Algorithms Associated with Factorization Machines

Yanyu Liang
Computational Biology Department
Carnegie Mellon University
Pittsburgh, PA 15213
yanyul@andrew.cmu.edu

Xupeng Tong
Computational Biology Department
Carnegie Mellon University
Pittsburgh, PA 15213
xtong@andrew.cmu.edu

Xin Lu
Computational Biology Department
Carnegie Mellon University
Pittsburgh, PA 15213
xlu2@andrew.cmu.edu

1 Introduction

Copy proposal?

OR: (please feel free to improve it!) Factorization machines were proposed in Rendle (2010), which has been used heavily in recommendation system. FMs introduce higher-order term to model interaction between features which works reasonably well in recommendation system context, where input data is sparse but contains some low-dimensional structure, e.g. user tends to rate more movies in a specific genre. For $x \in \mathbb{R}^p$, FMs define $f : \mathbb{R}^p \to \mathbb{R}$ using the formalization shown in Equation (1).

$$\hat{y} = w_0 + w^T x + \sum_{i=1}^{p} \sum_{j=i+1}^{p} v_i^T v_j x_i x_j \tag{1}$$

$$\hat{y} = w_0 + w^T x + x^T W x \tag{2}$$

, where v_i is k-by-1 vector and intuitively every feature is mapped to a $\mathbb{R}^k$ space where the inner product between two vectors describes the strength of interaction between two features. Here, FMs have two potential drawbacks: i) k as hyperparameter of the model is needed to be chosen in practice; ii) FMs is not convex in V. From another perspective, if we let $V = (v_1, v_2, \ldots, v_p)$, then $\sum_{i=1}^{p} \sum_{j=i+1}^{p} v_i^T v_j x_i x_j = x^T V V^T x - x^T \text{diag}(VV^T)x = x^T(VV^T - \text{diag}(VV^T))x = x^T W x$. If we model x_i^2 term as well, W is equivalent to VV^T and $\text{rank}(W) = k$. Therefore, we can think of FMs as a linear model with second-order term where coefficients of second-order term are regularized by a low rank constraint (see Equation (2). With this idea, Yamada et al. (2015) proposed a convex formalization of FMs (cFMs), where they introduced trace norm to get rid of picking hyperparameter k and instead of using V they used W directly to formalize the problem, which leads the whole problem be convex. To solve cFMs problem, they proposed a coordinate descent method where they iteratively optimize w_0, w and W greedily. However, the introduction of W with trace norm regularizer makes the optimization expensive, because we need to deal with W directly, which is a p-by-p matrix. Additionally, Lin and Ye (2016) re-formed the FMs (referred as gFMs) by removing the implicit constraint that W should be positive semi-definite and W has zeros in diagonal entries. Namely, they replace $W = V^T V$ with $U^T V$. To solve gFMs, they proposed a mini-batch algorithm which guarantees to converge with $O(\epsilon)$ reconstruction error when the sampling complexity is $O(k^3 p \log(1/\epsilon))$.

2 Problem set up

The goal of the project is to explore the optimization method for FMs, cFMs, and gFMs in regression setting with squared error loss as criteria, see Equation (3),

$$L(w_0, w, V)/L(w_0, w, W) = \frac{1}{n}\sum_{i=1}^{n}(y^i - \hat{y}(x^i, w_0, w, V/W))^2 \tag{3}$$

with various smooth and non-smooth regularizations on w, V and W (depends on the formalization) in regression case. Here, we define FMs with the form in Equation (4). And we consider the following regularizations on w: i) $\|w\|_2^2$; ii) $\|w\|_1$; iii) $\|\text{vec}(V)\|_2^2$.

$$\hat{y} = w_0 + w^T x + x^T V^T V x \tag{4}$$

$$\hat{y} = w_0 + w^T x + x^T U^T V x \tag{5}$$

$$\text{, where } U, V \in \mathbb{R}^{k \times p}$$

For cFMs formalization defined in Equation (2), we would like to first explore the case with trace norm penalty as stated in Yamada et al. (2015). Additionally, we would like to add sparsity constraint on interaction term, because the interaction between features should be sparity under the assumption that only features in the same genres have strong interaction and the interaction between different genres is relatively minor. Therefore, we plan to explore i) $\|W\|_{\text{tr}}$; ii) $\|\text{vec}(W)\|_1$, especially we want to explore the case where we need low rank and sparsity at the same time. For gFMs case, we plan to implement the algorithm proposed in Lin and Ye (2016) and compare its performance with others empirically.

