IBM MACHINE LEARNING

**Fatima, Sayeda**

**1/13/2022**

­­­­­­

DEEP AND REINFORCEMENT LEARNING MODELS FOR DIABETES PREDICTION

**Table of Contents**

[**1) Project Overview and Data Description** 2](#_Toc124443063)

[**1a) Problem Overview** 2](#_Toc124443064)

[**1b) About the Dataset** 2](#_Toc124443065)

[**1b-i) Brief Description of Chosen Data Set** 2](#_Toc124443066)

[**1b-ii) Summary of Data Attributes** 2](#_Toc124443067)

[**1c) Data Exploration, Data Cleansing and Features Engineering** 3](#_Toc124443068)

[**1c-i) Data Exploration** 3](#_Toc124443069)

[**1c-ii) Data Cleansing Actions & Features Engineering** 6](#_Toc124443070)

[**2) Main Objectives of the Analysis** 11](#_Toc124443071)

[**2a) Primary Objective** 11](#_Toc124443072)

[**2a) Secondary Objectives** 11](#_Toc124443073)

[**3) Summary of Training Different Deep Learning Models** 12](#_Toc124443074)

[**3a) Keras Hyperparameter Tuning using Grid Search** 12](#_Toc124443075)

[**3b) Automated Model Building** 13](#_Toc124443076)

[**3c) Summarizing Employed Models** 14](#_Toc124443077)

[**1)** **Keras Algorithm with Optimizer RMSprop:** 14](#_Toc124443078)

[**2)** **Keras Algorithm with Optimizer SGD:** 16](#_Toc124443079)

[**3)** **Keras Algorithm with Optimizer Nadam:** 18](#_Toc124443080)

[**4) Keras Algorithm with Optimizer Adamax:** 20](#_Toc124443081)

[**5) Keras Algorithm with Optimizer Adadelta:** 22](#_Toc124443082)

[**6) Keras Algorithm with Optimizer Adam:** 24](#_Toc124443083)

[**7) Keras Algorithm with Optimizer Adagrad:** 26](#_Toc124443084)

[**8) Base Model: Random Forest:** 28](#_Toc124443085)

[**3d) Model Choice** 29](#_Toc124443086)

[**3d-i) Result Summary** 29](#_Toc124443087)

[**3d-ii) Model Ranking, Choice and Justification** 29](#_Toc124443088)

[**4) Summary Key Findings and Insights** 30](#_Toc124443089)

[**4a) Feature Weights within Models** 30](#_Toc124443090)

[**4b Overall Feature Prominence** 31](#_Toc124443091)

[**5) Model Shortcomings and Future Directions** 32](#_Toc124443092)

[**5a) Model Shortcomings** 32](#_Toc124443093)

[**5b) Plan of Action to Revisit Analysis** 32](#_Toc124443094)

[**6 Useful Links** 33](#_Toc124443095)

[**6a) Guide to Keras** 33](#_Toc124443096)

[**6b) Keras Hyperparameter Tuning Approaches** 33](#_Toc124443097)

[**6c) Model Checkpoints** 33](#_Toc124443098)

[**6d) Get Feature Importance in Keras** 33](#_Toc124443099)

[**6e) Shap Plots: Some Good Examples** 33](#_Toc124443100)

[**6f) Matplot Lib** 33](#_Toc124443101)

[**6f) Other Useful Models** 33](#_Toc124443102)

[**7) Github Links** 34](#_Toc124443103)

[**7a) Link to Main Folder** 34](#_Toc124443104)

[**7b) Link to Assignment Notebook** 34](#_Toc124443105)

[**8) References** 35](#_Toc124443106)

**1) Project Overview and Data Description**

## **1a) Problem Overview**

The drastic and often fatal impacts of diabetes not only are not only worrying but its steady incline expected to reach at 629 million by 2045 has also turned it into global threat (Naz and Ahuja, 2020). Despite its alarming increase, diabetes is principally a preventable disease which can be prevented by adopting healthier lifestyle changes which may also decrease probability of developing other diseases like cancer or heart problems. Hence, early detection of diabetes through a reliable prognosis tool is crucial to either prevent disease onset or stop its further progression.

## **1b) About the Dataset**

### **1b-i) Brief Description of Chosen Data Set**

This project uses a hypothetical dataset ‘UCI Pima Diabetes Dataset’ which has been acquired for identifying risk of diabetes and was downloaded from the following link:

<https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database>

### **1b-ii) Summary of Data Attributes**

The PIMA dataset exhibits 768 data points (rows) and 9 features (columns) reflecting on patients’ characteristics where, based on various factors, each patient has been assigned a Diabetes Score.

Of these, the main features are:

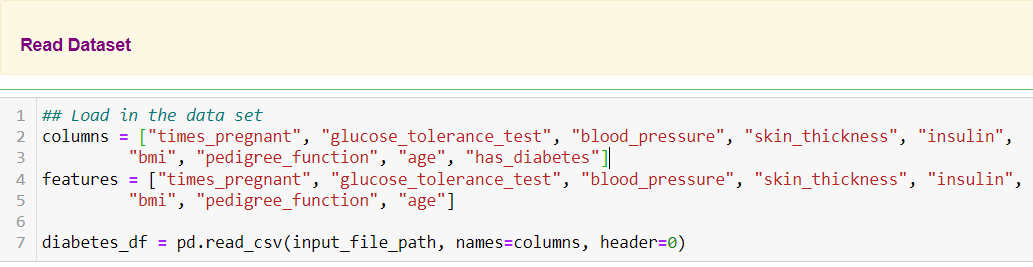
1. times\_pregnant,
2. glucose\_tolerance\_test,
3. blood\_pressure,
4. skin\_thickness,
5. insulin,
6. bmi,
7. pedigree\_function
8. age

## **1c) Data Exploration, Data Cleansing and Features Engineering**

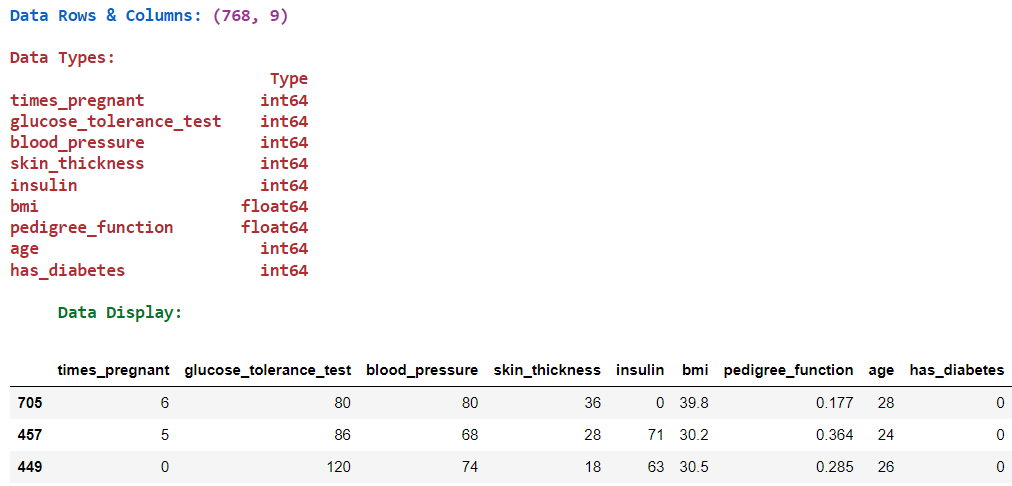
Since the quality of any machine learning model highly depends on quality of data, hence, this stage is not only most important but is also time consuming. Hence, it was conducted in a step-by-step process.

