Project Report

AI algorithm System for Prediction and detection of Diabetes using PWA and Cloud

Under the guidance of:

Dr. Mohammed Abdul Raheem

Professor, ECE Dept, MJCET

Authors:

1. Sayed Abdul Zakir (1604-17-735-090)
2. Mohammad Faiz Ahmed Subhani (1604-17-735-092)
3. Shaik Ehetesham (1604-17-735-098)

B13

Muffakham Jah College of Engineering and Technology,

Banjara Hills, Hyderabad - 500034

Abstract

The most common and chronic disease in the world currently is Diabetes. Diabetes mellitus, commonly known as diabetes, is a metabolic disease that causes high blood sugar in the body. The hormone insulin moves sugar from blood to the cells to be stored or used for energy. More than 77 million people in India have diabetes and 1 in 5 of them doesn't know that they have it. So, this critical problem can be solved by the use of Powerful Artificial Intelligence Algorithms. The motive of this research is to design a model which can predict and detect diabetes in the early stage, to avoid complications like coronary heart diseases, neuropathy, nephropathy etc. Therefore, the three Machine learning algorithms namely Logistic Regression, Random Forest Classification and Lasso Regression are used to detect the Diabetes.

To maintain the record of the diabetic patients we're using Artificial Neural Networks algorithms and the patient will receive a notification whenever the diabetic levels go high. The prime objective of this project is to predict diabetes based on diagnostic measurements available in the Pima Indian Diabetes (PID) dataset, by identifying which type of algorithm model works best for this prediction. To reach out to common people and to detect diabetes in early stage, a PWA(Progressive Web App) is built using AWS Cloud, Docker and Flask microframeworks. Keywords: Diabetes, Artificial Intelligence, Machine Learning algorithms, Artificial Neural Networks, AWS cloud, Progressive Web App, Docker, Flask microframework.

TABLE OF CONTENTS

Introduction

Introduction and significance of the field:

Artificial intelligence (AI) is wide-ranging branch of computer science concerned with building smart machines capable of performing tasks that typically require human intelligence. AI is an interdisciplinary science with multiple approaches, but advancements in machine learning and deep learning are creating a paradigm shift in virtually every sector of the tech industry. "Machine learning is the science (and art) of programming computers so they can learn from data," writes Aurelian Geron. It provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

Applying machine learning and data mining methods in DM research is a key approach to utilizing large volumes of available diabetes-related data for extracting knowledge. The severe social impact of the specific disease renders DM one of the main priorities in medical science research, which inevitably generates huge amounts of data. Undoubtedly, therefore, machine learning and data mining approaches in DM are of great concern when it comes to diagnosis, management and other related clinical administration aspects. Hence, in the framework of this study, efforts were made to review the current literature on machine learning and data mining approaches in diabetes research.

Current trends:

The healthcare sector has long been an early adopter of and benefited greatly from technological advances. These days, machine learning (a subset of artificial intelligence) plays a key role in many health-related realms, including the development of new medical procedures, the handling of patient data and records and the treatment of chronic diseases. As computer scientist Sebastian Thrum told the New Yorker in a recent article titled "A.I. Versus M.D., "Just as machines made human muscles a thousand times stronger, machines will make the human brain a thousand times more powerful."

Potential for work in the area

The power and effectiveness of these approaches are derived from the ability of commensurate methods to extract patterns and create models from data. The aforementioned fact is particularly significant in the big data era, especially when the dataset can reach terabytes or petabytes of data. Consequently, the abundance of data has strengthened considerably data-oriented research in biology. In such a hybrid field, one of the most important research applications is prognosis and diagnosis related to human-threatening and/or life quality reducing diseases. One such disease is diabetes mellitus (DM).

Problem statement

Health care is always a big issue for any nation and is always challenging thing to provide. Better, the health care of a nation better is the condition of the inhabitants living there. Improvement in the health care can directly result in economic growth because a healthy person can prove to be a big asset to the nation and can conduct activities effectively in the workforce than any unhealthy person. Assisting people for prediction of Diabetes using powerful Machine Learning algorithms by Cloud and Progressive web app.

