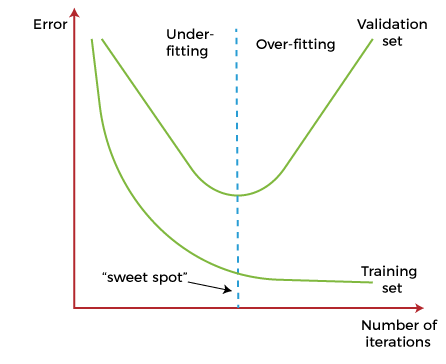
1. ****Early Stopping****
2. ****Train with more data****
3. ****Feature Selection****
4. ****Cross-Validation****
5. ****Data Augmentation****
6. ****Regularization****

### Early Stopping

In this technique, the training is paused before the model starts learning the noise within the model. In this process, while training the model iteratively, measure the performance of the model after each iteration. Continue up to a certain number of iterations until a new iteration improves the performance of the model.

After that point, the model begins to overfit the training data; hence we need to stop the process before the learner passes that point.

Stopping the training process before the model starts capturing noise from the data is known as ****early stopping.****



However, this technique may lead to the underfitting problem if training is paused too early. So, it is very important to find that "sweet spot" between underfitting and overfitting.

### Train with More data

Increasing the training set by including more data can enhance the accuracy of the model, as it provides more chances to discover the relationship between input and output variables.

It may not always work to prevent overfitting, but this way helps the algorithm to detect the signal better to minimize the errors.

When a model is fed with more training data, it will be unable to overfit all the samples of data and forced to generalize well.

But in some cases, the additional data may add more noise to the model; hence we need to be sure that data is clean and free from in-consistencies before feeding it to the model.

### Feature Selection

While building the ML model, we have a number of parameters or features that are used to predict the outcome. However, sometimes some of these features are redundant or less important for the prediction, and for this feature selection process is applied. In the feature selection process, we identify the most important features within training data, and other features are removed. Further, this process helps to simplify the model and reduces noise from the data. Some algorithms have the auto-feature selection, and if not, then we can manually perform this process.

### Cross-Validation

Cross-validation is one of the powerful techniques to prevent overfitting.

In the general k-fold cross-validation technique, we divided the dataset into k-equal-sized subsets of data; these subsets are known as folds.

### Data Augmentation

Data Augmentation is a data analysis technique, which is an alternative to adding more data to prevent overfitting. In this technique, instead of adding more training data, slightly modified copies of already existing data are added to the dataset.

The data augmentation technique makes it possible to appear data sample slightly different every time it is processed by the model. Hence each data set appears unique to the model and prevents overfitting.

### Regularization

If overfitting occurs when a model is complex, we can reduce the number of features. However, overfitting may also occur with a simpler model, more specifically the Linear model, and for such cases, regularization techniques are much helpful.

Regularization is the most popular technique to prevent overfitting. It is a group of methods that forces the learning algorithms to make a model simpler. Applying the regularization technique may slightly increase the bias but slightly reduces the variance. In this technique, we modify the objective function by adding the penalizing term, which has a higher value with a more complex model.

The two commonly used regularization techniques are L1 Regularization and L2 Regularization.

### Ensemble Methods

In ensemble methods, prediction from different machine learning models is combined to identify the most popular result.

The most commonly used ensemble methods are ****Bagging and Boosting.****

In bagging, individual data points can be selected more than once. After the collection of several sample datasets, these models are trained independently, and depending on the type of task-i.e., regression or classification-the average of those predictions is used to predict a more accurate result. Moreover, bagging reduces the chances of overfitting in complex models.

In boosting, a large number of weak learners arranged in a sequence are trained in such a way that each learner in the sequence learns from the mistakes of the learner before it. It combines all the weak learners to come out with one strong learner. In addition, it improves the predictive flexibility of simple models.