### **1c-i) Data Exploration**

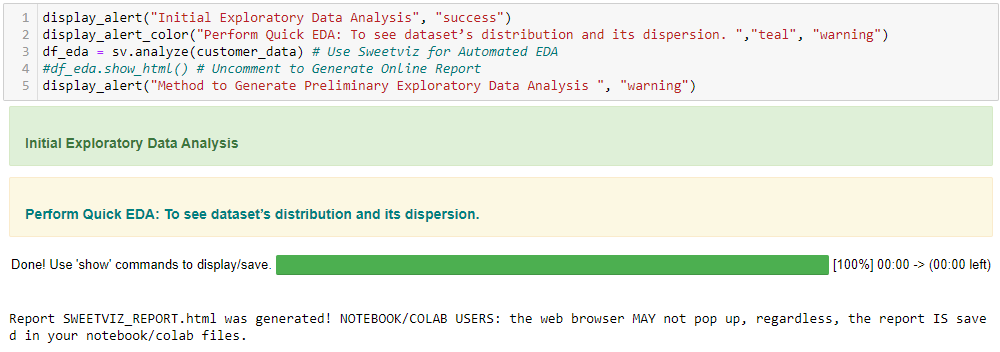
* Data was first loaded into pandas dataframe



* Column types were then explored

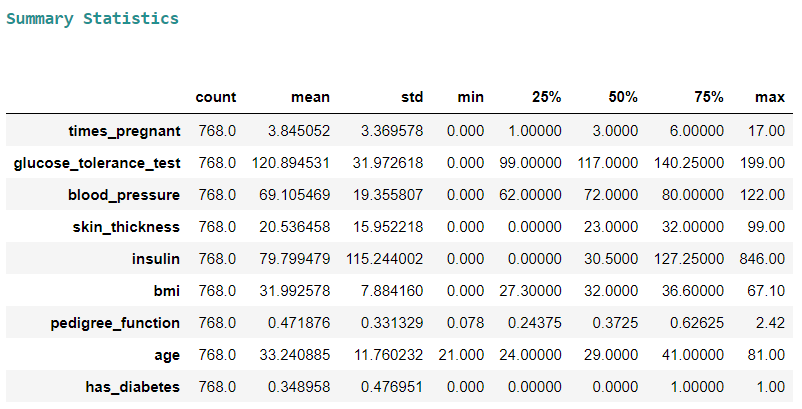


* Automated Exploratory Data Analysis was performed using Sweetviz





* Additional descriptive statistics were computed to summarize shape of a dataset’s distribution, its dispersion and central tendency



### **1c-ii) Data Cleansing Actions & Features Engineering**

In machine learning, feature selection is the method to reduce the number of input variables during developing predictive modelling. This reduction in input variables is necessary not only to minimize computational cost of modeling but also to achieve performance improvement of the model.

Among widely practices feature selection approaches include statistical-based feature selection methods which use statistical measures to evaluate relationship between each input variable and the target variable and then select those exhibiting strongest relationship with the latter. While these methods can be both speedy and effective, however, the ultimate choice of statistical measure is largely dependent on data types of both of these variables.

Irrespective of the statistical measure being employed, two dominant feature selection techniques, that is supervised and unsupervised, exist where the former can be further categorized into wrapper, filter and intrinsic techniques. Filter-based feature selection methods employs statistical measures to evaluate correlation between input and output variables so that those exhibiting highest correlations are selected. Statistical measures employed in filter-based feature selection are normally univariate in nature since they evaluate relationship of single input variables one by one with target variable, disregarding their interaction with each other.

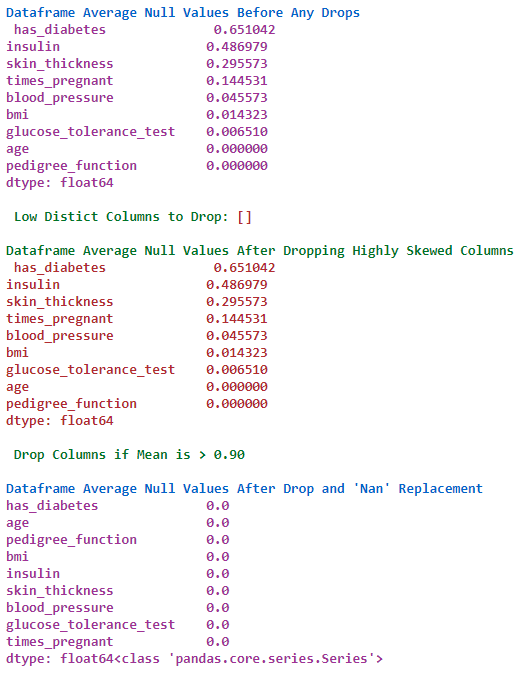
Consequently, adopting filter-based feature selection methods, the project approached filter engineering in the following steps.

An automated data cleansing method was created to do the following:

* Drop Columns with Unique Values Less than threshold of 2
* Drop Highly Skewed & Low Correlation Columns with target
* Drop Columns with High Nan Values

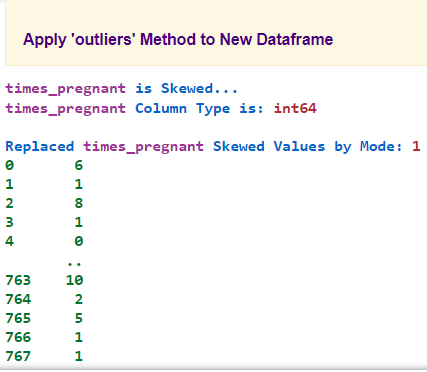


Null values were summed and were automatically managed by the above function.



**Outlier Treatment:** Like Supervised learning, deep learning models are also sensitive to outliers. Hence, an automated method was created to replace outliers with “Mode”, that is the most common value.

­­­­



**Data Splitting & Normalization:** To speed up algorithms’ learning, data normalization was carried out.



# **2) Main Objectives of the Analysis**

Even though healthcare organizations normally collect big data including electronic health records and images, nevertheless, its robust analysis to acquire meaningful and reliable insights remains a key challenge. Given promising results evidenced by deep and other machine learning methodologies, such medical data can be automatically analysed to discover hidden patterns and factors which may aid in diabetes diagnosis at an early stage.

## **2a) Primary Objective**

Hence, the main objective of this project is to present an automated methodology for employing different variations of Keras deep learning model for diabetes prediction using the PIMA dataset.

