

```
result = df.transpose()
```

8. Pandas function for Shape and reshape Data

Shape:

The DataFrame. shape attribute in Pandas enables us to obtain the shape of a DataFrame.

For example, if a DataFrame has a shape of (80, 10), this implies that the DataFrame is made up of 80 rows and 10 columns of data.

Syntax: `dataframe.shape` or `dataframe.shape(5)`

Reshape:

Pandas has two methods that aid in reshaping the data into a desired format. Pandas has two methods namely, `melt()` and `pivot()`, to reshape the data.

`melt()`

Melt in pandas reshape dataframe from wide format to long format. It uses the “`id_vars[‘col_names’]`” for melt the dataframe by column names.

```
df_melt = df.melt(id_vars='Category')
df_melt
```

`pivot()`

This method does the reverse of what `melt()` did. It transforms the key-value pairs into columns. Reshape data (produce a “pivot” table) based on column values. Uses unique values from specified index / columns to form axes of the resulting DataFrame. This function does not support data aggregation, multiple values will result in a MultiIndex in the columns. See the User Guide for more on reshaping.

syntax: `pandas.pivot_table(data, values=None, index=None, columns=None)`

Example: `pivot_tab=pd.pivot_table(df, values='likes', index=['Type', 'Category'])`

Conclusion: Hence, students have performed about various operations using python on the facebook metrics datasets.

FAQ's:

- 1) What is datasets? Define Features and instances in datasets?
- 2) What are different data types in Pandas?
- 3) Explain Python libraries which used in Data Science?
- 4) What can you do with Pandas?
- 5) What can you do with NumPy?
- 6) What are the difference between ‘`loc()`’ and ‘`iloc()`’ functions.
- 7) Which methods are used for reshape data?

Source Code with output:



Assignment No. 02

Performance	Understanding	Regularity	Total	Dated Sign of Subject Teacher
03	01	01	05	

Date of Performance:**Date of Completion:****Title** Perform the various operations using Python on the Air quality and Heart disease data sets**Objectives:**

To understand and apply the Analytical concept of Big data using Python.

Problem Statement:

Perform the following operations using Python on the Air quality and Heart disease data sets

- Data cleaning
- Data Integration
- Data Transformation
- Error Correcting
- Data model building

Outcomes:

Students will be able to,

1. Apply the Analytical concept of Big data using Python.

Software and Hardware requirements:

1. Software: Ubuntu OS, Anaconda, Jupyter Notebook
2. Hardware: Processor, Ethernet Connection or WiFi, RAM 1GB, HDD, Sound Card, camera, microphone (depending upon website selection)

Theory:**1) Data Cleaning:**

Data cleaning is one part of data quality. The aim of Data Quality (DQ) is to have the following:

- Accuracy (data is recorded correctly)
- Completeness (all relevant data is recorded)
- Uniqueness (no duplicated data record)
- Timeliness (the data is not old)
- Consistency (the data is coherent)



Data cleaning attempts to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. Data cleaning is usually an iterative two-step process consisting of discrepancy detection and data transformation.

The process of data mining contains two steps in most situations. They are as follows:

- The first step is to perform audition on the source dataset to find the discrepancy.
- The second step is to choose the transformation to fix (based on the accuracy of the attribute to be modified and the closeness of the new value to the original value). This is followed by applying the transformation to correct the discrepancy

Methods of Data Cleaning:

There are many data cleaning methods through which the data should be run. The methods are described below:



1. **Ignore the tuples:** This method is not very feasible, as it only comes to use when the tuple has several attributes is has missing values.
2. **Fill the missing value:** This approach is also not very effective or feasible. Moreover, it can be a time-consuming method. In the approach, one has to fill in the missing value. This is usually done manually, but it can also be done by attribute mean or using the most probable value.
3. **Binning method:** This approach is very simple to understand. The smoothing of sorted data is done using the values around it. The data is then divided into several segments of equal size. After that, the different methods are executed to complete the task.
4. **Regression:** The data is made smooth with the help of using the regression function. The regression can be linear or multiple. Linear regression has only one independent variable, and multiple regressions have more than one independent variable.
5. **Clustering:** This method mainly operates on the group. Clustering groups the data in a cluster. Then, the outliers are detected with the help of clustering. Next, the similar values are then arranged into a "group" or a "cluster".