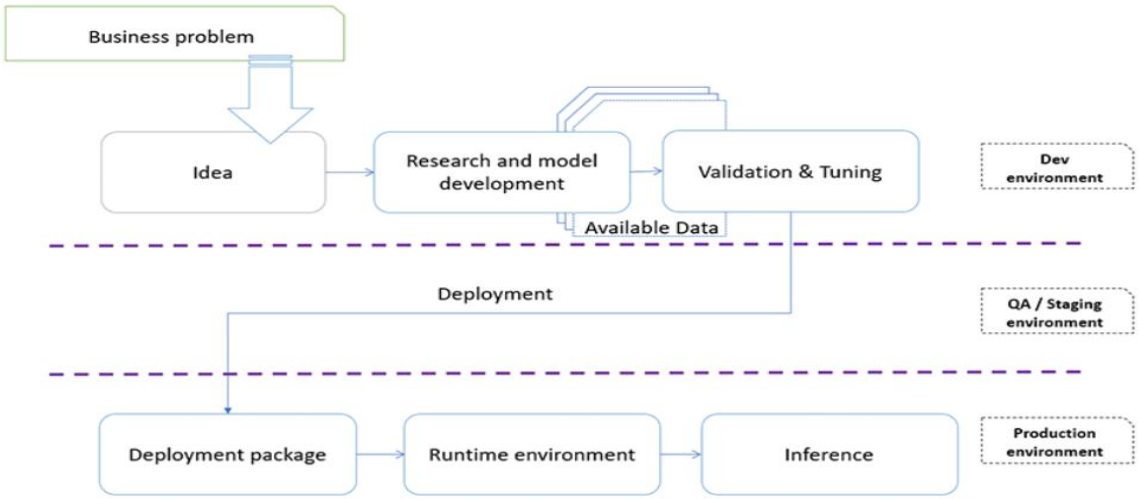
Objectives

Usage of technologies in the improvement of the health care is proved very beneficial. Machine learning can help to prevent, detect and treat various health conditions. Machine learning and data mining techniques are the best sources to improve healthcare system. Manual detection or diagnosis done by doctors is time consuming and lacks accuracy. Machines which are made to learn how to detect diseases using machine learning and data mining can better diagnose the problem and that too with high accuracy.

Machine Learning (ML) not only can prove beneficial in diagnosis and prognosis of a disease but can also prove to be helpful in personalized treatment and behavioral modification, manufacturing of drugs and discovering new patterns resulting in new medications and treatments, clinical trial research and smart electronic health records. With machine learning we can do all this and in a better way. Diabetes is typically resulting from genetics, way of life and surroundings. Eating a dangerous weight loss plan, being overweight play role in developing the diabetes. High blood sugar tiers can also result in kidney diseases, coronary heart illnesses. The excess of sugar in the blood can harm the tiny blood vessels in your frame.

Work Flow

The first step is to identify the problem statement and convert it into a business model, which can give a possible solution. The motto of our project is to Predict diabetes and provide assistance to the diabetic patients. So the next step is to do research in the field of Diabetes and build an appropriate model. After building the model, we need to integrate our ML models with the AWS cloud in the deployment/QA stage using Flask and Docker. Then the deployment package is tested in the dry run phase. Then the PWA is made as the end product .



Literature Survey

Parvin Soleymani [1] from Ryerson University, Canada, conducted an experimental study using 3 Machine Learning Classifiers namely Naïve Bayes, Logistic Regression and Decision tree to predict the likeliness of diabetes. These models were used to compare their performance in terms of accuracy, precision, recall and ROC Score. The final result of this experiment showed that the Logistic Regression classifier plays the best performance in this prediction of diabetes with a highest accuracy of 78% in comparison to the other models.

Abdullah A.Aljumah[2] and his team, as a part of the Journal of King Saud University(KSA); used data mining and machine learning tools to analyse the trend of Diabetes among various age groups. Their aim was to predict the type of diabetes and also suggest the appropriate treatment relevant to the case based more significantly on the age of the patient. They were focused on six types of treatments that were identified in the 2005 World Health Organization's NCD report of Ministry of Health, Saudi Arabia. The treatments included: Drug, Diet, Weight reduction, Smoke cessation, Exercise, Insulin.

Deepti Sisodia and Dilip Singh Sisodia [3] from NIT Raipur worked with the WEKA tool for the prediction of diabetes and their research was reviewed by the International Conference on Computational Intelligence and Data Science (ICCIDS 2018). WEKA is a software which is designed in the country New Zealand by University of Waikato, which includes a collection of various machine learning methods for data classification, clustering, regression, visualization etc. One of the biggest advantages of using WEKA is that it can be personalized according to the requirements. This proposed project was evaluated on Diabetes Dataset namely (PIDD), which is taken from UCI Repository.

Gauri D. Kalyankar[4] and her team used HADOOP along with other machine learning algorithms to predict diabetes. They implemented Hadoop MapReduce based machine learning algorithms for Pima Indian diabetes data set to find out missing values in it and to discover patterns from it. Their work suggested that implemented algorithms were able to impute missing values and to recognize patterns from the data set. Further pattern matching was employed by applying the discovered patterns on testing data set to predict diabetic prevalent and risk levels associated with it.

S.R. Surya [5] a professor from SRM University, performed a research on Predicting diabetes at an early stage. He used predictive analytic methods of Big data to predict the diabetes. Due to the unstructured nature of Big Data form health industry, it is necessary to structure and emphasis their size into nominal value with possible solution. Healthcare industry faces many challenges that make us to know the importance to develop the data analytics of the diabetes mellitus. So he implemented SVM and KNN algorithms and has obtained an accuracy of 92.34% and 86.5% respectively.

