

**PROJECT REPORT ON**

**DATA SCIENCE AUTOMATION TOOL**

Submitted By

**PRATIK URKUDE**

Batch

**INNOMINIONS-OCT-21**

**DATA SCIENCE AUTOMATION TOOL**

**Introduction**

Data science is an amalgamation of different scientific methods, algorithms and systems which enable us to gain insights and derive knowledge from data in various forms. Although data science techniques have been conceptualized and in use for several decades now, the current demand for data science is fuelled by the high availability of digital data, and resources for computation.

Data Science Automation Tool is a Data Science process automation application created using web technologies like HTML, CSS, Python and Flask. Given any dataset, the application is able to perform the tasks that are as part of a normal Machine Learning model. This application can be used in order to reduce the work load by reducing the effort to write codes.

Given any dataset, the application can perform the below mentioned tasks –

1. Read the dataset

2. Display the dataset information

3. Data Pre-processing and Cleaning

a. Type Conversion

b. Missing Value Treatment

c. Outlier Treatment

d. Feature Transformation and Scaling

e. Train – Test Split

4. Model Building

5. Hyper Parameter Tuning

6. Evaluation Metrics

7. Testing

**Purpose and goal**

The purpose and goal of the project is to build an automated web application where for any given dataset, it can automate and able to perform the functions that a normal machine learning algorithm can do. Effectively, it reduces the work load thereby decrease writing codes to perform data analysis and model building.

**Pre – requirements**

1. Python

2. Flask

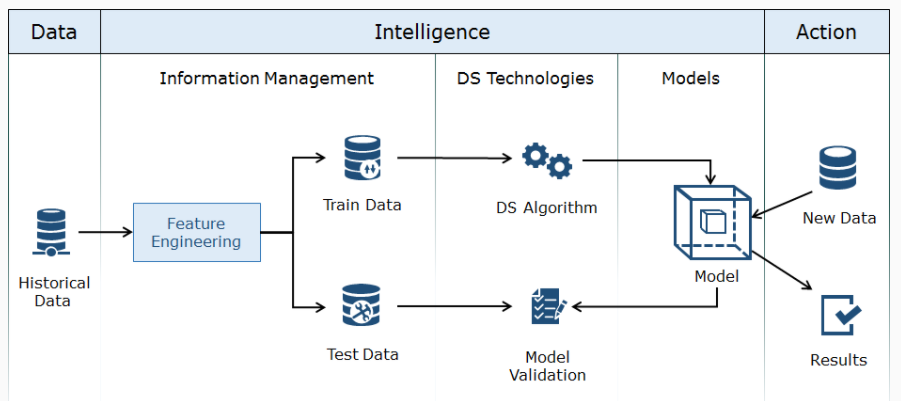
3. Web Technologies – HTML, CSS

4. Exploratory Data Analysis

5. Python Libraries – NumPy, Pandas, Scikit – Learn

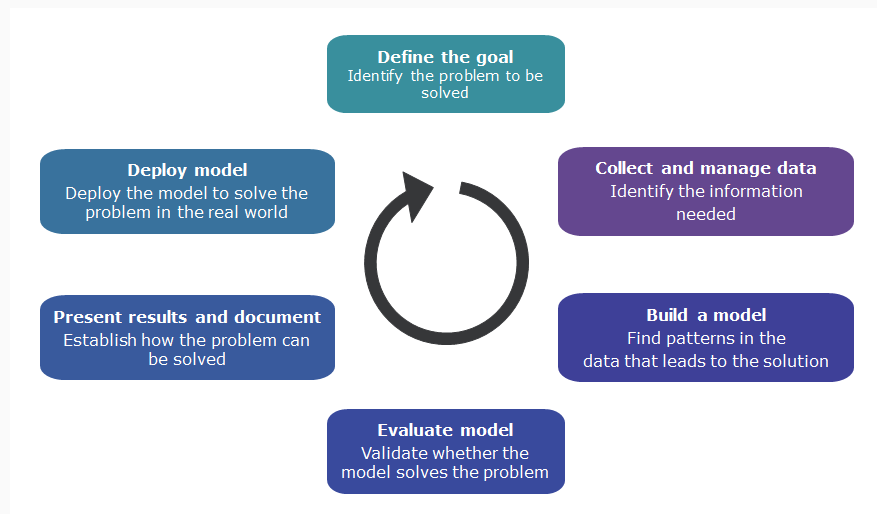
6. Machine Learning

**Data Science Process and Architecture**



Historical data or past data from various sources are cleaned and subjected to Feature Engineering. Feature Engineering is the process of using domain knowledge to select or create significant features from the historical data relevant to the problem statement. This engineered data is divided into two sets: Train data and Test data. Data science models are built using train data, and then the performance of the model is evaluated on the test data. This validated model is used for taking various decisions on new/unseen data points.

