Enhancing Predictive Models for Hospital Readmission: A Machine Learning Approach to Analysing HbA1c Impact

Luis Irisarri*
Institute for Cross-Disciplinary Physics and Complex Systems (IFISC), CSIC-UIB, Palma de Mallorca, Spain
(Dated: April 23, 2024)

Building upon the foundational work of [1], this study reanalyses the impact of HbA1c measurement on hospital readmission rates using advanced data analysis techniques on a subset of the original dataset. By applying various machine learning (ML) models, we aimed to elucidate the patterns and predictors of patient readmissions within 30 days post-discharge. Our comprehensive analysis involved data preparation, feature selection, and rigorous assessment of each model's performance, efficiency, and interpretability. The findings indicate that XGB excels in accuracy, AUC, and cost-effectiveness, whereas DT offers the best interpretability, making it preferable in contexts requiring clear decision-making. This study not only enhances our understanding of factors influencing hospital readmissions but also underscores the importance of selecting appropriate ML models based on specific medical needs.

I. INTRODUCTION

Hyperglycaemia, characterized by high blood glucose levels, significantly impacts both diabetic and non-diabetic patients by increasing risks of morbidity and mortality. The widespread availability of vast clinical databases today enables the utilization of advanced data analysis techniques, enhancing our ability to predict patient outcomes and assist in clinical decision-making. Building upon the foundational work of Strack *et al.*, which analysed the impact of HbA1c measurement on hospital readmission rates [1], this study utilizes similar large-scale clinical datasets to delve deeper into the patterns and predictors of patient readmission within 30 days post-discharge.

II. MATERIALS & METHODS

In this section, we describe the experimental setup. We start with data exploration and processing in §II A, detailing the steps taken to prepare the dataset. For alternative data preparation methods, refer to [2–5]. We then discuss the ML models used in this study in Section §II B. All computational analyses were performed using `Python` and its data analysis libraries. More details about the coding may be found in `https://github.com/liris8`.

A. Data Exploration & Processing

We analysed a dataset comprising 5000 patient instances and 37 features, detailed in tab. I. This dataset is a subset of a larger one referenced in [6], where a more comprehensive explanation of feature definitions can be found. The feature variable names are largely self-explanatory.

Feature name	Type
Encounter ID	Numeric
Patient number	Numeric
Race	Categorical
Gender	Categorical
Age	Categorical
Admission type	Categorical
Discharge disposition	Categorical
Admission source	Categorical
Time in hospital	Numeric
Num. lab procedures	Numeric
Num. procedures	Numeric
Num. medications	Numeric
Num. outpatient visits	Numeric
Num. emergency visits	Numeric
Num. inpatient visits	Numeric
Diagnosis 1	Numeric
Diagnosis 2	Numeric
Diagnosis 3	Numeric
Diagnosis 4	Numeric
Num. diagnoses	Numeric
Medications (15)	Categorical
Change of medications	Categorical
Diabetes medications	Categorical
Readmitted	Categorical

Table I. **Dataset features and types**. See [6] for further details on the specific definition of the features.

To enhance clarity, we decoded the originally encoded features: `admission_type_id`, `discharge_disposition_id`, `admission_source_id`, and `diag_x` for $x \in \{1, 2, 3\}$ back to their original categorical attributes. We excluded the `diag_4` feature due to its spurious data, which is discussed later. The mappings for ID fields are available in the `IDS_mapping.csv` file from the original database, and the diagnoses mappings are detailed in table 2 of [1]. For ease of reference and readability, these mappings are included in the supplementary materials under tables V to VIII. Moreover, tables V to VII also feature an additional *Classification* column, which indicates the

* luis.irisarri1@estudiant.uib.es

new categorical encoding used in our analysis to simplify the number of potential values for each feature, similar to the approach in [1] (refer to table 3 of the paper).

Having defined our data, we now examine and process it, beginning with the distribution of missing values. Our analysis reveals missing entries in the following features: `race` (112), `admission_type_id` (515), `discharge_disposition_id` (231), `admission_source_id` (316), and `tolbutamide` (3993), where the numbers in parentheses indicate the count of missing values. Then, we evaluated the distribution of values per feature. We identified several issues:

1. **Mono-value Features**: `acetohexamide`, `tolbutamide`, `troglitazone`, `examide`, and `metformin-rosiglitazone` are redundant, having only one value across all instances.

2. **Nearly Mono-value Features**: `chlorpropamide`, `tolazamide`, and `glipizide-metformin` exhibit $\sim$ 99.5% of instances with a single value, rendering them almost redundant. We will retain these features initially and rely on machine learning models for feature selection to reassess their relevance, though we anticipate minimal contribution to model performance.

