PROJECT REPORT
on
AutoVizML

NAS1001 - NASSCOM Future Skills- Associative Data Analyst
[E21+E22+E23 slot]

Submitted By:
19MIM10024 - Raghav Agarwal
19MIM10046 - Indrashis Paul
19MIM10088 - Kshitij Singh Chouhan
19MIM10038 - Anshika Mishra
19MIM10111 - Shashwat Jha
19MIM10078 - Raunak Ansari

Guided By:
Dr. Nilamadhab Mishra

Index table:

Introduction

There are many different aspects to learning. Declarative learning, the growth of physical and cognitive skills through teaching or practice, the organizing of new knowledge into universal, useful representations, and the discovery of novel facts and hypotheses through observation and experimentation are all examples of learning processes. Researchers have been working to incorporate these capabilities into computers since the dawn of the computer era. It has been and still is a challenging and interesting long-term objective in artificial intelligence to solve this issue (AI). Machine learning is the study and computer modeling of learning processes in their varied manifestations.

Due to the lack of data scientists, automated machine learning as a concept will become a popular issue in 2022. More than ever, it's critical to enhance productivity by having existing data scientists automate as much of their work as possible and get new data scientists up and running as quickly as possible. AutoML lets data scientists build models very quickly and also lets new data scientists on the ramp very quickly. They can concentrate on learning how to prepare data for AutoML algorithms rather than having to learn how to do so for each algorithm.

Problem Statement

With about 1,000 petabytes of data being generated each day across the globe, the need of extracting something useful out of it is the need of the hour. A trained domain expert can easily look into the data, find insights out of it and even model it to predict the predictions. But, an individual with no domain knowledge can not handle this data. Therefore, there is a need for a platform upon which anyone can choose the dataset, explore it to find some insights, and even model it to predict the predictions.

Existing work and Literature Review

There are a few available solutions out there and one of the best ones is a custom web application written in R programming language and uses a shiny user interface known as Radiant.

Radiant is a highly optimized web application that provides a wide variety of machine learning models and analytical tools which can be used with just some clicks using which any person with no prior knowledge of these tools working can extract some key information and can find some valuable insights that normal base level tools can't provide. And the most amazing thing is that it does not require a single line of code from the user. Radiant has a high dimensionality which makes it for everyone, which means it will prove to be a useful application for someone new in the field as well as for the experts in the field. It is a very useful application as it helps to displace some very long and repetitive processes and has a collection of some special tools collectively in one place. Radiant can also handle a large amount of data and process it efficiently as the user wants. It can easily channel the data through the required selection that the user selects.

If we talk about some more background proofed/ trusted solutions we take Google Cloud AutoML. It is a little bit different when we compare it to our last solution as it provides the power of machine learning to the user even if the user is having very little knowledge about machine learning. It has features like AutoMl Tables which allow users to build top-of-the-line machine learning models on structured data and also it has some advanced features which help like AutoML Vision and Video which increase the user reach to the graphical data as well.

Proposed Solution

Machine Learning is a complex process that requires a fair amount of technical knowledge even to build little dummy models. Our platform "AutoVizML" is a platform built using the Shiny package present in R programming language, that helps the user to automate the data visualization and model training part. Our platform has the following web pages:

1. **About page:**
   The about section gives a brief overview of our project that is about this application. It states the Steps to complete and evaluate a machine learning model. And also has a link to our source code and some sample data.