Dataset

About one in seven adults has diabetes now, according to the Centers for Disease Control and Prevention. But by 2050, that rate could skyrocket to as many as one in three. So we're using the PIMA Indian dataset.

The diabetes data set was originated from UCI Machine Learning Repository.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline


diabetes = pd.read_csv('diabetes.csv')
print(diabetes.columns)
```

The dimensions used in our dataset are:

- Pregnancies
- Glucose
- Blood Pressure
- Skin Thickness
- Insulin
- BMI
- Diabetes Pedigree Function
- Age
- Outcome

So to fetch the dataset, we need to use: Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'], dtype='object')

```
diabetes.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

The diabetes data set consists of 768 data points, with 9 features each:

dimension of diabetes data: (768, 9)

“Outcome” is the feature we are going to predict, 0 means No diabetes, 1 means diabetes. Of these 768 data points, 500 are labeled as 0 and 268 as 1:

```
print(diabetes.groupby('Outcome').size())
```

```
Outcome
0    500
1    268
dtype: int64
```

PYTHON LIBRARIES USED

NumPy

NumPy is shortened from Numerical Python, it is the most universal and versatile library both for pros and beginners. Using this tool you are up to operate with multi-dimensional arrays and matrices with ease and comfort. Such functions like linear algebra operations and numerical conversions are also available.

Pandas

Pandas is a well-known and high-performance tool for presenting data frames. Using it you can load data from almost any source, calculate various functions and create new parameters, build queries to data using aggregate functions akin to SQL. What is more, there are various matrix transformation functions, a sliding window method and other

methods for obtaining information from data. So it's totally an indispensable thing in the arsenal of a good specialist.

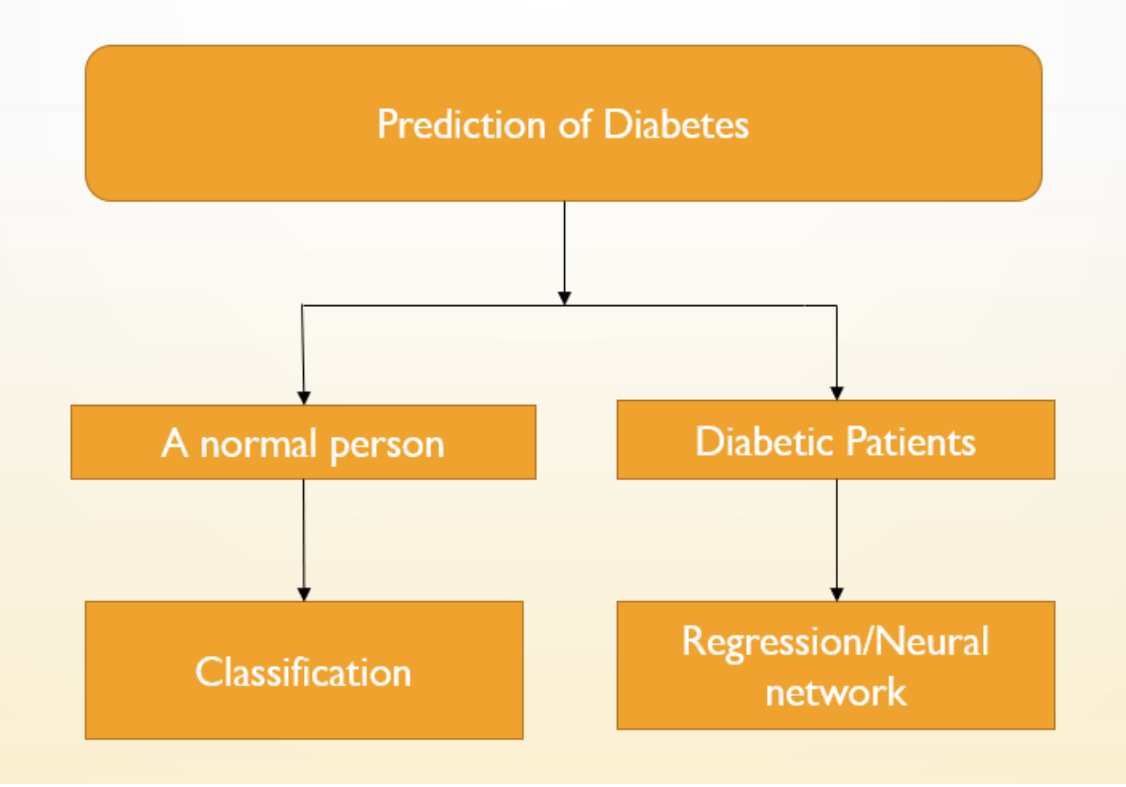
Matplotlib

Matplotlib is a flexible library for creating graphs and visualization. It is powerful but somewhat heavy-weight. At this point, you can skip Matplotlib and use Seaborn to get started (see Seaborn below).