## **2a) Secondary Objectives**

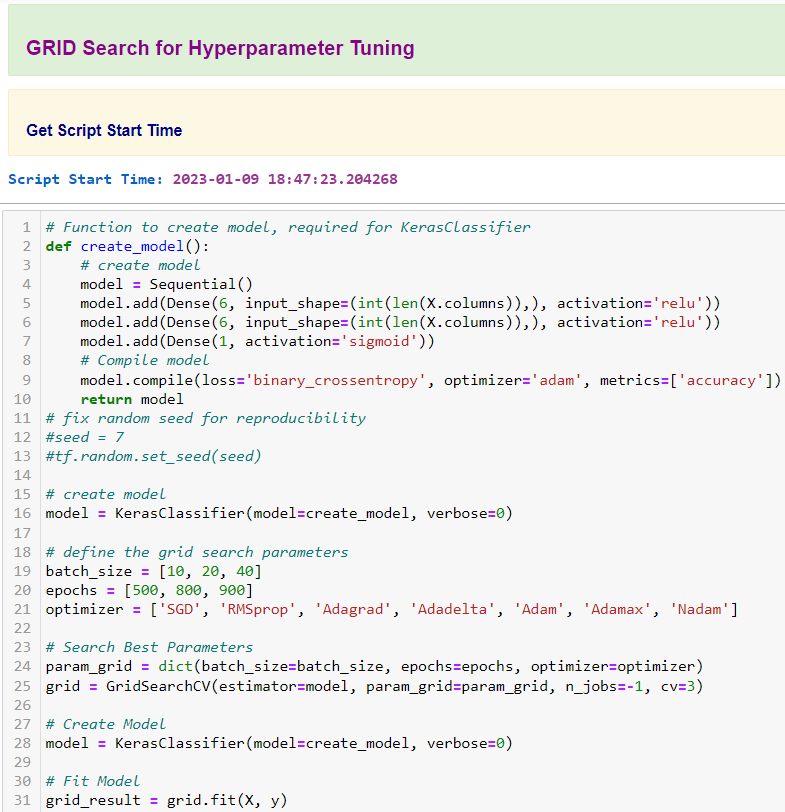
Secondary objective are as follows:

1. Build a baseline performance using Random Forest Model (RFM) for comparison
2. Develop automated grid search method for selecting best hyperparameters for Keras model development
3. Evaluate Keras results and compare with those of RFM to validate model robustness

# **3) Summary of Training Different Deep Learning Models**

## **3a) Keras Hyperparameter Tuning using Grid Search**

An automated method was created to find optimal parameters for various Keras Optimizers:



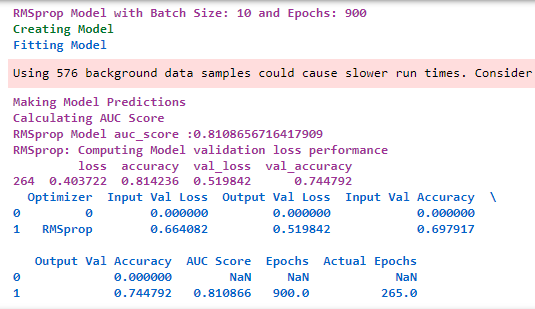
## **3b) Automated Model Building**

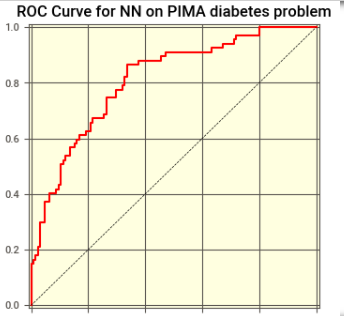


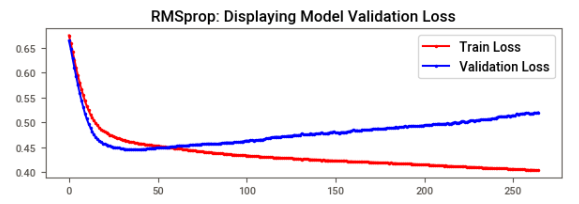
## **3c) Summarizing Employed Models**

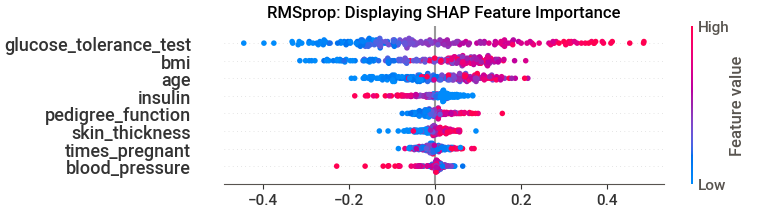
Following is a summary of all Keras Model Variations which have been used to predict onset of diabetes.

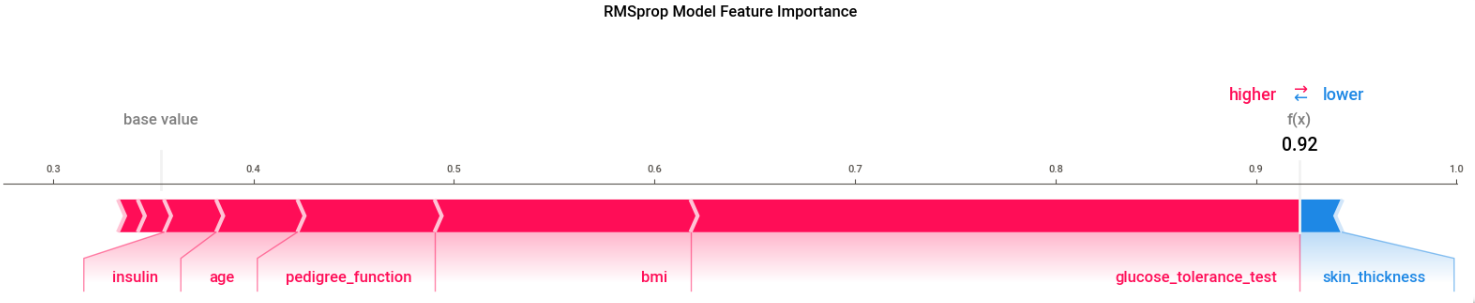
### **Keras Algorithm with Optimizer RMSprop:**

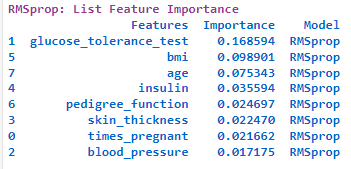




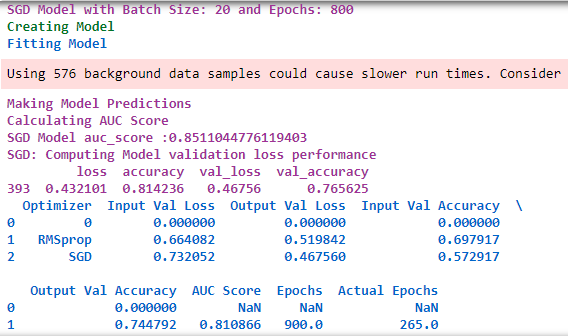


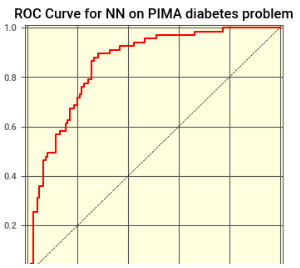


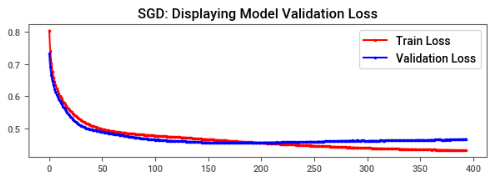


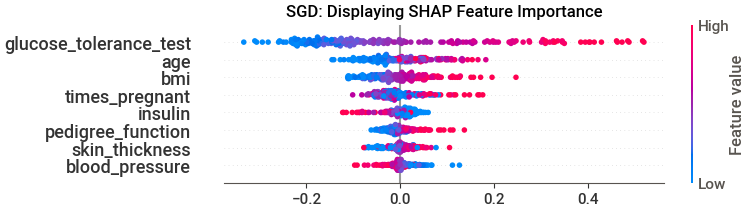


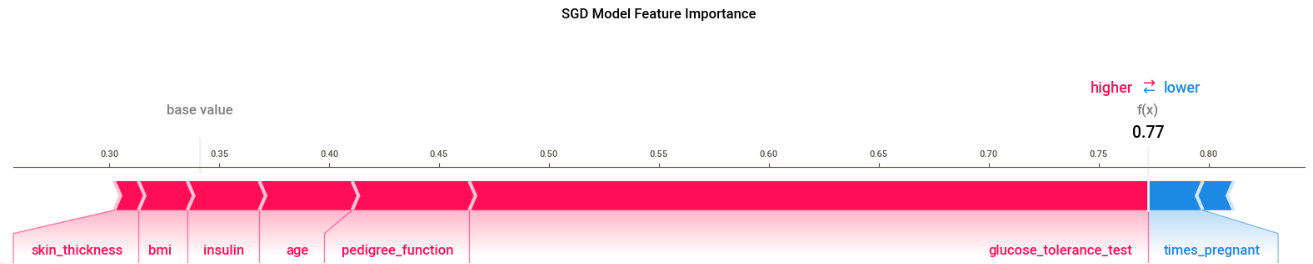
### **Keras Algorithm with Optimizer SGD:**

