Monotonicity Detection and Enforcement in Longitudinal Classification

Sergey Ovchinnik, Fernando E. B. Otero, and Alex A. Freitas

School of Computing, University of Kent, Canterbury, UK
{S.Ovchinnik,F.E.B.Otero,A.A.Freitas}@kent.ac.uk

Abstract. Longitudinal datasets contain repeated measurements of the same variables at different points in time, which can be used by researchers to discover useful knowledge based on the changes of the data over time. Monotonic relations often occur in real-world data and need to be preserved in data mining models in order for the models to be acceptable by users. We propose a new methodology for detecting monotonic relations in longitudinal datasets and applying them in longitudinal classification model construction. Two different approaches were used to detect monotonic relations and include them into the classification task. The proposed approaches are evaluated using data from the English Longitudinal Study of Ageing (ELSA) with 10 different age-related diseases used as class variables to be predicted. A gradient boosting algorithm (XGBoost) is used for constructing classification models in two scenarios: enforcing and not enforcing the constraints. The results show that enforcement of monotonicity constraints can consistently improve the predictive accuracy of the constructed models. The produced models are fully monotonic according to the monotonicity constraints, which can have a positive impact on model acceptance in real world applications.

1 Introduction

Longitudinal data mining is a branch of data mining that is concerned with longitudinal datasets, where repeated measurements of the same variables are taken at different points in time. Such datasets can provide deeper insight into the nature of the data being explored and thus can be used to construct predictive models that not only take into account the individual attribute values, but also the changes that occurred in those values over time and general time-based trends. Longitudinal data mining is becoming increasingly important in the context of human ageing research (the target domain in this work) with more and more datasets becoming available [10, 17].

Monotonic relations are relations between attributes in data that occur when the value of one attribute always changes in the same direction (increasing or decreasing) as the value of another does (or stays the same). Monotonic relations often represent natural dependencies or correlations that occur in data, such as a relation between a patient's blood cholesterol level and probability of chronic heart disease.

Monotonicity constraints are the most common domain constraints used in model construction and can have a significant effect on both the comprehensibility and the acceptability of a model [11]. While it is possible for monotonic constraints to improve the predictive accuracy of the model by improving generalization [3, 8], some practical experiments have shown the opposite effect [2].

There are currently many examples in the literature of longitudinal data mining methodologies [16] and many examples of use of monotonic relations in data mining [4]. There is, however, no example of the two approaches being used together in a single study. In this paper we propose a new methodology that automatically detects monotonic relations in longitudinal data and uses these in the construction of a longitudinal classification model. The proposed methodology utilises the longitudinal nature of the data by detecting monotonic relations that occur within individuals across different time points, which is a type of monotonic relation unexplored in the literature.

The dataset used in this study was created from data of the English Longitudinal Study of Ageing (ELSA) [7]. The attributes used were mainly from the "Nurse Visit" section of ELSA, representing various health-related measurements across four time points (referred to as "Waves" in ELSA). The class attributes used in this study were binary attributes representing whether an individual had a particular age-related disease. Ten age-related diseases were used as class attributes to be predicted, and were addressed as ten separate classification problems. A series of experiments was run using each of the classification problems to determine the effects of adding the proposed monotonicity detection approaches on the predictive accuracies of the constructed models.

The remainder of this paper is organized as follows. Section 2 presents the background on longitudinal classification, monotonic classification, and the XGBoost classification algorithm used in this work. Section 3 describes the experimental methodology and dataset creation. Section 4 reports the results obtained by using the proposed methods to detect monotonicity constraints that are exploited by the XGBoost algorithm, a state-of-the-art classification algorithm. Finally, Section 5 presents our conclusions and suggests future work.

2 Background

2.1 Longitudinal Classification

In recent years, a number of approaches for longitudinal classification were proposed. So far, most longitudinal classification studies use conventional non-longitudinal classification algorithms and rely heavily on data pre-processing algorithms that transform the longitudinal datasets to allow conventional algorithms to be used for class prediction [16].