Scikit-Learn

I can say it's the most well-designed ML package I've observed so far. It implements a wide-range of machine-learning algorithms and makes it comfortable to plug them into actual applications. You can use a whole slew of functions here like regression, clustering, model selection, preprocessing, classification and more. So, it's totally worth learning and using. The great advantage here is the high speed of work. So it's not surprising why such leading platforms like Spotify, Booking.com, J.P.Morgan are using scikit-learn.

DESIGN METHODOLOGY



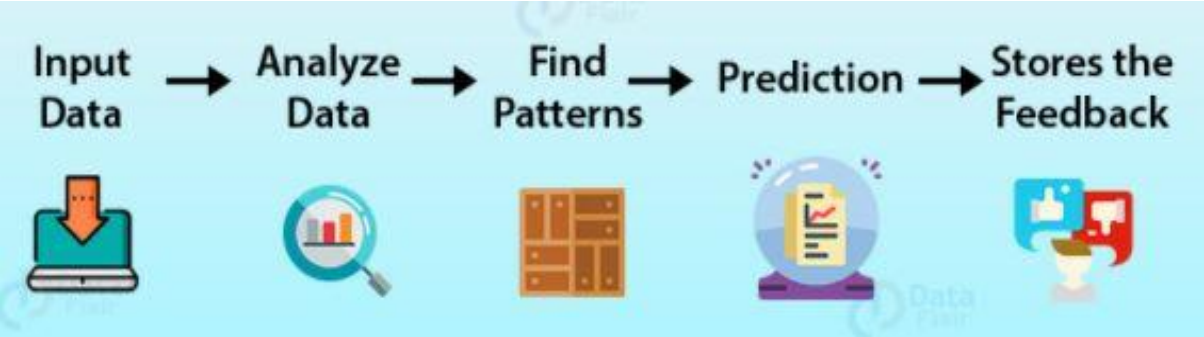
The project can be divided into two scenarios:

1) A normal person

2) Diabetic person

In scenario 1(Normal person) we aim to determine if the person is diabetic or not. So by the term 'Normal Person' it is intended to convey that the person in this scenario is unaware if he has the disease or not. Since it is a "yes/no" or "true/false" question that we are trying get an answer to, we will use Classification algorithms.

In scenario 2 (Diabetic person); the person is already aware that he diabetes, here we are aiming to identify what kind of diabetes that person is suffering from and we also intend to create a sort of a reminder system, which is efficient enough to predict any sort of adverse effect of Diabetes in advance using the historical data of the patient.



Machine learning is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence

There are various stages of machine learning:
- data collection
- data sorting
- data analysis
- algorithm development
- checking algorithm generated
- the use of an algorithm to further conclusions

To look for patterns, various algorithms are used, which are divided into two groups:
- Unsupervised learning
- Supervised learning

Algorithms Used

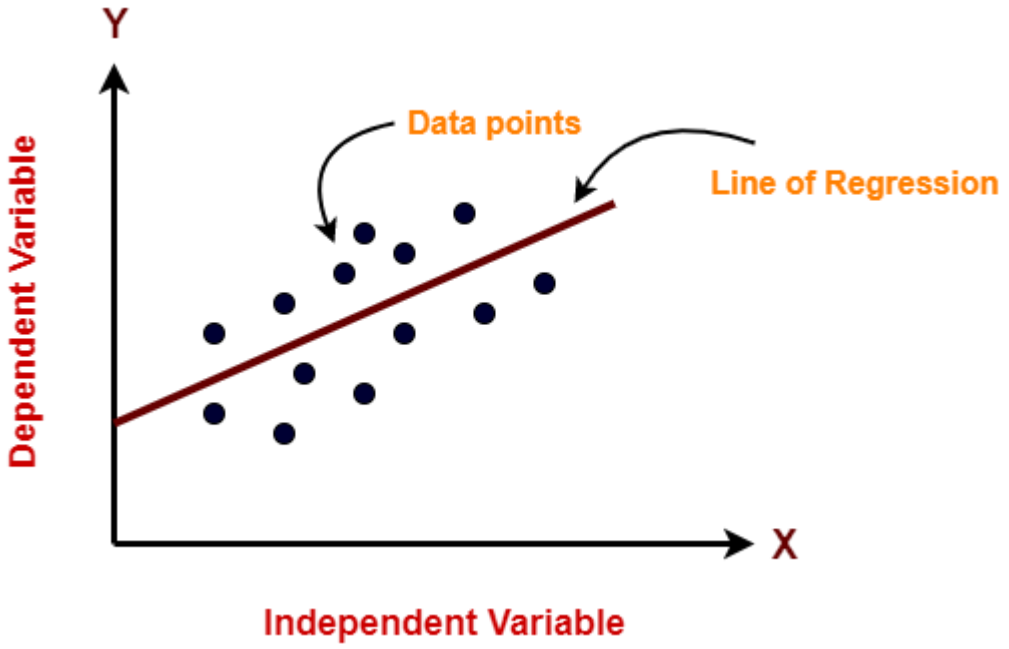
Supervised Machine Learning: The majority of practical machine learning uses supervised learning. Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output Y = f(X) . The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.