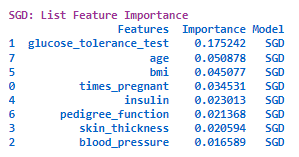




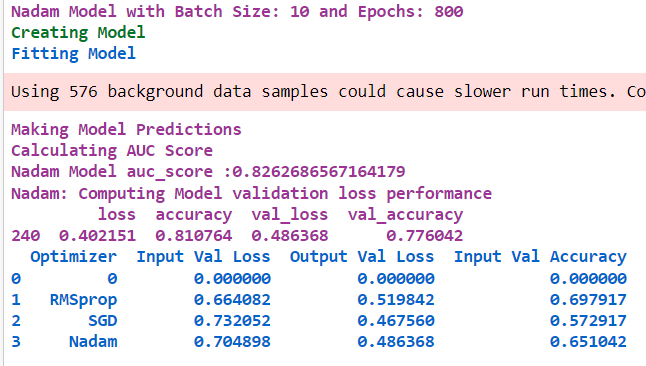


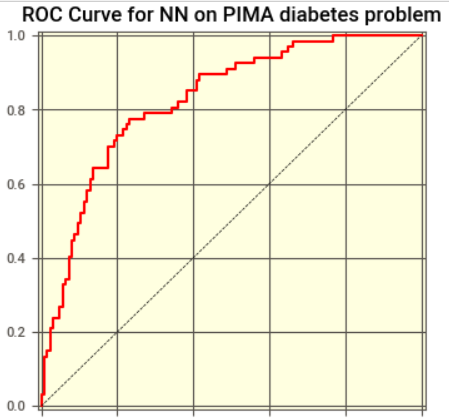


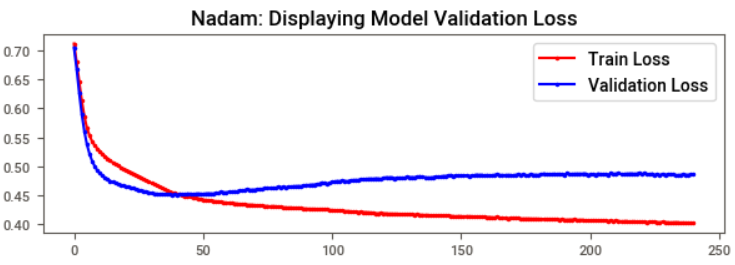


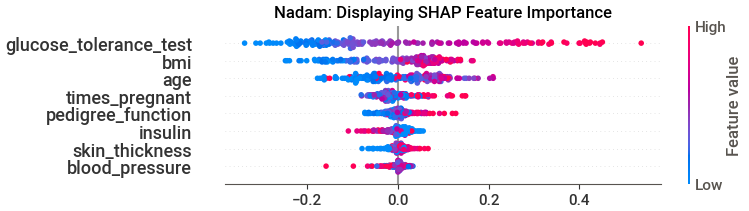


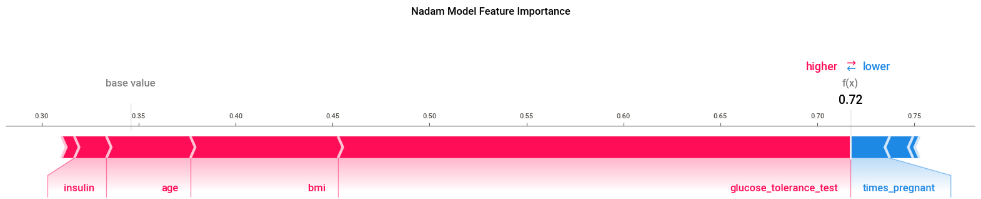
### **Keras Algorithm with Optimizer Nadam:**

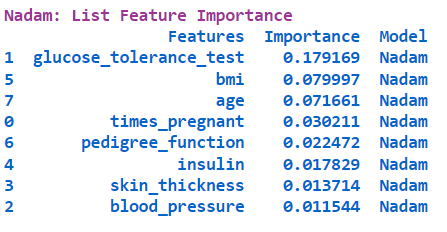




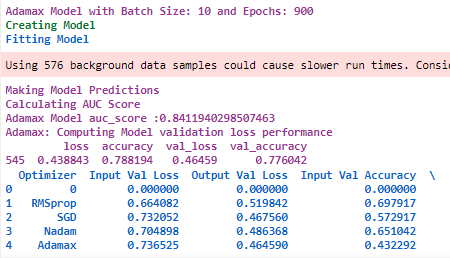


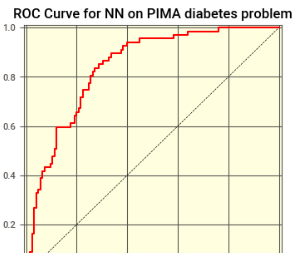


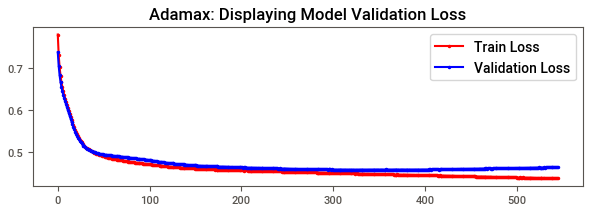


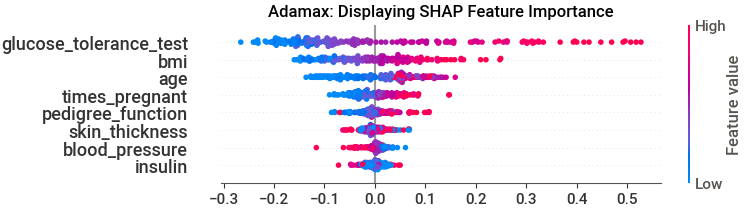


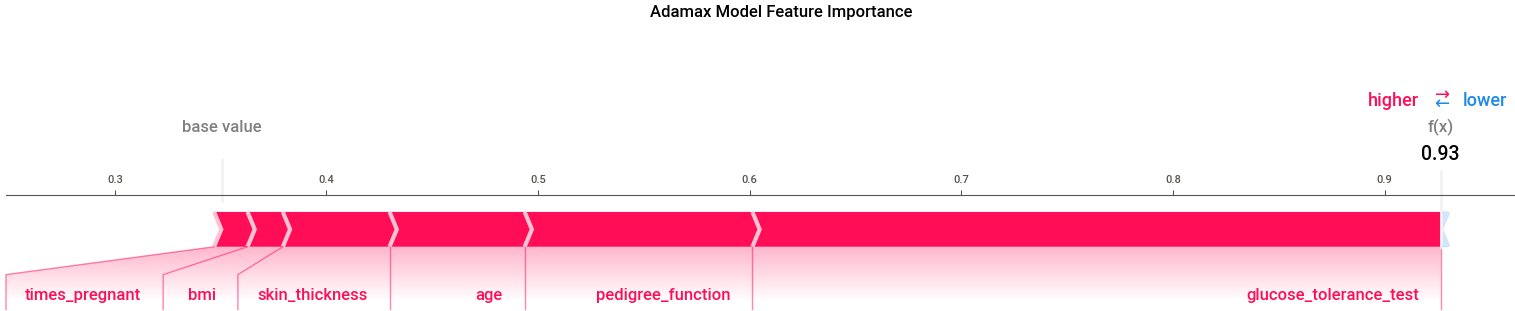
### **4) Keras Algorithm with Optimizer Adamax:**

