Factorization Machines

Domain	RecSys
tag	Sparse Data Tensor Factorization
Conference / Journal	ICDM
Publish year	2010
정리 날짜	@2024년 1월 15일
AI summary	Factorization Machines (FMs) as a versatile model that integrates the advantages of Support Vector Machines (SVMs) and factorization models, aimed at efficiently processing sparse data sets. It's particularly beneficial for tasks like recommendation systems, where data sparsity and variable interactions pose significant challenges.
AI key info	Factorization Machines, Background & Motivation, Factorization Machine Model, Expressiveness, Parameter Estimation Under Sparsity, Factorization Machines as Predictors, Learning Factorization Machines, d-way Factorization Machine, Conclusion

Background & Motivation

- SVM
  - 머신러닝, 데이터 마이닝에서 가장 널리 사용되는 예측 모델 중 하나
  - collaborative filtering에서는 중요하게 역할 하지 못함

    - 이러한 작업에서 매우 sparse data에서 complex kernel space에서 신뢰할 수 있는 파라미터를 학습할 수 없기 때문에 좋은 성능 못냄
        - Cannot learn reliable parameters (hyperplanes) in complex (non-linear) kernel spaces
- tensor factorized model, specialized factorization models의 단점
    1. 표준 예측 데이터에 적용 불가
    2. 일반적으로 학습 알고리즘의 모델링 및 설계에 노력이 필요한 특정 작업에 대해 개별적으로 도출됨

Factorization Machine

- Factorization Machine은 **general predictor**이며, **high sparsity에서도 reliable parameters를 예측**할 수 있다.
    - FM은 복잡한 interaction도 모델링하고, factorized parametrization를 사용
    - Linear Complexity이며, linear number of parameters (성능 좋고 스피드 빠르다)
        - 예측을 위한 파라미터를 저장할 필요없이 모델 파라미터를 직접 최적화, 저장 가능
- 장점
    - **Sparse data**(SVM은 실패)
    - **Linear Complexity**
    - 모든 실제 값의 특징 벡터에서 작동 가능한 **General Predictor**

Prediction under sparsity

- 대부분의 prediction task에서는 실수값을 가지는 feature vector x로부터 목표변수 T=R인 함수 y를 예측
    - 데이터 셋 $D=\{(x(1),y(1)),(x(2),y(2)),...\}$가 있다고 가정
    - 이 논문에서는 x가 매우 sparse한 문제(vector x의 거의 모든 원소 xi가 0인 문제)를 다룸
    - 큰 categorical variable 도메인을 다루기 때문에 huge sparsity가 생김
    - 일반적으로 볼 수 있는 영화 평점 데이터의 예시

Example 1 Assume we have the transaction data of a movie review system. The system records which user $u \in U$ rates a movie (item) $i \in I$ at a certain time $t \in \mathbb{R}$ with a rating $r \in \{1, 2, 3, 4, 5\}$. Let the users U and items I be:

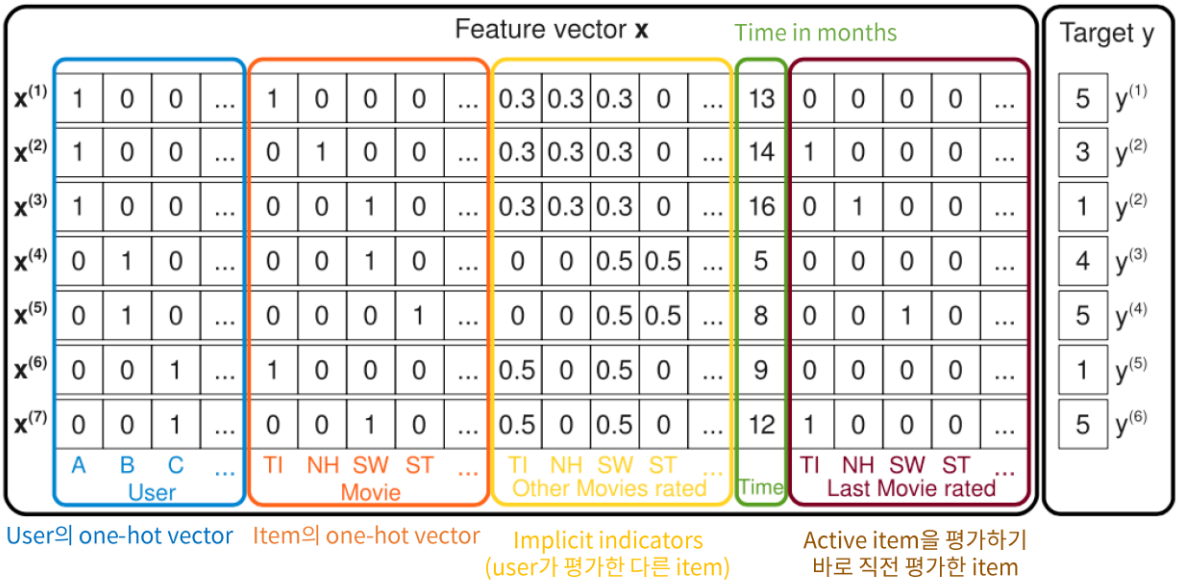
$$U = \{\text{Alice (A)}, \text{Bob (B)}, \text{Charlie (C)}, \ldots\}$$
$$I = \{\text{Titanic (TI)}, \text{Notting Hill (NH)}, \text{Star Wars (SW)}, \text{Star Trek (ST)}, \ldots\}$$

Let the observed data S be:

$$S = \{(\text{A}, \text{TI}, \text{2010-1}, 5), (\text{A}, \text{NH}, \text{2010-2}, 3), (\text{A}, \text{SW}, \text{2010-4}, 1), (\text{B}, \text{SW}, \text{2009-5}, 4), (\text{B}, \text{ST}, \text{2009-8}, 5), (\text{C}, \text{TI}, \text{2009-9}, 1), (\text{C}, \text{SW}, \text{2009-12}, 5)\}$$

- Matrix Factorization은 user와 movie, 그리고 해당 rating만 사용
- Factorization Machine은 더 다양한 정보들을 사용 가능

- transaction에서 생성된 특징 벡터 x 예제



- FM은 SVM과 같은 general predictor이므로 모든 실제 값 특징 벡터에 적용 가능, 추천시스템에 국한되지 않음

Methodology

Factorization Machine Model

$$\hat{y}(x) := w_0 + \Sigma_{i=1}^{n} w_i x_i + \Sigma_{i=1}^{n} \Sigma_{j=i+1}^{n} < v_i, v_j > x_i x_j$$

- w_0: global bias
- $w_i x_i$: i번째 weight 모델링
- $< v_i, v_j >:= \Sigma_{f=1}^{k} f_{i,f}.v_{j,f}$: size k의 두 벡터 내적
- $x_0 \in R, w \in R^n, V \in R^{n \times k}$ → 추정해야할 파라미터
- 2-way FM (degree=2) : 개별 변수 또는 변수 간 interaction 모두 모델링한다.
- 다항 회귀와 매우 흡사하지만, coefficient 대신 파라미터마다 embedding vector를 만들어서 내적

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

- Matrix Factorization → $W_u \times W_i$
  - user와 item latent vector
- Factorization Machine → $W_i \times x_i$
  - x_i마다 구한다

- 변수간 latent vector 조합을 고려한다
- 이 때, degree = 2인 경우 모든 interaction을 얻을 수 있다
- Pairwise feature interaction을 고려하기 때문에 sparse한 환경에 적합하다

Expressiveness

- 어떠한 양의 행렬 W에 대해 $W=V \cdot Vt$과 같은 행렬 V가 존재

- k가 충분히 클 때, FM은 모든 행렬 W를 표현 가능
- but sparse한 환경에서는 복잡한 interaction matrix W를 추정한 데이터가 충분하지 않기 때문에 일반적으로 작은 k를 선택해야함
- **k 제한 → FM 표현력 ↑ → sparse할 때 interaction matrix** 더 잘 **일반화** 가능