Techniques of Supervised Machine Learning algorithms include linear and logistic regression, multi-class classification, Decision Trees and support vector machines. Supervised learning requires that the data used to train the algorithm is already labeled with correct answers. For example, a classification algorithm will learn to identify animals after being trained on a dataset of images that are properly labeled with the species of the animal and some identifying characteristics.

Supervised learning problems can be further grouped into Regression and Classification problems. Both problems have as goal the construction of a succinct model that can predict the value of the dependent attribute from the attribute variables. The difference between the two tasks is the fact that the dependent attribute is numerical for regression and categorical for classification.

Regression

A regression problem is when the output variable is a real or continuous value, such as "salary" or "weight". Many different models can be used, the simplest is the linear regression. It tries to fit data with the best hyper-plane which goes through the points.



Regression models used:

- **Multi Linear Regression**

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

In essence, multiple regression is the extension of ordinary least-squares (OLS) regression because it involves more than one explanatory variable.

Formula and Calculation of Multiple Linear Regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \epsilon$$

where, for $i = n$ observations:

y_i = dependent variable

x_i = expanatory variables

β_0 = y-intercept (constant term)

β_p = slope coefficients for each explanatory variable

ϵ = the model's error term (also known as the residuals)

Simple linear regression is a function that allows an analyst or statistician to make predictions about one variable based on the information that is known about another variable. Linear regression can only be used when one has two continuous variables—an independent variable and a dependent variable. The independent variable is the parameter that is used to calculate the dependent variable or outcome. A multiple regression model extends to several explanatory variables.

The multiple regression model is based on the following assumptions:

There is a linear relationship between the dependent variables and the independent variables
The independent variables are not too highly correlated with each other
yi observations are selected independently and randomly from the population
Residuals should be normally distributed with a mean of 0 and variance σ

The coefficient of determination (R-squared) is a statistical metric that is used to measure how much of the variation in outcome can be explained by the variation in the independent variables. R2 always increases as more predictors are added to the MLR model, even though the predictors may not be related to the outcome variable.

R^2 by itself can't thus be used to identify which predictors should be included in a model and which should be excluded. R2 can only be between 0 and 1, where 0 indicates that the outcome cannot be predicted by any of the independent variables and 1 indicates that the outcome can be predicted without error from the independent variables.

When interpreting the results of multiple regression, beta coefficients are valid while holding all other variables constant ("all else equal"). The output from a multiple regression can be displayed horizontally as an equation, or vertically in table form.

- **Lasso Regression**

Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point, like the mean. The lasso procedure encourages simple, sparse models (i.e. models with fewer parameters). This particular type of regression is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination.
The acronym "LASSO" stands for Least Absolute Shrinkage and Selection Operator.
L1 Regularization
Lasso regression performs L1 regularization, which adds a penalty equal to the absolute value of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients; Some coefficients can become zero and eliminated from the model. Larger penalties result in coefficient values closer to zero, which is the ideal for producing simpler models. On the other hand, L2 regularization (e.g. Ridge regression) doesn't result in elimination of coefficients or sparse models. This makes the Lasso far easier to interpret than the Ridge.
Performing the Regression
Lasso solutions are quadratic programming problems, which are best solved with software (like Matlab). The goal of the algorithm is to minimize:

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Which is the same as minimizing the sum of squares with constraint Σ |Bj≤ s. Some of the βs are shrunk to exactly zero, resulting in a regression model that's easier to interpret. A tuning parameter, λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage:
When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.
As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, all coefficients are eliminated).
As λ increases, bias increases.
As λ decreases, variance increases.