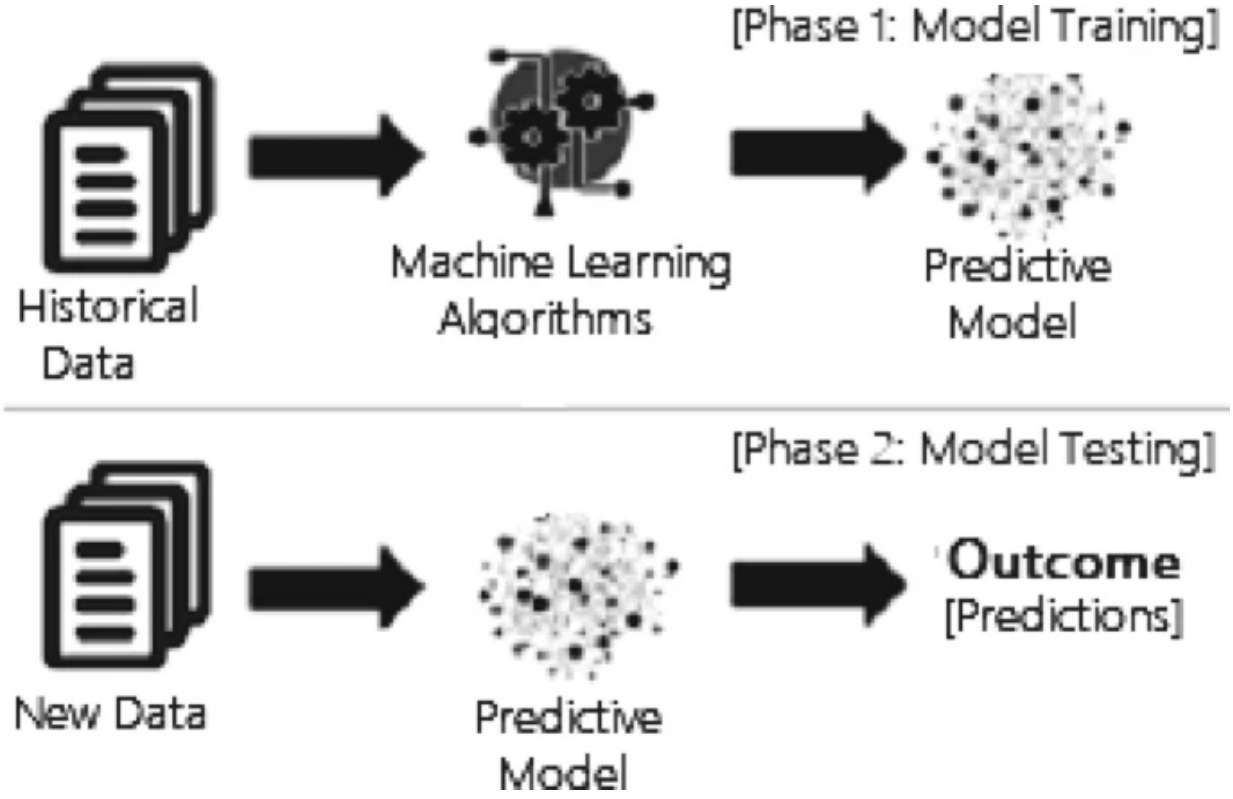
2. **Data Upload page:**
   This page consists of three sections:
   a. Data Upload
   b. Select the Outcome/Target Variable
   c. Data Table

3. **Data Summary page:**
   This page consists of three sections:
   a. Drop Columns
   b. Preprocess Data
   c. Data Description table

4. **Data Visualization page:**
   On this page, the user can select the type of the plot, the variable on the x-axis, the variable on the y-axis, and the variable to color by. Once these four values are selected our application will display the plot according to the user selections.

5. **Build Model Page:**
   In this page, we have used machine learning algorithms that include classification analysis, regression analysis, data clustering, association rule

learning, and feature engineering for dimensionality reduction. A general structure of a machine learning-based predictive model has been shown in the Figure given below where the model is trained from historical data in phase 1 and the outcome is generated in phase 2 for the new test data.



There are two types of methods:

1. Classification Analysis
2. Regression Analysis

K-Nearest Neighbors, Generalized Linear Model(logit), Random Forests, Gradient Boosting, and Naive Bayes are some of the machine learning algorithms used.