A recent study by Zhang et al. [19] used a conventional non-longitudinal decision tree algorithm to make predictions based on longitudinal data. Their approach used a data pre-processing algorithm that combined the data from two consecutive waves into a single dataset, disregarding the time indexes of

the features. A C5.0 decision tree classifier was built to predict the value of the class attribute in the first of the two waves using predictor attributes from both waves. A similar approach was used earlier by Mo et al. [12] where data from two waves was merged and the class attribute from the first wave was predicted using the predictor attributes of the second wave. The study used five non-longitudinal classification algorithms to build prediction models and evaluated their performance.

Niemann et al. [13] proposed an approach for longitudinal classification that used a pre-processing step to cluster all data instances based on their attribute values and generated new predictor attributes based on cluster data. After the clustering is completed and cluster data is added to every instance in each wave, the dataset is transformed by combining all waves into a single dataset, similarly to the two previous studies, omitting the time indexes. A number of non-longitudinal classification algorithms are used to construct the classification models and evaluate their performance.

Overall, the longitudinal classification literature is dominated by studies using data pre-processing methods to transform the longitudinal data to allow the use of conventional non-longitudinal classification algorithms. It is interesting to note that there is no example in the literature of a longitudinal classification approach enforcing monotonicity constraints.

2.2 Monotonic Classification

In recent years, many new monotonic classification algorithms have been developed, using various approaches to monotonicity enforcement [4].

Some applications use data pre-processing approaches to make the training data fully monotonic. This is done by relabeling the input dataset to enforce full monotonicity according to pre-defined monotonicity constraints. A set of algorithms for monotonic dataset relabeling was described by Pijls and Potharst [14]. The *Naive Relabel* is a greedy algorithm for relabeling datasets that produces fully monotonic outputs, but does not guarantee an optimal solution to the relabeling problem. The *Borders* algorithm is an extension of Naive Relabel that minimises the differences between the new and original labels and uses a simpler approach for selecting the new label values. The *Antichain* algorithm is a further extension to the Borders algorithm that minimises the total number of relabelings by constructing a monotonicity violation graph and finding the maximum antichain in it. The Antichain approach produces a fully monotonic dataset with a minimal number of relabelings.

Duivesteijn and Fielders [8] proposed an algorithm for monotonic classification based on the k-nearest neighbors (kNN) classification algorithm. The proposed Monotone kNN (MkNN) algorithm uses a procedure for re-labeling the instances in the training set to ensure full monotonicity before the kNN model is created. The algorithm used a monotonicity violation graph (MVG) as a representation of all instances within the training set and the ordinal relations between them. The graph is then used to determine the optimal set of instances to be re-labeled and the relabeling step is done. The authors argue that since

the kNN algorithm uses the training data as classification model, monotonising the training data at the pre-processing step is sufficient to produce a fully monotonic classification algorithm. Overall, monotonicity enforcement has improved the predictive accuracy of the kNN classifiers and a positive effect on model acceptability was suggested.

Some applications use model post-processing approaches to alter the model trained on non-monotonic data to make the model fully monotonic. This is done by altering the model after the training is completed to enforce full monotonicity in the output of the model. Verbeke at al. [18] proposed a post-processing algorithm to create rule-based and tree-based models. The proposed RUle LEarning of ordinal classification with Monotonicity constraints (RULEM) algorithm can guarantee full monotonicity of the constructed model by applying post-processing changes to any model constructed by a decision tree or rule induction classification algorithm. It evaluates the monotonicity of a rule set by generating a decision grid based on the rules and evaluating the Conflict score (C-score) of each cell on the grid and detects monotonicity violations. It then adds complementary rules to the rule set to resolve the monotonicity violations. The results of the experiments indicate that the proposed approach preserves the predictive power of the original rule induction techniques while guaranteeing monotone classification, at the cost of a small increase in the size of the rule set.