3 Methods

3.1 Solving FMs

Rendle (2010) proposed a stochastic gradient descent method to optimize it. The gradient of every term is as follow:

$$\nabla_{w_0}\hat{y} = 1$$
$$\nabla_{w}\hat{y} = x$$
$$\nabla_{V}\hat{y} = 2Vxx^T$$

The gradient of smooth regularizer $\|w\|_2^2$ and $\|\text{vec}(V)\|_2^2$ is:

$$\nabla\|w\|_2^2 = 2w$$
$$\nabla\|\text{vec}(V)\|_2^2 = 2V$$

And the proximal operator of $\|w\|_1$ is the basic soft-thresholding function:

$$\{\text{prox}_t(x)\}_i = \begin{cases} x_i - t & \text{, if } x_i > t \\ 0 & \text{, if } x_i \in [-t, t] \\ x_i + t & \text{, if } x_i < -t \end{cases}$$

Since the only non-smooth term is $\|w\|_1$, then we can optimize the criteria $L(w_0, w, V)$ with proximal gradient descent and accelerated proximal method.

3.2 Solving cFMs

Beyond trace norm penalty on W, the introduction of non-smooth sparsity constraint makes the whole algorithm described above fail. We plan to apply subgradient method as baseline method and test the performance of augmented Lagrange Multiplier method on this problem. Here, the optimization problem we consider is the following:

$$\min_{w_0, w, W} L(x, w_0, w, W) + \lambda_1\|w\|_2^2 + \lambda_2\|W\|_{\text{tr}} + \lambda_3\|\text{vec}(W)\|_1 \tag{6}$$

Since every term in the objective is convex, from the additive property of subgradient operator, a subgradient (let's use ∂f to denote a subgradient of f) of objective is given by the following quantities:

$$\partial_{W_{ij}} \|\text{vec}(W)\|_1 = \begin{cases} 1 & \text{, if } W_{ij} < 0 \\ -1 & \text{, if } W_{ij} > 0 \\ 0 & \text{, otherwise} \end{cases}$$

$$\partial \|W\|_{\text{tr}} = U^T V$$

$$\text{, where } U, V \text{ is given by } W = U^T \Sigma V$$

Additionally, by combining subgradient and proximal method, we can use the following iterator to update W:

$$W^k \leftarrow \text{prox}_{\|\text{vec}(W)\|_1, t_k}(W^{k-1} - t_k(\nabla_W L^k + \partial \|W^k\|_{\text{tr}}))$$

, where t_k can be set as $\frac{1}{k}$.

Li et al. (2015) solved the problem with both trace norm and l_1 penaly by augmented Lagrange multipliers (ALM), which was used in Lin et al. (2010) to solve matrix completion problem and robust PCA. Inspired by their works, we can reformalize eq. (6) as follow:

$$\min_{w_0, w, W} L(x, w_0, w, W) + \lambda_1 \|w\|_2^2 + \lambda_2 \|W\|_{\text{tr}} + \lambda_3 \|\text{vec}(P)\|_1 \tag{7}$$

$$\text{subject to} \quad W - P = 0 \tag{8}$$

The Lagrangian is:

$$\mathcal{L}(w_0, w, W, P, Y, \mu) = L(x, w_0, w, W) + \lambda_1 \|w\|_2^2 + \lambda_2 \|W\|_{\text{tr}} + \lambda_3 \|\text{vec}(P)\|_1 + \langle Y, W - P \rangle + \frac{\mu}{2} \|\text{vec}(W - P)\|_2^2 \tag{9}$$

, where Y is Lagrange multiplier. Then we can apply general ALM algorithm proposed in Lin et al. (2010) as stated in algorithm 1. Intuitively, μ^k can be seen as a parameter penalizing on the difference between W and P, which pushes the equality constraint to be satisfied as μ increases. In practice, we can increase μ^k geometrically (say with factor $\rho > 0$) and every time solving the subproblem in while loop, we use the previous solution as warm start.

Algorithm 1 ALM method solving eq. (7)

1: $\mu_0 > 0$
2: **while** not converge **do**
3: solve $\arg\min \mathcal{L}(w_0, w, W^{k-1}, P, Y^k, \mu^k)$ for w_0^k, w^k, P^k
4: solve $\arg\min \mathcal{L}(w_0^k, w^k, W, P^k, Y^k, \mu^k)$ for W^k
5: $Y_{k+1} \leftarrow Y^k + \mu^k(W^k - P^k)$
6: Update μ^k to μ^{k+1}
7: **end while**

4 Plan

References

S. Rendle, "Factorization machines," in *2010 IEEE International Conference on Data Mining*, Dec 2010, pp. 995–1000.

M. Yamada, A. Goyal, and Y. Chang, "Convex factorization machine for regression," *arXiv preprint arXiv:1507.01073*, 2015.

M. Lin and J. Ye, "A non-convex one-pass framework for generalized factorization machine and rank-one matrix sensing," *arXiv preprint arXiv:1608.05995*, 2016.

J. Li, X. Chen, D. Zou, B. Gao, and W. Teng, "Conformal and low-rank sparse representation for image restoration," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 235–243.

Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.