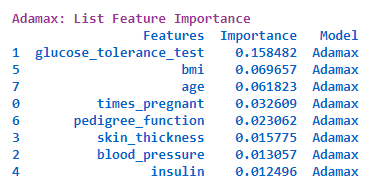




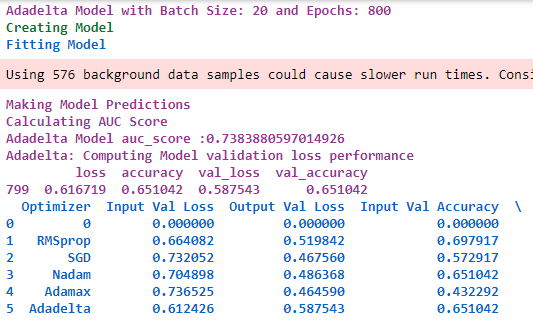


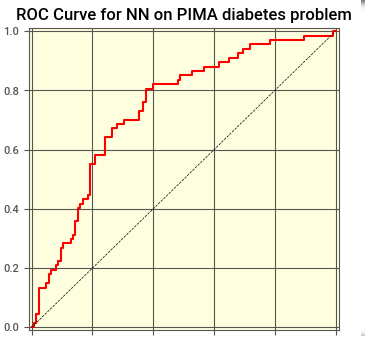


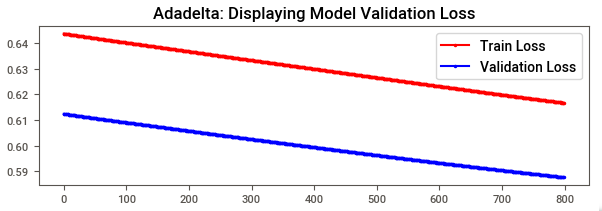


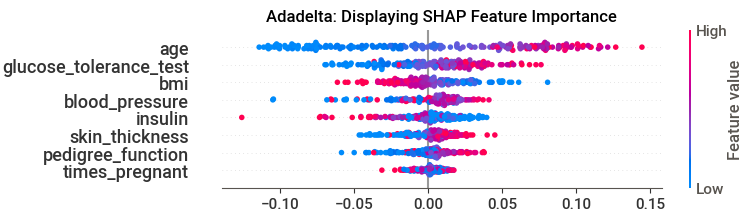


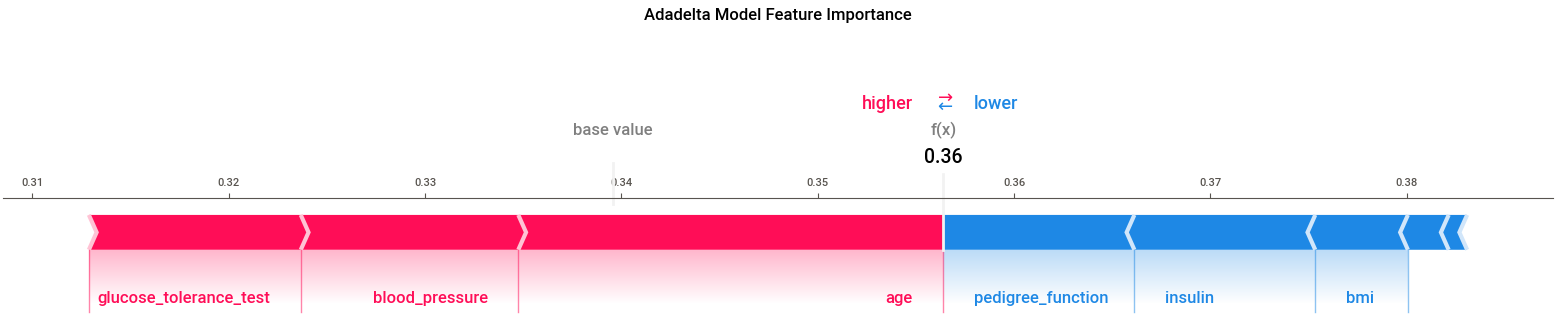
### **5) Keras Algorithm with Optimizer Adadelta:**