There is a small number of approaches that incorporate monotonic constraints on the model construction stage, thus using the monotonic constraints in training and producing fully monotonic models based on non-monotonic data and some pre-defined constraints. Zhu et al. [20] proposed a neural network-based monotonic classification algorithm, called Monotonic Classification Extreme Learning Machine (MCELM). The proposed algorithm is an extension of Extreme Learning Machine (ELM) [9] — a single hidden layer feed-forward neural network learning algorithm. The proposed algorithm approaches the classification model construction task as a quadratic programming problem, where the training error rate is used as the objective for optimisation. The algorithm can ensure that the model produced by ELM is fully monotonic and does not require training data to be fully monotonic. A similar approach was taken by Chen and Li [5] using a Support Vector Machine (SVM)-based classification algorithm. In this study, a Monotonicity Constrained Support Vector Machine (MC-SVM) classification algorithm was proposed for monotonic classification of financial credit rating data. It uses a quadratic programming approach in a similar fashion to MCELM, and approaches the classifier construction as an optimisation problem with pre-defined monotonicity constraints being added as conditions to the problem. This approach produces fully monotonic models but has the drawback of requiring a fully monotonic set of training data.

2.3 Monotonicity Measures

There are two main monotonicity measures that can be used for estimating monotonicity in datasets and classification models: Non-Monotonicity Index (NMI) and Spearman's rank correlation coefficient.

The Non-Monotonicity Index (NMI) is a monotonicity measure originally introduced for determining the degree of monotonicity that a decision tree model exhibits in relation to a pre-defined set of monotonic constraints [1]. NMI is a measure of how "non-monotonic" a given set of examples is in relation to a certain monotonic constraint (or a set of constraints). In the original paper, the measure was used to compare pair-wise all paths from the root to the leaves of a decision tree and calculate the proportion of such pairs that violated a pre-defined set of constraints. NMI is defined as a measure that ranges between 0 and 1, where low values (around 0) represent strong monotonic relations, middle values (around 0.5) represent no monotonic relation, and high values (around 1) represent an inverted monotonic relation that is the opposite of the one being evaluated.

The monotonic relation between two attribute value pairs (x_i, y_i) and (x_j, y_j) is evaluated using the following function, which takes the value 1 if the two attribute pairs contradict the monotonic relation and 0 otherwise:

$$non\_monotonic((x_i, y_i), (x_j, y_j)) \tag{1}$$

For example, if the positive monotonic relation is estimated, then the two value pairs $(1, 6)$ and $(4, 8)$ would have *non_monotonic* value of 0, while two value pairs $(1, 6)$ and $(4, 2)$ would have *non_monotonic* value of 1. NMI can then be estimated using the following formula:

$$NMI = \frac{\sum_{i=1}^{k} \sum_{j=1}^{k} non\_monotonic((x_i, y_i), (x_j, y_j))}{k \times (k-1)} \tag{2}$$

where k is the number of instances, i $\neq$ j.

The Spearman's rank correlation coefficient is a mathematical measure that estimates how well the relationship between two variables can be described using a monotonic function. It assesses the degree of the correlation between two variables by first converting each variable to a ranked variable. It then applies the Pearson's correlation coefficient to the two ranked variables to estimate the degree of linear correlation between them.

The Pearson's correlation coefficient is estimated using the covariance of two attributes X and Y divided by the product of their standard deviations (σ_X and σ_Y).

$$Pearson_{X,Y} = \frac{cov(X, Y)}{\sigma_X \times \sigma_Y} \tag{3}$$

While the NMI monotonicity measure was used in a large portion of monotonic classification studies, Spearman is currently under-explored by the monotonic classification literature despite being the main mathematical measure for estimating the monotonic relation between two variables.

2.4 The XGBoost Tree Boosting Algorithm

XGBoost is a highly accurate, scalable tree boosting system that uses the gradient boosting approach with tree-based classifiers serving as weak learners [6].

XGBoost has become a popular approach for data mining in recent years with many studies conducted using it as the main machine learning tool.

XGBoost supports monotonicity constraint enforcement during model construction, which allows users to add monotonic constraints to a set of predictor attributes to produce models that follow those constraints. The constraints are enforced during the model construction using a simple yet effective method of restricting the output values of the branches following a split on a constrained attribute. Using this approach, XGBoost can guarantee that after the value on an attribute is checked against the threshold, the outputs of all leaf nodes under the left branch will always be lower than outputs of all leaf nodes under the right branch, thus ensuring full monotonicity of the model. XGBoost always produces a fully monotonic model without requiring any data pre-processing steps to be taken, making it applicable to large non-monotonic datasets.