3. **Spurious Feature**: The `diag_4` feature is removed from the dataset due to its unique value per patient and non-standard encoding, which includes many floating points unlike other `diag_x` features.

4. **Imbalanced Distribution**: The `race` feature shows significant imbalance: Caucasian (3759), African American (923), Hispanic (95), Other (72), and Asian (39). We have consolidated Hispanic, Other, and Asian categories into a single 'Other' category to address this imbalance.

This careful consideration ensures a cleaner, more relevant dataset for subsequent modeling stages.

From our previous analysis, we identified 163 duplicate patient entries in the dataset. We initially used these duplicates to mitigate the impact of missing values. After addressing missing values, we removed the duplicates to prevent biases in our analysis, retaining only the first entry for each duplicated patient. Furthermore, we eliminated both the encounter ID and patient number from the dataset as these identifiers are irrelevant to our analysis, ensuring a focus on medically relevant features.

We excluded all patient records categorized as expired or hospice from our dataset. Patients who have expired are not pertinent to future readmission predictions, and those in hospice care, given their end-of-life status, are also not relevant to our analysis focused on potential readmissions. This ensures our study accurately targets a population where readmission is a feasible outcome.

To prepare the dataset for machine learning algorithms, which require numerical inputs, we encoded the categorical variables as follows:

1. **Binary Features**: The features `gender`, `admission_type_id`, `discharge_disposition_id`, `admission_source_id`, `change`, `diabetesMed` and `readmitted` were encoded as 0 for negative outcomes and 1 for positive outcomes.

2. **Ordinal Feature**: The age groups, originally formatted as ranges like $[x0 - (x+1)0)$, were mapped to a decimal representation, $0.x$, aligning each decade with a proportional incremental value.

3. **Nominal Features**:

   - **Race**: Applied one-hot encoding to transform the categorical race data into binary vectors, ensuring no ordinal implications influence the model.
   - **Diagnosis Codes** (`diag_x`): Mapped the diagnosis codes to decimal values based on their relevance to diabetes, with specific conditions assigned values that reflect their impact or relation to diabetes, e.g., injuries at 0.1 through to direct diabetes-related issues at 1.0.
   - **Medications**: Encoded medication statuses with a scale that quantifies changes in medication as follows: No: 0.0, Down: 0.33, Steady: 0.66, Up: 1.0. This encoding attempts to reflect the intensity or reduction of treatment.

Additionally, we applied Min-Max scaling to normalize the data, ensuring all features operate on a consistent scale to prevent undue influence on models sensitive to feature scaling. This preprocessing step facilitates fair comparison and evaluation across different machine learning algorithms, mitigating potential bias introduced by features operating on different scales.

Finally, for the missing values, we have followed two procedures: the *conservative* and the *probabilistic*. In the *conservative* approach, we have removed all the rows with missing values, which also removes lots of valuable data. Conversely, in the *probabilistic* approach, we have assigned a probability to each feature attribute according to its frequency, and we imputed the missing values based on that distribution. After testing the resulting datasets with our machine learning models, we found that the probabilistic approach performs better in all cases. Henceforth, we have exclusively considered the probabilistic procedure for handling missing data. Lastly, we have shuffled all the instances to avoid possible biases induced by the data gathering.

To sum up, we have removed duplicates, expired, and hospice patients, leaving us with a data frame of 4711 instances and 32 features, one of which is the class (`readmitted`). All data has been encoded numerically, and missing values have been filled using the probabilistic approach. The data is normalized and the instances

shuffled. Now the data is ready for the machine learning models.

B. Modelling

To assess the classification modelling, we considered a variety of ML algorithms. Specifically, we have included *K-Nearest Neighbours* (KNN), *Decision Trees* (DT), *Random Forest* (RF), *Gaussian Naive Bayes* (GNB), *Support Vector Machine* (SVM), *Logistic Regression* (LR), and *XGBoost* (XGB). Apart from the GNB, these models each have distinct hyper-parameters, and we explored most of the possible configurations to achieve the most accurate results. See tab. II for the final hyper-parameters considered for each model.

Model	Hyper-parameters
KNN	`n_neighbouhrs=25`, `weights='distance'`
DT	`criterion='entropy'`, `max_depth=4`
RF	`criterion='entropy'`, `max_depth=6`, `n_estimators=128`
SVM	`kernel='rbf'`, `probability=True`
LR	`solver='lbfgs'`, `max_iter=1000`
XGB	`n_estimators= 128`, `eta= 0.1`, `max_depth= 6`, `subsample= 0.8`, `colsample_bytree= 0.8`

Table II. **Hyper-parameters for the ML models**. This table displays the hyper-parameters considered for each model, selected to optimize performance. Note: GNB is not included as it does not have hyper-parameters.