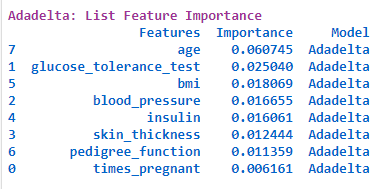




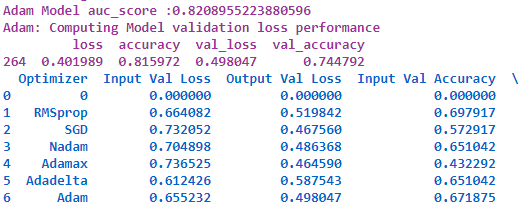


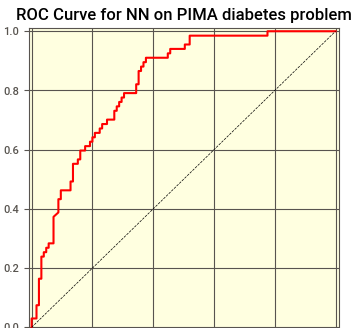


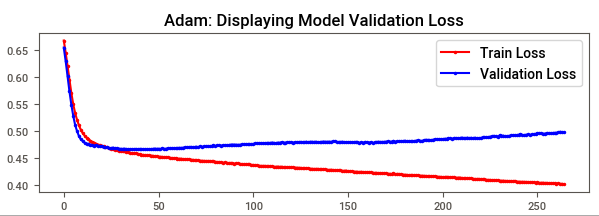


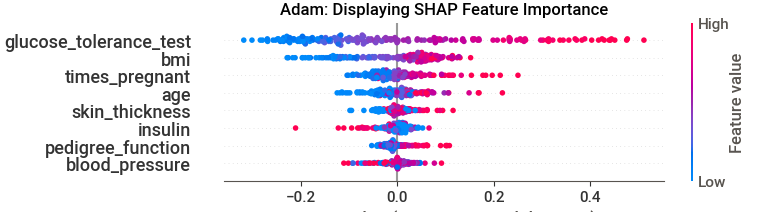


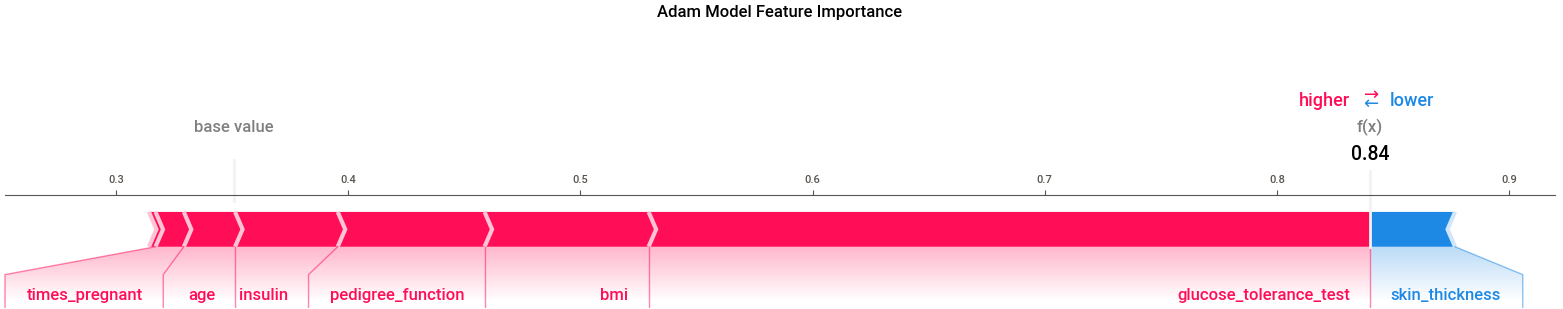
### **6) Keras Algorithm with Optimizer Adam:**

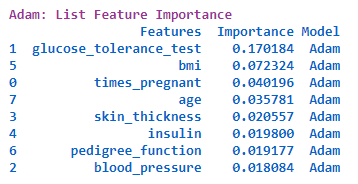




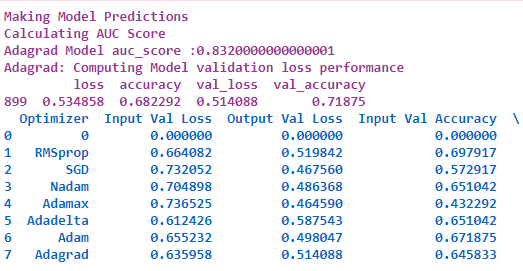


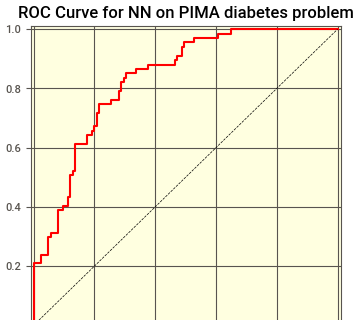


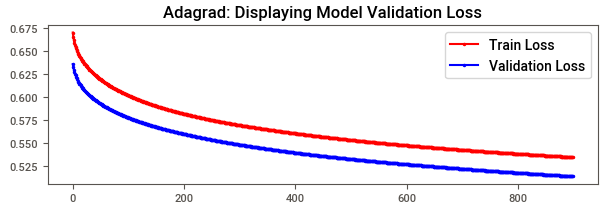


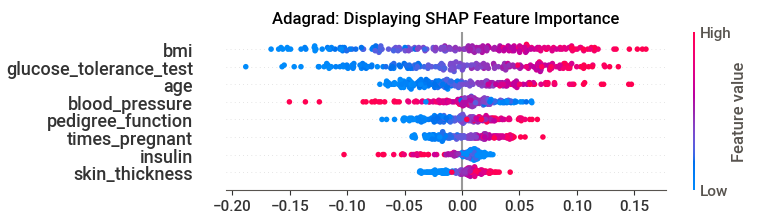


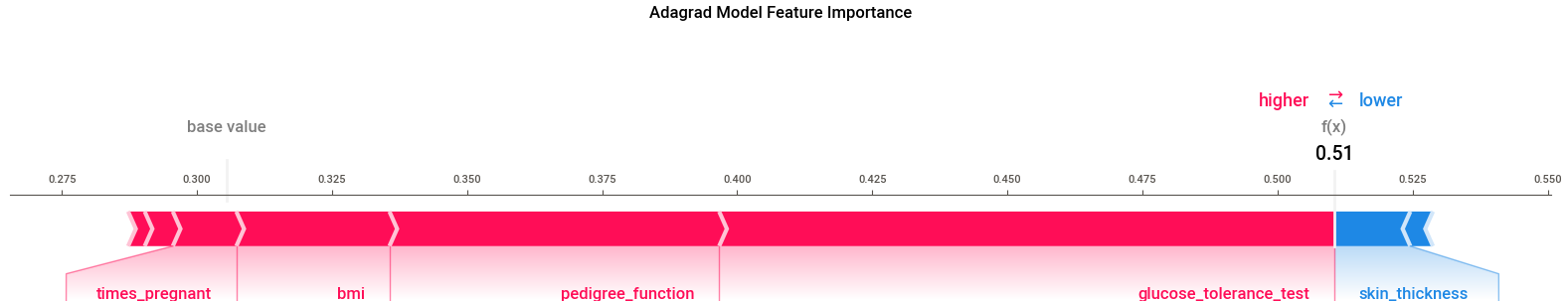
### **7) Keras Algorithm with Optimizer Adagrad:**