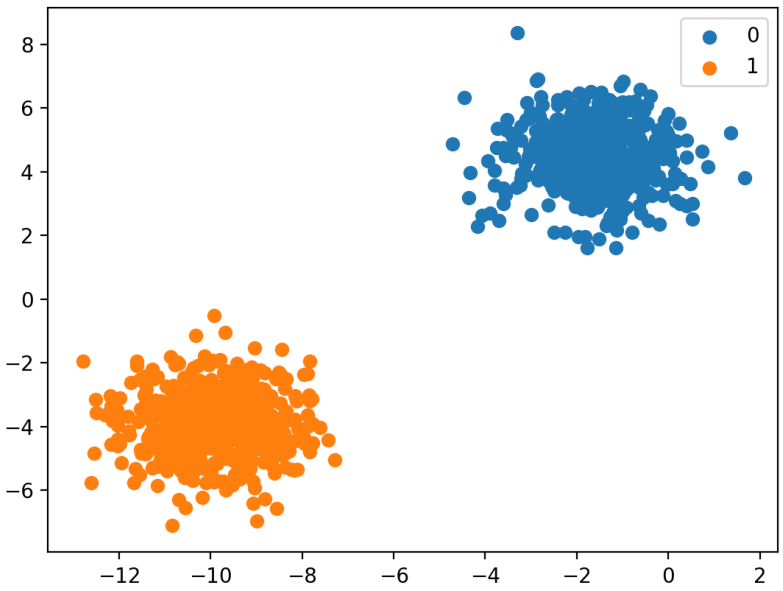
Classification

A classification problem is when the output variable is a category, such as “red” or “blue” or “disease” and “no disease”. 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.
For example, when filtering emails “spam” or “not spam”, when looking at transaction data, “fraudulent”, or “authorized”. In short Classification either predicts categorical class labels or classifies data (construct a model) based on the training set and the values (class labels) in classifying attributes and uses it in classifying new data. There are a number of classification models. Classification models include logistic regression, decision tree, random forest, gradient-boosted tree, multilayer perceptron, one-vs-rest, and Naive Bayes.

For example:
Which of the following is/are classification problem(s)?

- Predicting the gender of a person by his/her handwriting style
- Predicting house price based on area
- Predicting whether monsoon will be normal next year
- Predict the number of copies a music album will be sold next month

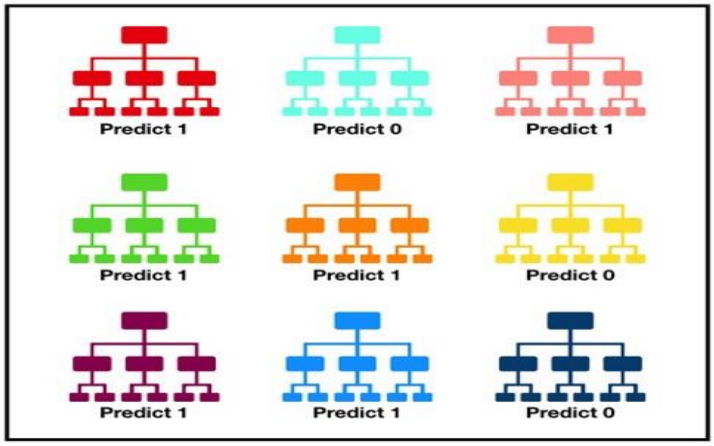
Solution: Predicting the gender of a person Predicting whether monsoon will be normal next year. The other two are regression.



Classification algorithms used:

- **Random Forest Classifier**

  A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.



Features and Advantages of Random Forest:

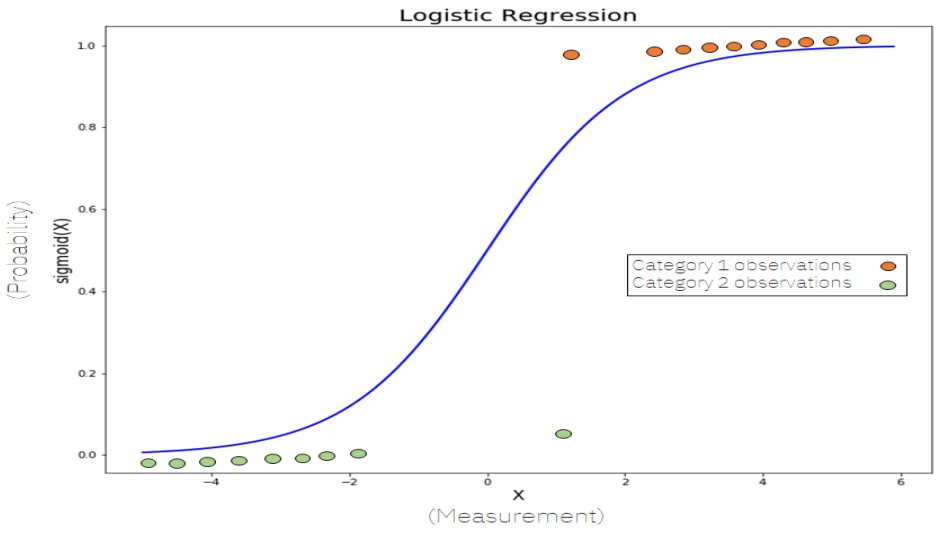
- It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.
- It runs efficiently on large databases.
- It can handle thousands of input variables without variable deletion. It gives estimates of what variables that are important in the classification.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing.

- **Logistic Regression**

Logistic Regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous

Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more independent variables.

$$\Phi(z) = \frac{1}{1+\exp(-z)}$$



If 'Z' goes to infinity, Y(predicted) will become 1 and if 'Z' goes to negative infinity, Y(predicted) will become 0.

