FDS - Final Project: Prediction of Diabetes patients

Using Linear, Logistic & Softmax Regression, Features Extraction, KNN and Transfer Learning

Arman Feili:
Matricola: 2101835
Email address:
armanfeili7@gmail.com

Rahim Rahimov
Matricola: 1921843
Email address:
rahimov.1921843@studenti.uniroma1.it

Begaiym Satarova
Matricola: 2056861
Email address:
satarova.2056861@studenti.uniroma1.it

Milad Torabi
Matricola: 2103454
Email address:
miladtorabi65@gmail.com

Matthieu TANG
Matricola: 2117284
Email address:
mata61743@eleve.isep.fr

***Abstract*— we attempted to train models that are useful for healthcare professionals in identifying patients who may be at risk of developing diabetes and in developing personalized treatment plans.**

Keywords*— *Linear Regression, Logistic Regression, SoftMax Regression, Features Extraction, KNN, Transfer Learning

I. Introduction

In this project, we used as many models as we have so far learned. We trained them for different purposes but mainly for reaching a model that can predict whether a patient has diabetes or not with the highest performance and accuracy.

The actual code for this project is accessible at [1]: https://github.com/armanfeili/FDS-Final-Project/tree/main

II. Dataset and Benchmark:

Previously, We obtained 'Diabetes prediction' dataset from the Kaggle website [2].

- The data includes medical and demographic features such as age, gender, body mass index (BMI), hypertension, heart
- disease, smoking history, HbA1c level (Hemoglobin A1), and blood glucose level.

III. Dataset and Benchmark:

We applied three steps for data understanding and data preparation, including:

- **Data Description:** Understanding the basic structure of the data, like the number of rows and columns, types of variables, and initial peek at the top rows.
- **Data Exploration:** Delving deeper into the data to identify patterns, trends, and relationships. This includes looking at summary statistics and distribution of variables.
- **Data Quality Assessment:** Identifying any issues in the data such as missing values, duplicate records, outliers, and inconsistent data formats

IV. Related work:

There are four other datasets in Kaggle website and developers worked on all of them to train models and predict diabetes patients. Mentioned datasets are as follows: Diabetes dataset 1 [3], Diabetes dataset 2 [4], Pima Indians Diabetes Database [5], Predict Diabetes dataset [6].

V. Metrics:

We used Mean Squared Error (MSE) and R-squared for Linear Regression, and we used Accuracy, Precision, Recall, and F1 score for evaluating Logistic & SoftMax Regression along with KNN.

- We used Mean Squared Error (MSE) to see how far off our predictions are from the actual values, squared (to make everything positive), and then averaged
- we used R-squared to observe how well our model fits the data
- The Accuracy metric is the overall correctness of the model
- Precision evaluation is the ratio of correctly predicted positive observations to the total predicted positive observations. It focuses on the accuracy of positive predictions.
- The Recall or Sensitivity metric is ratio of correctly predicted positive observations to the all observations in the actual class.
- Eventually, F1 score is the mean of precision and recall.

V. Methods:

A. Linear Regression

We trained Linear Regression model to find coefficients (theta) that minimizes the sum of the squared differences between predicted and actual values of blood glucose level. Since the usage of Linear Regression is more effective in predicting continuous features, we tried to predict blood_glucose_level based on other features. (A step by step explanation is provided in the project.)

Linear	***Mean Squared Error***	***R-squared***
Actual Features	1355.197070736374	0.18591360000386215
Normalized Features	**1355.197070736373**	**0.18591360000386226**

Fig. 1. Evaluation for Linear Regression.

For achieving better result, the Mean Squared Error should be decreased and R-squared should be increased. As can be seen, in Figure-1, there are no significant difference in using normalized or unnormalized featured.

B. *Logistic Regression*

We defined logistic regression model using gradient ascent that aims to maximize the likelihood function and optimize coefficients. Since Logistic Regression is better to be used for binary prediction, we used it to predict diabetes patients.

Logistic	***Accuracy***	***Precision***	***Recall***	***F1 Score***
Actual Features	0.911	=~ 0	=~ 0	=~ 0
Normalized Features	**0.912**	**0.505**	**0.442**	**0.472**

Fig. 2. Evaluation for Logistic Regression.

As it is illistrated in Figure-2, if we do not use normalized data as our features, we end up precision 0. However, if we use Normalized data, we can reach some values for Precision, Recall and F1 Score. Using Standardization method is much better than MinMax method of normalization. Finally, if we use both methods for normalization, we reach the best results among other evaluations.

C. *Multinomial Classification (Softmax Regression Model)*

We defined the Multinomial Logistic Regression model using gradient descent to optimize weights and biases. Since we needed more than two classes to predict, we merged all three 'diabetes,' 'heart_disease,' and 'hypertension' features as our target value. We calculate the model scores by multiplying input features with weights and adding biases. Finally, we compute probabilities for each class using the SoftMax function and Select the class with the highest probability as the predicted class for each input sample.

SoftMax	***Accuracy***	***Precision***	***Recall***	***F1 Score***
Actual Features	**0.837**	0.701	**0.837**	0.763
Normalized Features	0.832	**0.830**	0.832	**0.831**

Fig. 3. Evaluation for Multinomial Classification.

