DMRAAW: Diabetes Mellitus Risk Assessment App for Women

Ifeanyi Osuchukwu1, Samuel Yoon1, Jintong Hou2

1School of Biomedical Engineering, Drexel University, USA

2School of Public Health, Drexel University, USA

Course: BMES 550

Instructor: Ahmet Sacan

Date: 2021-12-7

### **ABSTRACT**

DMRAAW is web-based graphical user interface design for physicians to collect, store, and assess Diabetes risk factors for their female patients. The user is able to store their data into a SQLite-database and also calculate individual diabetes risk scores by logistic regression using five predictors (number of pregnancies, blood glucose level (mg/dL), BMI(kg/m2), Diabetes pedigree function value, age (in years). Risk factors are consistent with CDC report[1] , but the proportion of diabetes in the initial dataset (200 records) is higher than 9.5% prevalence in women as reported by CDC, 2020[2] . The application is implemented by php and python scripts. The information is then stored into a SQLite databse.

### **1 INTRODUCTION**

Diabetes is a metabolic disease in which a person has high blood sugar due to problems of processing or producing insulin. Diabetes can affect people of any age, race, sex, or any lifestyle. There are two types of diabetes, type 1 which is more common in children and type 2 which is more common in adults. Type 1 diabetes is caused by an autoimmune response to insulin producing beta cells in the pancreas and is associated with genes and environmental factors.

Type 2 diabetes is due to increased insulin resistance, associated with weight gain, inactive lifestyles, and poor diet. Type 2 diabetes is more frequently found in men, especially at ages of 35-54, where men are twice as likely to develop diabetes, with onset at a much lower average BMI.

Between 1971 and 2000, the death rate for men with diabetes fell, according to a study in Annals of Internal Medicine [3] . This decrease reflects advances in diabetes treatment. But the study also indicates the death rate for women with diabetes didn’t improve. In addition, the difference in death rates between women who had diabetes and those who didn’t more than doubled.

Many studies were performed to find the relationship between predictors and women's diabetes. Rapid lifestyle change, however, in these days would lead to significantly different results from old data of predictor contribution. Up to the date data can represent the current trend of diabetes accurately. With this in mind our goal was to create an app that could help physicians predict Diabetes risk for their female patients. The Diabetes Risk Assessment App will collect patient data using an intuitive GUI. The updated database will provide current diabetes predictor trends to health care professionals. With better databases, researchers and physicians can predict and provide better guidance for patients to control their diabetes.

**2 DATASET**

The dataset in Figure 1., contains 10 fields: ID (unique), First name, Last name, survey year, diabetes (Yes/No, outcome variable), five predictors (number of pregnancies, blood glucose level (mg/dL), BMI (kg/m2), Diabetes pedigree function value, age (in years). Data is collected from individuals’ medical/family history, physical examination (BMI), and blood test (glucose), and records are entered by a physician. The initial 200 records used for development and demonstration purposes were imported from the publicly available pima dataset, which is originally from the National Institute of Diabetes and Digestive and Kidney Diseases.

Table

Description automatically generated

**Figure 1.** **Database Schema.** ER diagram showing all ten fields used to store information on a patient. A physician, when inputting new records into the app, needs to input every field excluding the id as that will be auto generated.

**3 METHODS AND IMPLEMENTATION**

**Software**: apache, php, Python (version 3.0 or later). Python library used: sys, os, sqlite3, pandas, numpy , matplotlib, statsmodels, sklearn, statistics.

**Statistical method used**:

\*Summary statistics (Python module: statistics) :mean, standard deviation, median, minimum, and maximum for predictors; frequency and percentage for survey year).

\*Model (logistic regression, python module: statsmodels)

\*Plots :histogram for predictors, ROC curve (python module sklearn) for model, both plots use the matplotlib module. The model estimates are used to predict risk.

**Implementation details**: The project is developed in the Windows operating system. Programming languages: php, python, and SQLite were utilized; the flowchart in Figure 2 outlines the design of the app.

All files share the same database saved in a temporary folder in the user's local computer; the initial database will be copied from the datatemplate folder within the project folder if it does not exist in the temporary folder. The dbtestconn.php file does this and connects to all .php files. All temporary plots (ROC curve, histogram) are temporarily saved in the same temporary folder.