Mathematically this can be written as,

$h_\Theta(x) = P(Y=1 \mid X; \text{theta})$

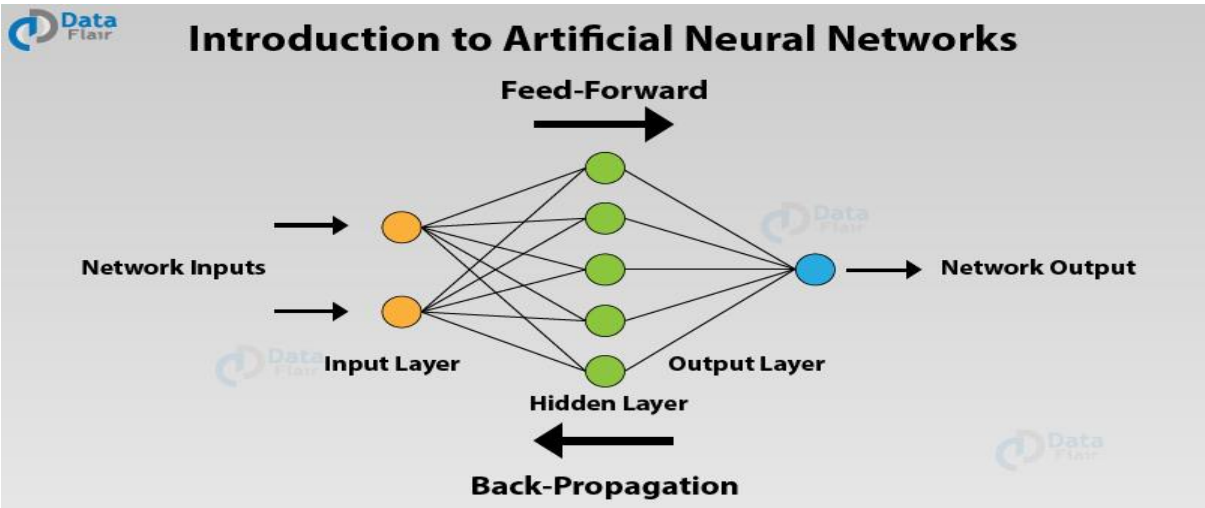
Probability that Y=1 given X which is parameterized by 'theta'.

$P(Y=1 \mid X; \text{theta}) + P(Y=0 \mid X; \text{theta}) = 1$

$P(Y=0 \mid X; \text{theta}) = 1 - P(Y=1 \mid X; \text{theta})$

Artificial Neural Networks (ANN)

- A neural network is a machine learning algorithm based on the model of a human neuron. The human brain consists of millions of neurons. It sends and process signals in the form of electrical and chemical signals.
- In this project, we're using LSTM Neural Network



- The structure of a neural network also referred to as its 'architecture' or 'topology'. It consists of the number of layers, Elementary units. It also consists of Interchanged Weight adjustment mechanism. The choice of the structure determines the results which are going to obtain. It is the most critical part of the implementation of a neural network.
- The simplest structure is the one in which units distributes in two layers: An input layer and an output layer. Each unit in the input layer has a single input and a single output which is equal to the input. The output unit has all the units of the input layer connected to its input, with a combination function and a transfer function. There may be more than 1 output unit. In this case, resulting model is a linear or logistic regression. This is depending on whether transfer function is linear or logistic. The weights of the network are regression coefficients.
- By adding 1 or more hidden layers between the input and output layers and units in this layer the predictive power of neural network increases. But a number of hidden layers should be as small as possible. This ensures that neural network does not store all information from learning set but can generalize it to avoid over fitting.
- Overfitting can occur. It occurs when weights make the system learn details of learning set instead of discovering structures. This happens when size of learning set is too small in relation to the complexity of the model.
- A hidden layer is present or not, the output layer of the network can sometimes have many units, when there are many classes to predict.

Integration of ML model on AWS cloud

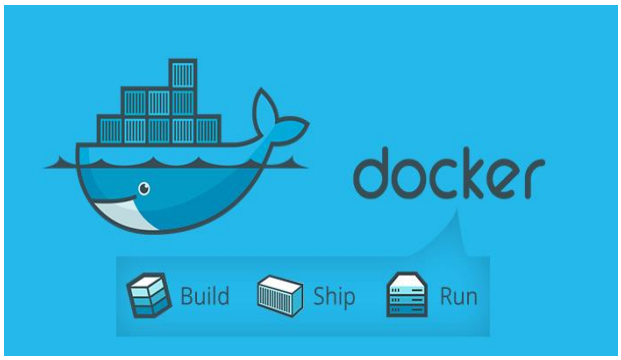
Where the DS creates a usable piece of software for the stakeholders to consume the machine learning models.

Tech-stack: **Python, Flask, Docker, AWS EC2**

The workflow can be broken down into following basic steps:

- Training a machine learning model on a local system.
- Wrapping the inference logic into a flask application.
- Using docker to containerize the flask application.
- Hosting the docker container on an AWS ec2 instance and consuming the web-service.

We need some machine learning model that we can wrap in a web-service. For demo purpose, I chose a logistic regression model to do multiclass classification on iris dataset (Yep, super easy! #LazinessFTW). The model was trained on a local system using python 3.6.



Docker is an open platform for developing, shipping, and running applications. Docker enables us to separate the applications from the infrastructure so that we can deliver software quickly. With Docker, we can manage our infrastructure in the same ways we manage your applications.