**Data Science Project Life Cycle**



Step 1: Define the goal

The first and foremost step in any project is to define a clear goal. Hence, at this point, it is important to learn every minute detail about the project such as:

1. Why is the project being started? What is missing currently and what exactly is required?

2.What all resources are needed? What kind of data is available? Is domain expertise available within the team? What are the computational resources available/required?

Step 2: Collect and Manage Data

Now that the goal has been set, the next step is to find, explore and clean the data necessary for analysis. This stage takes up a lot of time but helps in finding answers to many important questions, such as:

1. What all data is available?

2. Will it help in solving the problem?

3. Is the data enough to carry out analysis?

4. Is the quality of data up to the mark?

Step 3: Build a model

Once the data is ready, the next step is to find meaningful insights from the data. Depending on the nature of the business problem we are dealing with we can make use of any of the following data modelling techniques to gather such insights.

Step 4: Model Evaluation

Now that we have built our model, we need to determine whether it meets our goals by asking the following questions:

1. Is the model accurate enough for our needs?

2. Does the model meet the expectations? Is it better than the methodology being currently used?

 If for either of the above questions, the answer is NO, we need to revisit the previous steps.

Step 5: Present results and document

At this stage we have achieved a desirable model. A model that meets all the requirements and goals we set for ourselves at the beginning of the project.

The next step is to showcase the project to various audience as follows:

1. Present the details of the model to all the collaborators, clients and sponsors.

2. Provide everyone in charge of usage and maintenance of the model, once deployed, with documentation that covers all aspects of the working of the model.

Hence, specific data visualization techniques must be used for each of them.

Step 6: Deploy model

The last and final step is to deploy the model. But before they are off the job, they must make sure that the following are in place:

1. The model has been tested thoroughly and generalizes well.

2. The model should be able to adjust well to unforeseen environmental changes.

**Technologies Used**

**HTML** stands for**Hyper Text Markup Language**. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. It is a markup language that is used by the browser to manipulate text, images, and other content to display it in the required format.

**CSS** stands for Cascading Style Sheets. It is a stylesheet language used to design the webpage to make it attractive. The reason for using CSS is to simplify the process of making web pages presentable. It allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.

**Python** is a high-level, general-purpose and a very popular programming language. It is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry. Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. The biggest strength of Python is huge collection of standard library which can be used for Machine Learning, GUI Applications, Web Frameworks, Web Scraping, Scientific Computing and many more.

**Flask** is an API of Python that allows us to build up web-applications. It has less base code to implement a simple Web-Application. A Web-Application Framework or Web Framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as protocols, thread management, etc. Flask is based on WSGI (Web Server Gateway Interface) toolkit and Jinja2 template engine.

**Exploratory Data Analysis (EDA)**is an approach to analyse the data using visual techniques. It is used to discover trends, patterns, or it check assumptions with the help of statistical summary and graphical representations.

**Machine Learning** is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn.

**NumPy**is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.

**Pandas** is an open-source library that is built on top of NumPy library. It is a Python package that offers various data structures and operations for manipulating numerical data and time series.

It is mainly popular for importing and analysing data much easier. Pandas is fast and it has high-performance & productivity for users.

**Scikit – Learn** is an open-source Python library that implements a range of machine learning, pre-processing, cross-validation, and visualization algorithms using a unified interface. It is built on the top of NumPy, SciPy, and Matplotlib.

**Implementation**

When a dataset is uploaded into the application, it provides the shape of the dataset, column names in the dataset and the information of the dataset.

When you start the data analysis, several options like type conversion, missing value treatment, feature transformation, outlier detection and train-test split options appear to choose.

Each of the functionality is described below.