XGBoost has a built-in method for handling missing values in data. During the construction of tree models, each split in the tree has a default direction assigned for the instances where the attribute being split has no value. This allows the XGBoost algorithm to effectively classify instances with missing values by simply making default decisions where a value test can not be performed. This property is especially important when working with ELSA data, since it contains a large amount of missing attribute values.

XGBoost is highly customisable and allows creation of boosted models with any number of tree classifiers, and the maximum size of tree classifiers can be adjusted. XGBoost can be used for both regression and classification problems with minor parameter adjustments. Although XGBoost uses a combination of weak classification models and produces a non-interpretable ensemble model, it has a built-in measure of attribute importance which can be accessed directly form the model. This provides an insight on which attributes are involved in the predictions and scores them based on their impact.

3 Methodology

3.1 Quantitative Measures For Monotonicity Detection

Two quantitative measures are used in the proposed approach to evaluate the strength of monotonic relations between attributes: the Non-Monotonicity Index (NMI) and the Spearman's correlation coefficient.

Despite being originally developed as a measure for evaluating the degree of monotonicity of predictive models, NMI has also proven useful in detecting monotonic relations in data. In data, a similar approach can be taken by comparing the data instances pair-wise and calculating the proportion of those that violate an assumed monotonicity constraint. In the task of full monotonicity detection, the detection algorithm can be used to iterate through possible monotonic relations in the data and then estimate the value of NMI to asses if those relations exist.

Since in this study we are only concerned with monotonic relations between pairs of attributes where one attribute is the class, a full monotonicity scan only

requires the monotonicity detection algorithm to iterate through all pairs between predictor attributes and the class in a dataset and estimate the strength of their monotonic relation using NMI. Thus, a dataset with n predictor attributes would only require n NMI estimations.

NMI serves as a good monotonicity detection measure in many cases, but it has a flaw that causes it to become unusable in some cases. NMI always has to assume that a pair of examples is monotonic if the values of at least one attribute in a pair are equal. For example, the attribute pairs $(1, 6), (1, 3)$ and $(1, 12)$ would all be considered monotonic with each other since the value of the first attribute is the same. This causes no trouble on numeric attributes with barely any repeating values, but can have a huge impact on attributes with small domain space (e.g. binary attributes) and attributes that have many repeated values. Since NMI has to make an assumption that two examples with repeated attribute value are monotonic, it can not accurately estimate monotonicity when one or both attributes are largely dominated by a frequent repeated value. For example, a binary attribute that has 95% positive values and 5% negative values will always have very low NMI even when it is estimated against random noise.

As a solution for this issue, we propose an Entropy-Adjusted Non-Monotonicity Index (EANMI). It uses an entropy-based coefficient to adjust the monotonicity measure and increase the degree of non-monotonicity for the attributes that are largely dominated by a single value (and thus have low entropy). The adjustment uses the entropy of an attribute and the maximum entropy of the attribute. The maximum entropy is estimated as the maximum entropy an attribute can have, provided it has the same domain of values. For each pair of a predictor attribute x and the class attribute y, the adjustment is made only for x (considering that y is fixed as the second attribute in all attribute pairs used for monotonicity detection).

$$Entropy(X) = -\sum_{i=1}^{n} P(x_i) \times log_2 P(x_i) \tag{4}$$

$$Max\_Entropy(X) = log_2(n) \tag{5}$$

$$EANMI_{X,Y} = 1 - (1 - NMI_{X,Y}) \times \frac{Entropy(X)}{Max\_Entropy(X)} \tag{6}$$

where n is the number of unique values of x, x_i is the ith unique value of X, $P(x_i)$ is the relative frequency (empirical probability) of value x_i.

The Spearman's correlation coefficient can also be used as a monotonicity estimation measure using the same approach of estimating the strength of monotonic relation between two attributes in a dataset. Since it only estimates the strength of the linear correlation between ranked values of the attributes, no additional adjustment is required.

