







# Predicting Physical Properties of Woven Fabrics via Automated Machine Learning and Textile Design and Finishing Features

Rui Ribeiro<sup>1,4</sup> , André Pilastrí<sup>1</sup> , Carla Moura<sup>2</sup>, Filipe Rodrigues<sup>3</sup>, Rita Rocha<sup>3</sup>, José Morgado<sup>3</sup>, and Paulo Cortez<sup>4</sup>  

<sup>1</sup> EPMQ - IT Engineering Maturity and Quality Lab, CCG ZGDV Institute, Guimarães, Portugal

{[rui.ribeiro](mailto:rui.ribeiro@ccg.pt), [andre.pilastri](mailto:andre.pilastri@ccg.pt)}@ccg.pt

<sup>2</sup> Riopele, Pousada de Saramagos, Portugal

<sup>3</sup> CITEVE - Centro Tecnológico das Indústrias Têxtil e do Vestuário de Portugal, Famalicão, Portugal

<sup>4</sup> ALGORITMI Centre, Department of Information Systems, University of Minho, Guimarães, Portugal  
[pcortez@dsi.uminho.pt](mailto:pcortez@dsi.uminho.pt)

**Abstract.** This paper presents a novel Machine Learning (ML) approach to support the creation of woven fabrics. Using data from a textile company, two CRoss-Industry Standard Process for Data Mining (CRISP-DM) iterations were executed, aiming to compare three input feature representation strategies related with fabric design and finishing processes. During the modeling stage of CRISP-DM, an Automated ML (AutoML) procedure was used to select the best regression model among six distinct state-of-the-art ML algorithms. A total of nine textile physical properties were modeled (e.g., abrasion, elasticity, pilling). Overall, the simpler yarn representation strategy obtained better predictive results. Moreover, for eight fabric properties (e.g., elasticity, pilling) the addition of finishing features improved the quality of the predictions. The best ML models obtained low predictive errors (from 2% to 7%) and are potentially valuable for the textile company, since they can be used to reduce the number of production attempts (saving time and costs).

**Keywords:** Textile fabrics · Regression · Machine Learning

## 1 Introduction

The introduction of the Industry 4.0 concept is transforming diverse industry sectors due to the adoption of Information Technology (IT), such as Internet of Things (IoT), Big Data, Cloud Computing and Artificial Intelligence (AI) [10, 23]. In particular, the Industry 4.0 transformation can enhance the textile industry by improving the production efficiency (e.g., reducing costs) and assisting in the design of woven fabrics.

In this work, we address a textile company that is being transformed by the Industry 4.0. The company produces custom made woven fabrics for diverse clients. Currently, the fabric design is mostly based on the designer experience and intuition, which results in the execution of several trial-and-error production experiments that require resources (production materials, machines, human labour) and time. Each new fabric production attempt also requires laboratory quality tests, to verify if the produced fabric complies with quality standards and the client requirements. If a fabric is not approved, a new design attempt is set, resulting in an additional production time and costs. All these production steps generate data that can be explored by AI tools, namely Data Mining (DM) and Machine Learning (ML), to support the design of new woven fabrics.

In this paper, we report the implementation of a CRISP-Industry Standard Process for DM (CRISP-DM) [24] project for the prediction of the final fabric physical properties, as measured by nine laboratory quality tests (e.g., abrasion, pilling). The goal is to use a ML model as an “oracle”, providing estimates of the fabric real physical properties for several input design options, thus aiding the textile design experts and reducing the number of fabric production attempts. To better focus on input feature selection and transformation, we adopt an Automated Machine Learning (AutoML) procedure during the modeling stage of CRISP-DM, allowing to automatically select and tune the hyperparameters of the predictive ML models [19]. In particular, we focus on input variables that can be set during the textile design phase, namely based on fabric design (e.g., composition, amount of finished threads) and finishing (e.g., washing, drying, singeing) features. In total, we executed two major CRISP-DM iterations, in which we explored different input feature engineering strategies.

This paper is organized as follows: Sect. 2 introduces the related work; Sect. 3 presents the two CRISP-DM iterations; Sect. 4 details the obtained results; and finally Sect. 5 presents the main conclusions.