Using the familiar scikit-learn, the above-mentioned model can be trained quickly. For model development, refer the notebook 'Model_training.ipynb' in the github repo for this blog. There are only 2 important aspects of model development that I would like to highlight:

The model file generated after training is stored as a pickle file which is a serialized format for storing objects. (In the repo, the file is named 'iris_trained_model.pkl')

The inference call (.predict()) call requires 4 features per test sample in the form of a numpy array.

Wrapping the inference logic into a flask web service

Now that we have the trained model file, we are ready to query the model to get a class label for a test sample. The inference is as simple as calling a predict() function on the trained model with the test data. However, we would like to build the inference as a web-service. For this purpose, we would use Flask.

Flask is a powerful python microwebserver framework that allows us to build REST API based web-services quickly with minimum configuration hassle.

Let's dive into the code:

a. First, let's define a simple function to load the trained model file.

```
model = None

def load_model():
    global model
    # model variable refers to the global variable
    with open('iris_trained_model.pkl', 'rb') as f:
        model = pickle.load(f)
```

b. Next, we instantiate a Flask object called ‘app’:

```
app = Flask(__name__)
```

c. Now, we define a home endpoint, which when hit, returns a ‘Hello World!’ message.

```
@app.route('/')
def home_endpoint():
    return 'Hello World!'
```

d. Now, we define a ‘predict’ endpoint. The endpoint accepts a ‘POST’ request wherein the test data on which we wish to get a prediction is received by the endpoint. Keeping things simple, the function works only when a single test sample needs to be predicted (won’t work if multiple samples need to be predicted in a single call to the endpoint).

```
@app.route('/predict', methods=['POST'])
def get_prediction():
    # Works only for a single sample
    if request.method == 'POST':
        data = request.get_json()  # Get data posted as a json
        data = np.array(data)[np.newaxis, :]  # converts shape from (4,) to (1, 4)
        prediction = model.predict(data)  # runs globally loaded model on the data
    return str(prediction[0])
```

e. Finally, declare the main function:

```
if __name__ == '__main__':
    load_model()  # load model at the beginning once only
    app.run(host='0.0.0.0', port=80)
```

Design Objectives

- To build a machine learning model that is capable of automatic prediction and detection of Diabetes.
- Choosing the optimum model which gives the highest accuracy.
- Building a reminder mechanism using an app to alert the patient of the probable spike in blood sugar levels, to prevent further health risks.

Application areas

- This project can be used in
- Medical Applications
    - Self assessment of Diabetes
        - Prediction of Type-1 and Type-2 diabetes
    - Maintaining record of diabetic patients
        - Alerting the diabetic patients
        - Risk bias and quality assessment

REFERENCES

[1] Beyer, W. H. CRC Standard Mathematical Tables, 31st ed. Boca Raton, FL: CRC Press, pp. 536 and 571, 2002.

[2] Agresti A. (1990) Categorical Data Analysis. John Wiley and Sons, New York.

[3] Kotz, S.; et al., eds. (2006), Encyclopedia of Statistical Sciences, Wiley.

[4] Wheelan, C. (2014). Naked Statistics. W. W. Norton & Company.

[5] Sheetal Sharma(2017) Artificial Neural Network (ANN) in Machine Learning, Data Science Central.

[6]https://www.sciencedirect.com/topics/earth-and-planetary-sciences/artificial-neural-network

[7] https://ieeexplore.ieee.org/document/8058253?denied

[8]https://www.researchgate.net/publication/331672855_Prediction_of_Diabetes_Disease_Using_Classification_Data_Mining_Techniques

[9]https://www.sciencedirect.com/science/article/pii/S0933365702000519?via%3Dihub

[10]https://www.researchgate.net/publication/339543101_Diabetes_Prediction_using_Machine_Learning_Algorithms

[11] https://sci-hub.se/https://ieeexplore.ieee.org/document/8058253?denied=

[12]https://www.sciencedirect.com/science/article/pii/S0933365703000125

[13] https://acadpubl.eu/jsi/2018-118-7-9/articles/9/87.pdf

[15] https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0175-6

[16]https://www.datasciencecentral.com/profiles/blogs/artificial-neural-network-ann-in-machine-learning.

[17] Al Juma, A.L., Ahmad, M.G., Siddiqui, M.K., 2013. Application of data mining: Diabetes health care in young and old patients. Journal of King Saud University - Computer and Information Sciences 25, 127–136. doi:10.1016/j.jksuci.2012.10.003.

[18]Yu,W.,Liu,T.,Valdez,R.,Gwinn,M.,Khoury,M.J.,2010.Applicationofsupportvectormachinemodelingforpredictionofcom-mondiseases:Thecaseofdiabetesandpre-diabetes.BMCMedicalInformaticsandDecisionMaking10.doi:10.1186/1472-6947-10-16,arXiv:arXiv:1011.1669v3.