6. Model Evaluation page:

Different machine learning algorithms search for different trends and patterns. One algorithm isn't the best across all data sets or for all use

cases. To find the best solution, you need to conduct many experiments, evaluate machine learning algorithms, and tune their hyperparameters:

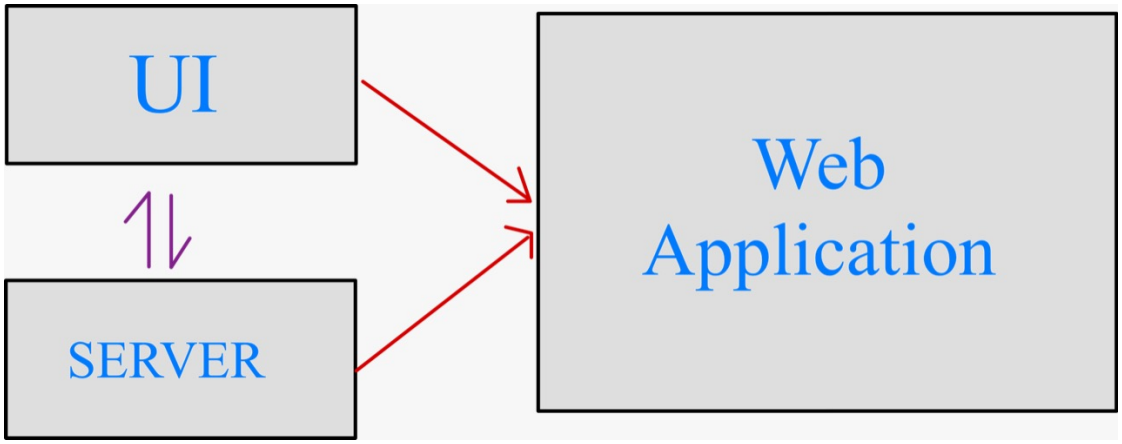
Model evaluation is the process of using different evaluation metrics to understand a machine learning model's performance, as well as its strengths and weaknesses. Model evaluation is important to assess the efficacy of a model during the initial research phases, and it also plays a role in model monitoring.

To understand if your model(s) is working well with new data, you can leverage a number of evaluation metrics.

The most popular metrics for measuring classification performance include accuracy, precision, and confusion matrix.

- **Accuracy** measures how often the classifier makes the correct predictions, as it is the ratio between the number of correct predictions and the total number of predictions.
- **Precision** measures the proportion of predicted Positives that are truly Positive. Precision is a good choice of evaluation metrics when you want to be very sure of your prediction.
- The **confusion matrix** (or confusion table) shows a more detailed breakdown of correct and incorrect classifications for each class.

Application Architecture DiagramUser Work Flow:



Methodology and Workflow explanation

The Automated Platform that is used is the Shiny App which helps users to automate data visualization and model training evaluation on different machine learning models as applied to the selected Data of Choice.

For evaluating a machine learning model, the following steps need to be followed:

1. Select the data - choose the data you want to use.
2. Examine the data summary - see what is in the data.
3. Explore the data - see what features to use in a model.
4. Build a prediction model - pre-process data, select features, and generate model.
5. Evaluate the prediction model - estimate in-sample and out-of-sample errors.
6. Predict outcomes for test data.

1. Upload

For Choosing the data, we have given two options:

- Choose a dataset from the drop-down menu, which contains a list of all the available datasets, first.
- The user can update their own dataset file in CSV format by utilizing the upload button, which is the second option.

2. Data Summary

In this page we are going to summarize the data following some steps:

- **Drop columns:** Under this function, we can select whatever column from the data we want to drop, and here we can see all the columns that are available in our dataset.

- **Pre-processing the data**: In this function, we are simply using the common pre-processing technique on the dataset. There are two functions that are used:

    1. Data preparation with the missing dataset: If any data has a missing value or NAN so it automatically removes and returns the data.
    2. Label encoder function: this functionality is used if there is any categorical dataset so that it converts into 0 and 1.

After pre-processing the data we can see the all features of attributes as well as we can see the drop column as a target.

