Towards Interactive Feature Selection with Human-in-the-loop

Maialen Larrañaga, Dimitra Gkorou, Thiago Guzella, Alexander Ypma,
Faegheh Hasibi, Robert Jan van Wijk

ASML, De Run 6501, Veldhoven 5504DR, the Netherlands

1 Introduction

Feature Selection (FS) has been applied to numerous domains, and shown to be effective in increasing the performance of machine learning algorithms. In the semiconductor industry, FS is part of various prediction tasks that aim at avoiding production stops and yield loss. For example, it can be used for: (i) *diagnostics*, wherein relevant features constitute potential root causes, with their identification being the initial step in a detailed investigation of process defects [3]; (ii) *control*, as the values of a small set of relevant features can be used to group objects and apply actions per group [1]; (iii) *improving prediction performance and interpretability*, by enforcing sparsity [4, 7]. Nevertheless, when analyzing manufacturing datasets, one faces two particular challenges, as in other real-world datasets:

- **High-dimensionality:** typically, the number of features p to be evaluated in FS can reach hundreds of thousands and it is much larger than the number of instances n.
- **Collinearity**: some features may be strongly correlated with one another.

These characteristics challenge the robustness of FS against spurious correlations. To overcome these challenges, we propose an interactive FS scheme in which a user provides expert feedback, by assessing the relevance of the features with respect to a performance metric and thereby separating relevant features from spurious correlations. An initial approach has been implemented as a web application in ASML, the leading manufacturer of lithography machines and major player in the semiconductor industry. Next, we give a detailed explanation of the proposed interactive scheme, while the supplementary material provides further details on the implementation.

2 Interactive Feature Selection

For integrating expert feedback, we use the knowledge elicitation framework proposed by Daee et al. [2]. It is based on a bayesian regression model with spike and slab (s&s) sparsity-enforcing priors [2, 5]. Let us denote by $\mathbf{y} \in \mathbb{R}^n$ the target of interest, and let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the feature set. Assume $\mathbf{w} \in \mathbb{R}^p$ to be the regression coefficients. The goal is to define the posterior distribution of $\mathbf{w}$, given that $\mathbf{y} \sim \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2 I)$ and $w_j \sim z_j \mathcal{N}(0, \psi^2) + (1 - z_j)\delta_0$ (i.e., s&s prior), where δ_0 is a Dirac delta, and z_j encodes the relevance of the features ($z_j = 1$ if feature j is expected to contribute to the regression and $z_j = 0$ otherwise). For ease of notation, we consider σ^2 and ψ^2 to be constants. The computation of the posterior distribution of w_j for all $j = 1, \dots, p$ depends on the s&s prior but also on the feedback provided by a domain expert (*relevant*, *irrelevant*, *do not know*). These

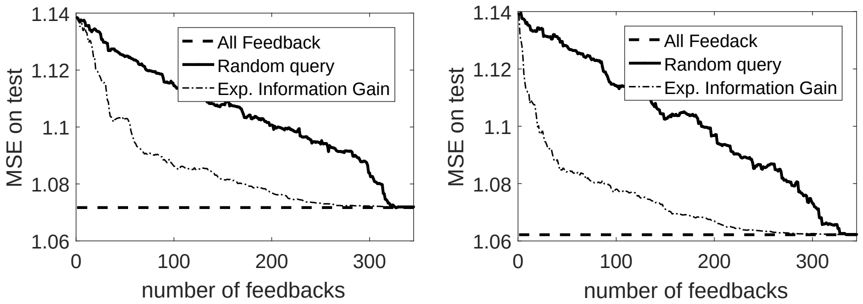


Fig. 1: Evaluation of FS approach for different query strategies and user feedback. Left: Evaluation with imperfect user feedback. Right: Evaluation with real expert feedback.

posteriors are sequentially adapted every time an expert is queried about the relevance of a feature. In order not to overwhelm the expert, we use smart query strategies that reduce the number of required user feedback. We implement the Expected Information Gain (EIG) strategy proposed in Daee et al. [2].