3.2 The Proposed Longitudinal Monotonicity Detection Approaches

Two longitudinal monotonicity techniques are used in this project:

- **Timeless** monotonicity detection is a method for detecting general trends occurring between attributes in the data. This approach ignores the longitudinal aspect of the dataset and combines all data entries from different waves into a single non-longitudinal dataset. It can then detect monotonic relations between attributes in the resulting dataset. This approach can detect strong correlations in data that occur regardless of time and individual effects. It is intended to be used iterating through all attribute pairs to provide a full insight into monotonic relations that occur in the dataset, although in the context of classification, only the attribute pairs containing the class attribute are of interest. A natural language example of such relation can be: "The higher a patient's blood sugar levels are, the higher their likelihood of diabetes is".
- **Time-Index Based Individual-wise (TIBI)** monotonicity detection is a method for detecting the time-based attribute trends that occur in measurements taken for a certain individual. This approach takes the different values of an attribute across time for each individual and estimates how those values have changed over time. It can provide an insight into time-based trends a certain individual may have and the strength measures of the monotonic relations can be used as additional predictive attributes. In a monotonicity detection task, it is intended to be used iterating through all individuals and all longitudinal attributes to cover all possible relations of this type. A natural language example of such relation can be: "Individual 12345 has their blood sugar levels steadily increasing over time". Note that this method ignores the class labels of examples and requires the data to be present for at least three waves in order to be significant.

3.3 Datasets

The dataset used in these experiments was constructed using biomedical data from ELSA [7]. Predictive attributes were created from the "Nurse Visit" data portion of the ELSA database. Most attributes use raw values of ELSA variables representing various patient health measurements such as blood test results and physical performance tests. There are 10 class attributes, each containing binary values representing either the presence or absence of a certain age-related disease in the final wave (Wave 8 in these experiments). A separate classification problem was constructed for each class attribute. Since most of these attributes in ELSA are only recorded on even numbered waves, only the data from waves 2, 4, 6 and 8 was used. A full description of attributes used and their meaning can be found in a related study that previously used the same data preparation techniques in the context of automatic feature selection [15].

The 10 classification problems used for experiments each contained records for 7097 individuals participating in the ELSA study. Each instance contains 143 attribute values in total (considering different values of an attribute across all waves) and a single class label. The 10 classification problems are presented in Table 1 along with the proportion of positive class values (presence of the disease) in each problem.

Table 1. Frequencies of positive class values

Class Attribute	Number of positive instances	Frequency
Angina	258	3.64%
Arthritis	3021	42.57%
Cataract	2322	32.72%
Dementia	148	2.09%
Diabetes	946	13.33%
High Blood Pressure	2854	40.21%
Heart attack	401	5.65%
Osteoporosis	654	9.22%
Parkinson's	66	0.93%
Stroke	421	5.93%

3.4 Experimental Setup

For each of the classification problems described in the previous section, a separate set of experiments was conducted, thus producing separate results for each class attribute used. Each set of experiments consisted of four separate experiments:

- One was performed on the baseline dataset without any changes (Baseline experiment).
- One experiment which included detection of Timeless monotonic relations using Entropy-Adjusted NMI (EANMI). Only the relations between the class attribute and all other attributes were used. The entropy adjustment was only made for each predictor (non-class) attribute. Only strong monotonicity constraints (with EANMI < 0.15, a threshold determined in preliminary experiments) were used as monotonicity constraints for the XGBoost algorithm.
- One experiment was performed with an addition of TIBI attributes representing monotonic trends in values of longitudinal attributes between waves. The TIBI attributes were detected using the Spearman's correlation coefficient.
- The final experiment introduced both the TIBI attributes and Timeless monotonic constraints. It is important to note that the Timeless constraints were introduced after the TIBI attributes were added, thus allowing those attributes to be considered for constraints as well.

During each experiment, a well-known 10-fold cross-validation approach was used to estimate the average accuracy of the XGBoost model built using the created dataset and monotonicity constraints if any (depending on the experiment). In order to get a reliable estimation of average predictive accuracy of the models, the cross validation process was repeated 30 times (using different random seeds for cross validation) and the average predictive accuracy was measured. The predictive accuracy measure used in model training and evaluation was area under ROC curve. In all experiments, the following parameters were

Table 2. $\overline{\text{AUROC}}$ values for each set of experiments. The highest $\overline{\text{AUROC}}$ value for each dataset is shown in bold.

