Future Improvement

Jiaxin Hu

August 11, 2020

1 Unsolved Review Summary

1.1 Comparisons

1. Add comparison on tensor-on-tensor regression (Lock, 2018; Llosa, 2018; Gahrooei et al., 2020; Raskutti et al., 2015). **NIPS R3**.
2. Add comparison on scalar-on-tensor regression (Chen et al., 2019; Zhou et al., 2013). **ICML R1**.
3. Add comparison on HOLRR (Rabusseau and Kadri, 2016). Though HOLRR is a special case of the proposed model, it is helpful to show the benefit of the incorporation of covariate on distinct modes, when all covariates are vectorized in a long vector in HOLRR. **AISTAT R3**.
4. Add comparison on HOPLS (Zhao et al., 2012) and TPG (Yu and Liu, 2016) from the perspective of the model. **AISTAT R1**.
5. Add comparison on paper (Smith et al., 2015). **NIPS R4**.

1.2 Experiments

1. Add comparisons on other scalar-on-tensor and tensor-on-tensor regressions in both simulations and real data analysis. **NIPS R2, R3**.
2. Add numerical results on large-scale or higher-order response. **NIPS R2, AISTAT R3**.
3. Show how many iterations and time are needed to converge. **NIPS R2**.

1.3 Explanations

1. Add explanations on hyperparameter selection for our method and other methods (e.g. HOLRR, TPG, HOPLS) we want to compare. **NIPS R2,R4**.
2. Add explanations on rank selection. Add necessary references. Explain why not use the proposed rank selection strategy in simulations. Explain the computational issues of grid search. **NIPS R3, ICML R1, R3**.
3. Discuss the existence of $\mathcal{B}_{true}$ and discuss the mismatching loss $\sum \hat{\mathcal{B}}_{ijk} \neq \mathcal{B}_{trueijk}$. **NIPS R4**.
4. Compare the computational complexity with other methods. **NIPS R2**.
5. Explain how the 136 subjects in HCP are selected from thousands of subjects in the original data. **NIPS R3**.

2 Model Equivalence

2.1 Our method

Let $\mathcal{Y} = [\![\mathcal{Y}_{i_1,...,i_k}]\!] \in \mathbb{R}^{d_1,\cdots,d_K}$ be the tensor observation, $\boldsymbol{X}_k \in \mathbb{R}^{d_k \times p_k}, k \in [K]$ be the matrix covariates. Our tensor regression model is of form

$$g\left(\mathbb{E}[\mathcal{Y}|\boldsymbol{X}_1,...,\boldsymbol{X}_K]\right) = \mathcal{B} \times_1 \boldsymbol{X}_1 \times_2 \cdots \times_K \boldsymbol{X}_K, \tag{1}$$

where $g(\cdot)$ is the inverse link function, $\mathcal{B} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ is the order-K tensor-valued coefficient. Reformulating the model (1) into $\prod_{i=1}^K d_k$ univariate regressions, we have

$$g\left(\mathbb{E}[\mathcal{Y}_{i_1,...,i_K}|\boldsymbol{X}_1,...,\boldsymbol{X}_K]\right) = \langle \mathcal{B}, \boldsymbol{X}_{1i_1} \circ \cdots \circ \boldsymbol{X}_{Ki_K} \rangle, \quad \text{for } i_k \in [d_k], k \in [K], \tag{2}$$

where $\boldsymbol{X}_{ki_k}$ refers to the i_k-th row of $\boldsymbol{X}_k$, for all $k \in [K]$, $\langle \cdot, \cdot \rangle$ refers to the inner product of two tensors, and $\circ$ refers to the outer product of two vectors.

2.2 Scalar-on-tensor regression

Let $y \in \mathbb{R}$ denote the scalar response, $\mathcal{X} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ denote the order-K tensor predictor, and $\boldsymbol{Z} \in \mathbb{R}^{p_0}$ denote the vector-valued covariate. Given the i.i.d. sample set $\{y^{(i)}, \mathcal{X}^{(i)}, \boldsymbol{Z}^{(i)}\}_{i=1}^n$, the scalar-on-tensor regression model is

$$g\left(\mathbb{E}[y^{(i)}|\mathcal{X}^{(i)}]\right) = \alpha + \gamma^T \boldsymbol{Z} + \langle \mathcal{B}, \mathcal{X}^{(i)} \rangle, \quad \text{for } i = 1,...,n \tag{3}$$

where $g(\cdot)$ is the inverse link function, $\alpha \in \mathbb{R}$ is the scalar coefficient, $\gamma \in \mathbb{R}^{p_0}$ is the vector-valued coefficient, and $\mathcal{B} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$ is the order-K tensor-valued coefficient.
Ignoring the scalar parameter α and the vector-valued covariate $\boldsymbol{Z}$, our model (2) is equal to the scalar-on-tensor model (3) by defining

$$\mathbb{E}[y^{(i)}|\mathcal{X}^{(i)}] := \mathbb{E}[\mathcal{Y}_{i_1,...,i_K}|\boldsymbol{X}_1,...,\boldsymbol{X}_K] \quad \text{and} \quad \mathcal{X}^{(i)} := \boldsymbol{X}_{1i_1} \circ \cdots \circ \boldsymbol{X}_{Ki_K}, \quad \text{for } i = 1,...,n,$$

where $n = \prod_{i=1}^K d_k$.