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Glucose	1	768.00	120.89	31.97	117.00	119.38	29.65	0.00	199.00	199.00	0.17	0.62	1.15
BloodPressure	2	768.00	69.11	19.36	72.00	71.36	11.86	0.00	122.00	122.00	-1.84	5.12	0.70
SkinThickness	3	768.00	20.54	15.95	23.00	19.94	17.79	0.00	99.00	99.00	0.11	-0.53	0.58
Insulin	4	768.00	79.80	115.24	30.50	56.75	45.22	0.00	846.00	846.00	2.26	7.13	4.16
BMI	5	768.00	31.99	7.88	32.00	31.96	6.82	0.00	67.10	67.10	-0.43	3.24	0.28
DiabetesPedigreeFunction	6	768.00	0.47	0.33	0.37	0.42	0.25	0.08	2.42	2.34	1.91	5.53	0.01
Age	7	768.00	33.24	11.76	29.00	31.54	10.38	21.00	81.00	60.00	1.13	0.62	0.42
Outcome	8	768.00	0.35	0.48	0.00	0.31	0.00	0.00	1.00	1.00	0.63	-1.60	0.02

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Pregnancies	1.00	768.00	3.85	3.37	3.00	3.46	2.97	0.00	17.00	17.00	0.90	0.14	0.12

3. Exploring the data

In Exploring the dataset, EDA's primary goals are to find errors and outliers in the data as well as to recognize various patterns. It enables Analysts to comprehend the data more thoroughly before assuming anything.

Using statistical graphs and other visualization tools, data scientists and analysts look for various patterns, relationships, and anomalies in the data.

First, we have to upload the Dataset(given the file is in CSV, Excel, and JSON format). The dataset can be selected from the drop-down menu or the user can upload their own dataset.

Then we will check for any missing data/columns or outliers present in the dataset. If present, remove all the missing columns and outliers. Sometimes data given to you is coming from a data entry form that has default values like select, none, all, etc. While filling out the form if a user does not select anything then these values go into the database directly. In fact, they are null values. We need to treat them properly.

After cleaning our data and searching for all null/missing values present in the columns, We will find the dataset more balanced and clean. But if you still observe some imbalance or distribution issue then you can take a call and perform previous steps once again on this clean dataset. Perform this analysis to know whether you should move ahead or once again iterate the data cleaning cycle.

Data distribution of each field. – Univariate analysis

Categorical Features - Univariate analysis

Distribution of each continuous data field for target field (continuous) – Univariate analysis

Distribution of each continuous data field for each value of the target field (categorical). – Multivariate analysis

Distribution of each categorical data field for the target field (continuous). – Multivariate analysis

Distribution of each Categorical data field for each value of the target field (categorical). – Multivariate analysis

Check the co-linearity of all fields. – Bivariate analysis.

Now we will be comparing different variables on both X-axis and Y- the axis and forming different plots accordingly. First, we will be selecting the type of graph the user need and then mention the variables on the X-Axis and Y-Axis. For Graphical representation of the variables, we will be using the plots mentioned below:

- PointPlot

  A point plot uses the position of the dot to indicate an estimate of the central tendency for a numerical variable, and error bars are used to show the degree of uncertainty surrounding that estimate.

- Histogram

The location, spread, and skewness of a dataset are all visually represented by a histogram, which also makes it easier to see if the distribution is symmetric or left- or right-skewed. If it is unimodal, bimodal, or multimodal as well. Additionally, it can highlight any anomalies or data gaps.

- Boxplot

  A box plot, also known as a box-and-whisker plot, illustrates the distribution of quantitative data in a form that makes it easy to compare one variable to

another. The box displays the dataset's quartiles, while the whiskers expand to display the remaining distribution.