We evaluated the implementation of our FS scheme with an ASML expert. We used a dataset with 344 features and 100 instances from the logs of ASML machines. The results are shown in Fig. 1, for a simulated expert (Fig. 1, left) and for a knowledgeable domain expert (Fig. 1, right), with the former being less trustworthy than the latter. In these results, “All feedback” refers to the performance of the bayesian regression with s&s priors after all expert feedback is received; we note that the best achieved MSE in Fig. 1 left is higher than the best MSE in Fig. 1 right, but in both cases the EIG strategy reaches the best MSE after $\sim$ 250 feedbacks. Even with a simulated expert (Fig. 1, left) we see an improvement on the predictive performance with respect to not having an expert in the loop (i.e., when the number of feedback equals 0). This is crucial for the robustness of our application.

3 Open questions and future work

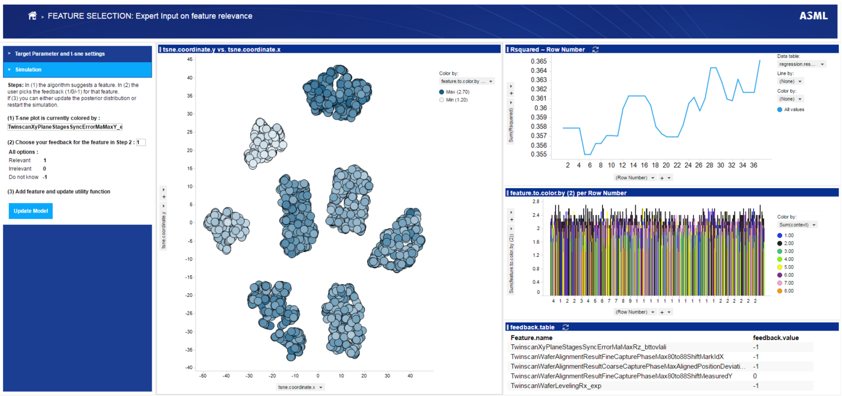
At present, we still face several challenges in the implementation of the proposed scheme. First, our datasets have typically thousands of features, such that asking feedback from a domain expert for all these features is unfeasible. Second, domain experts are often not sure of the feedback for some particular features, and their input might also be biased. To address these challenges, we are adopting the following steps:

- group features that relate to the same effects/machines in a single category and ask the expert to provide feedback on the entire category at once.
- visualize the data in an informative way to help the expert make a decision on the relevance of a particular feature category and avoid biases.

How to group features into categories and how to visualize the data in the *most informative* way remain an open question. In particular, the data visualization is a challenging task since (1) dimensionality reduction methods show artifacts, (2) assessing the quality of a visualization is not straightforward and (3) often only well-known effects and structures emerge in the visualization. One needs to get rid of the latter to discover the underlying patterns.

References

1. H. E. Cekli, J. Nije, A. Ypma, V. Bastani, D. Sonntag, H. Niesing, L. Zhang, Z. Ullah, V. Subramony, R. Somasundaram, W. Susanto, M. Matsunobu, J. Johnson, C. Tabery, C. Lin, and Y. Zou. A novel patterning control strategy based on real-time fingerprint recognition and adaptive wafer level scanner optimization. volume 10585, 2018.
2. Pedram Daee, Tomi Peltola, Marta Soare, and Samuel Kaski. Knowledge elicitation via sequential probabilistic inference for high-dimensional prediction. *Machine Learning*, 106:1599–1620, 2017.
3. M. Giollo, A. Lam, D. Gkorou, X. Lan Liu, and R. van Haren. Machine learning for fab automated diagnostics. *Proc.SPIE*, 10446, 2017.
4. F. Hasibi, L. van Dijk, M. Larrañaga, A. Pastol, A. Lam, and R. van Haren. Towards fab cycle time reduction by machine learning based overlay metrology. *Proc. SPIE*, 2018.
5. J. M. Hernández-Lobato, D. Hernández-Lobato, and A. Suárez. Expectation propagation in linear regression models with spike-and-slab priors. *Machine Learning*, 99(3):437–487, 2015.
6. L. Kirsch, N. Riekenbrauck, D. Thevessen, M. Pappik, A. Stebner, J. Kunze, A. Meissner, A. K. Shekar, and E. Müller. Framework for exploring and understanding multivariate correlations. In *Machine Learning and Knowledge Discovery in Databases*, pages 404–408, 2017.
7. A. Lam, A. Ypma, M. Gatefait, D. Deckers, A. Koopman, R. van Haren, and J Beltman. Pattern recognition and data mining techniques to identify factors in wafer processing and control determining overlay error. *Proc. SPIE*, 9424, 2015.



