Case Studies from Industry on Big Data

Presented by group 4

African Institute for Mathematical Sciences, AIMS-Senegal

Supervised by Dr Bubacarr Bah
from MRC Unit The Gambia at LSHTM

May 16, 2024

Group members

The group members are:

- Mamady KONATE
- Ndeye Anta SISSOKHO
- BAYANA NDOBA Merveille

Overview

1. Introduction
2. Problem
3. Data Preparation
4. Data visualization
5. Build Model
6. Result
7. Conclusion

Introduction

The introduction introduces diabetes mellitus type 1 (formerly IDDM), an autoimmune disease causing insulin deficiency. It describes the aim to develop a suitable machine learning model for a dataset of type 1 diabetes patients, opting for linear regression due to its simplicity and effectiveness. The structure of the project includes dataset description, data preprocessing, ML algorithm choice, performance comparison with existing literature, and discussion of challenges faced.

1. hyperglycemia (When the average blood glucose level exceeds 200 over several years.)
2. hypoglycemia (low BG) symptoms fall into two classes. Between 40-80 mg/dl
3. normal .

Problem

Patients with IDDM can experience insulin deficiency due to either low production by the pancreas's beta islet cells after an autoimmune attack or insulin resistance, commonly seen in older individuals and those with obesity. This deficiency leads to metabolic effects, primarily hyperglycemia, which poses risks for vascular problems like retinopathy over time. The therapeutic goal for IDDM is to maintain average blood glucose levels as close to normal as possible, despite the challenge posed by BG level variations. Lowering the mean BG increases the frequency of low BG levels, which can be unpleasant or dangerous for patients.

Outpatient management of IDDM

Outpatient management of IDDM relies principally on three interventions: diet, excercise and exogenous insulin. Proper treatment requires careful consideration of all three inter-ventions.

Objectives

Our goal in this project is to use machine learning techniques to classify patients into one of the following categories:

- **Hyperglycemia:** When the average blood glucose level exceeds 200 over several years.
- **Hypoglycemia:** Low BG symptoms fall into two classes. Between 40-80 mg/dl.
- **Normal:** To ensure appropriate treatment for patients with Type I diabetes (IDDM), the goal is to maintain average blood glucose levels as close as possible to the normal range. This includes keeping pre-meal BG between 80-120 mg/dl and post-meal BG between 80-140 mg/dl. While the target range for individuals with diabetes is debated, it's considered beneficial to have 90 measurements below 200 mg/dl and an average BG of 150 mg/dl or less.

Domain Description

The data used in this project describes the physiology and pathophysiology of diabetes mellitus and its treatment. Data-[01-70]: These datasets span several weeks to several months of outpatient care for 70 patients. File Names and Format:

1. Date in MM-DD-YYYY format
2. Time in XX:YY format
3. Code
4. Value

Domain Description

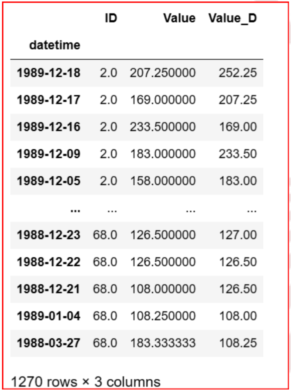
The Code field is deciphered as follows:

- **33:** Regular insulin
- **34:** NPH insulin dose
- **35:** UltraLente insulin dose
- **48:** Unspecified blood glucose measurement
- **57:** Unspecified blood glucose measurement
- **58:** Pre-breakfast blood glucose measurement
- **59:** Post-breakfast blood glucose measurement
- **60:** Pre-lunch blood glucose measurement
- **61:** Post-lunch blood glucose measurement
- **62:** Pre-supper blood glucose measurement
- **63:** Post-supper blood glucose measurement
- **64:** Pre-snack blood glucose measurement

Data Preprocessing and Exploration

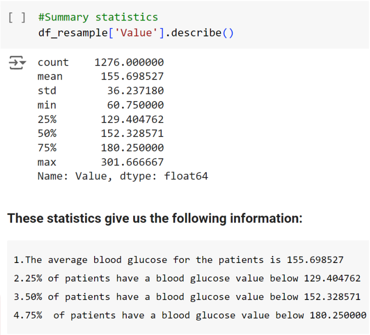
In this project, our focus is on time series analysis to predict daily blood glucose levels. To achieve this, we performed several tasks. Firstly, we converted the target variable (Value) to numeric format. Since our data involves time series, we combined the Time and Date columns into one datetime column and created an index typical of time series analysis. Since the data points are irregularly spaced, we generated 24-hour lags to facilitate daily predictions. Additionally, we created an independent variable D based on the target variable for these 24-hour lags.