Class Attribute	Baseline	Timeless	TIBI	TIBI+Timeless
Angina	0.545	**0.682**	0.660	0.668
Arthritis	0.610	0.610	0.610	**0.611**
Cataract	**0.655**	**0.655**	**0.655**	**0.655**
Dementia	0.755	**0.767**	0.750	0.758
Diabetes	0.825	**0.829**	0.824	**0.829**
High Blood Pressure	**0.704**	**0.704**	**0.704**	**0.704**
Heart attack	0.679	0.696	0.688	**0.698**
Osteoporosis	0.666	**0.677**	0.664	0.670
Parkinson's	0.575	**0.621**	0.578	0.606
Stroke	0.666	**0.681**	0.673	0.680
Average Rank	3.3	1.65	3.2	1.85

Table 3. Results of the Friedman test with post-hoc Holm test. Statistically significant results at the 5% significance level are shown in bold, indicating that the result of a particular approach is statistically significaantly worse than the control approach.

Approach	Average Rank	p-value	Holm
Timeless (Control)	1.65	-	-
Timeless+TIBI	1.85	0.729	0.05
TIBI	**3.2**	**0.007**	**0.025**
Baseline	**3.3**	**0.004**	**0.017**

used for XGBoost: num_round=10, max_depth=10, objective="binary:logistic", eval_metric="auc", eta=1.0. The parameter values were selected based on parameter descriptions in the XGBoost documentation; no attempt was made to optimise the parameters for this specific task.

4 Results of the Experiments and Comparative Analysis

4.1 Predictive Accuracies of Constructed Models

The results for the four sets of experiments over the ten classification problems were collected. Table 2 shows the average area under ROC curve ($\overline{\text{AUROC}}$) values for each set of experiments. In addition, the average rank value for each monotonicity type combination is included.

From Table 2 we can see that the introduction of monotonicity constraints led to an improvement of the AUROC or had no substantial negative effect across all classification problems. Both approaches enforcing monotonicity constraints (Timeless and TIBI+Timeless) achieved the best rankings. The use of monotonicity-based TIBI attributes alone had no significant effects.

Further statistical analysis was performed on the $\overline{\text{AUROC}}$ rankings, where Table 3 shows the results of the non-parametric Friedman test with the Holm

Table 4. Average number [standard deviation] of Timeless monotonicity constraints used by XGBoost per classification problem

Class Attribute	Number of constraints
Angina	65.0 [0.50]
Arthritis	0.003 [0.005]
Cataract	0.0 [0.0]
Dementia	73.06 [0.50]
Diabetes	20.12 [0.39]
High Blood Pressure	0.0 [0.0]
Heart attack	61.89 [0.50]
Osteoporosis	39.42 [0.46]
Parkinson's	80.87 [0.49]
Stroke	61.61 [0.50]

post-hoc test — statistically significant differences at the 5% significance level are shown in bold. Both TIBI and Baseline approaches are statistically significantly worse than Timeless, the combination with the best ranking (control).

4.2 Monotonicity Constraints and their Effects on Model Sizes

Monotonic relations were detected separately in each training set prior to the experiments and then used as monotonicity constraints during model construction. The results have shown that no monotonic relations were detected between TIBI attributes and the class in any of the TIBI experiments. Table 4 shows the average number of monotonicity constraints used by XGBoost for each of the classification problems as well as the corresponding standard deviation (over all training sets produced by cross-validation iterations with different random seeds) of each measurement.

It can bee seen from the Table 4 that some classification problems used a large number of monotonicity constraints while others used none at all. Standard deviations remain very low for each measurement, meaning that there was only a small variation in number of detected constraints between different training sets.

The number of constraints can be correlated to the percentage of positive class labels in each dataset as shown in Table 1. Generally, the classification problems that used class attributes with lower number of positive class labels had a larger number of monotonic constraints. The reason for this trend may be related to the adjustment used by EANMI, which took into account the entropies of predictor attributes but not the entropy of the class. Therefore, class attributes with lower entropy generally had a high chance of having a strong monotonic relation ($EANMI < 0.15$) and thus had more monotonic constraints.

The average model sizes were measured for each of the experiments to determine the effect of introduction of monotonic approaches to the classification task. Overall, the introduction of both Timeless monotonicity constraints and

TIBI attributes resulted in a small decrease in model size across all experiments. The decrease was consistent across all classification problems, yet not very significant: Introduction of Timeless constraints decreased the model size on average by 13% and introduction of TIBI attributes resulted in a 2% decrease in average model sizes. In the experiments where both approaches were used, the model size effects were similar to that of just using Timeless constraints.