## 2 Related Work

The creation of a new woven fabric is composed of several phases (e.g., design, production, testing). In particular, fabric testing has a crucial role in assessment product quality and performance, ensuring regulatory compliance and it provides information about the properties of the fabrics [18]. The overall process of new fabric creation generates large amounts of data, which under the Industry 4.0 concept can be used by AI tools (DM and ML) to extract valuable knowledge [17]. Following the increasing interest in DM, the CRISP-DM was proposed as a standard methodology to support the execution of real DM projects [24]. The methodology involves interactions between business domain and DM/ML experts and several iterations that can include up to six main phases: business understanding, data understanding, data preparation, modeling, evaluation and deployment. Regarding the textile domain, use of DM techniques is more recent, involving mainly classification tasks, such as defect detection and estimating the quality of yarns [26].

In what concerns a data-driven modeling of textile quality tests, the research is more scarce. Fan and Hunter [12] used a backpropagation Neural Network (NN) with one single hidden layer with 30 inputs based on fibre, yarn, and fabric constructional parameters to predict nine fabric properties (e.g., abrasion, seam slippage). A similar NN model was adopted in [1] to predict the pilling propensity of fabrics. In other study, Support Vector Machines (SVM) with 17 input features related to fiber and yarn were used to predict 8 different rates of pilling [25]. In the same study, backpropagation NN were also proposed to predict other textile properties, such as seam strength and elongation. In a more recent study [11], a simple multiple regression model was used to estimate the relationship between fabric tear strength and other input variables, such as yarn tensile strength, yarn count and fabric linear density. In previous work [22], we performed an initial exploration of ML algorithms, such as Random Forests (RF) and Gradient Boosting Machines (GBM), to predict two fabric properties (tear strength in warp and weft directions). While interesting results were achieved, the study explored a very limited set of inputs (e.g., no finishing data was used).

The performance of a ML algorithm is dependent on a correct feature engineering and ML model selection [9]. Most of the mentioned state-of-the-art works adopt a simple and fixed set of input variables (defined *a priori*) when predicting the textile physical properties. Moreover, the prediction models were obtained by using an empirical trial-and-error process that often requires a substantial effort from the ML expert [15]. In contrast with the related works (e.g., [1, 12, 25]), we employ in this paper an AutoML that automatically selects the best among several state-of-the-art ML algorithms. The adoption of the AutoML procedure allowed us to better focus on feature engineering, which is a non-trivial task in this domain. For instance, finishing features were considering a future challenge and thus were excluded from the predictive study performed in [12]. Woven fabric feature engineering is a complex task due to two main reasons. First, the design and finishing processes of the fabric creation includes a variable number of input features (e.g., yarns, finishing operations) that can influence the targeted textile physical properties. Second, most of these features are nominal and often present a high cardinality. Since most regression ML algorithms work only work with numeric values, a nominal to numeric transform is needed.

### 3 CRISP-DM Methodology

#### 3.1 First CRISP-DM Iteration

**Business Understanding:** The creation of a new woven fabric starts with the definition desired characteristics. The fabric developer uses its experience and intuition, taking into account the textile requirements and starts to analyse the most similar fabrics already produced. Then, a several design elements are initially set, such as the type and number of fibers and the pick count. Some of these design elements involve a single value per fabric (e.g., number of picks), while others involve a variable number of choices (e.g., which and how many yarns to use). Next, a physical sample is produced using several materials

(e.g., yarns) and machines (e.g., loom). The final production stage includes a variable sequence of finishing operations (e.g., washing, drying). Then, the produced fabric is tested via laboratory instruments, allowing to infer the physical properties and check if it meets the desired characteristics. If the fabric does not comply with the quality standards or client requirements, then the whole fabric creation process is repeated. In practice, several iterations are executed until a quality fabric is achieved, which results in additional production time and costs. The analyzed textile company expressed the need to get a fast and cheap estimate of the true fabric physical properties by adopting a ML approach. The goal is to use the predictive ML models as “oracles”, quickly checking some fabric design and finishing alternative choices, thus reducing the number of attempts necessary to produce a woven fabric.