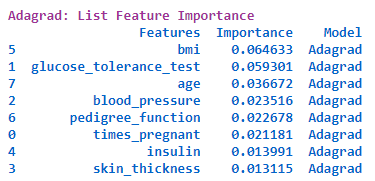




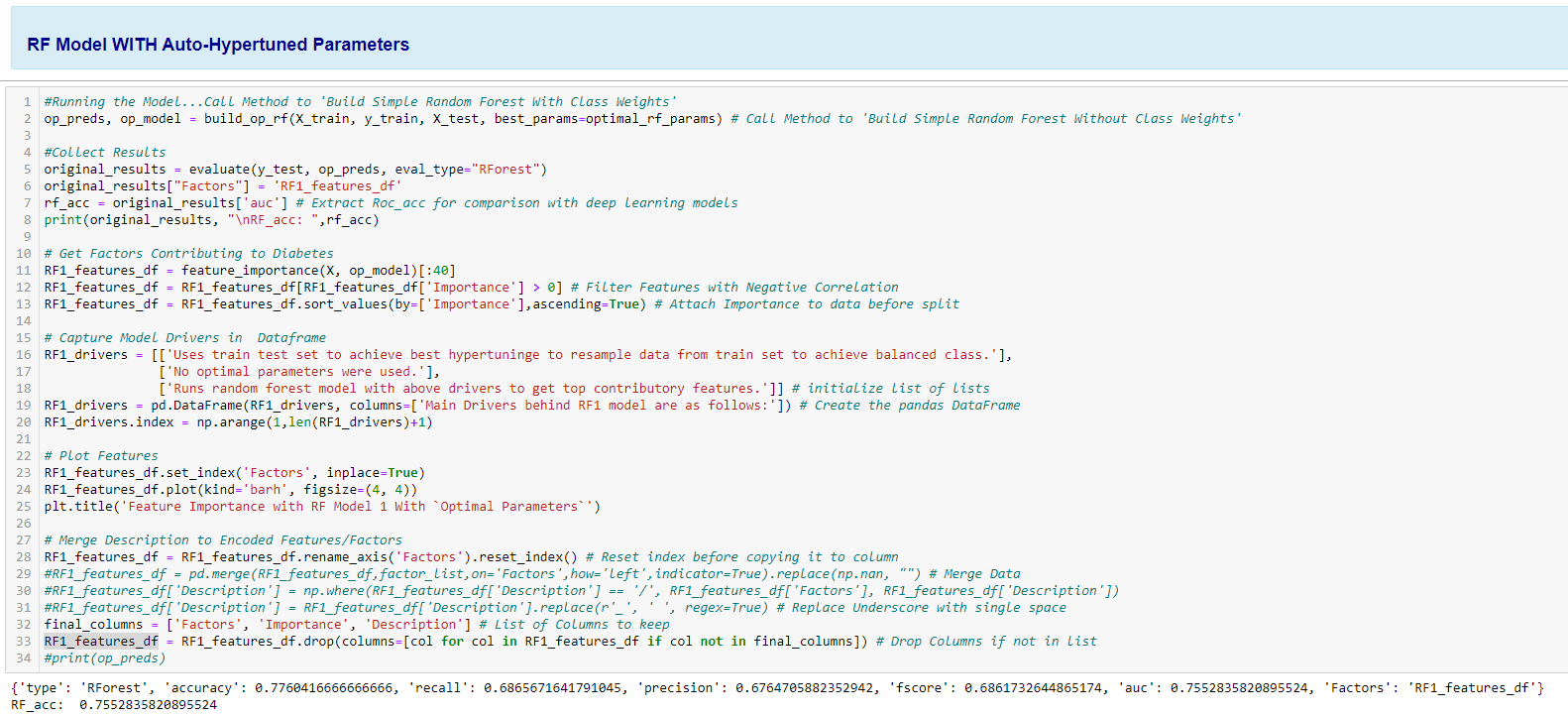


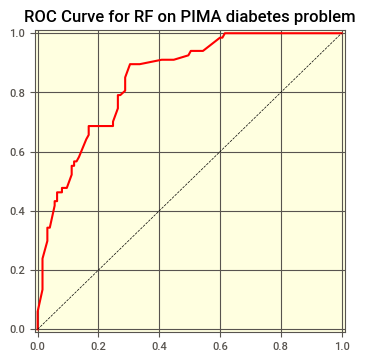


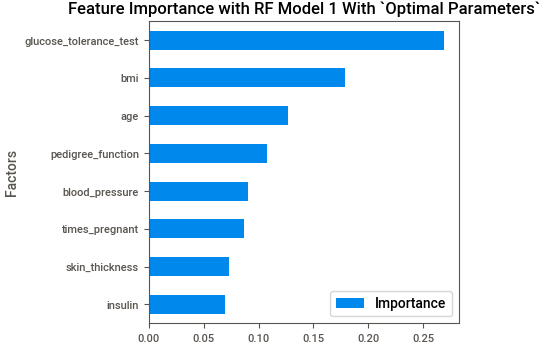




### **8) Base Model: Random Forest:**

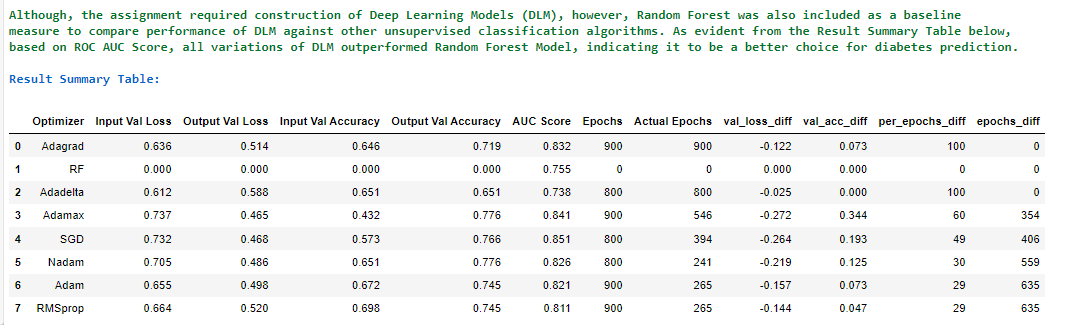




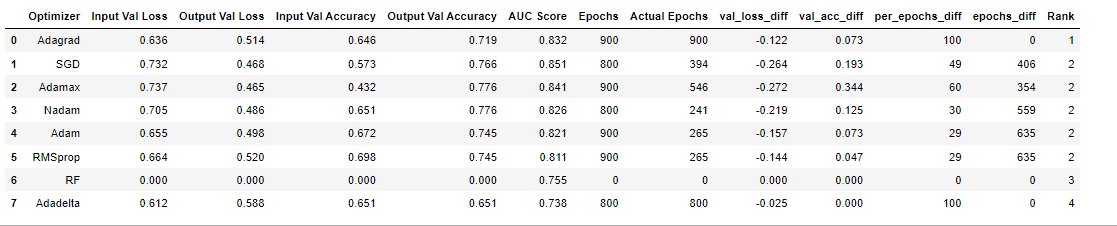


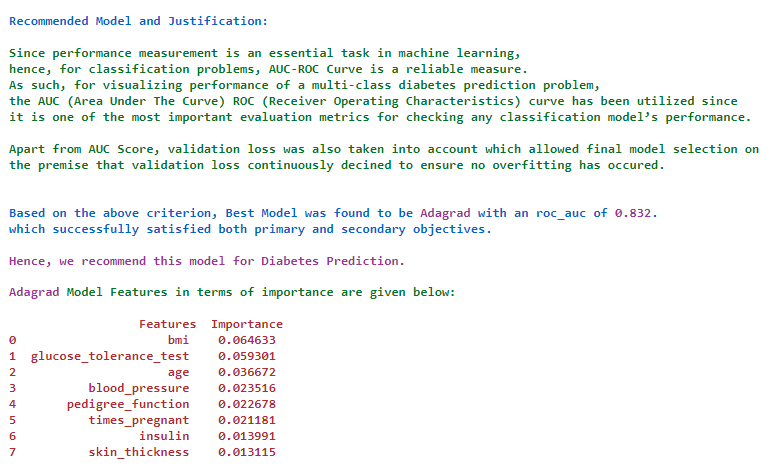
## **3d) Model Choice**

### **3d-i) Result Summary**



### **3d-ii) Model Ranking, Choice and Justification**

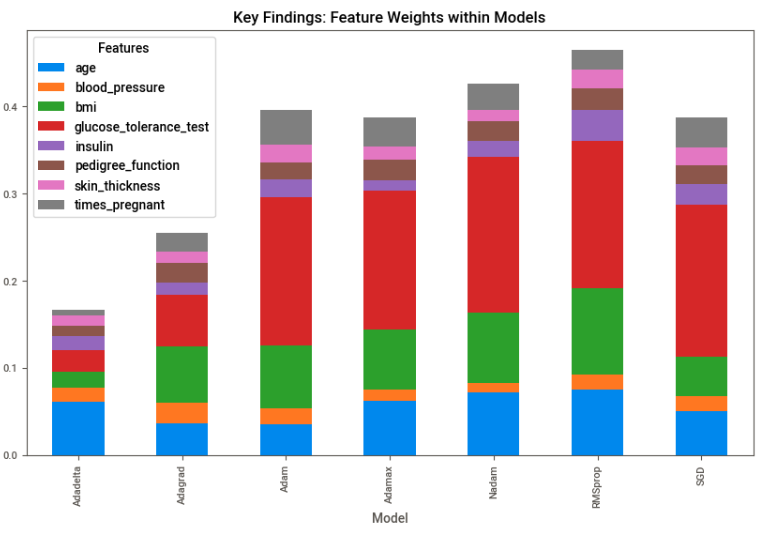




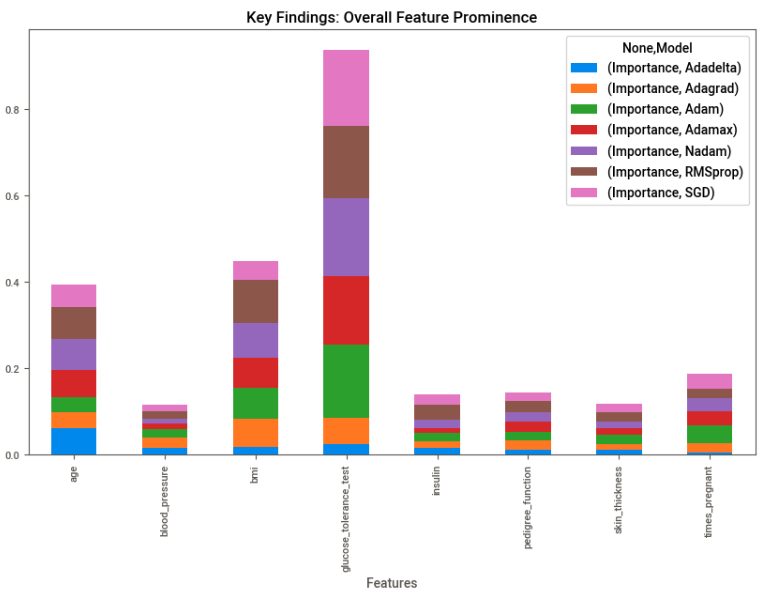
# **4) Summary Key Findings and Insights**

Model output clearly shows glucose tolerance test, bmi and age as most prominent features in most models irrespective of operator variation or auc score. Consequently, feature weights and their individual prominence is depicted as follows:

### **4a) Feature Weights within Models**



### **4b Overall Feature Prominence**



# **5) Model Shortcomings and Future Directions**

## **5a) Model Shortcomings**

Despite its novel approach to automatically pick best model with best parameters, the project is not without its shortcomings.

To begin with, it only utilized a very small dataset with merely nine features.

Hence, it may have discounted other important factors that may impact on set of diabetes.

Furthermore, the project could have employed other deep learning models to ensure even more improved results.

Then, employed Grid Search Hyperparameter Tuning proved to be computationally intensive and could have been replaced by more efficient methods. This also led to a basic model architecture which discounted incorporating other parameters like different learning rates, momentum, etc.

## **5b) Plan of Action to Revisit Analysis**

Future recommendations are, hence, as follows:

1) Use a larger dataset with more features.

2) Employ more unsupervised models like LSTM etc.

3) Try other Hyperparameter Tuning methods like Keras Tuner or Bayesian optimization (See, Section 5).

4) Include additional parameter search like learning rate, momentum, neuron activation function, etc. to make a more robust model.

# **6 Useful Links**

## **6a) Guide to Keras**

[**https://machinelearningmastery.com/use-keras-deep-learning-models-scikit-learn-python/**](https://machinelearningmastery.com/use-keras-deep-learning-models-scikit-learn-python/)

## **6b) Keras Hyperparameter Tuning Approaches**

[**https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/**](https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/)

[**https://medium.datadriveninvestor.com/hyperparameter-tuning-with-deep-learning-grid-search-8630aa45b2da**](https://medium.datadriveninvestor.com/hyperparameter-tuning-with-deep-learning-grid-search-8630aa45b2da)

[**https://towardsai.net/p/l/stop-using-grid-search-the-complete-practical-tutorial-on-keras-tuner**](https://towardsai.net/p/l/stop-using-grid-search-the-complete-practical-tutorial-on-keras-tuner)

## **6c) Model Checkpoints**

[**https://machinelearningmastery.com/how-to-stop-training-deep-neural-networks-at-the-right-time-using-early-stopping/**](https://machinelearningmastery.com/how-to-stop-training-deep-neural-networks-at-the-right-time-using-early-stopping/)