As it is obvious in Figure-3, using Normalized data can improve all Accuracy, Precision, Recall, and F1 Score. Again, Standardization method worked better in compared to MinMax normalization method. If we Consider both MinMax and Standardization normalized features, we get the best result possible.

D. *Histogram Features Extraction*

For “Blood Glucose Level” and “BMI”, we calculated the Mean, Variance, and Skewness. - we compared results in both Linear Regression model and Logistic Regression Model.

Linear regression	***Mean Squared Error***	***R-squared***
Actual Features	**1354.9938168420365**	**0.18603569754576132**
Extracted Features	1355.2540408736168	0.18587937722184966

Fig. 4.1. Evaluation of Linear regression using extracted features.

Logistic regression	***Accuracy***	***Precision***	***Recall***	***F1 Score***
Actual Features	0.91	0	0	0
Extracted Features	0.91	0	0	0

Fig. 4.2. Evaluation of Logistic regression using extracted features.

As it is demonstrated in Figure-4, using mean, variance, and skewness for BMI instead of just BMI itself does NOT improve the results of model regarding Linear and Logistic regression.

E. *k-Nearest Neighbors (KNN) Classifier:*

We iterated through each test point and calculated the distances between this test point and all the training points (square root of the sum of squared differences to measure the distance between points in a multi-dimensional space). Then, we found the k nearest neighbors based on the smallest distances calculated. We performed majority voting and assigns the most frequently occurring label among its k nearest neighbors as the prediction for the test point. Finally we returned an array of predicted labels for all the test points.

KNN	***Accuracy***	***Precision***	***Recall***	***F1 Score***
Actual Features	0.953	0.883	0.545	0.674
Normalized Features	**0.992**	**0.998**	**0.917**	**0.956**

Fig. 5. Evaluation of (KNN) Classifier.

As can be seen, in Figure-5, using normalized data can improve KNN better than other models. Here we can observe that using MinMax normalized features, in comparison with Standardization normalized features, can reach the most Accuracy, Precision, Recall and F1-score. Besides, Using both MinMax and Standardization normalized features at the same time, wan NOT a good idea and although the results are better than real features, they are worse than MinMax and Standardization as individual normalization models.

F. *Transfer learning (Using Adam neural network model)*

Adam stands for Adaptive Moment Estimation and combines techniques from two other extensions of stochastic gradient descent: AdaGrad and RMSProp. It computes adaptive learning rates for each parameter by keeping track of both the first and second moments of the gradients. Adam optimizer is an optimization algorithm used to update the weights of the network during training.

Models	***Accuracy***	***Precision***	***Recall***	***F1 Score***
Logistic Model	0.896	0.403	0.341	0.369
Adam Model	**0.969**	**0.985**	**0.667**	**0.795**
Softmax Model	0.441	0.865	0.441	0.479
Adam Model	**0.892**	**0.841**	**0.415**	**0.556**

Fig. 6. Evaluation of Adam optimizer model compared to our Linear and Logistic models.

As can be seen, in Figure-6, the pre-trained Tensorflow model as "adam" works way better than our simple Linear and Logistic Regression in every metrics.

VI. CONCLUSION

After trying multiple models to predict diabetes patients, we found different approaches work better for different models. Some procedures including feature selection and normalization, made the models better. especially using Standardization in Logistic and SoftMax models. However, for KNN, using MinMax made a big difference. Feature extraction in our case, did not help us with the results, and finally, trying Transfer Learning was much better than the simple models we started with. Overall, by testing different ways, we can find better methods to help practitioners treat patients better

VII. TEAM ASSIGNMENTS:

Arman Feili:

- Merging, debugging and combining all parts of the project together.
- Logistic Regression
- SoftMax Regression
- Transfer Learning
- Two PowerPoints
- Presented twice
- Report

Rahim Rahimov:

- Data Understanding
- Data Preparation
- Cooperating in report

Begaiym Satarova:

- Cooperating in Transfer Learning
- Cooperating in PowerPoints
- Cooperating in presentation

Milad Torabi:

- Linear Regression
- Cooperating in presentation
- Adding explanations
- Debugging in evaluation parts

Matthieu Tang:

- KNN
- Participating in presentations

VIII. REFERENCES

[1] Arman Feili, Rahim Rahimov, Begaiym Satarova, Milad Torabi, and Matthieu TANG, "FDS-Final-Project-Feili-Satarova-Rahimov-Torabi-Tang." Accessed: Dec. 27, 2023. [Online]. Available: https://github.com/armanfeili/FDS-Final-Project/tree/main

[2] "Diabetes prediction dataset." Accessed: Dec. 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset/data

[3] "Diabetes Dataset - 1", Accessed: Dec. 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset

[4] "Diabetes Dataset - 2", Accessed: Dec. 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/mathchi/diabetes-data-set

[5] "Pima Indians Diabetes Database", Accessed: Dec. 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[6] "Predict Diabetes", Accessed: Dec. 27, 2023. [Online]. Available: https://www.kaggle.com/datasets/whenamancodes/predict-diabities