In total, the company identified nine target properties: abrasion, seam slippage (warp and weft directions), elasticity (warp and weft directions), dimensional stability to steam (warp and weft directions), bias distortion and pilling. All these nine properties are measured using numeric values. In this work, each property is measured as a separate regression task.

**Data Understanding and Preparation:** The data was collected from two main data sources: the company Enterprise Resource Planning (ERP), with fabric production records, and the laboratory testing database, with fabric quality tests performed between February 2012 to March 2019. The ERP data included 90,034 examples with 2,391 features per row. Using a manual analysis and domain expert knowledge, the ERP features were filtered into a total of 805 potentially relevant attributes. The laboratory dataset had 149,388 examples with the results for the nine selected physical tests. To aggregate all data, a Data Warehouse system was implemented, using an Extraction, Transform, Load (ETL) process to merge and preprocess the two data sources. During the ETL, some records were discarded since they had missing features (e.g., no yarns or no composition values).

When analysing the obtained historical data, we identified a small fraction of laboratory database entries (around 1%) that included slight different physical test values for the same fabric. After consulting the laboratory analysts, it become clear that the differences were due to the execution of laboratory tests at different fabric finishing procedures (e.g., before or after drying). Since the laboratory database did not include when such tests were executed, we opted to compute average values, in order to get a single number per fabric and test.

The initial set of input features explored in this CRISP-DM iteration is presented in Table 1. Figure 1 exemplifies how some of these features are related with the textile fabric. The first 11 rows of the table are related with a fixed set of design attributes that are defined for all fabrics. Each fabric is composed by two main elements warp and weft, each including a variable mixture of yarns, from 1 (minimum) to a maximum of 21 (in our database). Moreover, each yarn has four main characterizing features plus the number of its repetitions in the warp or weft (these features are shown in the last five rows of Table 1). Thus, the

proper preprocessing of yarn data, to feed the regression models, is a non-trivial issue. In this work, we propose the following yarn representation. For each fabric, we use a sequence that has a maximum of  $max_y$  yarns for warp and then another sequence of  $max_y$  yarns for weft. Each sequence is thus composed by the elements  $\langle y_1, \dots, y_{max_y} \rangle$ , where  $y_i$  denotes the  $i$ -th yarn representation data. In this work, we adopt the threshold of  $max_y = 6$  yarns per warp and weft. This value allows the representation of 99.7% of the fabrics without any information loss, while using a larger threshold would increase the sparseness of the input space, increasing the complexity of the predictive models. When a fabric does not have 6 yarns, we use a zero padding to fill the “empty” yarn values, which is a popular text preprocessing technique that adds null values (e.g., 0) to non-existent features. Finally, we explore two yarn alternative representations: **A**, use of the code (unique value) and its number of repetitions, where  $y_i$  is set as the tuple (code $_i$ , repetitions $_i$ ); and **B**, use of all yarn characterizing elements except the code, where  $y_i$  is set as (composition $_i$ , folds $_i$ , count $_i$ , repetitions $_i$ ). In total, the **A** representation assumes 35 input variables ( $11 + 2 \times 6 \times 2$ ), while the **B** encoding results in 59 input features ( $11 + 2 \times 6 \times 5$ ). Before feeding the ML algorithms, all the numeric inputs were standardized to a zero mean and one standard deviation. As for the nominal variables, several of them contain a high cardinality. For instance, the analyzed database includes 6,265 distinct types of yarns. A popular nominal to numeric transform is the one-hot encoding, which assigns one boolean value per nominal level. However, this transformation would highly increase the input space, resulting in a very sparse representation that would prejudice the learning of the regression models, also enlarging the computational memory and effort. To handle this issue, in this work we transform all nominal attributes with the Inverse Document Frequency (IDF) function:

$$IDF(l) = \ln(n/n_l) \quad (1)$$

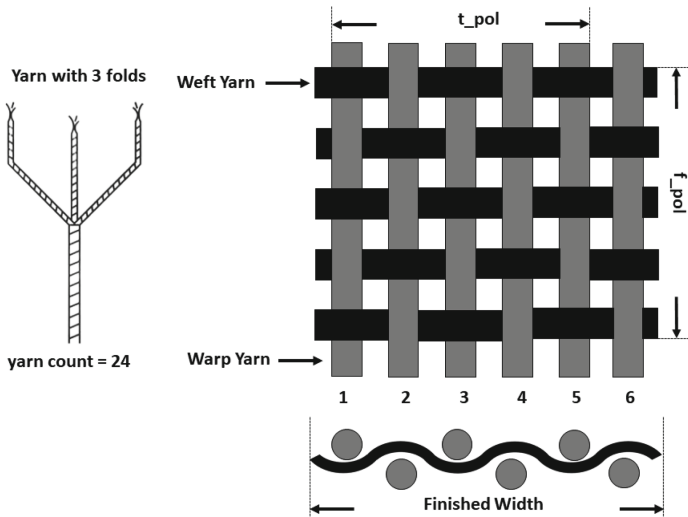
where  $n$  is the total number of examples in the training set and  $n_l$  is the number of examples that contain the level  $l$  in the analyzed attribute [5]. The advantage of this transform is that it encodes a nominal attribute into a single numeric value, with the most frequent levels being set near the zero (but with a larger “space” between them), and the less frequent ones being more close to each other and near a  $IDF(l)$  maximum value.

Table 2 presents the nine output targets. The last column (**Range**) shows the admissible range values for each target, as defined by the textile company. All examples outside such range were considered outliers (e.g., uncommon military fabrics) and thus removed from the dataset. Since different quality tests can be assigned to different fabrics (depending on the client requirements), a variable number of examples is presented for each output (column **Examples**).

**Modeling:** The experiments were conducted in a personal computer using two different computational environments: the R statistical tool and its **rminer** package for data manipulation and evaluation of ML algorithms [8], and H2O software which implements a AutoML procedure [7]. As previously discussed, during the

**Table 1.** Initial list of fabric design input features.

Attribute	Description (data type)	Range
composition	Composition of the fabric (nominal code)	660 levels
t_pol	Number of finished threads per centimeter (numeric)	[18, 1321]
p_pol	Number of finished picks per centimeter (numeric)	[7, 510]
weight/m <sup>2</sup>	Weight (in grams) per square meter (numeric)	[22, 1690]
finished width	Width in centimeters (numeric)	[122, 168]
weave design	Weave pattern of the fabric (nominal code)	20 levels
reed width	Width of the reed in centimeters (numeric)	[30, 242]
denting	Number of the reed dents per centimeter (numeric)	[5, 252]
ends/dent	Number of yarns per dent (numeric)	{0, 1, ..., 8}
n_picks	Number of picks on loom per centimeter (numeric)	[0, 3450]
warp total ends	Total number of threads on the warp (numeric)	[1026, 6862]
yarn code	Identification code of the yarn (nominal code)	6,265 levels
yarn composition	Composition of the yarn (nominal code)	88 levels
n_folds	Number of single yarns twisted (numeric)	{0, 1, ..., 12}
yarn count	Mass per unit length of the yarn (numeric)	[0, 268]
yarn repetitions	Number of yarn repetitions in warp or weft (numeric)	{0, 1, ..., 8}



**Fig. 1.** Visualization of some woven fabric features.

first iteration of CRISP-DM we explored the issue of yarn representation, thus two main strategies as compared: **A** and **B**. During this modeling stage, to find the best ML algorithm we adopt an AutoML procedure.

The AutoML was configured to automatically select the regression model and its hyperparameters based on the best Mean Absolute Error (MAE), using a internal 5-fold cross-validation applied over the training data. We adopted the same

**Table 2.** List of output target variables.

Test	Examples	Range
Abrasion	456	[5000, 30000]
Seam slippage (warp)	10,605	[1, 20]
Seam slippage (weft)	10,279	[1, 20]
Elasticity (warp)	7,901	[5, 55]
Elasticity (weft)	12,698	[5, 70]
Dimensional stability to steam (warp)	8,773	[-4, 2]
Dimensional stability to steam (weft)	8,871	[-4, 2]
Bias distortion	15,141	[1, 14]
Pilling	11,912	[1, 4.5]