For each model, we assessed both performance and efficiency. Regarding performance, we measured the *Accuracy* (A), AUC (Area Under the Curve), and the Cost (C) using a predefined cost matrix (see tab. III for details).

Cost Matrix		Predicted Label 0	Predicted Label 1
True Label	0	-1	1
	1	10	-1

Table III. **Cost Matrix**.

In terms of efficiency, we assessed two time metrics: *Training Time* t_t, the duration required to train the model on the entire dataset, and *Prediction Time* t_p, the duration to predict the labels for a single instance.

For performance metrics, we performed cross-validation with 10 folds and recorded the mean and standard deviation of each measure. For the efficiency metrics, due to the small values obtained, we calculated the mean and standard deviation of the training and prediction times over 10 runs.

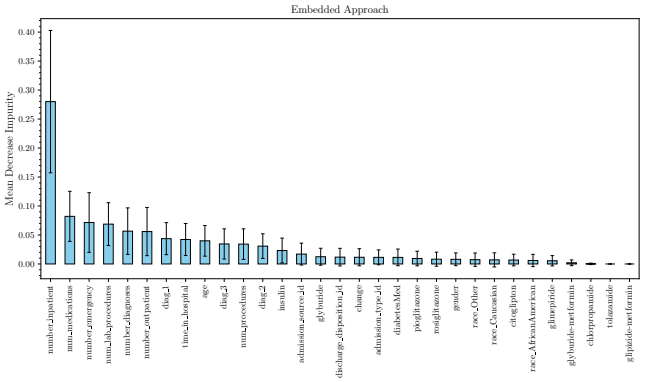

Figure 1. **Feature Importances via Embedded Approach**. In this figure, we show the feature importances obtained from the RF model with the hyper-parameters specified in table II. The importance is measured in terms of the mean decrease impurity.

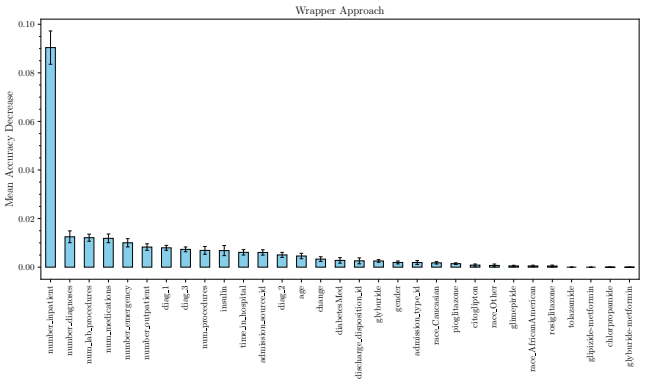

Figure 2. **Feature Importances via Wrapper Approach**. In this figure, we show the feature importances obtained from the RF model with the hyper-parameters specified in table II. The importance is measured in terms of the mean accuracy decrease.

We also implemented random shuffling to assess the significance of our modelling. Specifically, we shuffled the class feature, retrained the model, and evaluated the performance. In all cases, we achieved the accuracy expected of a random classifier, confirming the significance of our results.

Finally, we performed *Feature Selection* to identify the most relevant features for classification. We utilized both *embedded* and *wrapper* methods, limiting our analysis to the RF model for simplicity. The results are depicted in figs. 1 and 2. While there are some discrepancies between the two methods, they generally concur on the most crucial features.

For the embedded approach, we included all features contributing more than 1% to the mean decrease in impurity. For the wrapper approach, we considered

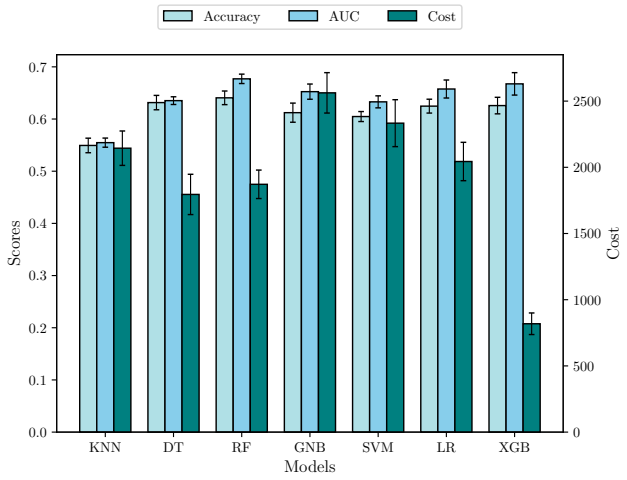

Figure 3. **Models Comparisons**. In this figure, we show the performance and efficiency metrics for the different ML models considered in this study. The metrics are the Accuracy (A), the Cost (C), the Training Time (t_t), and the Prediction Time (t_p).