4.3 Feature Importance

The average feature importance was estimated for each feature in each experiment using the built-in feature importance measure of XGBoost (based on average predictive accuracy increase) and the most important features were analysed. The results were sensible in most experiments: in Diabetes experiments the most important features were patient blood glucose levels; in HBP experiments the blood pressure features were most important; blood cholesterol levels had high importance in the Angina, Heart Attack and Stroke experiments. In most experiments, the age feature was among the top five most important features and the sex feature generally had a high feature importance in all models.

Interestingly, some of the TIBI features achieved very high feature importance in some of the TIBI experiments. In the Cataract experiments, the TIBI feature for diastolic blood pressure has achieved 4th highest feature importance; the TIBI feature for blood total cholesterol level had high importance in Heart Attack and Stroke TIBI experiments, and in the Heart Attack experiment with TIBI+Timeless approach it was the most important feature.

Some categorical ordinal attributes were also highly important in classification models constructed using training sets with TIBI features. In the Angina experiments, the TIBI feature for the outcome of side-by-side stand test was the feature with the highest average importance; a TIBI feature of a binary feature representing whether the patient had any respiratory infection in the last 3 weeks was the 4th most important feature in Heart Attack TIBI experiments.

Table 5 shows the features that achieved high feature importance in the constructed models across all experiments for a given class. The features in bold were often detected as being monotonic with the class and were used as monotonic constraints in Timeless and TIBI+Timeless experiments.

5 Conclusion

A set of experiments was conducted using the proposed monotonicity detection approaches and their effect on the predictive accuracies of the constructed classification models was evaluated. Enforcement of monotonicity constraints detected using the proposed Entropy Adjusted Non-Monotonicity Index was shown to generally result in a significant improvement of the predictive accuracy of the model without any significant negative effects. The addition of Time-Index Based Individual-wise (TIBI) monotonic attributes was shown to have a very minor effect on the constructed models. TIBI attributes, despite being frequently used in

Table 5. High-importance features for each classification problem. The features that were constrained by Timeless constraints are displayed in bold.

Class Attribute	Common features
Angina	**Age**, **Arterial pressure**, **Cholesterol level**, Chair rises∗, Side-by-side stands∗
Arthritis	Age, Chair raises∗, Leg raises∗, Main hand grip∗, Hip measurement
Cataract	Age, Leg raises with eyes open∗, Diastolic blood pressure
Dementia	**Age**, Clotting disorder, **Blood ferritin level**, **Grip strength**∗
Diabetes	Fasting blood glucose level, **LDL cholesterol level**, **Waist measurement**
High Blood Pressure	Age, Systolic blood pressure, Diastolic blood pressure
Heart attack	**Cholesterol level**, **LDL cholesterol level**, **Arterial pressure**
Osteoporosis	**Age**, **Main hand grip**∗, **Weight**
Parkinsons	Blood fibrinogen level, **White blood cell count**, **Non-dominant hand grip**∗
Stroke	**Age**, **Cholesterol level**, **LDL cholesterol level**,

∗ Outcomes of physical exercise tests

some classification models, were not detected as being monotonic with the class attributes and thus have not been used as monotonicity constrained attributes in any of the models.

Overall, our approach used the longitudinal dataset to make class predictions and effectively detected and enforced monotonicity constraints. The proposed approach for automatic monotonicity detection in datasets and enforcement in classification models worked well in this context, but further studies are needed to determine the optimal monotonicity detection approach and to define methodologies for longitudinal monotonicity detection more appropriate for general use. Additionally, it would be interesting to evaluate the use of the entropy adjustment on the class attribute to cope with class imbalance.