Steps: In (1) the algorithm suggests a feature. In (2) the user picks the feedback (1/0/-1) for that feature. If (3) you can either update the posterior distribution or restart the simulation.

(1) T-sne plot is currently colored by : TwinscanXyPlaneStagesSyncErrorMaMaxY_e

(2) Choose your feedback for the feature in Step 2 : 1

All options :

Relevant	1
Irrelevant	0
Do not know	-1

(3) Add feature and update utility function

Feature.name	feedback.value
TwinscanXyPlaneStagesSyncErrorMaMaxRz_bttovlali	-1
TwinscanWaferAlignmentResultFineCapturePhaseMax80to88ShiftMarkIdX	-1
TwinscanWaferAlignmentResultCoarseCapturePhaseMaxAlignedPositionDeviati...	-1
TwinscanWaferAlignmentResultFineCapturePhaseMax80to88ShiftMeasuredY	0
TwinscanWaferLevelingRx_exp	-1

Fig. 2: Snapshot of the visualization of the proposed FS framework

4 Supplementary Material: web application

In this work, we sought to use smart query strategies to reduce the number of required user feedback. While some state-of-the-art tools allow for user feedback, none of them are directly applicable to our problem setting. RapidMiner, Weka, and FEXUM [6] focus on supervised and unsupervised problems allowing only offline user feedback. KNIME has an active learning component, but it cannot be directly applied to FS. Therefore, we have implemented a web application in ASML for the interactive Feature Selection scheme.

The expected users of our framework are domain experts without Machine Learning knowledge but with scientific background e.g., domain experts, field engineers, etc. The experts can use our tool to easily find the relevant features to a selected target variable. The User Interface of the proposed FS scheme is presented in Fig. 2. The expert initializes a workflow by selecting one or more performance metrics to monitor (targets $\mathbf{y}$) on the left panel in Fig. 2. The application, with non-linear embeddings (Fig. 2 middle plot), helps the user understand complex relations and interconnections between multiple features and the target. Particularly, users can select multiple features and observe interactively, via a t-sne embedding, their 2D visualization colored by the values of the target.

In each iteration a user is provided with a feature (left most panel) for which he/she is required to give feedback (*relevant* feature, *irrelevant* feature or *do not know*). As mentioned in the abstract, this feature is selected sequentially via knowledge elicitation with the EIG query strategy. The user interacts with the application to provide feedback on a certain number of features. To help the decision-making process, on Fig. 2 (right), three plots are shown: (i) the R^2 value of the linear regression model after each expert feedback. This provides information on the predictive power of the model in each iteration; (ii) the values of the feature that the user is evaluating colored per cluster. This gives information on how the target value relates to the underlying structure of the data; (iii) a list with all the feedback that he/she has already provided. Once the decision has been made, the feedback is given on the left panel, the posterior distribution of the regression is updated and the algorithm suggests a new feature.

In this way, actively querying the expert for feedback results in better user experience as, the users are not expected to examine all the features to find those whose relevance they should assess.