Describe the features of Python

Python provides many useful features which make it popular and valuable from the other programming languages. It supports object-oriented programming, procedural programming approaches and provides dynamic memory allocation. We have listed below a few essential features.

### 1) Easy to Learn and Use

Python is easy to learn as compared to other programming languages. Its syntax is straightforward and much the same as the English language. There is no use of the semicolon or curly-bracket, the indentation defines the code block. It is the recommended programming language for beginners.

### 2) Expressive Language

Python can perform complex tasks using a few lines of code. A simple example, the hello world program you simply type ****print("Hello World")****. It will take only one line to execute, while Java or C takes multiple lines.

### 3) Interpreted Language

Python is an interpreted language; it means the Python program is executed one line at a time. The advantage of being interpreted language, it makes debugging easy and portable.

### 4) Cross-platform Language

Python can run equally on different platforms such as Windows, Linux, UNIX, and Macintosh, etc. So, we can say that Python is a portable language. It enables programmers to develop the software for several competing platforms by writing a program only once.

### 5) Free and Open Source

Python is freely available for everyone. It is freely available on its official website [www.python.org](https://www.python.org/" \t "https://www.javatpoint.com/blank). It has a large community across the world that is dedicatedly working towards make new python modules and functions. Anyone can contribute to the Python community. The open-source means, "Anyone can download its source code without paying any penny."

### 6) Object-Oriented Language

Python supports object-oriented language and concepts of classes and objects come into existence. It supports inheritance, polymorphism, and encapsulation, etc. The object-oriented procedure helps to programmer to write reusable code and develop applications in less code.

### 7) Extensible

It implies that other languages such as C/C++ can be used to compile the code and thus it can be used further in our Python code. It converts the program into byte code, and any platform can use that byte code.

### 8) Large Standard Library

It provides a vast range of libraries for the various fields such as machine learning, web developer, and also for the scripting. There are various machine learning libraries, such as Tensor flow, Pandas, Numpy, Keras, and Pytorch, etc. Django, flask, pyramids are the popular framework for Python web development.

### 9) GUI Programming Support

Graphical User Interface is used for the developing Desktop application. PyQT5, Tkinter, Kivy are the libraries which are used for developing the web application.

### 10) Integrated

It can be easily integrated with languages like C, C++, and JAVA, etc. Python runs code line by line like C,C++ Java. It makes easy to debug the code.

### 11. Embeddable

The code of the other programming language can use in the Python source code. We can use Python source code in another programming language as well. It can embed other language into our code.

### 12. Dynamic Memory Allocation

In Python, we don't need to specify the data-type of the variable. When we assign some value to the variable, it automatically allocates the memory to the variable at run time. Suppose we are assigned integer value 15 to ****x,**** then we don't need to write ****int x = 15.**** Just write x = 15.

Design a program to display the fifibonacci sequences up to nth term where n is provided by the user

Or

Design a Python code to generate the Fibonacci series upto n.( 0 1 1 2 3 5....n)

# Program to display the Fibonacci sequence up to n-th term

nterms = int(input("How many terms? "))

# first two terms

n1, n2 = 0, 1

count = 0

# check if the number of terms is validif nterms <= 0:

print("Please enter a positive integer")# if there is only one term, return n1elif nterms == 1:

print("Fibonacci sequence upto",nterms,":")

print(n1)# generate fibonacci sequenceelse:

print("Fibonacci sequence:")

while count < nterms:

print(n1)

nth = n1 + n2

# update values

n1 = n2

n2 = nth

count += 1

Design Python code to solve the quadratic equation ax\*\*2 + bx + c = 0 by getting input for coeffiffifficients from the user.

