# Factorization Machines

Tianjian Li

March 2022

## 1 FM

To understand Factorization Machines, 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 If $w^\top x + b \neq -1$, 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 $v_j$.

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

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.*

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

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

## 2 Derivative Generalization

The derivative of FM is calculated as

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

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.

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

which can also be computed in linear time.