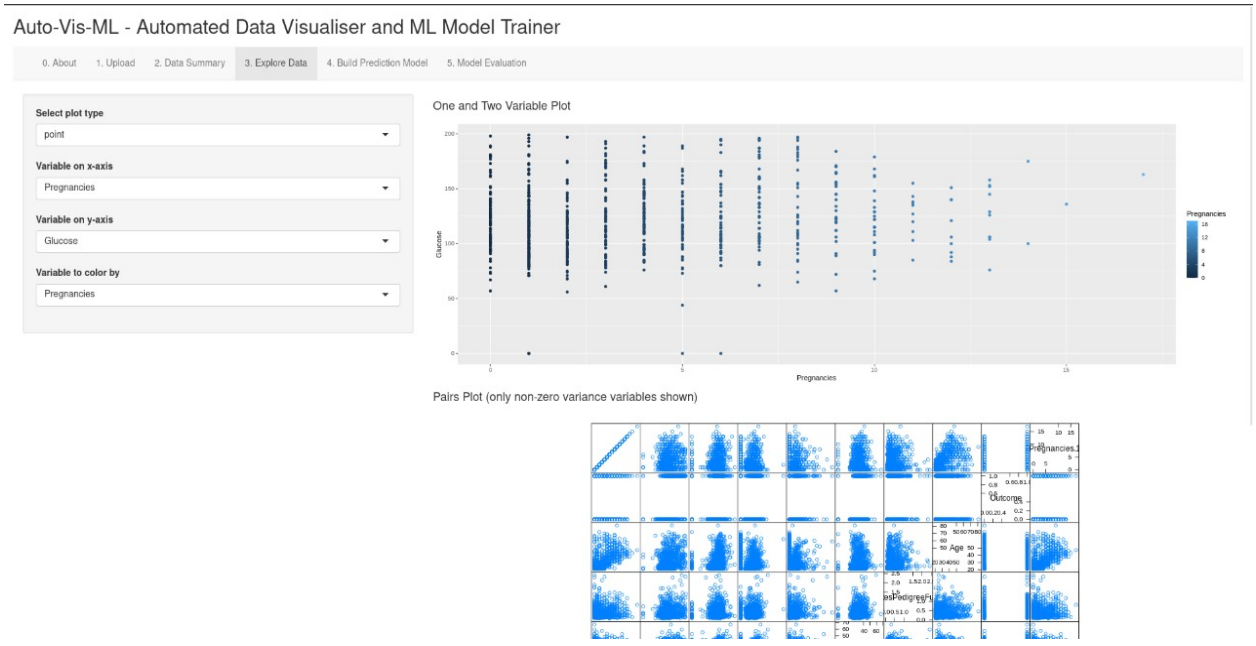
- Density Plot

A density plot is a representation of a numeric variable's distribution that displays the probability density function of the variable using a kernel density estimate. The density() function in the R language is used to estimate kernel densities.

- Jitter

  In a manner similar to a scatter plot, a jitter plot displays data points as single dots. The jitter plot, on the other hand, makes it easier to see how a measurement variable and a categorical variable relate to one another.

The pairs plot is also mentioned below, which is a Static Graph for all the variables.