# Python program to find roots of quadratic equation

**import** math

# function for finding roots

**def** equationroots( a, b, c):

    # calculating discriminant using formula

    dis **=** b **\*** b **-** 4 **\*** a **\*** c

    sqrt\_val **=** math.sqrt(abs(dis))

    # checking condition for discriminant

**if** dis > 0:

        print(" real and different roots ")

**print**((**-**b **+** sqrt\_val)**/**(2 **\*** a))

**print**((**-**b **-** sqrt\_val)**/**(2 **\*** a))

**elif** dis **==** 0:

**print**(" real and same roots")

        print(**-**b **/** (2 **\*** a))

    # when discriminant is less than 0

**else**:

        print("Complex Roots")

        print(**-** b **/** (2 **\*** a), " + i", sqrt\_val)

        print(**-** b **/** (2 **\*** a), " - i", sqrt\_val)

# Driver Program

a **=** 1

b **=** 10

c **=** **-**24

# If a is 0, then incorrect equation

**if** a **==** 0:

**print**("Input correct quadratic equation")

**else**:

    equationroots(a, b, c)

Differentiate between Classifification and Regression?

|  |  |
| --- | --- |
| Regression Algorithms | Classification Algorithms |
| The output variable must be either continuous nature or real value. | The output variable has to be a discrete value. |
| The regression algorithm’s task is mapping input value (x) with continuous output variable (y). | The classification algorithm’s task mapping the input value of x with the discrete output variable of y. |
| They are used with continuous data. | They are used with discrete data. |
| It attempt to find the best fit line, which predicts the output more accurately. | Classification tries to find the decision boundary, which divides the dataset into different classes. |
| Regression algorithms solve regression problems such as house price predictions and weather predictions. | Classification algorithms solve classification problems like identifying spam e-mails, spotting cancer cells, and speech recognition. |
| We can further divide Regression algorithms into Linear and Non-linear Regression. | We can further divide Classification algorithms into Binary Classifiers and Multi-class Classifiers. |

Differentiate between Supervised, Unsupervised and Reinforcement Learning

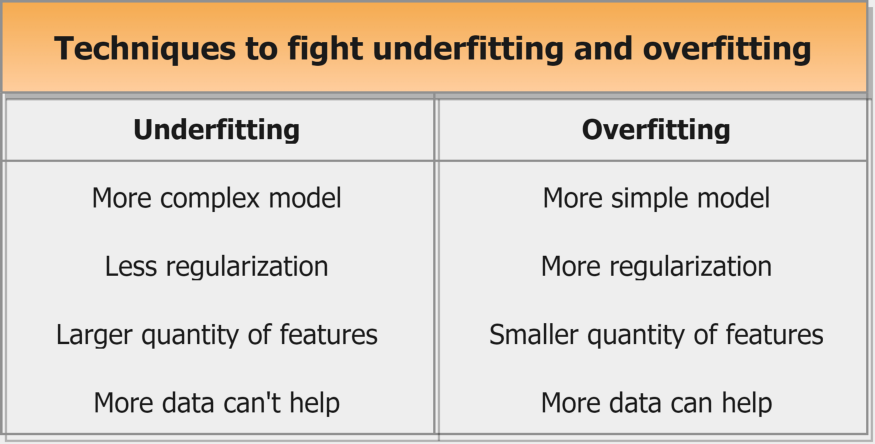
| **Criteria** | **Supervised ML** | **Unsupervised ML** | **Reinforcement ML** |
| --- | --- | --- | --- |
| Definition | Learns by using labelled data | Trained using unlabelled data without any guidance. | Works on interacting with the environment |
| Type of data | Labelled data | Unlabelled data | No – predefined data |
| Type of problems | Regression and classification | Association and Clustering | Exploitation or Exploration |
| Supervision | Extra supervision | No supervision | No supervision |
| Algorithms | Linear Regression, Logistic Regression, SVM, KNN etc. | K – Means, C – Means, Apriori | Q – Learning, SARSA |
| Aim | Calculate outcomes | Discover underlying patterns | Learn a series of action |
| Application | Risk Evaluation, Forecast Sales | Recommendation System, Anomaly Detection | Self Driving Cars, Gaming, Healthcare |