AutoML configuration executed in [13]. The computational experiments were executed on a desktop computer and each ML algorithm was trained using a maximum running time of 3,600 s. After selecting the best ML algorithm, its best set of hyperparameters are fixed and the ML is retrained with all training data. A total of 6 different regression families are searched by the AutoML tool: RF, Extremely Randomized Trees (XRT), Generalized Linear Models (GLM), GBM, XGBoost (XG) and Stacked Ensemble (SE). RF is an ensemble method that typically combines a large set of tree predictors, such that each tree depends on a random sample of features and training examples [4]. XRT is another tree ensemble that consists of randomizing both attribute and cut-point choices when splitting a tree node [14]. GLM estimates regression models for outcomes following exponential distributions (e.g., Gaussian, Poisson, gamma) [21]. GBM performs an ensemble of weak successive decision trees, sequentially building regression trees for all data features [20]. XG is another popular boosting decision tree algorithm [6]. Finally, the SE combines the predictions of the previous individual ML algorithms by using a second-level ML algorithm [3]. The H20 tool sets RF and XRT with their default hyperparameters, performs a grid search to set the hyperparameters for GLM (1 hyperparameter), GBM (9 hyperparameters) and XG (10 hyperparameters), and uses GLM as the second-level learner for SE.

**Evaluation:** An external 3-fold cross-validation was executed to evaluate the regression models. Several metrics were selected to measure the quality of the predictions: MAE, Normalized MAE (NMAE), Adjusted R2 (Adj.R<sup>2</sup>) and classification accuracy for a given tolerance  $T$  (Acc@ $T$ ). Regarding MAE and NMAE, the lower the values, the better are the predictions. The NMAE measure normalizes the MAE by the range of the output target on the test set, thus it provides a percentage that is easy to interpret and is scale independent. In the case of Adj.R<sup>2</sup> and Acc@ $T$  (from 0 to 1), higher values indicate better predictions. The Acc@ $T$  value is based on the Regression Error Characteristic (REC) curves and it measures the percentage of correctly classified examples when assuming a

fixed absolute error tolerance ( $T$ ) [2]. In this work we use  $T \in \{5\%, 10\%, 20\%\}$ . We note that the percentage of error tolerance is computed by considering the range of the target values. The first CRISP-DM iteration results are discussed in Sect. 4.

### 3.2 Second CRISP-DM Iteration

After showing the Sect. 4 results to the textile experts, it was decided to perform a second CRISP-DM iteration to check the utility of finishing features. During a new business understanding phase, it became clear that the finishing process should influence the final fabric properties. The finishing consists of a predefined sequence of operations that are applied to a fabric with the goal to increase the attractiveness or serviceability of the textile product [16].

At the new data understanding and preparation stages, we identified that the company had a total of 61 different types of finishing operations. Moreover, the sequence of finishing operations can be different for each fabric and it can include repetitions of the operations (e.g., several wash and dry cycles). In the analyzed database, the number of executed finishing operations ranged from 1 (minimum) to 39 (maximum), with an average of 6.82. Table 3 presents the top ten most used types of finishing and the respective number of usages (column **Examples**). In the table, we added the special value “Others” to represent a merge of distinct finishing operations for which there was no description data.

**Table 3.** Ten most used fabric finishing operations.

Rank	Finishing	Examples	Rank	Finishing	Examples
1	Dry	145,008	6	Finish_Fixate	53,717
2	Wash	122,513	7	Dyeing	44,162
3	Sanforization	73,401	8	Others	31,704
4	Finish	67,774	9	Shear_Right	21,600
5	Singeing	58,875	10	Decatizing	20,685

Similarly to the yarn encoding strategy, in this work we will assume a sequence with a maximum of  $max_f$  finishing operations to represent the finishing process:  $\langle f_1, \dots, f_{max_f} \rangle$ , where  $f_i$  denotes the  $i$ -th finishing operation. In this work, we set  $max_f = 10$  as a reasonable value that represents around 85% of all fabrics without information loss, helping to reduce the number of inputs that are fed into the ML models. To encode each finishing operation (nominal attribute) we adopt the same IDF transform (Eq. 1). In the modeling phase, the best previous input encoding (**A**) is compared with the new encoding **C** that merges all **A** inputs with the finishing features, resulting in 45 (35 + 10) input variables. The evaluation phase was executed similarly to the first CRISP-DM iteration.