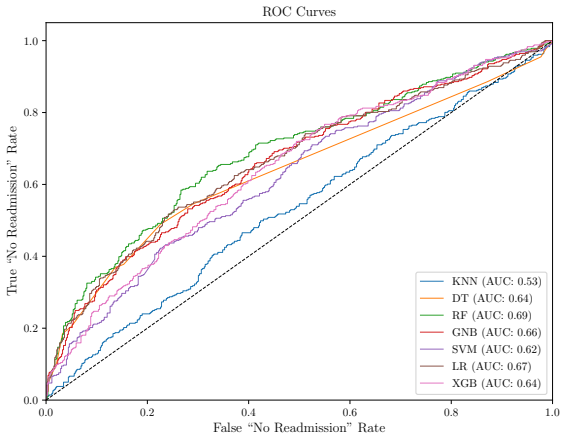

Figure 4. **ROC Curves**.

those with a contribution greater than 10^{-3}. We then identified the intersection of non-important features from both methods and decided to remove the following: `chlorpropamide`, `citoglipton`, `glimepiride`, `glipizide-metformin`, `glyburide-metformin`, `pioglitazone`, `race_Other`, and `tolazamide`. Notably, this list of non-important features includes those hypothesized to be almost redundant. On the other hand, we considered the intersection of the most important features and obtained: `number_inpatient`, `num_lab_procedures`, `number_diagnoses`, `num_medications`, `number_emergency`, `number_outpatient`, `diag_1`, `diag_2`, `diag_3`, `admission_source_id`, `admission_type_id`, `age`, `discharge_disposition_id`, `glyburide`, `insulin`, and `num_procedures`.

III. RESULTS & DISCUSSION

In this section, we present the main results of this study. We begin by displaying the performance and efficiency metrics for the different ML models considered in tab. IV and fig. 3. Subsequently, as commonly practiced in medical contexts, we plot the ROC curves for the different models in fig. 4.

Regarding the model's performance, from tab. IV and fig. 3, it is observed that all models, except for KNN which has an accuracy of 0.55, perform better than a random classifier. Specifically, the RF model outshines others in terms of accuracy and AUC. However, when considering the cost of misclassification, which is particularly relevant in the medical context, the XGB model emerges as the best overall.

If we consider the efficiency metrics, we observe that the SVM is the least efficient method in terms of training time, while the KNN is the most efficient. On the other hand, regarding classification time, the LR proves to be the most efficient and the RF the least. Nonetheless, it is worth mentioning that for medical purposes, classification time is most critical, and all methods are quite efficient. Apart from the RF, all of them require less than a second for classification.

There is a final important aspect to consider: the *interpretability* and *actionability* of the results. From this perspective, the DT, GNB, SVM, and LR are deemed the most valuable models. DT, especially with small-sized trees like those considered in this study, are straightforward to interpret as one can visually represent and analyze their decision paths. GNB, on the other hand, provides a probabilistic output for each classification, which is particularly useful in the medical context for assessing risk levels. SVM and LR also offer high interpretability due to their clear decision boundaries that can be easily understood and acted upon.

In contrast, models like KNN, RF, and XGB are less interpretable, tending to operate as "black-box" models where the decision-making process is not as transparent. From the standpoint of interpretability and accuracy, the DT model stands out as the best option, offering direct insights into the criteria affecting its decisions.

IV. CONCLUSIONS

In this study, we have reanalyzed the work of [1] on the impact of HbA1c measurement on hospital readmission rates. We utilized a subset of the original dataset and applied various machine learning (ML) models to predict the likelihood of patient readmission within the next 30 days post-discharge. Following an initial data exploration phase, we cleaned and prepared the data for ML

	KNN	DT	RF	GNB	SVM	LR	XGB
A	0.55 ± 0.01	0.63 ± 0.01	0.64 ± 0.01	0.61 ± 0.02	0.60 ± 0.01	0.62 ± 0.01	0.62 ± 0.03
C	2144 ± 130	1795 ± 152	1872 ± 108	2562 ± 153	2333 ± 177	2044 ± 145	829 ± 116
t_t (s)	0.0013 ± 0.0002	0.0082 ± 0.0003	0.18 ± 0.01	0.003 ± 0.001	5.4 ± 0.3	0.02 ± 0.01	0.10 ± 0.04
$t_p \cdot 10^2$ (s)	0.10 ± 0.02	0.026 ± 0.002	1.5 ± 0.1	0.026 ± 0.004	0.081 ± 0.007	0.017 ± 0.004	0.062 ± 0.006

Table IV. **Models Comparisons**. In this table, we show the performance and efficiency metrics for the different ML models considered in this study. The metrics are the Accuracy (A), the Cost (C), the Training Time (t_t), and the Prediction Time (t_p).

modeling. We assessed a range of ML models in terms of their performance, efficiency, and interpretability, and also conducted feature selection to identify the most relevant features for the classification task.

