Breast Cancer Risk Prediction System

Project Description:

Breast cancer is one of the main causes of cancer death worldwide. Early diagnostics significantly increases the chances of correct treatment and survival, but this process is tedious and often leads to a disagreement between pathologists. Computer-aided diagnosis systems showed the potential for improving diagnostic accuracy. But early detection and prevention can significantly reduce the chances of death. It is important to detect breast cancer as early as possible.

We will be building a model in Watson Studio and deploying the model in IBM Watson Machine Learning. To interact with the model we will be using Node-Red and scoring Endpoint.

Solution Requirements:

Develop a model that is capable of detecting the Breast Cancer in early stages. The Machine learning model is trained and deployed on IBM Watson Studio and an endpoint is created. The web application is built using IBM Node-Red.

Project development steps:

Initially added data to do analysis, which is downloaded from kaggle repository, this data set consisted of six parameters namely mean_radius, mean_texture, mean_perimeter, mean_area, mean_smoothness and finally diagnosis fields. In the next step we have to build model, to do that we need to provided configuration details. In which I have tspecified column name to do prediction, classification details and optimized metric to run the experiment, then after i have run the auto ai model.

After successfully run the auto ai model, experiment successfully completed, here which show experiment summary and pipeline comparison. In the auto ai model, pipeline leader board which shows accuracy of all the algorithm, where it show star mark in the rank field which means that the particular algorithm gives high accuracy so thatwe considered it as best optimized algorithm of this project. In our experiment XGB Classifier algorithm has ranked 1, Gradient Boosting classifier has ranked 2, and subsequently remaining algorithms follows. Once identifying the algorithm save the algorithm details as model in ibm cloud. These details are helpful us to deploy the model.

After successfully saving the model it shows view in project, once we get in the view project we have to deploy it by clicking on promote to deployment space, where you have to select machine learning model and we should go to new space where we can deploy it. Here defined space information by providing project details, then selected cloud object storage and required machine learning service. After creating the deployment space it is required to creates the target space here we have to promote the deployment space, finally we reached to project model end point.

At the end point we can find url which is required in node red to create user interface.

In the next step we were deployed project using nodered which I have already created in IBM Cloud. Using cloud foundry apps opened the node red and added nodes like form, timestamp, pretoken function node, msg.payload, http request, pre prediction function node, parsing function node and prediction text node. Second http request is the end point node.

In the form node entered form elements (all the attributes of the data excluding target node i.e diagnosis) and defined variable names and its type, these details are input to the next nodes. In the pretoken node defined global variables, specified api key which is taken from manage accounts (IAM) where space is provide for validation. Here pre token node pass the input variable details to the http request node. In the second http request node added url of the which is taken from model where we are deployed.

After all the nodes are successfully configured in the node red, we deployed our project

and created user interface by using dashboard button finally we were redirected to Node red user interface.

In the user interface which is requested for the input values of mean_radius, mean_texture, mean_perimeter, mean_area, mean_smoothness. Finally when you entered the values in the text area, It shows the predicted values whether it is 0 (zero) or 1 (one) in the prediction area.