**Table 4.** AutoML predictive results (average NMAE test set values, in %; best results per CRISP-DM iteration in **bold**).

Target	First iteration		Second iteration	
	A	B	A	C
Abrasion	<b>4.81</b>	5.17	<b>4.81</b>	4.93
Seam slippage (warp)	<b>4.54</b>	5.00	4.54	<b>4.43</b>
Seam slippage (weft)	3.09	<b>2.62</b>	3.09	<b>2.56</b>
Elasticity (warp)	<b>2.94</b>	3.42	2.94	<b>2.59</b>
Elasticity (weft)	<b>2.39</b>	2.87	2.39	<b>2.15</b>
Dimensional stability to steam (warp)	<b>6.60</b>	7.52	6.60	<b>6.17</b>
Dimensional stability to steam (weft)	<b>4.27</b>	6.58	4.27	<b>4.12</b>
Bias distortion	<b>4.16</b>	4.33	4.16	<b>3.79</b>
Pilling	<b>6.80</b>	8.14	6.80	<b>6.70</b>
<b>Average</b>	<b>4.41</b>	5.07	4.41	<b>4.16</b>

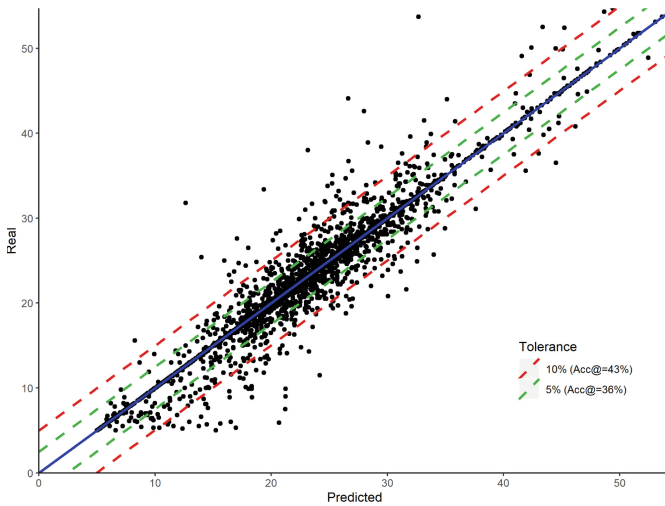
## 4 Results

In all experiments performed the AutoML always selected the GBM or SE algorithms. GBM provided the best overall results (lowest NMAE averaged over the external 3 cross-validation iterations), while for some targets and specific folds (e.g., Bias Distortion and third fold experiment), the selected model was SE. To compare the feature strategy results, we always assume the best algorithm (GBM or SE) per external fold validation, denoting this as the AutoML model.

Table 4 summarizes the predictive performance results, in terms of the 3-fold average NMAE values for the best AutoML model that were obtained during the first and second CRISP-DM iterations. For the first iteration, it becomes clear that **A** is the best yarn representation strategy. It provides the lowest NMAE results for eight of the nine fabric targets and it also obtains the lowest average value, over all output tasks (difference of 0.66% points when compared with **B**). Moreover, **A** has the additional advantage of producing less inputs (35 and not 59), leading to predictive models that require less computational memory and fitting effort. Following these results, we adopted the **A** encoding to represent the yarns. In the second iteration, the usage of fabric finishing features (**C**) improves the prediction results for eight of the nine targets. Overall, **C** provides the lowest average NMAE, with a 0.25% point improvement when compared with **A**. Table 5 complements the results by showing the other predictive measures for **A** and **C** (represented in column **Str.**). In general, when **C** obtains the lowest MAE error, it also outperforms the **A** strategy for the other measures (Acc@ $T$  and Adj.R<sup>2</sup>). For demonstrative purposes, Fig. 2 shows the AutoML elasticity (warp) predictions ( $x$ -axis) versus the target values for a particular external 3-fold iteration. The plot includes the tolerance ranges of the  $T = 5\%$  and  $T = 10\%$ , showing that an interesting percentage of the values are correctly predicted within those ranges (e.g., 43% of accuracy for  $T = 10\%$ ).

