Interaction Modelings for Industrial Data - for final quality estimation and proess integration

L. QIN, T. NAGATA, K. HANDA, K. WATANABE, Y. OGAWA, T. MAEDA, S. ARIMA
arima@sk.tsukuba.ac.jp
University of Tsukuba
[1-1-1, Tennodai, Tsukuba city, Ibaraki pref., 305-8573 Japan]
Phone: +81 -298-535-558 Fax: +81-298-535-558

Abstract – This paper introduced interaction modeling methods based on a sparse modeling. Some numerical evaluations proved to select promising variables and interactions at high accuracy. Scalable synthesis data and industrial data were used for the evaluations of effectiveness and computational efficiency. As the last, the possibility and issues for actual use are summarized.

I. INTRODUCTION

It has been seemed external challenge to estimate a final product quality and analyze its cause from more than 0.1-5 million explanatory variables and more than 0.1-10 billions interaction of whole of several hundred process steps. However, recently it becomes possible to use a big data, a powerful computational environment, and a sparse modeling responding to high dimensional data which general multivariate statistical analyses cannot solve. An industrial data will become to include more interaction effects because of the piled steps, advanced equipment/process control, and leading-edge product structure. That is a serious difficulty of process integrations beyond the knowledge and skills of human-being. This paper will introduce automation methods of interaction modeling as the first step.

The number of candidates of second-order interaction effects reaches about 200 million even if there are only 20000 first-order variables. SPRINTER (Sparse reluctant interaction modeling) [1] is one of hopeful solutions of second-order interaction modeling for such an actual data of high dimension. Residuals of a regression model of main effects (e.g. LASSO [2]) is used in selection step of interaction effects like the sure independence screening [3]. Fast computation by the top-m approach [4] and the various interaction selection ability are powerful for industrial data. Even an interaction without a variables of effective main effect can be selected.

The second interaction model is Pliable LASSO [5] in which interaction is expressed as a factorization model. It may be efficient in case less number of factors related to the interaction.

II. METHODS AND NUMERICAL EXPERIMENTS

The algorithm of SPRINTER [1] consists of three steps as shown in Fig.1. SPRINTER is based on reluctant interaction selection principle, and the fast computation and various interaction effect selection is guaranteed in that mechanism.

As a first numerical evaluations using synthesis data (Fig.2), Fig.3 shows accuracy of main and interaction effects of SPRINTER (SL) to compare with LASSO of only main effects (MEL), Two-stage LASSO (TwS-L) [6]. In summary, SL is the best of three. In addition, we proposed SPRINTER with Stability selection [7] (SSS) which can improve False positive (FP*) of SL by its ensemble mechanism. SSS can achieve much less FP* than others as we aimed.

Table 2 shows the result of actual data evaluation. The followings are summary of numerical results for actual data. (Data details are impossible be opened)

About efficiency, the main and interaction factors can be automatically selected by SPRINTER within 3.5 hours (84% reduced, without additional man-hour) for an actual data (Fig.2(b), $n = 17333$, $m = \lceil n/\log(n) \rceil$=1776). SSS is also applicable (6 hours) though it takes 1.7 times of SL because of its ensemble mechanism. On the other hand, TwS-L cannot work well because of memory shortage.

About effectiveness, from 20000 variables 10 main effects in oracle can be detected in 100% by MEL, SL, and SSS. In addition, three interaction effects are newly discovered by SL as a data-driven method. The discovered results are supported as effective by engineering knowledge. SPRINTER can detect the interaction effect even though no valid main factor is included in the interaction factor. That means we can use it not only in mass production phase but also in design and trial phases of process integration. On the other hand, SSS cannot detect the valid interaction effects because the categorical variables exist as a block in the time-series in the actual data. SSS's random sampling and the ensemble learning through data subsets do not become effective. Also for the actual data, SL and MEL tend to select too many variables but SSS can restrain the number of selected variables.

III. CONCLUSION

This paper introduced interaction modeling methods based on a sparse modeling. Some numerical evaluations proved to select promising variables and interactions at high accuracy for synthesis data and industrial data.

There are two future works. The first is to respond to infrequency and bias in time series of 0-1 variables of the actual industrial data. The second is a portfolio strategy to set the best interaction modeling method. We will prove advanced solutions of those issues near future.

REFERENCES

1. G. Yu, J. Bien, and R. Tibshirani.: Reluctant Interaction Modeling. arXiv preprint arXiv:1907.08414v1 (2019).
2. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) 58(1), 267-288 (1996).
3. Fan, J. & Lv, J. : Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society Series B (Statistical Methodology) 70(5), 849–911 (2008).
4. Niu, Y. S., Hao, N. and Zhang, H. H.: Interaction screening by partial correlation. Statistics and Its Interface 11(2), 317–325 (2018).
5. R. Tibshirani, and J. Friedman. "A pliable lasso." Journal of Computational and Graphical Statistics 29(1), 215-225 (2020).
6. Hao, N., Feng, Y., and Zhang, H. H. : Model selection for high-dimensional quadratic regression via regularization. Journal of the American Statistical Association 113(522), 615-625 (2018).
7. N. Meinshausen., and P. Bühlmann: Stability selection. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72(4), 417-473 (2010).