Data Preprocessing and Summary Statistics



datetime	ID	Value	Value_D
1989-12-18	2.0	207.250000	252.25
1989-12-17	2.0	169.000000	207.25
1989-12-16	2.0	233.500000	169.00
1989-12-09	2.0	183.000000	233.50
1989-12-05	2.0	158.000000	183.00
...	...	...	...
1988-12-23	68.0	126.500000	127.00
1988-12-22	68.0	126.500000	126.50
1988-12-21	68.0	108.000000	126.50
1989-01-04	68.0	108.250000	108.00
1988-03-27	68.0	183.333333	108.25

1270 rows × 3 columns



```
#Summary statistics
df_resample['Value'].describe()
```

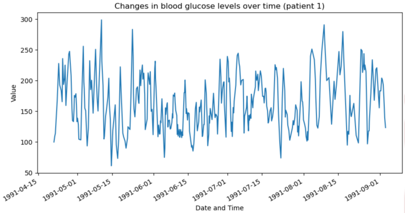
```
count    1276.000000
mean      155.698527
std        36.237180
min        60.750000
25%       129.404762
50%       152.328571
75%       180.250000
max       301.666667
Name: Value, dtype: float64
```

These statistics give us the following information:

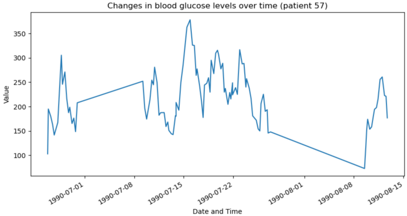
1.The average blood glucose for the patients is 155.698527
2.25% of patients have a blood glucose value below 129.404762
3.50% of patients have a blood glucose value below 152.328571
4.75% of patients have a blood glucose value below 180.250000

Data Visualisation

This plots represent the variation in blood glucose levels over time for each patient (moving average per day).



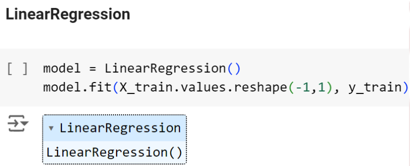
Changes in blood glucose levels over time (patient 1)



Changes in blood glucose levels over time (patient 57)

Model

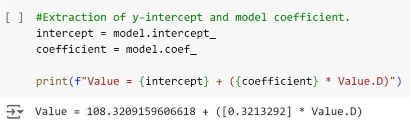
In statistics, linear regression is a statistical model which estimates the linear relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables).



LinearRegression

```
[ ]  model = LinearRegression()
     model.fit(X_train.values.reshape(-1,1), y_train)
```

```
▾ LinearRegression
LinearRegression()
```



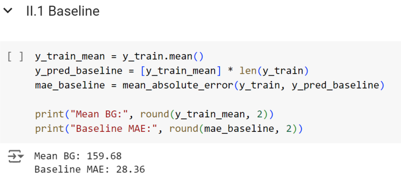
```
[ ]  #Extraction of y-intercept and model coefficient.
     intercept = model.intercept_
     coefficient = model.coef_

     print(f"Value = {intercept} + ({coefficient} * Value.D)")
```

```
Value = 108.3209159606618 + ([0.3213292] * Value.D)
```

Model Evaluation

For the evaluation, we use as metric the mean absolute error (MAE). The MAE for the training and test set are above the MAE baseline so we have a prediction.

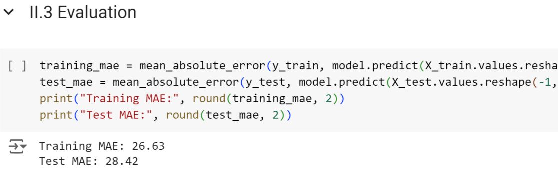


II.1 Baseline

```
[ ] y_train_mean = y_train.mean()
    y_pred_baseline = [y_train_mean] * len(y_train)
    mae_baseline = mean_absolute_error(y_train, y_pred_baseline)

    print("Mean BG:", round(y_train_mean, 2))
    print("Baseline MAE:", round(mae_baseline, 2))
```

```
Mean BG: 159.68
Baseline MAE: 28.36
```



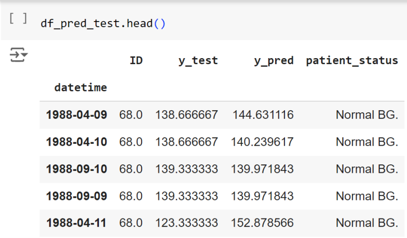
II.3 Evaluation

```
[ ] training_mae = mean_absolute_error(y_train, model.predict(X_train.values.resha
    test_mae = mean_absolute_error(y_test, model.predict(X_test.values.reshape(-1,
    print("Training MAE:", round(training_mae, 2))
    print("Test MAE:", round(test_mae, 2))
```

```
Training MAE: 26.63
Test MAE: 28.42
```

Result

After training our test set, we get the following predictions.



```
[ ] df_pred_test.head()
```

	ID	y_test	y_pred	patient_status
datetime				
1988-04-09	68.0	138.666667	144.631116	Normal BG.
1988-04-10	68.0	138.666667	140.239617	Normal BG.
1988-09-10	68.0	139.333333	139.971843	Normal BG.
1988-09-09	68.0	139.333333	139.971843	Normal BG.
1988-04-11	68.0	123.333333	152.878566	Normal BG.

Conclusion

Our job in this project is to alert patients to severe cases (hyperglycemia and hypoglycemia) based on the blood carbohydrate values collected for each patient. And in this work, the biggest challenge was to completely change our approach to machine learning projects, so we had to treat this project as it is customary for time series, it was very complicated but we managed it. We wanted to use other options such as ARMs, but due to time constraints we couldn't. So, all in all, it was challenging, but we think we've done the essential

THANK YOU FOR YOUR ATTENTION