References

1. Ben-David, A.: Monotonicity maintenance in information-theoretic machine learning algorithms. Machine Learning **19**(1), 29–43 (1995). https://doi.org/10.1023/a:1022655006810
2. Ben-David, A., Sterling, L., Tran, T.: Adding monotonicity to learning algorithms may impair their accuracy. Expert Systems with Applications **36**(3), 6627–6634 (Apr 2009). https://doi.org/10.1016/j.eswa.2008.08.021
3. Brookhouse, J., Otero, F.E.B.: Monotonicity in ant colony classification algorithms. In: Lecture Notes in Computer Science, pp. 137–148. Springer (2016)
4. Cano, J.R., Gutiérrez, P.A., Krawczyk, B., Woźniak, M., García, S.: Monotonic classification: An overview on algorithms, performance

measures and data sets. Neurocomputing **341**, 168–182 (May 2019). https://doi.org/10.1016/j.neucom.2019.02.024
5. Chen, C.C., Li, S.T.: Credit rating with a monotonicity-constrained support vector machine model. Expert Systems with Applications **41**(16), 7235–7247 (Nov 2014)
6. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD 2016. pp. 785–794. ACM Press (2016). https://doi.org/10.1145/2939672.2939785
7. Clemens, S., Phelps, A., Oldfield, Z., Blake, M., Oskala, A., Marmot, M., Rogers, N., Banks, J., Steptoe, A., Nazroo, J.: English longitudinal study of ageing: Waves 0-8, 1998-2017 (2019). https://doi.org/10.5255/ukda-sn-5050-16
8. Duivesteijn, W., Feelders, A.: Nearest neighbour classification with monotonicity constraints. In: Machine Learning and Knowledge Discovery in Databases, pp. 301–316. Springer Berlin Heidelberg (Sep 2008)
9. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: Theory and applications. Neurocomputing **70**(1-3), 489–501 (Dec 2006). https://doi.org/10.1016/j.neucom.2005.12.126
10. Kaiser, A.: A review of longitudinal datasets on ageing. Journal of Population Ageing **6**(1-2), 5–27 (Jun 2013). https://doi.org/10.1007/s12062-013-9082-3
11. Martens, D., Baesens, B.: Building acceptable classification models. In: Annals of Information Systems, pp. 53–74. Springer (Oct 2009)
12. Mo, J., Siddiqui, S., Maudsley, S., Cheung, H., Martin, B., Johnson, C.A.: Classification of Alzheimer Diagnosis from ADNI Plasma Biomarker Data. In: Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics - BCB 2013. ACM Press (2013)
13. Niemann, U., Hielscher, T., Spiliopoulou, M., Volzke, H., Kuhn, J.P.: Can we classify the participants of a longitudinal epidemiological study from their previous evolution? In: 2015 IEEE 28th International Symposium on Computer-Based Medical Systems. IEEE (Jun 2015). https://doi.org/10.1109/cbms.2015.12
14. Pijls, W., Potharst, R.: Repairing non-monotone ordinal data sets by changing class labels. Tech. rep., Erasmus University Rotterdam (Dec 2014)
15. Pomsuwan, T., Freitas, A.A.: Feature selection for the classification of longitudinal human ageing data. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW). IEEE (Nov 2017). https://doi.org/10.1109/icdmw.2017.102
16. Ribeiro, C., Freitas, A.A.: A mini-survey of supervised machine learning approaches for coping with ageing-related longitudinal datasets. In: 3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL), held as part of IJCAI-2019 (2019)
17. Ribeiro, C.E., Brito, L.H.S., Nobre, C.N., Freitas, A.A., Zárate, L.E.: A revision and analysis of the comprehensiveness of the main longitudinal studies of human aging for data mining research. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **7**(3), e1202 (mar 2017). https://doi.org/10.1002/widm.1202
18. Verbeke, W., Martens, D., Baesens, B.: RULEM: A novel heuristic rule learning approach for ordinal classification with monotonicity constraints. Applied Soft Computing **60**, 858–873 (Nov 2017). https://doi.org/10.1016/j.asoc.2017.01.042
19. Zhang, Y., Jia, H., Li, A., Liu, J., Li, H.: Study on prediction of activities of daily living of the aged people based on longitudinal data. Procedia Computer Science **91**, 470–477 (2016). https://doi.org/10.1016/j.procs.2016.07.122
20. Zhu, H., Tsang, E.C., Wang, X.Z., Ashfaq, R.A.R.: Monotonic classification extreme learning machine. Neurocomputing **225**, 205–213 (Feb 2017). https://doi.org/10.1016/j.neucom.2016.11.021