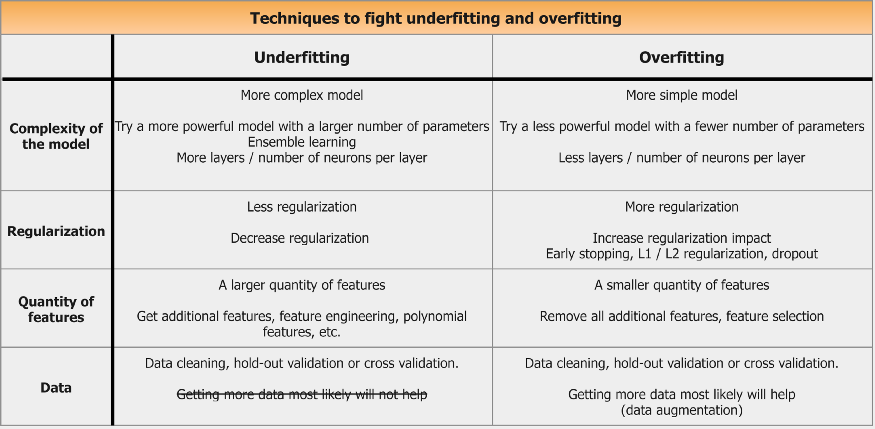
Differentiate between Training data and Testing Data

## Training data vs. Testing Data

* The main difference between training data and testing data is that training data is the subset of original data that is used to train the machine learning model, whereas testing data is used to check the accuracy of the model.
* The training dataset is generally larger in size compared to the testing dataset. The general ratios of splitting train and test datasets are ****80:20, 70:30, or 90:10.****
* Training data is well known to the model as it is used to train the model, whereas testing data is like unseen/new data to the model.

Differentiate between Underfifitting and Overfifitting.





Discuss Feature Selection?

feature Selection as, "**It is a process of automatically or manually selecting the subset of most appropriate and relevant features to be used in model building**." Feature selection is performed by either including the important features or excluding the irrelevant features in the dataset without changing them.

efore implementing any technique, it is really important to understand, need for the technique and so for the Feature Selection. As we know, in machine learning, it is necessary to provide a pre-processed and good input dataset in order to get better outcomes. We collect a huge amount of data to train our model and help it to learn better. Generally, the dataset consists of noisy data, irrelevant data, and some part of useful data. Moreover, the huge amount of data also slows down the training process of the model, and with noise and irrelevant data, the model may not predict and perform well. So, it is very necessary to remove such noises and less-important data from the dataset and to do this, and Feature selection techniques are used.

Selecting the best features helps the model to perform well. For example, Suppose we want to create a model that automatically decides which car should be crushed for a spare part, and to do this, we have a dataset. This dataset contains a Model of the car, Year, Owner's name, Miles. So, in this dataset, the name of the owner does not contribute to the model performance as it does not decide if the car should be crushed or not, so we can remove this column and select the rest of the features(column) for the model building.

Below are some benefits of using feature selection in machine learning:

* ****It helps in avoiding the curse of dimensionality.****
* ****It helps in the simplification of the model so that it can be easily interpreted by the researchers.****
* ****It reduces the training time.****
* ****It reduces overfitting hence enhance the generalization.****

## Feature Selection Techniques

There are mainly two types of Feature Selection techniques, which are:

* ****Supervised Feature Selection technique****  
  Supervised Feature selection techniques consider the target variable and can be used for the labelled dataset.
* ****Unsupervised Feature Selection technique****  
  Unsupervised Feature selection techniques ignore the target variable and can be used for the unlabelled dataset.

Discuss the basic Tuple operations with examples

A tuple is a collection of objects which ordered and immutable. Tuples are sequences, just like lists. The differences between tuples and lists are, the tuples cannot be changed unlike lists and tuples use parentheses, whereas lists use square brackets.