2) Data Integration:

Data integration combines data from multiple sources to form a coherent data store. The common issues here are as follows:

- **Heterogeneous data:** This has no common key
- **Different definition:** This is intrinsic, that is, same data with different definition, such as a different database schema



- **Time synchronization:** This checks if the data is gathered under same time periods
- **Legacy data:** This refers to data left from the old system
- **Sociological factors:** This is the limit of data gathering

There are several approaches that deal with the above issues

- **Entity identification problem:** Schema integration and object matching are tricky. This referred to as the entity identification problem.
- **Redundancy and correlation analysis:** Some redundancies can be detected by correlation analysis. Given two attributes, such an analysis can measure how strongly one attribute implies the other, based on the available data.
- **Tuples Duplication:** Duplication should be detected at the tuple level to detect redundancies between attributes
- **Data value conflict detection and resolution:** Attributes may differ on the abstraction level, where an attribute.

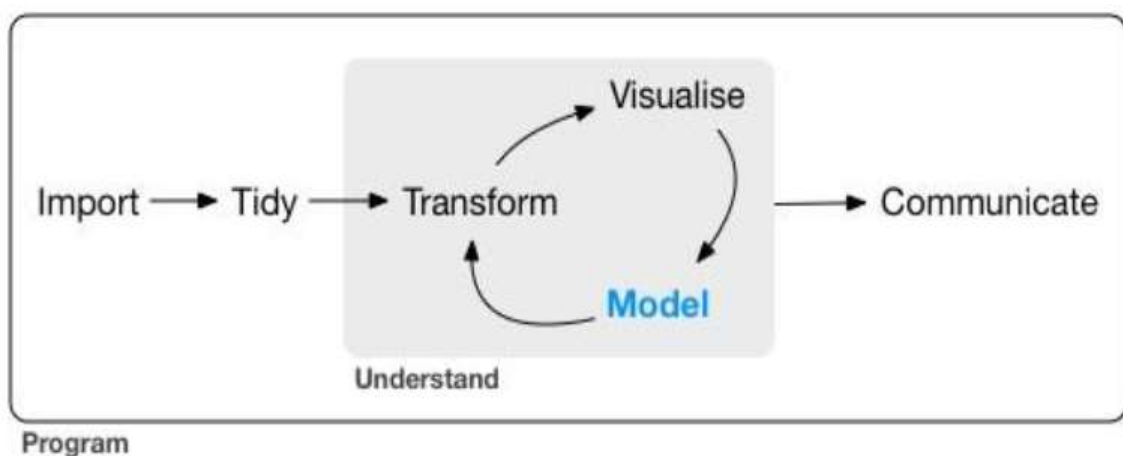
3) Data Transformation:

Data transformation is an approach to transform the original data to preferable data format for the input of certain data mining algorithms before the processing.

Data transformation routines convert the data into appropriate forms for mining. They're shown as follows:

- **Smoothing:** This uses binning, regression, and clustering to remove noise from the data
- **Attribute construction:** In this routine, new attributes are constructed and added from the given set of attributes
- **Aggregation:** In this summary or aggregation, operations are performed on the data
- **Normalization:** Here, the attribute data is scaled so as to fall within a smaller range
- **Discretization:** In this routine, the raw values of a numeric attribute are replaced by interval label or conceptual label
- **Concept hierarchy generation for nominal data:** Here, attributes can be generalized to higher level concepts

4) Data Model Building:



The goal of a model is to provide a simple low-dimensional summary of a dataset. Ideally, the model will capture true “signals” (i.e. patterns generated by the phenomenon of interest), and ignore “noise” (i.e. random variation that you’re not interested in). Here we only cover “predictive” models, which, as the name suggests, generate predictions and “data discovery” models. These models don’t make predictions, but instead help you discover interesting relationships within your data. (These two categories of models are sometimes called supervised and unsupervised).