Our findings indicate that while the KNN model performs comparably to a random classifier, the XGB model excels in terms of accuracy, AUC, and cost-effectiveness. However, when prioritizing interpretability and actionability, the Decision Tree (DT) emerges as the preferable model. This study not only enhances our understanding of the predictive factors influencing hospital readmissions but also underscores the importance of selecting appropriate ML models based on the specific requirements of the medical context.

[1] B. Strack, J. P. DeShazo, C. Gennings, J. L. Olmo, S. Ventura, K. J. Cios, and J. N. Clore, Impact of HbA1c measurement on hospital readmission rates: Analysis of 70,000 clinical database patient records, BioMed Research International **2014**, 781670 (2014).

[2] Data Preparation of Diabetes Dataset, https://yungchou.github.io/site/.

[3] S. Raj, Diabetes 130 US hospitals for years 1999–2008 (Hospital Readmission) (2020).

[4] S. Ranveer, Diabetes 130 US hospitals for years 1999–2008. (2021).

[5] J. Neff, Jonneff/Diabetes2 (2023).

[6] K. C. John Clore, Diabetes 130-US Hospitals for Years 1999-2008 (2014).

Supplemental Material of the manuscript: Enhancing Predictive Models for Hospital Readmission: A Machine Learning Approach to Analyzing HbA1c Impact

V. ID MAPPINGS

Admission type ID	Original meaning	Classification
1	Emergency	Urgency
2	Urgency	Urgency
3	Elective	Elective
4	Newborn	Newborn
5	Not Available	Unknown
6	NULL	Unknown
7	Trauma Center	Urgency
8	Not mapped	Unknown

Table V. Admission type ID mappings. See [6] for further details on the specific definition of the features.

Admission source ID	Original meaning	Classification
1	Physician Referral	Healthcare
2	Clinic Referral	Healthcare
3	HMO Referral	Healthcare
4	Transfer from a hospital	Healthcare
5	Transfer from a SNF	Healthcare
6	Transfer from another health care facility	Healthcare
7	Emergency room	Emergency
9	Not Available	Unknown
17	NULL	Unknown
20	Not Mapped	Unknown

Table VI. Admission source ID mappings. See [6] for further details on the specific definition of the features.

Discharge disposition ID	Original meaning	Classification
1	Dc. to home	Home
2	Dc./trf. to another short term hospital	Hospital
3	Dc./trf. to SNF	Hospital
4	Dc./trf. to ICF	Hospital
5	Dc./trf. to another type of inpatient care	Hospital
6	Dc./trf. to home with home health service	Hospital
7	Left AMA (Against medical advice)	Hospital
8	Dc./trf. to home under care of Home IV provider	Home
9	Admitted as an inpatient	Hospital
10	Neonate dc. to another hospital	Hospital
13	Hospice/home	Hospice
14	Hospice/medical facility	Hospice
15	Dc./trf. within this institution	Hospital
16	Dc./tf./rf. another institution	Hospital
17	Dc./tf./rf. to this institution	Hospital
18	NULL	Unknown
22	Dc./tf. to another rehab	Hospital
23	Dc./tf. to a long term care hospital	Hospital
24	Dc/tf. to a nursing	Hospital
25	Not mapped	Unknown
26	Unknown/invalid	Unknown
30	Dc./tf. to another type of health	Hospital
27	Dc./tf. to a federal health	Hospital
28	Dc./tf. to a critical	Hospital

Table VII. Discharge disposition ID mappings. Dc. stands for Discharged. Trf. stands for Transferred. Rf. stands for Referred. See [6] for further details on the specific definition of the features.

Diagnosis ID	Category
$390 - 459, 785$	Circulatory
250.xx	Diabetes
$460 - 519, 786$	Respiratory
$520 - 579, 787$	Digestive
$800 - 999$	Injury
$710 - 739$	Musculoskeletal
$580 - 629, 788$	Genitourinary

Table VIII. Diagnosis ID mappings. See [6] for further details on the specific definition of the features.