2.3 Tensor-on-tensor regression

Let $\mathbb{Y} \in \mathbb{R}^{d_{K+1} \times \cdots \times d_N}$ be the tensor observation, $\mathbb{X} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ be the tensor predictor. Given the i.i.d. sample set $\{\mathbb{Y}^{(i)}, \mathbb{X}^{(i)}\}_{i=1}^n$, the tensor-on-tensor regression is of form

$$g\left(\mathbb{E}[\mathbb{Y}^{(i)}|\mathbb{X}^{(i)}]\right) = \langle \mathbb{X}^{(i)}, \mathbb{B} \rangle_K, \quad \text{for } i = 1,...,n, \tag{4}$$

where $g(\cdot)$ is the inverse link function, $\mathbb{B} \in \mathbb{R}^{d_1 \times \cdots \times d_N}$ is the order-K tensor-valued coefficient, $\langle \cdot, \cdot \rangle_K$ refers to the contracted tensor product; i.e., for two tensors $\mathbb{P} \in \mathbb{R}^{I_1 \times \cdots \times I_L \times d_1 \times \cdots \times d_K}$ and $\mathbb{Q} \in \mathbb{R}^{d_1 \times \cdots \times d_K \times J_1 \times \cdots \times J_M}$, the contracted tensor product $\langle \mathbb{P}, \mathbb{Q} \rangle_K \in \mathbb{R}^{I_1 \times \cdots \times I_L \times J_1 \times \cdots \times J_M}$ and the $(i_1,...,i_L,j_1,...,j_M)$-the entry is

$$(\langle \mathbb{P}, \mathbb{Q} \rangle_K)_{i_1,...,i_L,j_1,...,j_M} = \sum_{a_1,...,a_K}^{d_1,...,d_K} \mathbb{P}_{i_1,...,i_L,a_1,...,a_K} \mathbb{Q}_{a_1,...,a_K,j_1,...,j_M}.$$

Let $N = K$. Then, the contracted product becomes usual inner product, and $\mathbb{Y}^{(i)}$ becomes a scalar for $i \in [n]$. Consequently, the model (4) degenerates to the scalar-on-tensor regression (3), which is equal to our model.

3 Iteration time table

See Table 1.

Setting	Gaussian	Bernoulli	Poisson
$d = 30, r = 6$	2.6	6.0	5.7
$d = 30, r = 3$	0.5	1.1	1.3
$d = 25, r = 6$	1.4	3.7	4.0
$d = 25, r = 3$	0.3	0.6	0.6

Table 1: Iteration time of Algorithm 1. Numerical values are the running time (seconds) required for one iteration in Algorithm 1 under different settings.

References

Chen, H., Raskutti, G., and Yuan, M. (2019). Non-convex projected gradient descent for generalized low-rank tensor regression. *The Journal of Machine Learning Research*, 20(1):172–208.

Gahrooei, M. R., Yan, H., Paynabar, K., and Shi, J. (2020). Multiple tensor-on-tensor regression: An approach for modeling processes with heterogeneous sources of data. *Technometrics*, pages 1–23.

Llosa, C. (2018). Tensor on tensor regression with tensor normal errors and tensor network states on the regression parameter.

Lock, E. F. (2018). Tensor-on-tensor regression. *Journal of Computational and Graphical Statistics*, 27(3):638–647.

Rabusseau, G. and Kadri, H. (2016). Low-rank regression with tensor responses. In *Advances in Neural Information Processing Systems*, pages 1867–1875.

Raskutti, G., Yuan, M., and Chen, H. (2015). Convex regularization for high-dimensional multiresponse tensor regression. *arXiv preprint arXiv:1512.01215*.

Smith, S. M., Nichols, T. E., Vidaurre, D., Winkler, A. M., Behrens, T. E., Glasser, M. F., Ugurbil, K., Barch, D. M., Van Essen, D. C., and Miller, K. L. (2015). A positive-negative mode of population covariation links brain connectivity, demographics and behavior. *Nature neuroscience*, 18(11):1565–1567.

Yu, R. and Liu, Y. (2016). Learning from multiway data: Simple and efficient tensor regression. In *International Conference on Machine Learning*, pages 373–381.

Zhao, Q., Caiafa, C. F., Mandic, D. P., Chao, Z. C., Nagasaka, Y., Fujii, N., Zhang, L., and Cichocki, A. (2012). Higher order partial least squares (HOPLS): a generalized multilinear regression method. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1660–1673.

Zhou, H., Li, L., and Zhu, H. (2013). Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108(502):540–552.