Tuples respond to the + and \* operators much like strings; they mean concatenation and repetition here too, except that the result is a new tuple, not a string.

In fact, tuples respond to all of the general sequence operations we used on strings in the prior chapter −

|  |  |  |
| --- | --- | --- |
| **Python Expression** | **Results** | **Description** |
| len((1, 2, 3)) | 3 | Length |
| (1, 2, 3) + (4, 5, 6) | (1, 2, 3, 4, 5, 6) | Concatenation |
| ('Hi!',) \* 4 | ('Hi!', 'Hi!', 'Hi!', 'Hi!') | Repetition |
| 3 in (1, 2, 3) | True | Membership |
| for x in (1, 2, 3): print x, | 1 2 3 | Iteration |

## Indexing, Slicing, and Matrixes

Because tuples are sequences, indexing and slicing work the same way for tuples as they do for strings. Assuming following input −

L = ('spam', 'Spam', 'SPAM!')

|  |  |  |
| --- | --- | --- |
| **Python Expression** | **Results** | **Description** |
| L[2] | 'SPAM!' | Offsets start at zero |
| L[-2] | 'Spam' | Negative: count from the right |
| L[1:] | ['Spam', 'SPAM!'] | Slicing fetches sections |

Python includes the following tuple functions −

|  |  |
| --- | --- |
| **Sr.No.** | **Function with Description** |
| 1 | [cmp(tuple1, tuple2)](https://www.tutorialspoint.com/python/tuple_cmp.htm)  Compares elements of both tuples. |
| 2 | [len(tuple)](https://www.tutorialspoint.com/python/tuple_len.htm)  Gives the total length of the tuple. |
| 3 | [max(tuple)](https://www.tutorialspoint.com/python/tuple_max.htm)  Returns item from the tuple with max value. |
| 4 | [min(tuple)](https://www.tutorialspoint.com/python/tuple_min.htm)  Returns item from the tuple with min value. |
| 5 | [tuple(seq)](https://www.tutorialspoint.com/python/tuple_tuple.htm)  Converts a list into tuple. |

Explain Feature Extraction in machine learning in brief

<https://www.javatpoint.com/feature-selection-techniques-in-machine-learning>

Explain logistic regression?

<https://www.javatpoint.com/logistic-regression-in-machine-learning>

Explain Receiver Operating Characteristic curve in brief?

<https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>

Explain the Confusion Matrix with Respect to Machine Learning Algorithms.

<https://www.javatpoint.com/confusion-matrix-in-machine-learning>

Explain the history of Python evolution

# Python History and Versions

* Python laid its foundation in the late 1980s.
* The implementation of Python was started in December 1989 by ****Guido Van Rossum**** at CWI in Netherland.
* In February 1991, ****Guido Van Rossum**** published the code (labeled version 0.9.0) to alt.sources.
* In 1994, Python 1.0 was released with new features like lambda, map, filter, and reduce.
* Python 2.0 added new features such as list comprehensions, garbage collection systems.
* On December 3, 2008, Python 3.0 (also called "Py3K") was released. It was designed to rectify the fundamental flaw of the language.
* ABC programming language is said to be the predecessor of Python language, which was capable of Exception Handling and interfacing with the Amoeba Operating System.
* The following programming languages influence Python:
  + ABC language.
  + Modula-3

Explain the List Slicing and List Mutability.

<https://www.geeksforgeeks.org/python-list-slicing/>

<https://bjc.edc.org/March2019/bjc-r/cur/programming/old-labs/python/list_mutability.html?topic=nyc_bjc%2FNA-python.topic>

Explain the Python Dictionary Comprehension with examples

<https://www.programiz.com/python-programming/dictionary-comprehension>

Explain the scope and lifetime of a variable in Python

<https://www.geeksforgeeks.org/python-scope-of-variables/#:~:text=In%20Python%2C%20variables%20are%20the,is%20not%20%E2%80%9Cstatically%20typed%E2%80%9D.>

Explain why python is considered as interpreted language?