The main page has four entries:

1. Review database: a web page showing all data in the database.
2. Enter new records: a web page allowing users to enter values for all fields of a patient; incomplete entries are allowed. Users will be directed to the review database web page after submission of a new record to view the most current database.
3. Search/update database: a web page allowing users to search patients by first name, last name, ID, and/or survey year. Result will show all fields of the selected records, with update/delete option.
4. Risk analysis: a web page allowing the user to view summary statistics for all predictors and survey year, and fit a model dynamically to predict the risk of developing diabetes and get model information from 0-5 predictors as determined by the user based on the most up-to-date data from the database. The user has the option to select subsamples to perform the analysis based on the survey year range. By default, negative values and non-numbers are discarded, and zeros are discarded for Glucose, BMI,Diabetes pedigree function value, and age. User inputs are sanitized either by removing non - numbers or set to missing, and sanitized inputs used for calculation will be returned to the output window.  
    The three python scripts (for calculating risk and summary statistics) receives the user inputs and the database absolute path sent from Analysis.php file to connect to database, and use the user inputs to build sql query or statistical model formula, then use the query to select valid records (e.g. non-negative numbers) from database for further processing and use the formula to fit the corresponding model. For summary statistics, some invalid values were removed using python loops after records were selected from the database.   
    After calculation, python scripts save figures (ROC curve, histogram) to the temporary folder for Analysis. php to use, and send statistics as a list of strings back to the Analysis.php file. The php file identifies the meaning of each element in the list of which the number of elements varies according to the number of user inputs, and then makes necessary transformations (e.g. transformation of the parameter estimates to odds ratios according to their mathematical relationship is done in php file) for the final outputs.

Diagram

Description automatically generated

**Figure 2. Flowchart.** Design of The Diabetes Mellitus Risk Assessment App and the scripts used to create the interface.

### **4 EXPERIMENTS AND RESULTS**

All features of the Diabetes Mellitus Risk Assessment App were tested. In Figure 3 the interface of adding new record to the database is shown. Figure 4 shows an example of the analysis portion of the app. A diabetes risk factor score is provided by a logistic regression and risk estimation based on 4 user inputs: after entering a new record with valid values in all fields entered in addition to the initial 200 records, go to risk analysis page, and enter 1 for Number of pregnancies, 120 for glucose level, 30 for BMI and age, click “calculate” button, a snapshot of result is shown in Figure 4. The result shows more pregnancies, higher glucose level, BMI, and age are associated with diabetes with odds ratio greater than one. This is consistent with the CDC report [1] . The association is significant for BMI and glucose level, not significant for the other two.

Graphical user interface

Description automatically generated

**Figure 3.** **New Records.** Shows what the user will see when entering new records into the database.

A screenshot of a computer

Description automatically generated with medium confidence

**Figure 4. Calculation outputs.** It shows the predicted risk in the first row, followed by the user input information of the individual being predicted. Sample uses all years by default. Three tables of model results and the plot of the ROC curve showing the prediction ability of the model are shown on the same page. Number of observations is 201, this is because one valid record is entered in addition to the initial 200 records.

The database can also be modified by a user as outlined in Figure 5. The app allows for the deletion and or editing of a record that exists.

Graphical user interface, application

Description automatically generated

**Figure 5. Database Search.** The user can search for existing records in the database if no results are found, a warning will appear. The app allows for editing and deleting records.

**5 DISCUSSION**

Our website allows users to enter, update, delete data and perform statistical analysis dynamically using the most up-to-date dataset, it shows summary statistics, plots, model information and predictions allowing physicians to manage patient data while doing research within the same system. Predictions are based on data within the system; this may be more precise if the individual seeking a risk score shares more common factors such as spatial, cultural, temporal, biological, and social economic factors with the sample population already in our database.

In its current state this app does have some limitations. The main database is housed locally in the user’s computer instead of remotely on a server. Consequently, changes to the database from different computers are not communicated between computers therefore the data is not synced. Searching the database is not fully dynamic yet. Users are only able to edit or delete on record at a time. Also, users are unable to view summary statistics for each specific model predictor.

Future extensions of our app would include adding security features, such as better protecting against SQL injections when entering records into the database. We would create a user login interface to protect patients’ personal information and give proper permissions to users as everyone shouldn’t have the same access to the database. The analysis would benefit more if there were summary statistics for each specific model predictor. More predictors used in the calculation would enhance the statistics and accuracy of the app.

### **6 REFERENCES**

1. Diabetes Risk Factors”, Centers for Disease Control and Prevention, https://www.cdc.gov/diabetes/basics/risk-factors.html
2. “National Diabetes Statistics Report 2020, Estimates of Diabetes and Its Burden in the United States”, Center for Disease Control and Prevention,
3. https://www.cdc.gov/diabetes/pdfs/data/statistics/national-diabetes-statistics-report.pdf
4. Edward W. Gregg, PhD, et al. “Mortality Trends in Men and Women with Diabetes, 1971to 2000” Annals of Internal Medicine, articles7 August 2007 https://doi.org/10.7326/0003-4819-147-3-200708070-00167. Accessed 5 Dec. 2021