[**https://pyimagesearch.com/2021/06/30/how-to-use-the-modelcheckpoint-callback-with-keras-and-tensorflow/**](https://pyimagesearch.com/2021/06/30/how-to-use-the-modelcheckpoint-callback-with-keras-and-tensorflow/)

## **6d) Get Feature Importance in Keras**

[**https://stackoverflow.com/questions/45361559/feature-importance-chart-in-neural-network-using-keras-in-python**](https://stackoverflow.com/questions/45361559/feature-importance-chart-in-neural-network-using-keras-in-python)

[**https://www.kdnuggets.com/2020/01/explaining-black-box-models-ensemble-deep-learning-lime-shap.html**](https://www.kdnuggets.com/2020/01/explaining-black-box-models-ensemble-deep-learning-lime-shap.html)

## **6e) Shap Plots: Some Good Examples**

[**https://shap.readthedocs.io/en/latest/example\_notebooks/api\_examples/plots/decision\_plot.html**](https://shap.readthedocs.io/en/latest/example_notebooks/api_examples/plots/decision_plot.html)

## **6f) Matplot Lib**

[**https://matplotlib.org/stable/gallery/subplots\_axes\_and\_figures/subplot.html**](https://matplotlib.org/stable/gallery/subplots_axes_and_figures/subplot.html)

[**https://medium.com/mlearning-ai/shap-force-plots-for-classification-d30be430e195**](https://medium.com/mlearning-ai/shap-force-plots-for-classification-d30be430e195)

## **6f) Other Useful Models**

[**https://github.com/AI-MOO/IBM-Machine-Learning-Professional-Certificate**](https://github.com/AI-MOO/IBM-Machine-Learning-Professional-Certificate)

[**https://github.com/topics/ibm-machine-learning**](https://github.com/topics/ibm-machine-learning)

[**https://samyzaf.com/ML/pima/pima.html**](https://samyzaf.com/ML/pima/pima.html)

# **7) Github Links**

## **7a) Link to Main Folder**

[**https://github.com/FATIMASP/IBM-MACHINE-LEARNING-CERTIFICATION/tree/main/Deep%20Learning%20and%20Reinforcement%20Learning**](https://github.com/FATIMASP/IBM-MACHINE-LEARNING-CERTIFICATION/tree/main/Deep%20Learning%20and%20Reinforcement%20Learning)

## **7b) Link to Assignment Notebook**

[**https://github.com/FATIMASP/IBM-MACHINE-LEARNING-CERTIFICATION/blob/main/Deep%20Learning%20and%20Reinforcement%20Learning/DEEP%20LEARNING%20%26%20REINFORCEMENT%20LEARNING%20FINAL%20MODEL.ipynb**](https://github.com/FATIMASP/IBM-MACHINE-LEARNING-CERTIFICATION/blob/main/Deep%20Learning%20and%20Reinforcement%20Learning/DEEP%20LEARNING%20%26%20REINFORCEMENT%20LEARNING%20FINAL%20MODEL.ipynb)

# **8) References**

Naz, H. & Ahuja, S., 2020. Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes and Metabolic Disorders,* 19(1), p. 391–403.

**The grading will center around 5 main points:**

1. **Does the report include a section describing the data?**
2. **Does the report include a paragraph detailing the main objective(s) of this analysis?**
3. **Does the report include a section with variations of a Deep Learning model and specifies which one is the model that best suits the main objective(s) of this analysis?**
4. **Does the report include a clear and well presented section with key findings related to the main objective(s) of the analysis?**
5. **Does the report highlight possible flaws in the model and a plan of action to revisit this analysis with additional data or different modeling techniques?**