 Python program runs directly from the source code.  
  
- Each time Python programs are executed code is required.  
  
- Python converts source code written by the programmer into intermediate language which is again translated into the native language / machine language that is executed. So Python is an Interpreted language.  
  
- It is processed at runtime by the interpreter.  
  
- The program need not be compiled before its execution.  
  
- It is similar to PERL and PHP.  
  
- Python is also interactive where it can prompt and interact with the interpreter directly to write the programs.  
  
- It supports the object-oriented style of the technique which encapsulates the code within the objects.

Give a short note on indexing and slicing?

<https://www.tutorialspoint.com/difference-between-indexing-and-slicing-in-python>

What do you mean by Concept Learning?

NOT FOUND

What do you understand by Feature Normalization? Explain

<https://www.javatpoint.com/normalization-in-machine-learning>

What Is a False Positive and False Negative and How Are They Signifificant?

NOT FOUND

What is a list in python? How to create a nested list? Demonstrate how to create and print a 3-dimentional matrix with list.

**<https://www.geeksforgeeks.org/python-creating-3d-list/>**

What is a tuple? How literals of type tuple are written?

A tuple literal is an expression that defines a [Tuple](https://docs.elementscompiler.com/Oxygene/Types/Tuples/) by providing a list of expressions, seperated by a comma (,) and surrounded by parenthesis (()). If assigned to a pre-defined tuple, the expressions need to match in type; when used in combination with type inference in a [var Statement](https://docs.elementscompiler.com/Oxygene/Statements/Var/), the individual types will be inferred.

var x := (15, 15, 12); // tuple of (Integer, Integer, Integer)var y: tuple of (Double, Double, Double) := (1, 1, 5); // tuple of 3 double types.

what is cloning of List?

<https://www.geeksforgeeks.org/python-cloning-copying-list/>

What is linear regression process?

<https://www.javatpoint.com/linear-regression-in-machine-learning#:~:text=Linear%20regression%20makes%20predictions%20for,hence%20called%20as%20linear%20regression.>

What is the difference between AUC and ROC?

From notes

What Python uses, static typing or dynamic typing? Justify your answer with an example.

<https://www.tutorialspoint.com/is-python-dynamically-typed-language#:~:text=Python%20is%20a%20dynamically%20typed,before%20assigning%20values%20to%20them.>

Where will you use Classifification over Regression? Explain

https://www.javatpoint.com/regression-vs-classification-in-machine-learning#:~:text=The%20main%20difference%20between%20Regression,Spam%20or%20Not%20Spam%2C%20etc.

Why is \* called string repetition operator?

[https://www.tutorialspoint.com/How-does-the-repetition-operator-work-on-a-tuple-in-Python#:~:text=The%20\*%20symbol%20is%20commonly%20used,can%20be%20extended%20to%20them.](https://www.tutorialspoint.com/How-does-the-repetition-operator-work-on-a-tuple-in-Python#:~:text=The%20*%20symbol%20is%20commonly%20used,can%20be%20extended%20to%20them.)

Write a Pandas program to demonstrate how to iterate over rows in Dataframes

import pandas as pd

import numpy as np

exam\_data = [{'name':'Anastasia', 'score':12.5}, {'name':'Dima','score':9}, {'name':'Katherine','score':16.5}]

df = pd.DataFrame(exam\_data)

for index, row in df.iterrows():

print(row['name'], row['score'])

Write note on data cleaning? Also discuss some data cleaning tools.

<https://www.geeksforgeeks.org/data-cleansing-introduction/>

Tools dekhne hai abhi

Write note on Regression and Classifification techniques in supervised machine learning?

From notes

Write python script to print current date and time

import datetime

now = datetime.datetime.now()

print ("Current date and time : ")

print (now.strftime("%Y-%m-%d %H:%M:%S"))