If you are serious about doing a confirmatory analysis, one approach is to split your data into three pieces before you begin the analysis:

1. 60% of your data goes into a training (or exploration) set. You’re allowed to do anything you like with this data: visualize it and fit tons of models to it.
2. 20% goes into a query set. You can use this data to compare models or visualizations by hand, but you’re not allowed to use it as part of an automated process.
3. 20% is held back for a test set. You can only use this data ONCE, to test your final model.

This partitioning allows you to explore the training data, occasionally generating candidate hypotheses that you check with the query set. When you are confident you have the right model, you can check it once with the test data.

There are two parts to a model:

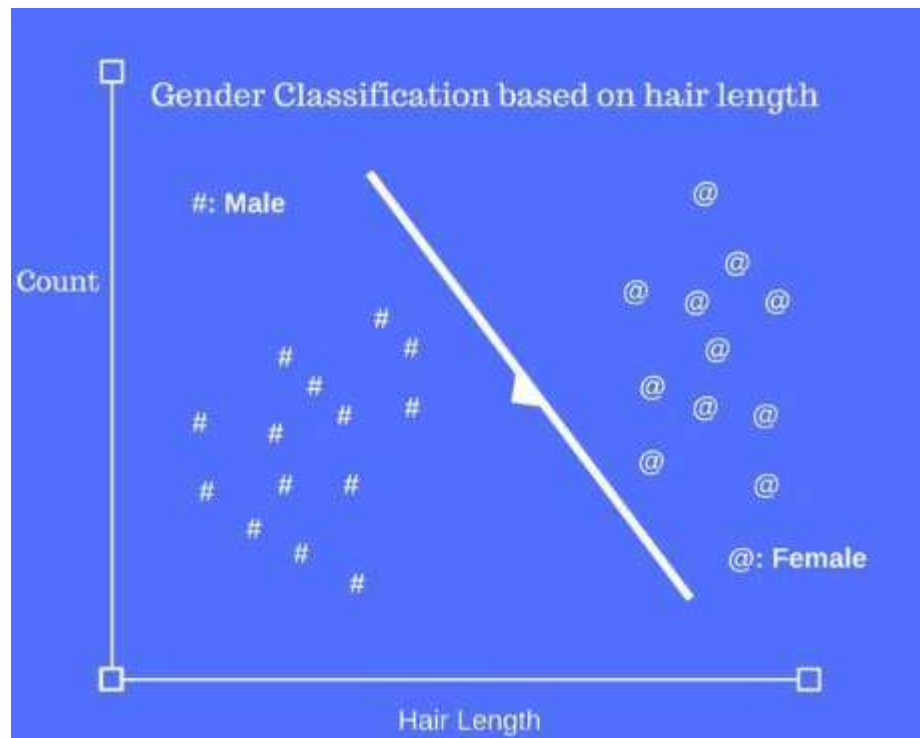
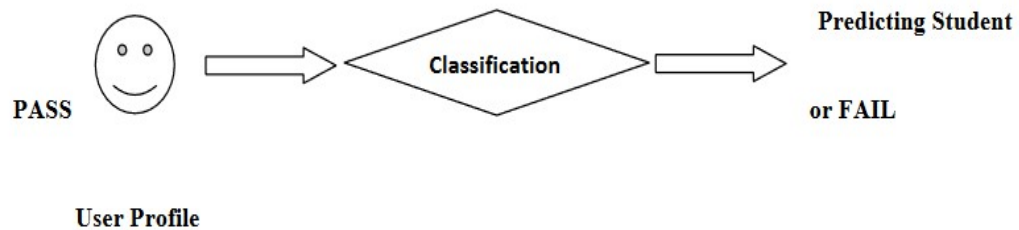
1. First, you define a **family of models** that express a precise, but generic, pattern that you want to capture. For example, the pattern might be a straight line, or a quadratic curve. You will express the model family as an equation like $y = a_1 * x + a_2$ or $y = a_1 * x^2 + a_2$. Here, x and y are known variables from your data, and a_1 and a_2 are parameters that can vary to capture different patterns.
2. Next, you generate a **fitted model** by finding the model from the family that is the closest to your data. This takes the generic model family and makes it specific, like $y = 3 * x + 7$ or $y = 9 * x^2$.