Parameter Estimation Under Sparsity

- sparse할 때는 변수간의 interaction을 추정하기 어려움
- Factorization machine은 상호작용 매개변수를 factorize하여 독립성을 깨뜨리기 때문에 이러한 환경에서도 상호작용 잘 추정 가능
- 하나의 interaction에 대한 데이터가 관련 interaction의 매개변수를 추정하는데도 도움을 줌

$$
\begin{aligned}
&\sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle \mathbf{v}_i, \mathbf{v}_j \rangle \, x_i \, x_j \\
=&\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \langle \mathbf{v}_i, \mathbf{v}_j \rangle \, x_i \, x_j - \frac{1}{2} \sum_{i=1}^{n} \langle \mathbf{v}_i, \mathbf{v}_i \rangle \, x_i \, x_i \\
=&\frac{1}{2} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{f=1}^{k} v_{i,f} \, v_{j,f} \, x_i \, x_j - \sum_{i=1}^{n} \sum_{f=1}^{k} v_{i,f} \, v_{i,f} \, x_i \, x_i \right) \\
=&\frac{1}{2} \sum_{f=1}^{k} \left(\left(\sum_{i=1}^{n} v_{i,f} \, x_i \right) \left(\sum_{j=1}^{n} v_{j,f} \, x_j \right) - \sum_{i=1}^{n} v_{i,f}^2 \, x_i^2 \right) \\
=&\frac{1}{2} \sum_{f=1}^{k} \left(\left(\sum_{i=1}^{n} v_{i,f} \, x_i \right)^2 - \sum_{i=1}^{n} v_{i,f}^2 \, x_i^2 \right)
\end{aligned}
$$

- Factorization of pairwise interaction
- model complexity : $O(kn^2) \rightarrow O(kn)$
- 2개 변수에 직접적으로 연관 있는 model parameter가 없다.
- Pairwise interaction식을 정리하면 다음과 같음

Factorization Machines as Predictors

- 다양한 예측 작업에 사용 가능
  1. **회귀(regression)** : y^(x)를 ㅈpredictor로 직접 사용, 최적화 기준은 minimal least square error
  2. **이진 분류 (binary classification)** : y^(x)의 부호 사용, hinge loss 또는 logit loss에 최적화
  3. **순위화(ranking)** : x는 y^(x)의 점수에 따라 정렬, 최적화는 pairwise classification loss가 있는 ($x(a)$,$x(b)$)$\in D$ 인스턴스 벡터 쌍에 대해 수행

Learning Factorization Machines

- 모델 파라미터는 SGD와 같은 경사하강법에 의해서 효과적인 학습 가능
- 모든 파라미터 업데이트는 sparse할 때 O(kn) or O(km(x))로 수행 가능

$$\frac{\partial}{\partial \theta}\hat{y}(\mathbf{x}) = \begin{cases} 1, & \text{if } \theta \text{ is } w_0 \\ x_i, & \text{if } \theta \text{ is } w_i \\ x_i \sum_{j=1}^{n} v_{j,f} x_j - v_{i,f} x_i^2, & \text{if } \theta \text{ is } v_{i,f} \end{cases} \quad (4)$$

d-way Factorization Machine

$$\hat{y}(x) := w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{l=2}^{d} \sum_{i_1=1}^{n} \cdots \sum_{i_l=i_{l-1}+1}^{n} \left(\prod_{j=1}^{l} x_{i_j} \right) \left(\sum_{f=1}^{k_l} \prod_{j=1}^{l} v_{i_j,f}^{(l)} \right) \quad (5)$$

- 2-way FM을 d-way FM으로 일반화할 수 있다.
- 마찬가지로 computation cost는 linear
- http://www.libfm.org 다양하게 활용할 수 있는 module 공개

Conclusion

- Factorized Interaction을 사용하여 feature vector x의 모든 가능한 interaction을 모델링
- FM은 SVM의 generality와 factorization models의 장점을 결합한 것
    1. (High) sparse한 상황에서 interaction을 추정할 수 있다. Unobserved Interaction에 대해서도 일반화할 수 있다.
    2. 파라미터 수, 학습과 예측 시간 모두 linear (Linear Complexity)
    3. 이는 SGD를 활용한 최적화를 진행할 수 있고 다양한 Loss Function을 사용할 수 있다.
- SVM, matrix, tensor and specialized factorization model 보다 더 나은 성능을 증명

Questions