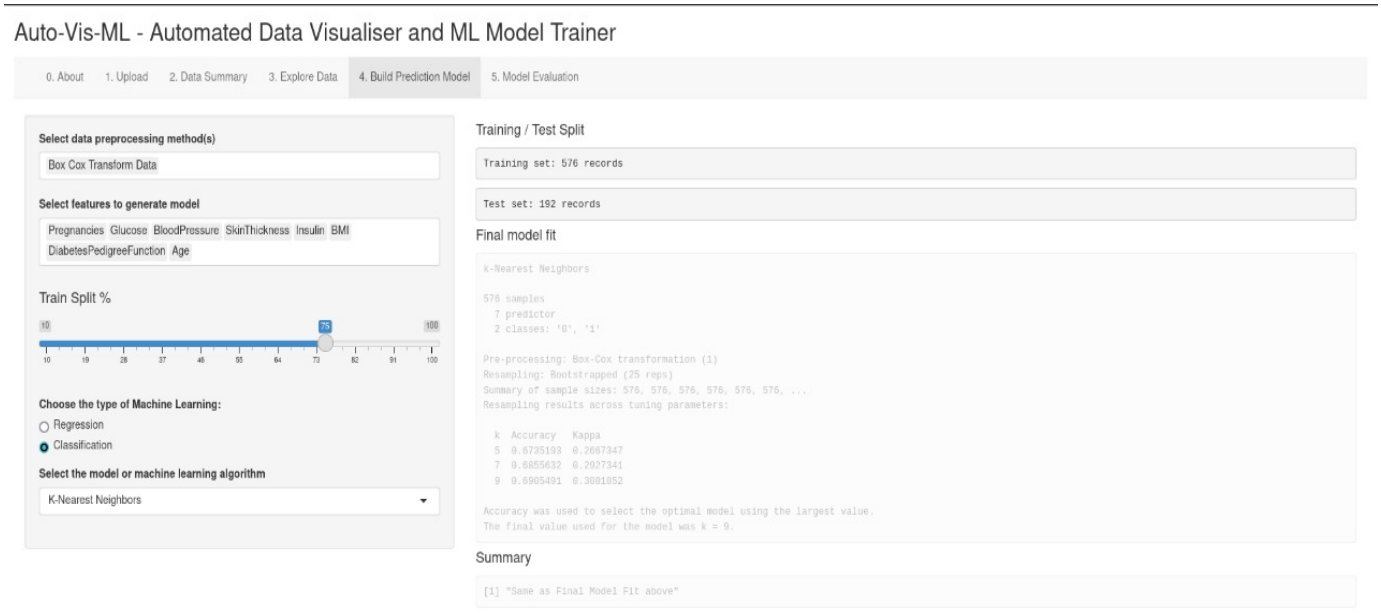
4. Model Prediction:

Under the model prediction page there are lots of functionalities that helps us to predict the model with better accuracy and it helps in choosing which algorithm is better or which parameter is the best fit for this your model. There is some common function that is used:

- **Selection of pre-processing method**: Here lots of pre-processing methods are available so we can choose the best method that is suitable for our model like: 'Box-Cox Transform Data', 'Yeo-Johnson Transform Data', 'Input missing data with k-nearest neighbors, 'Principle Component Analysis (95% variance), etc.
- **Future Selection**: Under this function, we can choose which attributes you want to take as a dependent variable and choose as per your model requirement.
- **Train-split %:** This is an amazing feature so that we can continuously choose any train split percentage and check which % is giving the best accuracy for the training model.

After choosing all the above parameters we can select any algorithm to predict our model some of the algorithms are mentioned below:
-Logistic Regression', 'K-Nearest Neighbour', 'Random Forest', 'Decision Tree', 'And Gradient Boosting.

After prediction, we can see how many training set record is generated as well as a test record.
At the end of the model evaluation, we can see all the necessary parameters as well as accuracy in the model fit block.



5. Model Evaluation

We compare multiple models against the test or validation dataset in this step. In general, we carry out these stages when evaluating a model.

- As it was done before training, transform the data. the transformer you developed for training data transformation should be used.
- Scale the validation and test dataset using the original scale. (On the test/validation dataset, transformation and scaling are performed shortly before the model is tested.)
- Using the model and test/validation data, forecast the outcomes. Utilize the chosen metrics to assess the model's performance.

Now, as we have been given machine learning algorithms like ‘Logistic Regression', 'K-Nearest Neighbour', 'Random Forest', 'Decision Tree', 'Gradient Boosting’, we will be finding all the accuracies in the form of a barplot.

Future Works

- Increasing the Number of Models
- Expanding Visualisations
- Including Model Hyperparameter Tuning
- Including Interactive Plots
- Scaling Up for bigger models
- Connection to Cloud Storage for Bigger Data

References

1. https://link.springer.com/article/10.1007/s42979-021-00592-x
2. https://shiny.rstudio.com/gallery/radiant.html
3. https://www.dominodatalab.com/data-science-dictionary/model-evaluation
4. https://cloud.google.com/automl