For model building any technique from machine learning can be used such as,

CLASSIFICATION:

The main goal of classification is to predict the target class (Yes/ No). If the trained model is for predicting any of two target classes. It is known as binary classification. Considering the student profile to predict whether the student will pass or fail. Considering the customer, transaction details to predict whether he will buy the new product or not. These kind problems will be addressed with binary classification. If we have to predict more the two target classes it is known as multi-classification. Considering all subject details of a student to predict which subject the student will score more. Identifying the object in an image. These kind problems are known as multi-classification problems.





Classification Algorithms:

- Linear classifiers
- Logistic regression
- Naive Bayes classifier
- Support vector machines
- Quadratic classifiers
- k-nearest neighbor (KNN)
- Decision trees
- Random Forest

Classification Example:

Suppose from your past data (**train data**) you come to know that your best friend likes the above movies. Now one new movie (**test data**) released. Hopefully, you want to know your best friend like it or not. If you strongly confirmed about the chances of your friend like the move. You can take your friend to a movie this weekend.

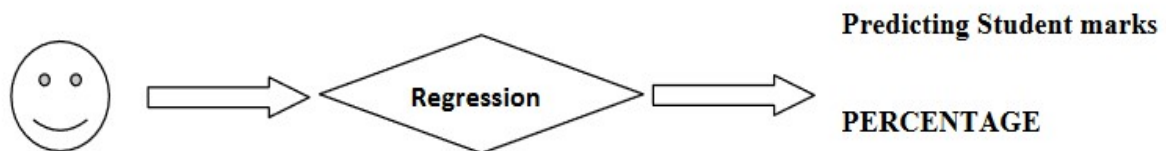


If you clearly observe the problem it is just whether your friend **like or not**. Finding a solution to this type of problem is called as classification. This is because we are classifying the things to their belongings (yes or no, like or dislike). Here we are forecasting **target class** (classification) and the other thing this **classification** belongs to **supervised learning**. This is because you are learning this from your **train data**.

In this case, the problem is a **binary classification** in which we have to predict whether output belongs to class 1 or class 2 (**class 1: yes, class 2: no**) We can use classification for predicting more classes too. Like (**Colour Prediction: RED, GREEN, BLUE, YELLOW, And ORANGE**)

REGRESSION:

The main goal of regression algorithms is the predict the discrete or a continues value. In some cases, the predicted value can be used to identify the linear relationship between the attributes. Suppose the increase in the product advantage budget will increase the product sales. Based on the problem difference regression algorithms can be used. Some of the basic regression algorithms are linear regression, polynomial regression, and multiple regressions.



User Profile

Regression Example:

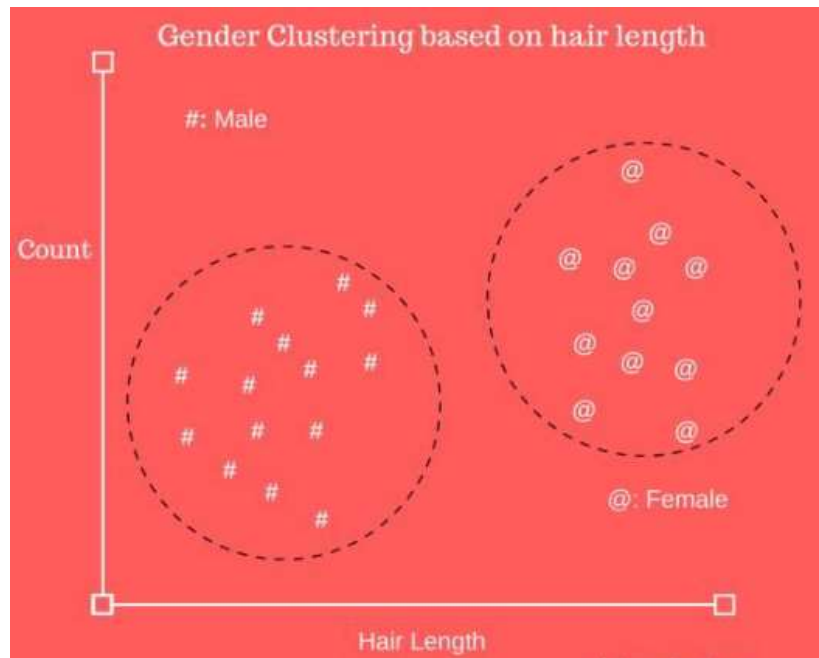
Suppose from your past data (train data) you come to know that your best friend likes some movies. You also know how many times each particular movie seen by your friend. Now one new movie (test data) released. Now you're are going to find how many times this newly released movie will your friend watch. It could be 5 times, 6 times, 10 times etc... If you clearly observe the problem is about finding the count, sometimes we can say this as predicting the value. Here we are forecasting a value (Prediction) and the other thing this prediction also belongs to supervised learning. This is because you are learning this from you train data.

- If forecasting target class (Classification)
- If forecasting a value(Regression)

Clustering:

In clustering the idea is not to predict the target class as like classification, it's more ever trying to group the similar kind of things by considering the most satisfied condition all the items in the same group should be similar and no two different group items should not be similar. To group the similar kind of items in clustering, different **similarity measures** could be used.



**Group items Examples:**

- While grouping similar language type documents (Same language documents are one group.)
- While categorizing the news articles (Same news category(Sport) articles are one group)

With clustering genders based on hair length example. To determine gender, different similarity measure could be used to categorize male and female genders. This could be done by finding the similarity between two hair lengths and keep them in the same group if the similarity is less (Difference of hair length is less). The same process could continue until all the hair length properly grouped into two categories.

Clustering Algorithms:

Clustering algorithms can be classified into two main categories linear clustering algorithms and Non-linear clustering algorithms

- Linear clustering algorithm
 - k-means clustering algorithm
 - Fuzzy c-means clustering algorithm
 - Hierarchical clustering algorithm
 - Gaussian(EM) clustering algorithm
 - Quality threshold clustering algorithm
- Non-linear clustering algorithm
 - MST based clustering algorithm
 - kernel k-means clustering algorithm
 - Density-based clustering algorithm

Application of Clustering Algorithms

- Recommender systems
- Anomaly detection
- Human genetic clustering
- Genom Sequence analysis
- Analysis of antimicrobial activity
- Grouping of shopping items
- Search result grouping
- Slippy map optimization
- Crime analysis
- Climatology

FAQ:

- 1) What do you understand from the term data cleaning?
- 2) What is Data Integration? What are the benefits of data integration?
- 3) Is Data integration And ETL programming is same?
- 4) Mention what is the responsibility of a Data analyst?
- 5) Mention what is data cleansing?

Viva Questions

- 1) List of some best tools that can be useful for data-analysis?
- 2) List out some common problems faced by data analyst?
- 3) Which functions are included in package caret, e1071, catools, class and gmodels name it
- 4) How to handle missing values?
- 5) How do you create log linear models in Python language?
- 6) What is meant by K-nearest neighbor explain with example?
- 7) Write a function in Python language to replace the missing value in a vector with the mean of that vector.
- 8) Which function is used to create histogram visualization in Python programming language?