**Table 5.** AutoML predictive results (other regression measures; best results in **bold**).

Test	Str.	Regression metrics				
		MAE	Acc@5%	Acc@10%	Acc@20%	Adj.R <sup>2</sup>
Abrasion	A	<b>924.03</b>	<b>1%</b>	<b>1%</b>	<b>2%</b>	<b>0.76</b>
	C	948.60	0%	0%	1%	0.75
Seam slippage (warp)	A	0.83	24%	35%	47%	0.79
	C	<b>0.81</b>	<b>28%</b>	<b>42%</b>	<b>54%</b>	<b>0.80</b>
Seam slippage (weft)	A	0.55	28%	52%	65%	0.83
	C	<b>0.46</b>	<b>61%</b>	<b>67%</b>	<b>73%</b>	<b>0.84</b>
Elasticity (warp)	A	1.46	11%	17%	24%	<b>0.92</b>
	C	<b>1.29</b>	<b>26%</b>	<b>34%</b>	<b>42%</b>	<b>0.92</b>
Elasticity (weft)	A	1.25	10%	17%	25%	<b>0.92</b>
	C	<b>1.12</b>	<b>13%</b>	<b>20%</b>	<b>30%</b>	<b>0.92</b>
Dimensional stability to steam (warp)	A	0.40	17%	31%	50%	0.63
	C	<b>0.37</b>	<b>30%</b>	<b>40%</b>	<b>53%</b>	<b>0.65</b>
Dimensional stability to steam (weft)	A	<b>0.25</b>	<b>55%</b>	61%	67%	0.73
	C	<b>0.25</b>	<b>55%</b>	<b>62%</b>	<b>68%</b>	<b>0.75</b>
Bias distortion	A	0.50	13%	25%	40%	0.53
	C	<b>0.46</b>	<b>14%</b>	<b>26%</b>	<b>44%</b>	<b>0.59</b>
Pilling	A	0.24	<b>35%</b>	<b>48%</b>	<b>63%</b>	0.76
	C	<b>0.23</b>	25%	37%	54%	<b>0.78</b>



**Fig. 2.** Predicted versus real elasticity (warp) values.

## 5 Conclusions

This paper addresses a textile company that is being transformed under the Industry 4.0 concept and that identified the need to reduce the number of production attempts when designing new woven fabrics by using a Machine Learning (ML) approach. To handle this goal, we implemented two iterations of the CRoss-Industry Standard Process for Data Mining (CRISP-DM) methodology. Each iteration focused on a feature engineering task, aiming to check the value of input fabric yarn and finishing feature representations. During the modeling stage of CRISP-DM, an Automated ML (AutoML) was used to select the best among six state-of-the-art ML algorithms. The best results were achieved by an input set of features that includes a fixed sequence with a simple yarn code representation and another fixed sequence with fabric finishing operations (strategy C). Interesting predictive results were achieved for nine targeted fabric properties, with an average NMAE error that ranges from 2% to 7%. The results were shown to the textile company, which considered them valuable to reduce the number of fabric creation attempts, thus having a potential to save the production time and costs. In future work, we intend to apply a similar approach in the prediction of other fabric quality tests, such as residual extension and traction.

**Acknowledgments.** This work was carried out within the project “TexBoost: less Commodities more Specialities” reference POCI-01-0247-FEDER-024523, co-funded by *Fundo Europeu de Desenvolvimento Regional* (FEDER), through Portugal 2020 (P2020).

## References

1. Beltran, R., Wang, L., Wang, X.: Predicting the pilling propensity of fabrics through artificial neural network modeling. *Text. Res. J.* **75**(7), 557–561 (2005)
2. Bi, J., Bennett, K.P.: Regression error characteristic curves. In: Proceedings of the 20th International Conference on Machine Learning (ICML 2003), pp. 43–50 (2003)
3. Breiman, L.: Stacked regressions. *Mach. Learn.* **24**(1), 49–64 (1996). <https://doi.org/10.1007/BF00117832>
4. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
5. Campos, G.O., et al.: On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data Min. Knowl. Discov.* **30**(4), 891–927 (2016). <https://doi.org/10.1007/s10618-015-0444-8>
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, pp. 785–794 (2016)
7. Cook, D.: *Practical Machine Learning with H2O: Powerful, Scalable Techniques for Deep Learning and AI*. O’Reilly Media Inc., Sebastopol (2016)
8. Cortez, P.: *Modern Optimization with R*. Springer, Cham (2014). <https://doi.org/10.1007/978-3-319-08263-9>
9. Domingos, P.: A few useful things to know about machine learning. *Commun. ACM* **55**(10), 78–87 (2012)