Algorithm: SPRINTER

Require: Main effect $\boldsymbol{X} \in \mathbb{R}^{n\times p}$, response $\boldsymbol{y} \in \mathbb{R}^{n}$, $\eta > 0$

Step1:

Fit a lasso of response $\boldsymbol{y}$ on $\boldsymbol{X}$.

Compute the residuals $\mathbf{r} = \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\theta}}$

Step2:

For a tuning parameter η, screen based on the residual which computed in Step1.

$$\hat{L}_{\eta} = \{l \in [q]: \overline{\mathrm{sd}}(r)|\overline{\mathrm{cor}}(\boldsymbol{Z}_l, r)| > \eta\}$$

Here, let $\overline{\mathrm{cor}}(\boldsymbol{X}_j, \boldsymbol{X}_k)$ stand for the sample correlation between variables j and k, and let $\overline{\mathrm{sd}}(\boldsymbol{X}_j)$ to be the sample standard deviation of $\boldsymbol{X}_j$. And $[p]$ is the set $\{1, 2, 3, \cdots, p\}$.

Note that the top-m approach [4] is used for the fast computation for the screening.

Step3:

Fit a Lasso of the response $\boldsymbol{y}$ on $\boldsymbol{X}$ and $\boldsymbol{Z}_{\hat{X}_{\eta}}$.

Fig. 1. SPRINTER algorithm

$\boldsymbol{X} \in \mathbb{R}^{n\times p}$, $\boldsymbol{Z} \in \mathbb{R}^{n\times \frac{p^2+p}{2}}$, $\boldsymbol{y} \in \mathbb{R}^{n}$, $\epsilon \in \mathbb{R}^{n}$
$n = 1000$, $p \in \{5000, 10000, 20000\}$
$X_{i,j} \sim \mathcal{N}(0,1)$ for the machine variables (identically distributed)
$X_{i,k} \sim Bern(0.3)$ for categorical variables
(identically distributed Bernoulli random variables)
$Z = (X_1 * X_1, X_1 * X_2, \ldots, X_p * X_p)$ for interaction effect candicates
$\epsilon_i \sim \mathcal{N}(0,1): error$ (identically distributed)
$i=1,\ldots,n$, $j = 2,4,6,\ldots,\frac{p}{2}$, $k = 1,3,5,\frac{p+1}{2},\ldots,p$
$\boldsymbol{T}_{main}$, $\boldsymbol{T}_{int}$: index set of indexes of main or interaction effects in oracle.
$\boldsymbol{\beta}_{main}$, $\boldsymbol{\beta}_{int}$: non-zero coefficient designed and generated based on actual data
Simulation data model 1: $\boldsymbol{y} = \boldsymbol{\beta}_{main}\boldsymbol{X}_{T_{main}} + \boldsymbol{\beta}_{int}\boldsymbol{X}_{T_{int}} + \epsilon$.

Example of instance:

T_{main}	Coefficient (β_{main})	T_{int}	Coefficient (β_{int})
$11,\ldots,20$	2	(1,2),(3,4)	5
$21,\ldots,30$	3	(5,6),(7,8)	6
$31,\ldots,40$	4	(9,10)	7

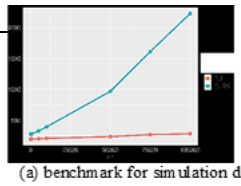
Fig. 2. Synthesis data (simulation data)

Table 1. Typical result of accuracy for simulation data 1 ($p = 10000$)

Method	Total # of variables selected	TP_{main}	FP_{main}	FN_{main}	TP_{int}	FP_{int}	FN_{int}
MEL	271.3	29.9	241.4	0.1	0	0	5
TwS-L	279.2	29.9	239.4	0.1	0	9.9	5
SL	349.5	30	269.5	0	4.5	45.5	0.5
SSS	72.5	26.7	32.2	3.3	4.2	9.4	0.8
Oracle	35.0	30	0	0	5	0	0

Table 2. Result of variable selection in actual industrial data 1 ($p \sim 20000$)

Method	# of Main factors	# of Interactions	The total number of non-zero variables
SL	1893	162	2055
SSS	70	14	84
MEL	1924	0	1924



(a) benchmark for simulation data

Methods	Computational time [seconds]
SL	13744.08 (about 3.5 hours)
SSS	22721.58 (about 6 hours)
MEL	570.09
TwS-L	Impossible to execute (memory shortage)

(b) comparison to actual data

Computer Environment:
(a): Intel (R) Core (TM) i7-8700CPU @3.20 GHz 3.19 GHz, memory: 32.0 GB
(b): Computer environment: Amazon Linux 2 EC2 (Machine type: t3.2xlarge, vCPU:8, Memory:32 GB

Fig. 2. Computational time [sec.]

The result will be shown in the final version of this manuscript

Fig. 3. Evaluation of Pliable LASSO (Case of Synthesis data in Fig.2)