

Factorization Machines

Tianjian Li

March 2022

1 FM

To understand Factorization Machines[1], we first need to understand how we make a prediction with a simple SVM. The SVM algorithm first assigns a feature vector $x_i \in \mathbf{R}^d$ for each training example labeled with y_i . Then the feature vector is multiplied by a trained weight vector $w \in \mathbf{R}^d$.

Given a test example x If $w^\top x + b \geq 1$, then the SVM model predicts x to be in the positive category, if $w^\top x + b \leq -1$, then the SVM model predicts x to be in the negative category.

However, SVM assumes that the features are independent, failing to take correlated features into account. If we were to assign a weight to any feature pair of (x_i, x_j) , the equation would become

$$\hat{y} = b + \sum_i w_i x_i + \sum_i \sum_{j>i} w_{ij} x_i x_j$$

Assuming there are d different features, this method would require $d + d^2$ parameters, which is not scalable to a large number of features. Therefore, we assign a vector $v_i \in \mathbf{R}^k$ to each feature, and express the weight of any feature pair (x_i, x_j) as the dot product of v_i and v_j .

$$w_{ij} = \langle v_i, v_j \rangle = \sum_{f=1}^k v_{i,f} v_{j,f}$$

Instead of using d^2 parameters to express the interactions, we only use d vectors of size k , which can be computed in $O(kd)$ time.

Lemma 1. *The equation above can be computed in $O(kd)$ time.*

Proof.

$$\begin{aligned}
& \sum_{i=1}^d \sum_{j=i+1}^d v_i \cdot v_j x_i x_j \\
&= \frac{1}{2} \sum_{i=1}^d \sum_{j=1}^d v_i \cdot v_j x_i x_j - \frac{1}{2} \sum_{i=1}^d v_i \cdot v_i x_i x_i \\
&= \frac{1}{2} \sum_{i=1}^d \sum_{j=1}^d \sum_{r=1}^k v_{i,r} \cdot v_{j,r} x_i x_j - \frac{1}{2} \sum_{i=1}^d \sum_{r=1}^k v_{i,r}^2 x_i^2 \\
&= \frac{1}{2} \sum_{r=1}^k \left(\left(\sum_{i=1}^d v_{i,r} x_i \right) \left(\sum_{j=1}^d v_{j,r} x_j \right) - \sum_{i=1}^d v_{i,r}^2 x_i^2 \right) \\
&= \frac{1}{2} \sum_{r=1}^k \left(\left(\sum_{i=1}^d v_{i,r} x_i \right)^2 - \sum_{i=1}^d v_{i,r}^2 x_i^2 \right)
\end{aligned}$$

which computation time complexity is $O(kd)$ □

2 Derivative and Generalization of FM

The derivative of FM is calculated as

$$\frac{\partial}{\partial w} \hat{y} \begin{cases} 1, & \text{if } w = w_0 \\ x_i, & \text{if } w = w_i \\ x_i \sum_{j=1}^d v_{j,r} x_j - v_{i,r} x_i^2, & \text{if } w = v_{i,r} \end{cases} \quad (1)$$

Here $\sum_{j=1}^d v_{j,r} x_j$ can be pre-computed since it is independent of i .

The 2-way FM can also be generalized to n-way FM to capture the relations between n different features.

$$\hat{y}(x) = w_0 + \sum_{i=1}^d w_i x_i + \sum_{l=2}^n \sum_{i_1=1}^d \dots \sum_{i_l=1}^d \left(\prod_{j=1}^l x_{i_j} \right) \left(\sum_{r=1}^k \prod_{j=1}^l v_{i_j,r}^{(l)} \right)$$

which can also be computed in linear time.

3 FFM

Factorization Machines(FM) attempts to measure the correlations between different features. Field Aware Factorization Machines(FFM) makes an improvement to FM in assigning a field to each feature. For example if we are building a recommender system with three different features of a training sample: Gender

of buyer, Item Category and Time of the Year, we denote as (G, C, T). In FM, the third term that measures the correlation between feature pairs is

$$w_g \cdot w_c x_g x_c + w_g \cdot w_t x_g x_t + w_c \cdot w_t x_c x_t$$

However, this method fails to recognize that the influence of Gender on Category is different from the influence of Gender on Purchase Time. Therefore we should just use a single vector w_g on the male gender. In this case, we use two different vectors $w_{g,c}, w_{g,t}$. In FFM, the third term becomes:

$$w_{g,c} \cdot w_{c,g} x_g x_c + w_{g,t} \cdot w_{t,g} x_g x_t + w_{c,t} \cdot w_{t,c} x_c x_t$$

To address in a more mathematical way:

$$\phi_{FFM}(w, x) = \sum_{i=1}^d \sum_{j=i+1}^d (w_{i,f_j} \cdot w_{j,f_i} x_i x_j)$$

In the original paper[2], FFM is trained with stochastic gradient descent(SGD) with a RMSprop optimizer.

References

- [1] Steffen Rendle. “Factorization Machines”. In: *2010 IEEE International Conference on Data Mining*. 2010, pp. 995–1000. DOI: 10.1109/ICDM.2010.127.
- [2] Yuchin Juan et al. “Field-Aware Factorization Machines for CTR Prediction”. In: *Proceedings of the 10th ACM Conference on Recommender Systems*. RecSys ’16. Boston, Massachusetts, USA: Association for Computing Machinery, 2016, pp. 43–50. ISBN: 9781450340359. DOI: 10.1145/2959100.2959134. URL: <https://doi.org/10.1145/2959100.2959134>.