10. Drath, R., Horch, A.: Industrie 4.0: Hit or hype? [Industry Forum] (2014). <https://doi.org/10.1109/MIE.2014.2312079>
11. Eltayib, H.E., Ali, A.H., Ishag, I.A.: The prediction of tear strength of plain weave fabric using linear regression models. *Int. J. Adv. Eng. Res. Sci.* **3**(11), 151–154 (2016)
12. Fan, J., Hunter, L.: A worsted fabric expert system: Part II: an artificial neural network model for predicting the properties of worsted fabrics. *Text. Res. J.* **68**(10), 763–771 (1998). <https://doi.org/10.1177/004051759806801010>
13. Ferreira, L., Pilastrri, A., Martins, C., Santos, P., Cortez, P.: An automated and distributed machine learning framework for telecommunications risk management. In: *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 2, Valletta, Malta, February*, pp. 99–107. SciTePress (2020)
14. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. *Mach. Learn.* **63**(1), 3–42 (2006)
15. Gibert, K., Izquierdo, J., Sánchez-Marrè, M., Hamilton, S.H., Rodríguez-Roda, I., Holmes, G.: Which method to use? An assessment of data mining methods in environmental data science. *Environ. Model. Softw.* **110**, 3–27 (2018)
16. Hall, M.E.: Finishing of technical textiles. In: *Handbook of Technical Textiles*, p. 152 (2000)
17. Han, J., Kamber, M., Pei, J.: *Data Mining: Concepts and Techniques*. Elsevier Inc., Amsterdam (2012). <https://doi.org/10.1016/C2009-0-61819-5>
18. Hu, J.: *Fabric Testing*. Elsevier Ltd., Amsterdam (2008). <https://doi.org/10.1533/9781845695064>
19. Le, T.T., Fu, W., Moore, J.H.: Scaling tree-based automated machine learning to biomedical big data with a feature set selector. *Bioinformatics* (2019). <https://doi.org/10.1093/bioinformatics/btz470>
20. Natekin, A., Knoll, A.: Gradient boosting machines, a tutorial. *Front. Neurobotics* **7**, 21 (2013)
21. Nelder, J.A., Wedderburn, R.W.: Generalized linear models. *J. R. Stat. Soc. Ser. (Gen.)* **135**(3), 370–384 (1972)
22. Ribeiro, R., Pilastrri, A., Moura, C., Rodrigues, F., Rocha, R., Cortez, P.: Predicting the tear strength of woven fabrics via automated machine learning: an application of the CRISP-DM methodology. In: *Proceedings of the 22th International Conference on Enterprise Information Systems – ICEIS2020, Prague, Czech Republic*. SciTePress, May 2020
23. Wang, S., Wan, J., Li, D., Zhang, C.: Implementing smart factory of Industrie 4.0: an outlook. *Int. J. Distrib. Sens. Netw.* **2016** (2016). <https://doi.org/10.1155/2016/3159805>
24. Wirth, R., Hipp, J.: CRISP-DM: towards a standard process model for data mining. In: *Proceedings of the 4th International Conference on the Practical Applications Of Knowledge Discovery and Data Mining*, pp. 29–39. Springer, London (2000)
25. Yap, P.H., Wang, X., Wang, L., Ong, K.L.: Prediction of wool knitwear pilling propensity using support vector machines. *Text. Res. J.* **80**(1), 77–83 (2010). <https://doi.org/10.1177/0040517509102226>
26. Yildirim, P., Birant, D., Alpyildiz, T.: Data mining and machine learning in textile industry. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **8**(1), e1228 (2018)