1. Type conversion

The type conversion function converts the original column datatypes into best possible datatypes for the respective columns.

2. Missing value treatment

The dataset may contain some missing values. This treatment can be done in four ways.

a. Drop missing value rows

This function removes the rows containing missing values.

b. Drop missing value columns

This function removes the columns containing missing values.

c. Impute with mean

This function imputes the missing values in a numerical column with its mean value. It also imputes the missing values in a categorical column with its mode value.

d. Impute with median

This function imputes the missing values in a numerical column with its median value. It also imputes the missing values in a categorical column with its mode value.

3. Outlier Detection

This function detects the outliers in the dataset and treats them in two ways.

a. Remove outliers

This function removes the rows containing outliers from the dataset.

b. Replace outliers with median

This function replaces the outliers in a numerical column with its median value.

4. Feature Transformations and Scaling

Feature pre-processing is one of the most crucial steps in building a Machine learning model. Sometimes feature transformation is the only way to gain a better score, so it is a crucial point how you represent your data and feed it to a target model.

Feature transformation is a mathematical transformation in which we apply a mathematical formula to a particular column(feature) and transform the values which are useful for our further analysis. Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units.

There are various techniques like Min Max Scaling, Robust Scaling, Log Transformation, Max Abs Scaling, Label Encoding and many more.

5. Train – Test Split

Training data builds up the machine learning algorithm. The data scientist feeds the algorithm input data, which corresponds to an expected output. The model evaluates the data repeatedly to learn more about the data’s behaviour and then adjusts itself to serve its intended purpose.

After the model is built, testing data once again validates that it can make accurate predictions. If training and validation data include labels to monitor performance metrics of the model, the testing data should be unlabelled. Test data provides a final, real-world check of an unseen dataset to confirm that the ML algorithm was trained effectively.

This function splits the dataset into train and test data according to the user inputs. Generally, the train – test split will be 75: 25.

Once the dataset is cleaned, we start building a model. The model is chosen based on the type of the target feature. Generally, there are three types of models.

1. Regression models

A regression model provides a function that describes the relationship between one or more independent variables and a response, dependent, or target variable. It indicates the strength of impact of multiple independent variables on a dependent variable. The target will be a continuous or a numerical column. Various regression models include Linear Regression, Random Forest Regression, Bayesian Ridge Regression, Lasso Regression and many more.

2. Classification models

A classification model attempts to draw some conclusion from observed values. Given one or more inputs a classification model will try to predict the value of one or more outcomes. Outcomes are labels that can be applied to a dataset. The target will be a categorical column. Various classification models include Logistic Regression, Decision Trees, Random Forest, K Nearest Neighbours and many more.

3. Clustering models

**Clustering** is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them. Clustering is very much important as it determines the intrinsic grouping among the unlabelled data present. Various clustering models include K Means Clustering, Agglomerative Clustering, DBSCAN Clustering, OPTICS Clustering and many more.

When one of the models is chosen, we need to give some inputs for hyperparameters. In statistics, hyperparameter is a parameter from a prior distribution; it captures the prior belief before data is observed. In any machine learning algorithm, these parameters need to be initialized before training a model. They are the properties that govern the entire training process. They are important because they directly control the behaviour of the training algorithm and have a significant impact on the performance of the model is being trained*.*

The model is then build using the model functions from the sci-kit library. The model is applied on the train data and predictions are generated using the test data. After the model is applied, we get the accuracy of the model along with the evaluation metrics for the model.

Evaluation metrics are used to measure the quality of the statistical or machine learning model. Evaluating machine learning models or algorithms is essential for any project. It is very important to use multiple evaluation metrics to evaluate your model. This is because a model may perform well using one measurement from one evaluation metric, but may perform poorly using another measurement from another evaluation metric. Using evaluation metrics are critical in ensuring that your model is operating correctly and optimally.

Various metrics include classification report, confusion matrix for classification models, explained variance, mean squared error, r2 score for regression models, silhouette score, Calin ski harabasz score, Dunn index for clustering models and many more.

**Summary**

As far our experiments are concerned, our application is able to take in any kind of dataset and able to perform exploratory data analysis and finally able to build a model with hyperparameter tuning with respect to user inputs and generated the accuracy and evaluation metrics of the model.

**Results**

